

TW 7697820



# THE UNITED STATES OF AMERICA

TO ALL TO WHOM THESE PRESENTS SHALL COME;

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office

November 01, 2018

THIS IS TO CERTIFY THAT ANNEXED IS A TRUE COPY FROM THE  
RECORDS OF THIS OFFICE OF THE FILE WRAPPER AND CONTENTS  
OF:

APPLICATION NUMBER: *09/736,825*

FILING DATE: *December 14, 2000*

PATENT NUMBER: *6,654,507*

ISSUE DATE: *November 25, 2003*

By Authority of the  
Under Secretary of Commerce for Intellectual Property  
and Director of the United States Patent and Trademark Office

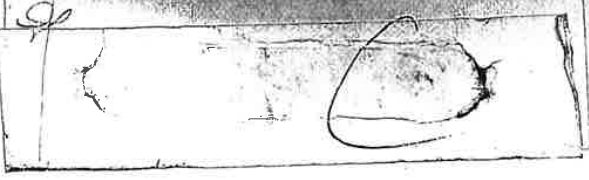


M. TARVER  
Certifying Officer  
PART ( / ) OF ( 2 ) PART(S)

32-4 912

IC784 U.S. PTO  
097736825  
12/14/00

382	Subclass
382	Class
ISSUE CLASSIFICATION	



PATENT NUMBER  
**6654507**

U.S. UTILITY Patent Application

O.I.P.E. PATENT DATE  
M.H. R. W. T. S. Q.A. AA Tm NOV 25 2003

CLASS <b>382</b>	SUBCLASS	ART UNIT <b>2621</b>	EXAMINER
---------------------	----------	-------------------------	----------

TITLE OF INVENTION:

APPLICANT(S):

Serial No: RF038455370  
SIP No: RF038455370  
Continuation of: 0103197934  
Location: N-46-01-2-12-0017-3-01-06  
File Date: 09736825  
Date: 09/21/2003  
Box Type: 1,2  
File Sect: 000002  
1439F  
SP

ISSUING CLASSIFICATION			
ORIGINAL		CROSS REFERENCE(S)	
CLASS	SUBCLASS	CLASS	SUBCLASS (ONE SUBCLASS PER B)
382	282	382	173
INTERNATIONAL CLASSIFICATION		345	620
G06K	9/20	358	453
H04N	1/387		

Continued on Issue Slip Inside File Jacket

10/4/03 Final Drawings (10 shnts) set 03/16/01

<input checked="" type="checkbox"/> <b>TERMINAL DISCLAIMER</b>	DRAWINGS			CLAIMS ALLOWED	
	Sheets Drwg. 10	Figs. Drwg. 19	Print Fig. 3	Total Claims 28	Print Claim for O.G. 1
<input type="checkbox"/> The term of this patent subsequent to _____ (date) has been disclaimed.	Aaron Carter (AUC) 6/16/03 (Assistant Examiner) (Date)			NOTICE OF ALLOWANCE MAILED 6.18.03	
<input type="checkbox"/> The term of this patent shall not extend beyond the expiration date of U.S Patent. No. _____	BHAVESH M. MEHTA SUPERVISORY PATENT EXAMINER TECHNOLOGY CENTER 2600 (Primary Examiner) 6/16/03 (Date)			ISSUE FEE Amount Due: 1,900 Date Paid: 9-10-03	
<input checked="" type="checkbox"/> The terminal _____ months of this patent have been disclaimed.	(Legal Instruments Examiner) 7/05 (Date)			ISSUE BATCH NUMBER	

**WARNING:**  
The information disclosed herein may be restricted. Unauthorized disclosure may be prohibited by the United States Code Title 35, Sections 122, 181 and 368. Possession outside the U.S. Patent & Trademark Office is restricted to authorized employees and contractors only.

Form PTO-436A (Rev. 6/99) FILED WITH:  DISK (CRF)  FICHE  CD-ROM  
(Attached in pocket on right inside flap)

ISSUE FEE IN FILE

(FACE)

PATENT APPLICATION SERIAL NO. \_\_\_\_\_

U.S. DEPARTMENT OF COMMERCE  
PATENT AND TRADEMARK OFFICE  
FEE RECORD SHEET

---

---

12/20/2000 EFLORES 00000126 050225 09736825

01 FC:101	710.00 CH
02 FC:103	144.00 CH

PTO-1556  
(5/87)

\*U.S. GPO: 2000-468-987/39595



UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS  
 UNITED STATES PATENT AND TRADEMARK OFFICE  
 WASHINGTON, D.C. 20231  
 www.uspto.gov




Data Sheet

CONFIRMATION NO. 8670

<b>SERIAL NUMBER</b> 09/736,825	<b>FILING DATE</b> 12/14/2000 <b>RULE</b>	<b>CLASS</b> 382	<b>GROUP ART UNIT</b> 2621	<b>ATTORNEY DOCKET NO.</b> 81595WFN
<b>APPLICANTS</b> Jiebo Luo, Rochester, NY;				
** CONTINUING DATA *****				
** FOREIGN APPLICATIONS *****				
<b>IF REQUIRED, FOREIGN FILING LICENSE</b> GRANTED ** 02/06/2001				
Foreign Priority claimed <input type="checkbox"/> yes <input checked="" type="checkbox"/> no	35 USC 119 (a-d) conditions met <input type="checkbox"/> yes <input checked="" type="checkbox"/> no <input type="checkbox"/> Met after Allowance	STATE OR COUNTRY NY	SHEETS DRAWING 10	TOTAL CLAIMS 28
Verified and Acknowledged	Examiner's Signature <i>[Signature]</i>	Initials RW	INDEPENDENT CLAIMS 2	
<b>ADDRESS</b> Patent Legal Staff Eastman Kodak Company 343 State Street Rochester ,NY 14650-2201				
<b>TITLE</b> Automatically producing an image of a portion of a photographic image				
<b>FILING FEE RECEIVED</b> 854	FEES: Authority has been given in Paper No. _____ to charge/credit DEPOSIT ACCOUNT No. _____ for following:	<input type="checkbox"/> All Fees <input type="checkbox"/> 1.16 Fees ( Filing ) <input type="checkbox"/> 1.17 Fees ( Processing Ext. of time ) <input type="checkbox"/> 1.18 Fees ( Issue ) <input type="checkbox"/> Other _____ <input type="checkbox"/> Credit		



<b>UTILITY PATENT APPLICATION TRANSMITTAL UNDER 37 CFR 1.53(b)</b>	<b>APPlicant NEY DOCKET 81595WFN Customer No. 01333</b>
To: Commissioner for Patents Box Patent Application Washington, D.C. 20231	Express Mail Label No. <b>EL267106180US</b>
AUTOMATICALLY PRODUCING AN IMAGE OF A PORTION OF A PHOTOGRAPHIC IMAGE	Date: <u>December 14, 2000</u>
First Named Inventor (or Application Identifier):  Jiebo Luo	

Enclosed are:

- |   |  |
|---|--|
| 1. <input checked="" type="checkbox"/> Specification  | 6. <input checked="" type="checkbox"/> Assignment of the invention to <u>Eastman Kodak Company</u>   |
| 2. <input type="checkbox"/> 15 Sheet(s) of drawing(s)   | 7. <input type="checkbox"/> Certified copy of a priority document  |
| 3. <input type="checkbox"/> Information Disclosure Statement Under 37 CFR 1.97.   | 8. <input type="checkbox"/> Associate Power of Attorney  |
| 4. Combined Declaration for Patent Application and Power of Attorney:   |  |
| 4a. <input checked="" type="checkbox"/> New   |  |
| 4b. <input type="checkbox"/> Copy from a prior application (37 CFR 1.63(d) (for continuation/divisional with Box 11 completed)  |  |
| 5. <input type="checkbox"/> Incorporation by Reference (useable if Box 4b is checked) The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein. | 9. <input type="checkbox"/> Deletion of Inventor(s). Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b). |

10.  If a 111A application prior to examination of the above-identified application, amend the specification at Page 1, after the title, by inserting the following:  
 --CROSS REFERENCE TO RELATED APPLICATION  
 Reference is made to and priority claimed from U.S. Provisional Application Serial No. , filed , entitled .

If a CONTINUING APPLICATION, check appropriate box and supply the requisite information:

11.  Continuation  Divisional  Continuation-in-part (CIP) of prior application No. ,
12.  Please address all written communications to Thomas H. Close, Patent Legal Staff, Eastman Kodak Company, 343 State Street, Rochester, NY 14650-2201. Please Direct all telephone calls to William F. Noval at (716) 477-5272.

The filing fee has been calculated as shown below:

FOR:	NO. FILED	NO. EXTRA	RATE	FEE
BASIC FEE				\$ 710
TOTAL CLAIMS	28 - 20 =	8	x 18 =	\$ 144
INDEPENDENT CLAIMS	2 - 3 =	0	x 80 =	\$ 0
MULTIPLE DEPENDENT CLAIM PRESENTED			+ 270	\$0
			<b>TOTAL</b>	<b>\$ 854</b>

- Please charge my Eastman Kodak Company Deposit Account No. 05-0225 in the amount of \$ 854 .  
 A duplicate copy of this sheet is enclosed
- The Commissioner is hereby authorized to charge any additional filing fees required under 37 CFR 1.16 or credit any overpayment to Eastman Kodak Company Deposit Account No. 05-0225.  
 A duplicate copy of this sheet is enclosed.

William F. Noval/law  
 Telephone: (716) 477-5272  
 Facsimile: (716) 477-4646

William F. Noval  
 Attorney for Applicants  
 Registration No. 22,049

PATENT APPLICATION BASED ON:

Docket No: 81595/WFN

Inventors: Jiebo Luo

Attorney: William F. Noval

---

**AUTOMATICALLY PRODUCING AN IMAGE OF A PORTION OF A  
PHOTOGRAPHIC IMAGE**

Commissioner for Patents  
Attn: Box Patent Application  
Washington, DC 20231

Express Mail Label No: *EL267106180US*  
Date: *December 14, 2000*

004727 " 62346460

**AUTOMATICALLY PRODUCING AN IMAGE OF A PORTION OF A  
PHOTOGRAPHIC IMAGE**

**CROSS REFERENCE TO RELATED APPLICATION**

5                   Reference is made to commonly assigned U.S. Patent Application  
Serial No. 09/490,915, filed January 25, 2000, entitled "Method for Automatically  
Creating Cropped and Zoomed Versions of Photographic Images" by Jiebo Luo et  
al., and assigned U.S. Patent Application Serial No. 09/223,860, filed December  
31, 1998, entitled "Method for Automatic Determination of Main Subjects in  
10 Photographic Images", by Jiebo Luo et al., the disclosures of which are  
incorporated herein by reference.

**FIELD OF THE INVENTION**

This invention relates in general to producing an image of a portion  
of a photographic image by using digital image processing.

**BACKGROUND OF THE INVENTION**

15                   For many decades, traditional commercial photofinishing systems  
have placed limits on the features offered to consumers to promote mass  
production. Among those features that are unavailable conventionally, zooming  
and cropping have been identified by both consumers and photofinishers as  
20 extremely useful additional features that could potentially improve the quality of  
the finished photographs and the subsequent picture sharing experiences. With  
the advent of, and rapid advances in digital imaging, many of the technical  
barriers that existed in traditional photography no longer stand insurmountable.

25                   Hybrid and digital photography provide the ability to crop  
undesirable content from a picture, and magnify or zoom the desired content to fill  
the entire photographic print. In spite of the fact that some traditional cameras  
with zoom capability provide consumers greater control over composing the  
desired scene content, studies have found that photographers may still wish to  
perform a certain amount of cropping and zooming when viewing the finished  
30 photograph at a later time. Imprecise viewfinders of many point-and-shoot

cameras, as well as simply second-guessing their initial compositions, are factors in the desirability of zoom and crop. In addition, it may be desirable to use some other regular border templates such as ovals, heart shapes, squares, etc. In another scenario, some people commonly referred to as "scrapbookers" tend to perform more aggressive crop in making a scrapbook, e.g., cutting along the boundary of objects.

There are significant differences in objectives and behaviors between these two types of cropping, namely album-making and scrapbook making, with the latter more difficult to understand and summarize. The invention described below performs automatic zooming and cropping for making photographic prints. One customer focus group study indicated that it would be beneficial to provide customers a double set of prints -- one regular and one zoom. Moreover, it is preferred that the cropping and zooming be done automatically. Most customers do not want to think about how the zooming and cropping is being done as long as the content and quality (e.g., sharpness) of the cropped and zoomed pictures is acceptable.

There has been little research on automatic zoom and crop due to the apparent difficulty involved in performing such a task. None of the known conventional image manipulation software uses scene content in determining the automatic crop amount. For example, a program entitled "XV", a freeware package developed by John Bradley at University of Pennsylvania, USA (Department of Computer and Information Science), provides an "autocrop" function for manipulating images and operates in the following way:

the program examines a border line of an image, in all of the four directions, namely from the top, bottom, left and right sides;

the program checks the variation within the line. In grayscale images, a line has to be uniform to be cropped. In color images, both the spatial correlation and spectral correlation have to be low, except for a small percentage of pixels, for the line to be qualified for cropping. In other words, a line will not be cropped if it contains a significant amount of variation;

if a line along one dimension passes the criterion, the next line (row or column) inward is then examined; and

the final cropped image is determined when the above recursive process stops.

This program essentially tries to remove relatively homogeneous margins around the borders of an image. It does not examine the overall content of the image. In practice, the XV program is effective in cropping out the dark border generated due to imprecise alignment during the scanning process.

However, disastrous results can often be produced due to the apparent lack of scene understanding. In some extreme cases, the entire image can be cropped.

Another conventional system, described by Bollman et al. in U.S. Patent 5,978,519 provides a method for cropping images based upon the different intensity levels within the image. With this system, an image to be cropped is scaled down to a grid and divided into non-overlapping blocks. The mean and variance of intensity levels are calculated for each block. Based on the distribution of variances in the blocks, a threshold is selected for the variance. All blocks with a variance higher than the threshold variance are selected as regions of interest. The regions of interest are then cropped to a bounding rectangle. However, such a system is only effective when uncropped images contain regions where intensity levels are uniform and other regions where intensity levels vary considerably. The effectiveness of such a system is expected to be comparable to that of the XV program. The difference is that the XV program examines the image in a line by line fashion to identify uniform areas, while Bollman examines the image in a block by block fashion to identify uniform areas.

In summary, both techniques cannot deal with images with non-uniform background.

In addition, in the earlier invention disclosed in U.S. Patent Application Serial No. 09/490,915, filed January 25, 2000, the zoom factor needs to be specified by the user. There is, therefore, a need for automatically determining the zoom factor in order to automate the entire zoom and crop process.

Some optical printing systems have the capability of changing the optical magnification of the relay lens used in the photographic copying process. In U.S. Patent 5,995,201, Sakaguchi describes a method of varying the effective



magnification of prints made from film originals utilizing a fixed optical lens instead of zoom lens. In U.S. Patent 5,872,619, Stephenson et al. describe a method of printing photographs from a processed photographic filmstrip having images of different widths measured longitudinally of the filmstrip and having heights measured transversely of the filmstrip. This method uses a photographic printer having a zoom lens and a printing mask to provide printed images having a selected print width and a selected print height. In U.S. Patent 4,809,064, Amos et al. describe an apparatus for printing a selected region of a photographic negative onto a photosensitive paper to form an enlarged and cropped photographic print. This apparatus includes means for projecting the photographic negative onto first and second zoom lenses, each of the zoom lenses having an adjustable magnification. In U.S. Patent 5,872,643, Maeda et al. describe a film reproducing apparatus that can effectively perform zoom and crop. This apparatus includes an image pick-up device which picks up a film frame image recorded on a film to generate image data, an information reader which reads information about photographing conditions of the film frame image, and a reproducing area designator which designates a reproducing area of the film frame image. However, the reproducing area of the film frame image is determined based on pre-recorded information about the position of the main object, as indicated by which zone of the photograph the automatic focusing (AF) operation in the camera was on – part of the recorded information about photographing conditions. In all the above-mentioned optical printing systems, the position of the photographic film sample and magnification factor of the relay lens are pre-selected.

#### SUMMARY OF THE INVENTION

According to the present invention, there is provided a solution to the problems of the prior art. It is an object of the present invention to provide a method for producing a portion of a photographic image by identifying the main subject of the photographic image.

According to a feature of the present invention, there is provided a method of producing an image of at least a portion of a digital image,

comprising the steps of:

- a) providing a digital image having pixels;
- b) computing a belief map of the digital image, by using the pixels of the digital image to determine a series of features, and using such features to assign the probability of the location of a main subject of the digital image in the belief map;

- c) determining a crop window having a shape and a zoom factor, the shape and zoom factor determining a size of the crop window; and
- d) cropping the digital image to include a portion of the image of high subject content in response to the belief map and the crop window.

#### **ADVANTAGEOUS EFFECT OF THE INVENTION**

One advantage of the invention lies in the ability to automatically crop and zoom photographic images based upon the scene contents. The digital image processing steps employed by the present invention includes a step of identifying the main subject within the digital image. The present invention uses the identified main subject of the digital image to automatically zoom and crop the image. Therefore, the present invention produces high-quality zoomed or cropped images automatically, regardless whether the background is uniform or not.

#### **BRIEF DESCRIPTION OF THE DRAWINGS**

The foregoing and other objects, aspects and advantages will be better understood from the following detailed description of a preferred embodiment of the invention with reference to the drawings, in which:

Fig. 1 is a schematic diagram of a system embodiment of the invention;

Fig. 2 is a schematic architectural diagram of an embodiment of the invention;

Fig. 3 is a schematic architectural diagram of an embodiment of the invention;

Fig. 4 is a schematic architectural diagram of an embodiment of the invention;

Fig. 5 illustrates the application of the invention to a simulated photograph;

Fig. 6 illustrates the application of the invention to a simulated photograph;

Fig. 7 illustrates the application of the invention to a simulated photograph;

5 Fig. 8 illustrates the application of the invention to a simulated photograph;

Fig. 9 illustrates the application of the invention to a simulated photograph;

10 Fig. 10 illustrates the application of the invention to a simulated photograph;

Fig. 11 illustrates the application of the invention to a simulated photograph;

Fig. 12 illustrates the application of the invention to a simulated photograph;

15 Fig. 13 is an exemplary uncropped photograph;

Fig. 14 is a belief map of the image shown in FIG. 13;

Fig. 15 is a cropped version of the image shown in FIG. 13;

Fig. 17 is a belief map of the image shown in FIG. 16; and

Fig. 18 is a cropped version of the image shown in FIG. 16.

20 **DETAILED DESCRIPTION OF THE INVENTION**

The invention automatically zooms and crops digital images according to an analysis of the main subject in the scene. Previously, a system for detecting main subjects (e.g., main subject detection or "MSD") in a consumer-type photographic image from the perspective of a third-party observer has been  
25 developed and is described in U.S. Patent Application Serial No. 09/223,860, filed December 31, 1998, the disclosure of which is incorporated herein by reference. Main subject detection provides a measure of saliency or relative importance for different regions that are associated with different subjects in an image. Main subject detection enables a discriminative treatment of the scene content for a  
30 number of applications related to consumer photographic images, including automatic crop and zoom.

Conventional wisdom in the field of computer vision, which reflects how a human observer would perform such tasks as main subject detection and cropping, calls for a problem-solving path via object recognition and scene content determination according to the semantic meaning of recognized objects. However, generic object recognition remains a largely unsolved problem despite decades of effort from academia and industry.

The MSD system is built upon mostly low-level vision features with semantic information integrated whenever available. This MSD system has a number of sub-tasks, including region segmentation, perceptual grouping, feature extraction, and probabilistic and semantic reasoning. In particular, a large number of features are extracted for each segmented region in the image to represent a wide variety of visual saliency properties, which are then input into a tunable, extensible probability network to generate a belief map containing a continuum of values.

Using MSD, regions that belong to the main subject are generally differentiated from the background clutter in the image. Thus, automatic zoom and crop becomes possible. Automatic zoom and crop is a nontrivial operation that was considered impossible for unconstrained images, which do not necessarily contain uniform background, without a certain amount of scene understanding. In the absence of content-driven cropping, conventional systems have concentrated on simply using a centered crop at a fixed zoom (magnification) factor, or removing the uniform background touching the image borders. The centered crop has been found unappealing to customers.

The output of MSD used by the invention is a list of segmented regions ranked in descending order of their likelihood (or belief) as potential main subjects for a generic or specific application. This list can be readily converted into a map in which the brightness of a region is proportional to the main subject belief of the region. Therefore, this map can be called a main subject "belief" map. This "belief" map is more than a binary map that only indicates location of the determined main subject. The associated likelihood is also attached to each region so that regions with large values correspond to regions with high confidence or belief of being part of the main subject.

To some extent, this belief map reflects the inherent uncertainty for humans to perform such a task as MSD because different observers may disagree on certain subject matter while agreeing on other subject matter in terms of main subjects. However, a binary decision, when desired, can be readily obtained by using an appropriate threshold on the belief map. Moreover, the belief information may be very useful for downstream applications. For example, different weighting factors can be assigned to different regions (subject matters) in determining the amount of crop.

For determination of crop, the invention uses the main subject belief map instead of a binarized version of the map to avoid making a bad cropping decision that is irreversible. Furthermore, using the continuous values of the main subject beliefs helps trade-off different regions under the constraints encountered in cropping. A binary decision on what to include and what not to include, once made, leaves little room for trade-off. For example, if the main subject region is smaller than the crop window, the only reasonable choice, given a binary main subject map, is to leave equal amounts of margin around the main subject region. On the other hand, secondary main subjects are indicated by lower belief values in the main subject belief map, and can be included according to a descending order of belief values once the main subject of highest belief values are included. Moreover, if an undesirable binary decision on what to include/exclude is made, there is no recourse to correct the mistake. Consequently, the cropping result becomes sensitive to the threshold used to obtain the binary decision. With a continuous-valued main subject belief map, every region or object is associated with a likelihood of being included or a belief value in its being included.

To reduce the degrees of freedom in determining the amount of crop, and to limit the amount of resolution loss incurred in the zoom process, in particular for making photographic prints, in one embodiment, the invention restricts the set of allowable zoom factors to the range of [1.2, 4]. This is based on the findings in the customer focus studies. Those skilled in the art would recognize that the present invention could be used with any the zoom factor.



To reduce the degrees of freedom in determining the amount of crop, in particular for making photographic prints, in one embodiment, the invention restricts the set of allowable zoom factors to the range of [1.2, 4.0]. This is based on the findings in the customer focus studies. In addition, an extremely large zoom factor usually leads to blurry and unacceptable picture due to the limit imposed by the resolution of the original image. If a zoom factor determined by the present invention falls within the range of acceptable zoom factors (e.g., between 1.2 and 4.0), it will be used in the subsequent cropping process. Otherwise, the zoom factor is clipped to 1.2 at the lower end and 4.0 at the higher end.

***General Description of Digital and Optical Printer System***

Referring to Fig. 1, the following description relates to a digital printing system. A source digital image 10 is received by a digital image processor 20. The digital image processor 20 may be connected to a general control computer 40 under operator control from an input control device 60. The monitor device 50 displays diagnostic information about the digital printing system. The general digital image processor 20 performs the needed image processing to produce a cropped and zoomed digital image 99.

Referring to Fig. 1a, the following description relates to an optical printing system. A photographic film sample 31 is received by a film scanner 32 which produces a source digital image 10 relating to the spatial density distribution of the photographic film sample. This source digital image is received by a digital image processor 20. The digital image processor 20 may be connected to a general control computer 40 under operator control from an input control device 60. The monitor device 50 displays diagnostic information about the optical printing system. The general control computer 40 keeps track of the lens magnification setting.

Referring to Fig. 2, a zoom factor 11, which corresponds to the lens magnification setting may also be received by the image processor 20 from the general control computer 40 under operator control. The image processor 20 receives the source digital image 10 and uses the zoom factor 11 and the source digital image 10 to calculate the proper position for the photographic film sample

in the form of a film sample position 9. The photographic film sample is positioned in a gate device 36 which holds the film negative in place during the exposure. The gate device 36 receives the film sample position 9 to position the photographic film sample to adjust which portion of the imaging area of the photograph will be printed.

Referring to Fig. 1a, a lamp house 34 provides the illumination source which is transmitted through the photographic film sample 31 and focused by a lens 12 onto photographic paper 38. The time integration device 13 opens and closes a shutter for a variable length of time allowing the focused light from the lamp house 34 to expose the photographic paper 38. The exposure control device 16 receives a brightness balance value from the digital image processor 20. The exposure control device 16 uses the brightness balance value to regulate the length of time the shutter of the time integration device stays open.

A block diagram of the inventive cropping process (e.g., the digital image understanding technology) is shown in Fig. 3, which is discussed in relation to Figs. 5-12. Figs. 5-12 illustrate the inventive process being applied to an original image shown in Fig. 5.

In item 200, the belief map is created using MSD. The present invention automatically determines a zoom factor (e.g. 1.5X) and a crop window 80 (as shown in Fig. 7), as referred to in item 201 of Fig. 3. This zoom factor is selected by an automatic method based directly on the main subject belief map (e.g., an estimate of the size of the main subject). The crop window is typically a rectangular window with a certain aspect ratio. After the zoom factor is determined by the digital image processor 20, the value of the zoom factor is used subsequently by the digital image processor 20 shown in Fig. 1. In Fig. 1a, the zoom factor is used to communicate with the lens 12 to adjust the lens magnification setting. This adjustment allows the lens 12 to image the appropriate size of the photographic film sample 31 onto the photographic paper 38.

In item 201, regions of the belief map are clustered and the lowest belief cluster (e.g., the background belief) is set to zero using a predefined threshold. As discussed in greater detail below, sections of the image having a belief value below a certain threshold are considered background sections. In

item 202 such sections are given a belief of zero for purposes of this embodiment of the invention.

Then, in item 202 the centroid, or center-of-mass (used interchangeably hereon forth), of nonzero beliefs are computed. More specifically, in Fig. 5 the subject having the highest belief in the belief map is the woman and the stroller. Fig. 7 illustrates that the centroid of this subject is approximately the top of the baby's head.

The centroid  $(\hat{x}, \hat{y})$  of a belief map is calculated using the following procedure:

$$\hat{x} = \sum_i x_i \text{bel}(x_i, y_i), \hat{y} = \sum_i y_i \text{bel}(x_i, y_i),$$

where  $x_i$  and  $y_i$  denote that coordinates of a pixel in the belief map and  $\text{bel}(x_i, y_i)$  represents the belief value at this pixel location.

Before the crop window is placed, a proper crop window is determined in item 203. Referring to Fig. 4, there is shown a block diagram of a method that automatically determines a zoom factor in response to the belief map. In item 301, two second-order central moments,  $c_{xx}$  and  $c_{yy}$ , with respect to the center-of-mass, are computed using the following procedure:

$$c_{xx} = \frac{\sum_i (x_i - \hat{x})^2 \times \text{bel}(x_i, y_i)}{\sum_i \text{bel}(x_i, y_i)}, c_{yy} = \frac{\sum_i (y_i - \hat{y})^2 \times \text{bel}(x_i, y_i)}{\sum_i \text{bel}(x_i, y_i)}.$$

Note that these two terms are not the conventional central moments that are computed without any weighting functions. In the preferred embodiment, a linear weighting function of the belief values is used. However, the conventional central moments, or central moments by a nonlinear function of the belief values, can also be used.

An effective bounding rectangle (MBR) of the regions of high subject content can be calculated using the following procedure, where the dimensions of the MBR are calculated by:

$$D_x = 2 \times \sqrt{3 \times c_{xx}}, D_y = 2 \times \sqrt{3 \times c_{yy}}$$

Fig. 6 illustrates that the effective bounding rectangle 70 is centered at approximately the top of the boy's head and approximately encompasses the region of high subject content. In general, the aspect ratio of the original image is maintained. Therefore, a crop window 80 is determined in item 5 303 such that it is the smallest rectangle of the original aspect ratio that encompasses the effective MBR 70.

In item 204, the initial position of the crop window p 80 is centered at the centroid, as shown in Fig. 7.

The crop window is 80 then moved so that the entire crop window 10 is within the original image (e.g. item 205) as shown in Fig. 8. In item 206, the crop window 80 is moved again so that all the regions of the highest belief values ("main subject") are included within the crop window and to create a margin 81, as shown in FIG. 9. This process (e.g., 206) captures the entire subject of interest. Therefore, as shown in Fig. 9, the top of the woman's head is included in the crop 15 window. Compare this to Fig. 8 where the top of the woman's head was outside the crop window.

Decision box 207 determines whether an acceptable solution has been found, i.e., whether it is possible to include at least the regions of the highest belief values in the crop window.

If an acceptable solution exists, the window is again moved, as 20 shown in item 208, to optimize a subject content index for the crop window. The preferred embodiment of the present invention defines the subject content index as the sum of belief values within the crop window. It should be noted that the present invention specifies higher numerical belief values corresponding to higher 25 main subject probability. Therefore, finding a numerical maximum of the sum of the belief values is equivalent to finding an optimum of the subject content index. This is shown in Fig. 10 where the secondary objects (e.g. flowers) are included within the crop window 80 to increase the sum of beliefs. The sum of beliefs for a crop window is computed as follows.

30 
$$sum(w) = \sum_{(x,y) \in w} bel(x,y),$$

where  $bel(x, y)$  represents the belief value at a given pixel location  $(x, y)$  within the crop window  $w$ .

Provided that the primary subjects are included, moving the crop window so that more of the secondary subjects are included would increase the sum of belief values within the crop window. Recall that the primary subjects are indicated by the highest belief values and the secondary subjects are indicated by belief values lower than those of the primary subjects but higher than those of the background subjects. The goal is to find the crop window that has the highest sum of belief values while ensuring that the primary subjects are completely included in the crop window, i.e.,

$$\tilde{w} = \max_{w \in W} sum(w),$$

where  $W$  denotes the set of all possible crop windows that satisfy all the aforementioned constraints (e.g., those that are completely within the uncropped image and those that encompass the entire primary subjects).

Then, in item 212 (in place of item 209, not shown), the position of the center of the crop window is used to calculate the translational component of the film sample position 9. The gate device 36, shown in Fig. 1a, receives the film sample position 9 and uses this information to control the position of the photographic film sample 31 relative to the lens 12. Those skilled in the art will recognize that either or both of the lens 12 and the photographic film sample 31 may be moved to achieve the centering of the effective cropped image region on the photographic paper 38.

Referring to Fig. 3, if decision box 207 does not produce an acceptable solution, the final position of the crop window is restored to that of item 205. Then, referring to Fig. 1a, the position of the center of the crop window is used to calculate the translational component of the film sample position 9. The gate device 36, shown in Fig. 1, receives the film sample position 9 and uses this information to control the position of the photographic film sample 31 relative to the lens 12.

The simulated image example shown in Figs. 5-12 illustrates the progress the invention makes as it moves through the process shown in Fig. 3.



One could formulate the problem as a global exhaustive search for the best solution. The procedure used in the invention is considered a “greedy” searching approach and is certainly more efficient than conventional processes.

5 The invention utilizes a built-in “k-means” clustering process to determine proper thresholds of MSD beliefs for each application. The invention also uses clustering, as discussed below to enhance the cropping process. In one preferred embodiment, it is sufficient to use three levels to quantize MSD beliefs, namely “high”, “medium”, and “low.” As would be known by one ordinarily skilled in the art, the invention is not limited to simply three levels of  
10 classification, but instead can utilize a reasonable number of classification levels to reduce the (unnecessary) variation in the belief map. These three levels allow for the main subject (high), the background (low), and an intermediate level (medium) to capture secondary subjects, or uncertainty, or salient regions of background. Therefore, the invention can perform a k-means clustering with  
15  $k = 3$  on the MSD belief map to “quantize” the beliefs. Consequently, the belief for each region is replaced by the mean belief of the cluster in that region. Note that a k-means clustering with  $k = 2$  essentially produces a binary map with two clusters, “high” and “low,” which is undesirable for cropping based on earlier discussion.

20 There are two major advantages in performing such clustering or quantization. First, clustering helps background separation by grouping low-belief background regions together to form a uniformly low-belief (e.g., zero belief) background region. Second, clustering helps remove noise in belief ordering by grouping similar belief levels together. The centroiding operation  
25 does not need such quantization (nor should it be affected by the quantization). The main purpose of the quantization used here is to provide a threshold for the background.

The k-means clustering effectively performs a multi-level thresholding operation to the belief map. After clustering, two thresholds can be  
30 determined as follows:

highest resolution of the original data. In both cases, the invention uses an interpolation process to resample the data in order to retain a maximum amount of image detail. In general, edge or detail-preserving image interpolation processes such as cubic-spline interpolation are preferred because they tend to preserve the detail and sharpness of the original image better.

Example consumer photographs and their various cropped versions are shown in pictures "house" (e.g., FIGS. 13-15) and "volleyball" (Figs. 16-18). More specifically, Figs. 13 and 16 illustrate uncropped original photographic images. Figs. 14 and 17 illustrate belief maps, with lighter regions indicating higher belief values. As would be known by one ordinarily skilled in the art given this disclosure, the light intensity variations shown in Figs. 14 and 17 are readily converted into numerical values for calculating the sum of the belief values discussed above. Finally, Figs. 15 and 18 illustrate images cropped according to the invention.

For the "house" picture, both Bradley and Bollman (U.S. Patent 5,978,519) would keep the entire image and not be able to produce a cropped image because of the shadows at the bottom and the tree extending to the top border of the uncropped image (Fig. 13). There are no continuous flat background regions extending from the image borders in this picture, as required by U.S. Patent 5,978,519. Similarly, the top of the tree in Fig. 16 would not be cropped in the system disclosed in U.S. Patent 5,978,519.

Secondary subjects can lead to a more balanced cropped picture. For the "volleyball" picture (Fig. 16), the inclusion of some parts of the tree by the algorithm leads to more interesting cropped pictures than simply placing the main subjects (players) in the center of the cropped image (Fig. 18). The invention was able to do so because the trees are indicated to be of secondary importance based on the belief map Fig. 17. It is obvious that the art taught by Bradley and Bollman in U.S. Patent 5,978,519 would not be able to produce such a nicely cropped image. In fact, both Bradley and Bollman (U.S. Patent 5,978,519) would at best remove the entire lower lawn portion of the picture and keep the tree branches in the upper-left of the uncropped image.

$$threshold_{low} = (C_{low} + C_{medium}) / 2,$$

$$threshold_{high} = (C_{medium} + C_{high}) / 2$$

where  $\{C_{low}, C_{medium}, C_{high}\}$  is the set of centroids (average belief values) for the three clusters, and  $threshold_{low}$  and  $threshold_{high}$  are the low and high thresholds, respectively.

Regions with belief values below the lower threshold are considered “background” and their belief values are set to zero in items 202, 302 and 402 discussed above. Regions with belief values above the higher threshold are considered part of the main subject and need to be included in their entirety, whenever possible. Regions with intermediate belief values (e.g., less than or equal to the higher threshold and greater than or equal to the lower threshold) are considered part of the “secondary subject” and will be included as a whole or partially, if possible, to maximize the sum of main subject belief values retained by the crop window. Note that the variance statistics on the three clusters can be used to set the thresholds more accurately to reflect cluster dispersions.

The invention initializes the k-means process by finding the maximum value  $bel_{maximum}$  and minimum values  $bel_{minimum}$  of the belief map, computing the average value  $bel_{average}$  of the maximum and minimum values for item in the belief map, and setting the initial centroids (denoted by a superscript of 0) at these three values, i.e.,

$$C_{low}^0 = bel_{minimum}, C_{medium}^0 = bel_{average}, C_{high}^0 = bel_{maximum}$$

Other ways of initialization may apply. For more about the k-means process, see Sonka, Hlavac, and Boyle, Image Processing Analysis, and MachineVision, PWS Publishing, 1999 page 307-308. For typical MSD belief maps, the k-means process usually converges in fewer than 10 iterations.

In applications where a zoom version of the cropped area is desired, there are two scenarios to consider. First, the zoom version effectively requires higher spatial resolution than the highest resolution of the original data. However, a visible loss of image sharpness is likely of concern in the situation. Second, the zoom version effectively requires lower spatial resolution than the



**PARTS LIST**

9	film sample position
10	source digital image
11	zoom factor
12	lens
13	time integration device
20	digital image processor
31	photographic film sample
32	film scanner
34	lamp house
36	gate device
38	photographic paper
40	general control computer
50	monitor device
60	input control device
80	crop window
81	margin
99	cropped digital image
200	image and belief map
201	decision box for performing clustering of the belief map
202	decision box for computing the center-of-mass
203	decision box for determining a zoom factor and a crop window
204	decision box for positioning the crop window
205	decision box moving a window
206	decision box for moving a window to contain the highest belief
207	decision box for determining if a solution exists
208	decision box for moving a window to optimize the sum of beliefs
209	decision box for cropping the image
210	cropped image
211	decision box for cropping the image
300	belief map
301	decision box for computing weighted central moments of the belief

CONFIDENTIAL



map with respect to the center-of-mass

302 decision box for computing an effective bounding rectangle (MBR)  
of the main subject content

303 decision box for determining a zoom factor and a crop window that  
encompasses the MBR

---

0041001 "S333E260"

**WHAT IS CLAIMED IS:**

- Sub AI
1. A method of producing an image of at least a portion of a digital image, the digital image comprising the steps of:
    - a) providing a digital image having pixels;
    - b) computing a belief map of the digital image, by using the pixels of the digital image to determine a series of features, and using such features to assign the probability of the location of a main subject of the digital image in the belief map;
    - c) determining a crop window having a shape and a zoom factor, the shape and zoom factor determining a size of the crop window; and
    - d) cropping the digital image to include a portion of the image of high subject content in response to the belief map and the crop window.
  2. The method of claim 1 wherein step c) further comprises the steps of:
    - i) computing a weighted center-of-mass of the belief map, weighted by the belief values of the belief map;
    - ii) computing weighted central moments of the belief map, relative to the center-of-mass and weighted by a weighting function of each belief value of the belief map;
    - iii) computing an effective rectangular bounding box according to the central moments; and
    - iv) determining a crop window having a shape and a zoom factor, the shape and zoom factor determining a size of the crop window
  3. The method of claim 1 wherein step d) further comprises the steps of:
    - i) selecting an initial position of the crop window at a location which includes the center of mass;
    - ii) using the belief values corresponding to the crop window to select the position of the crop window to include a portion of the image of high subject content in response to the belief map; and

CONFIDENTIAL

iii) cropping the digital image according to the position of the crop window.

4. The method of claim 2 wherein step d) further comprises the steps of:

i) selecting a crop window of a rectangular shape and of an identical aspect ratio to the (uncropped) digital image; and

ii) selecting a zoom factor to determine the size of the crop window such that it encompasses the effective bounding box.

5. The method of claim 2 wherein the weighting function in step b) is a linear weighting function.

6. The method of claim 2 wherein the weighting function in step b) is a constant function.

7. The method of claim 3 wherein step b) further comprises the steps of:

i) calculating a subject content index for the crop window derived from the belief values;

ii) following a positioning procedure of repeating step i) for at least two positions of the crop window; and

iii) using the subject content index values to select the crop window position.

8. The method of claim 1 wherein the crop window is completely within the digital image.

SUB 427 9. The method of claim 2 wherein step b) further comprises the step of performing a clustering of the belief map to identify at least a cluster of highest belief values corresponding to main subject, a cluster of intermediate

belief values corresponding to secondary subjects, and a cluster of lowest belief values corresponding to the background.

10. The method of claim 9 wherein said clustering includes setting said background portions to a zero belief value.

11. The method of claim 5 further comprising positioning said crop window such that the subject content index of said crop window is at an optimum.

12. The method of claim 3 further comprising positioning said crop window such that said crop window includes all of said main subject cluster.

13. The method of claim 12 further comprising positioning said crop window to include a buffer around said main subject cluster.

14. A computer storage product having at least one computer storage medium having instructions stored therein causing one or more computers to perform the method of claim 1.

*Pub  
A3* 15. A method of producing an image of a portion of at least a portion of a photographic image onto a photographic receiver, comprising the steps of:

a) receiving a digital image corresponding to the photographic image, the digital image comprising pixels;

b) computing a belief map of the digital image, by using the pixels of the digital image to determine a series of features, and using such features to assign the probability of the location of a main subject of the digital image in the belief map;

c) determining a crop window having a shape and a zoom factor, the shape and zoom factor determining a size of the crop window; and

d) locating the relative optical position of a photographic

image, a lens assembly, and a photographic receiver in response to the belief map and illuminating a portion of the photographic image of high subject content to produce an image of such portion onto the photographic receiver.

16. The method of claim 15 wherein step c) further comprises the steps of:

- i) computing a weighted center-of-mass of the belief map, weighted by the belief values of the belief map;
- ii) computing weighted central moments of the belief map, relative to the center-of-mass and weighted by a weighting function of each belief value of the belief map;
- iii) computing an effective rectangular bounding box according to the central moments; and
- iv) determining a crop window having a shape and a zoom factor, the shape and zoom factor determining a size of the crop window.

17. The method of claim 15 wherein step d) further comprises the steps of:

- i) selecting an initial position of the crop window at a location which includes the center of mass;
- ii) using the belief values corresponding to the crop window to select the position of the crop window to include a portion of the image of high subject content in response to the belief map; and
- iii) cropping the digital image according to the position of the crop window.

18. The method of claim 16 wherein step d) further comprises the steps of:

- i) selecting a crop window of a rectangular shape and of an identical aspect ratio to the (uncropped) digital image; and
- ii) selecting a zoom factor to determine the size of the crop

window such that it encompasses the effective bounding box.

19. The method of claim 16 wherein the weighting function in step b) is a linear weighting function.

20. The method of claim 16 wherein the weighting function in step b) is a constant function.

21. The method of claim 17 wherein step b) further comprises the steps of:

- i) calculating a subject content index for the crop window derived from the belief values;
- ii) following a positioning procedure of repeating step i) for at least two positions of the crop window; and
- iii) using the subject content index values to select the crop window position.

22. The method of claim 15 wherein the crop window is completely within the digital image.

23. The method of claim 16 wherein step b) further comprises the step of performing a clustering of the belief map to identify at least a cluster of highest belief values corresponding to main subject, a cluster of intermediate belief values corresponding to secondary subjects, and a cluster of lowest belief values corresponding to the background.

24. The method of claim 23 wherein said clustering includes setting said background portions to a zero belief value.

25. The method of claim 19 further comprising positioning said crop window such that the subject content index of said crop window is at an optimum.

00447-00000000







As below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name, I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled:

**AUTOMATICALLY PRODUCING AN IMAGE OF A PORTION OF A PHOTOGRAPHIC IMAGE**

The specification of which (check only one item below):

- is attached hereto.
- was filed as United States Application Serial No. \_\_\_\_\_ on \_\_\_\_\_ and was amended on \_\_\_\_\_ (if applicable).
- was filed as PCT international application Number \_\_\_\_\_ on \_\_\_\_\_ and was amended under PCT Article 19 on \_\_\_\_\_ (if applicable).

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose to the U.S. Patent & Trademark Office all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, §1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 of any foreign application(s) for patent or inventor's certificate or of any PCT international application(s) designating at least one country other than the United States of America listed below and have also identified below any foreign applications(s) for patent or inventor's certificate or any PCT international application(s) designating a least one country other than the United States of America filed by me on the same subject matter having a filing date before that of the application(s) of which priority is claimed:

**PRIOR FOREIGN/PCT APPLICATION(S) AND ANY PRIORITY CLAIMS UNDER 35 U.S.C. 119:**

COUNTRY (if PCT, indicate PCT)	APPLICATION NUMBER	DATE OF FILING (day month year)	PRIORITY CLAIMED UNDER 35 USC §119	
			YES	NO

I hereby claim the benefit under Title 35, United States Code, 119 §(e) of any United States provisional application(s) listed below:

**PRIOR PROVISIONAL APPLICATION(S) AND ANY PRIORITY CLAIMS UNDER 35 U.S.C. §119 (e):**

PROVISIONAL APPLICATION NUMBER	FILING DATE

I hereby claim the benefit under Title 35, United States Code, §120 of any prior United States application(s) or PCT international application(s) designating the United States of America that is/are listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in that/those prior application(s) in the manner provided by the first paragraph of Title 35, §112, I acknowledge the duty to disclose to the U.S. Patent & Trademark Office all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations §1.56, which became available between the filing date of the prior application(s) and the national or PCT international filing date of this application:

**PRIOR US APPLICATIONS OR PCT INTERNATIONAL APPLICATIONS DESIGNATING THE U.S FOR BENEFIT UNDER 35USC§120:**

U.S. APPLICATIONS			STATUS (Check one)		
U.S. APPLICATION NUMBER	U.S. FILING DATE		PATENTED	PENDING	ABANDONED
PCT APPLICATIONS DESIGNATING THE U.S.					
PCT APPLICATION NO.	PCT FILING DATE	U.S. SERIAL NUMBERS ASSIGNED (if any)			

**POWER OF ATTORNEY:** As a named inventor, I hereby appoint the attorney(s) and/or agent(s) associated with Eastman Kodak Company Customer No. 01333 to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

**Send Correspondence to:**

Patent Legal Staff  
Eastman Kodak Company  
343 State Street  
Rochester, NY 14650-2201

**Direct Telephone Calls to:**  
*(name and telephone number)*

William F. Noval  
(716) 477-5272  
FAX: (716) 477-4646

	FULL NAME OF INVENTOR	FAMILY NAME	FIRST GIVEN NAME	SECOND GIVEN NAME
2		Luo	Jiebo	
0	RESIDENCE & CITIZENSHIP	CITY	STATE OR FOREIGN COUNTRY	COUNTRY OF CITIZENSHIP
		Rochester	New York 14620 USA	PRC
1	BUSINESS ADDRESS	BUSINESS ADDRESS	CITY	STATE & ZIP CODE (COUNTRY)
		Eastman Kodak Company	343 State Street, Rochester	New York 14650 USA
2	FULL NAME OF INVENTOR	FAMILY NAME	FIRST GIVEN NAME	SECOND GIVEN NAME
0	RESIDENCE & CITIZENSHIP	CITY	STATE OR FOREIGN COUNTRY	COUNTRY OF CITIZENSHIP
2	BUSINESS ADDRESS	BUSINESS ADDRESS	CITY	STATE & ZIP CODE (COUNTRY)
2	FULL NAME OF INVENTOR	FAMILY NAME	FIRST GIVEN NAME	SECOND GIVEN NAME
0	RESIDENCE & CITIZENSHIP	CITY	STATE OR FOREIGN COUNTRY	COUNTRY OF CITIZENSHIP
3	BUSINESS ADDRESS	BUSINESS ADDRESS	CITY	STATE & ZIP CODE (COUNTRY)
2	FULL NAME OF INVENTOR	FAMILY NAME	FIRST GIVEN NAME	SECOND GIVEN NAME
0	RESIDENCE & CITIZENSHIP	CITY	STATE OR FOREIGN COUNTRY	COUNTRY OF CITIZENSHIP
4	BUSINESS ADDRESS	BUSINESS ADDRESS	CITY	STATE & ZIP CODE (COUNTRY)
2	FULL NAME OF INVENTOR	FAMILY NAME	FIRST GIVEN NAME	SECOND GIVEN NAME
0	RESIDENCE & CITIZENSHIP	CITY	STATE OR FOREIGN COUNTRY	COUNTRY OF CITIZENSHIP
5	BUSINESS ADDRESS	BUSINESS ADDRESS	CITY	STATE & ZIP CODE (COUNTRY)
2	FULL NAME OF INVENTOR	FAMILY NAME	FIRST GIVEN NAME	SECOND GIVEN NAME
0	RESIDENCE & CITIZENSHIP	CITY	STATE OR FOREIGN COUNTRY	COUNTRY OF CITIZENSHIP
6	BUSINESS ADDRESS	BUSINESS ADDRESS	CITY	STATE & ZIP CODE (COUNTRY)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issuing thereon.

SIGNATURE OF INVENTOR 201	SIGNATURE OF INVENTOR 202	SIGNATURE OF INVENTOR 203
DATE	DATE	DATE
SIGNATURE OF INVENTOR 204	SIGNATURE OF INVENTOR 205	SIGNATURE OF INVENTOR 206
DATE	DATE	DATE

007736825 12/14/2000

582

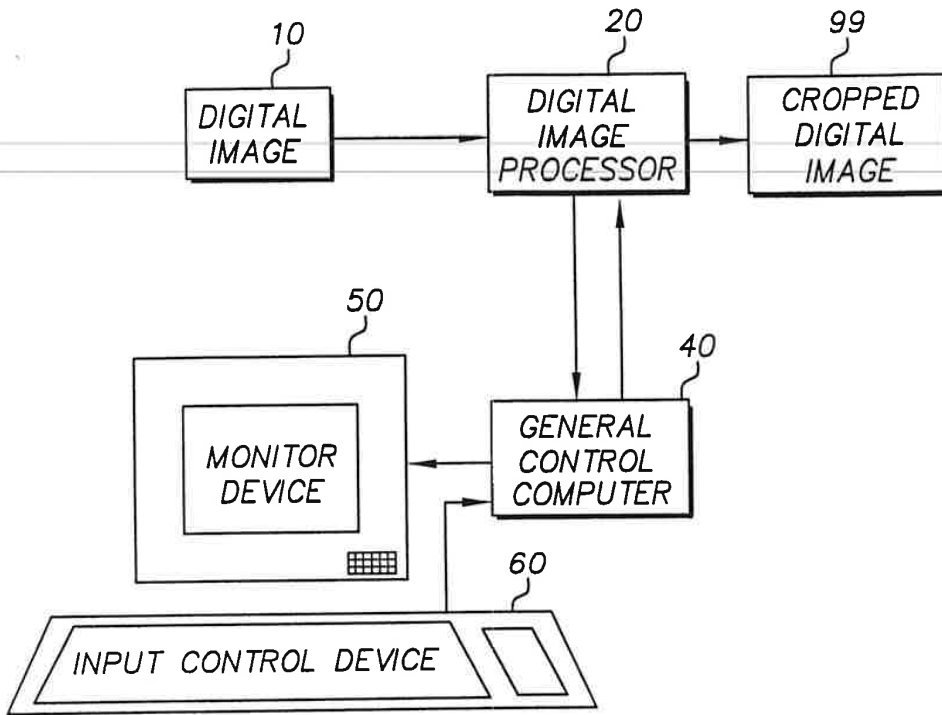


FIG. 1

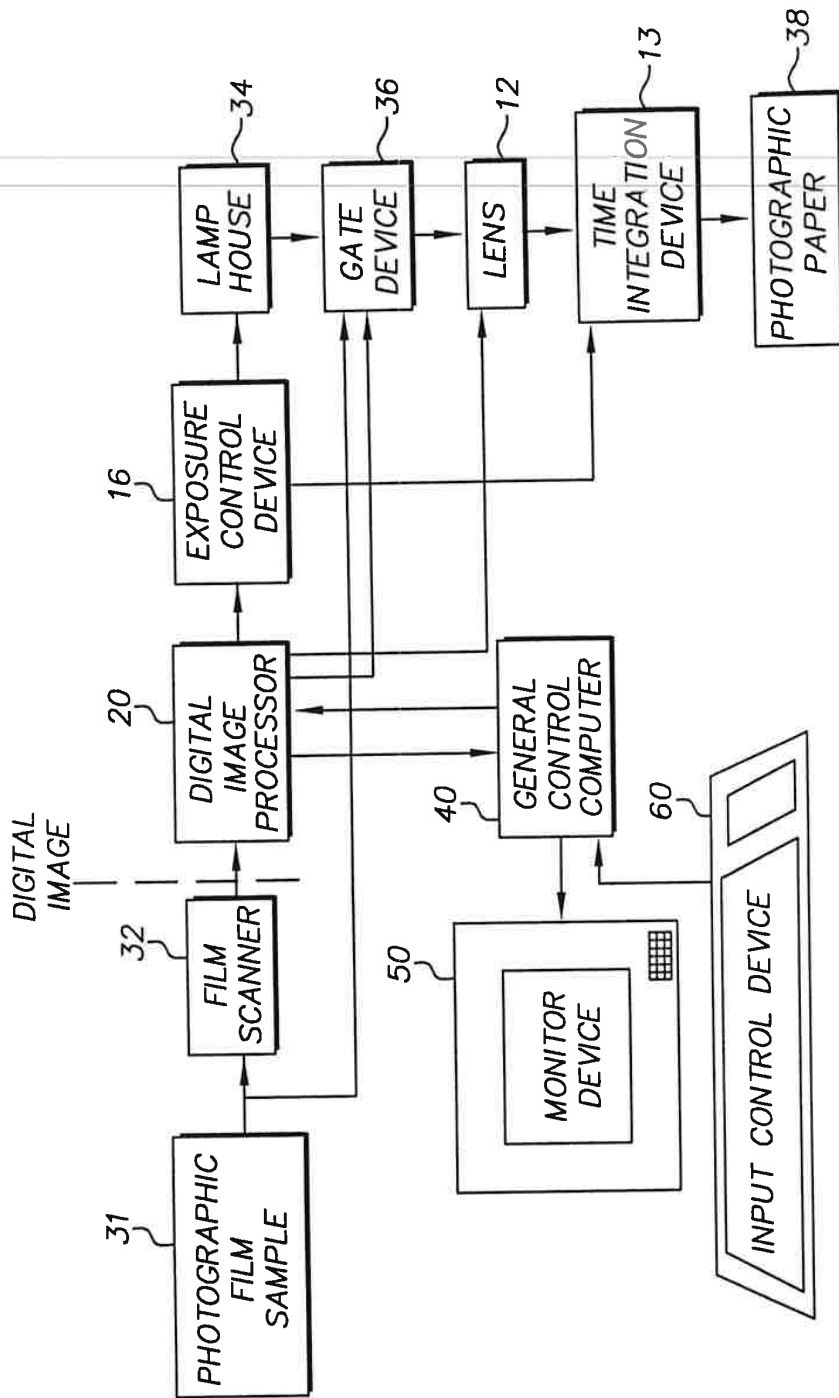


FIG. 1A

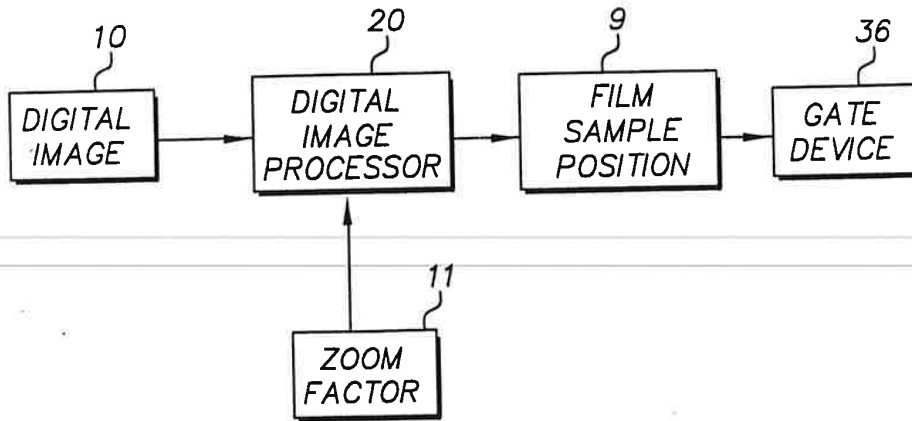


FIG. 2

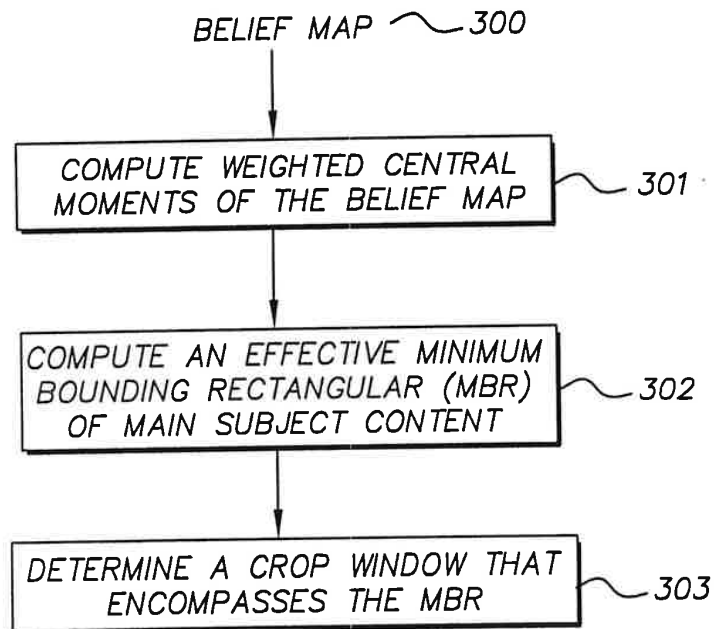


FIG. 4

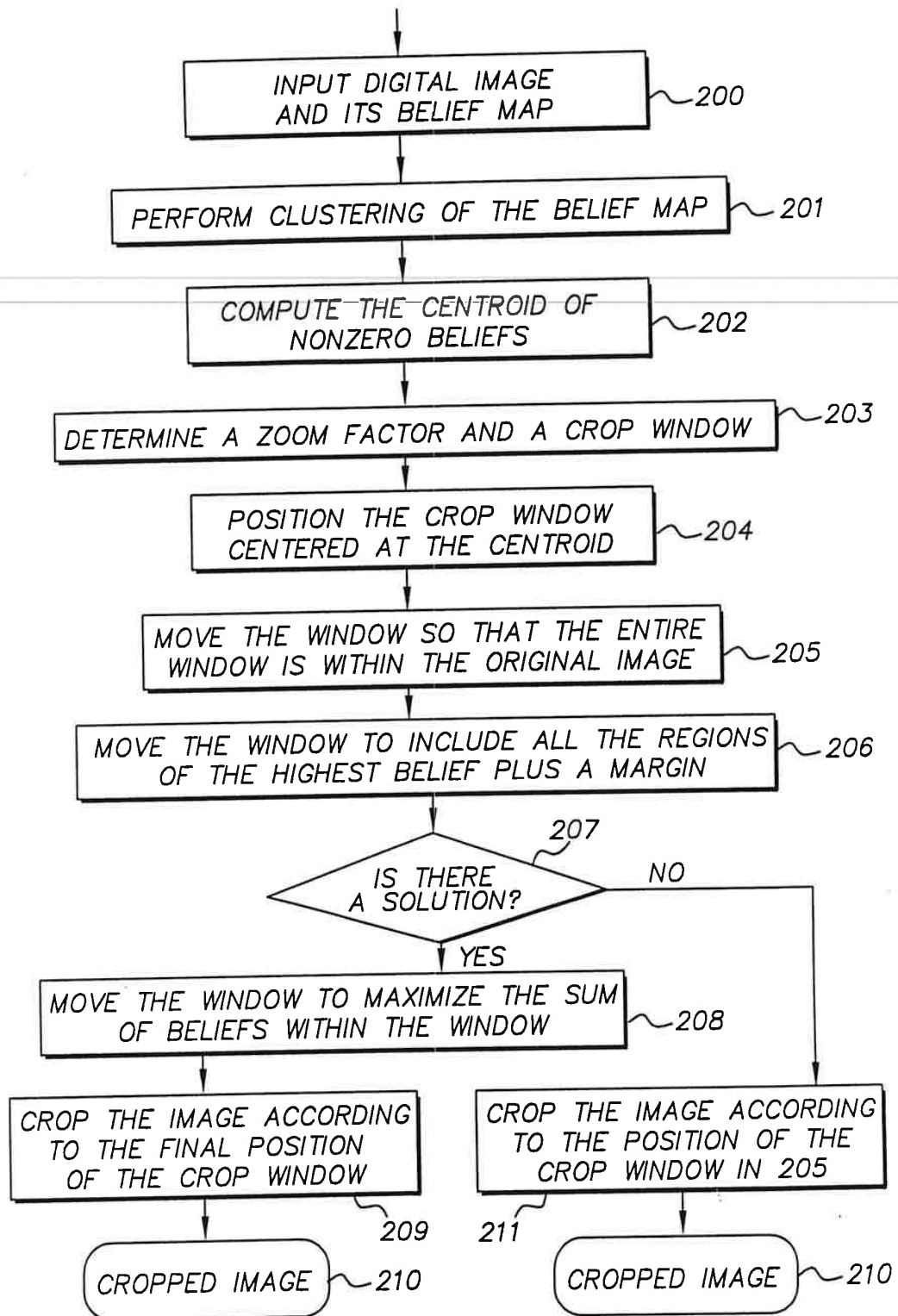


FIG. 3

09/27/2008 10:40:00 AM



Fig. 5

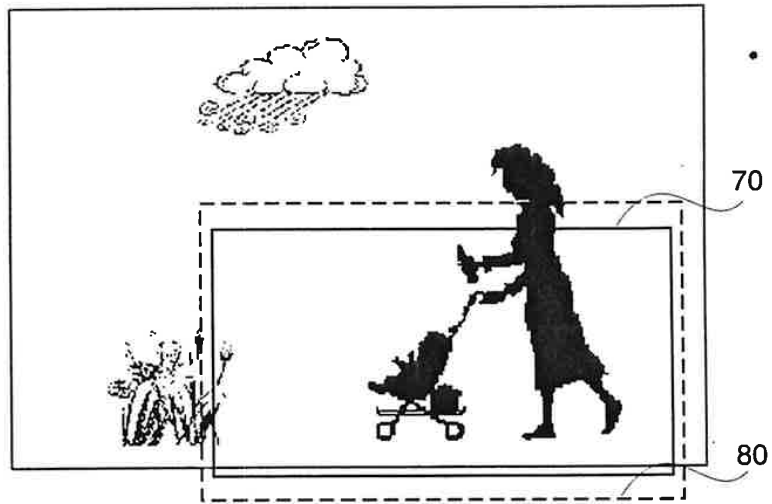


Fig. 6

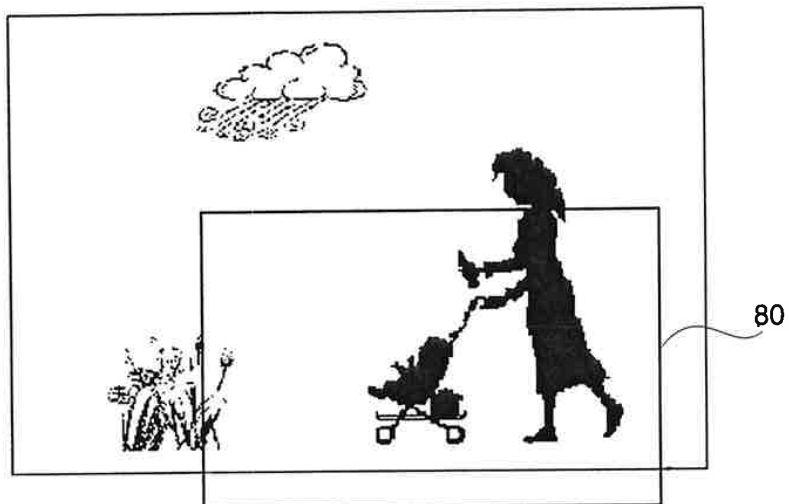


Fig. 7



Fig. 8



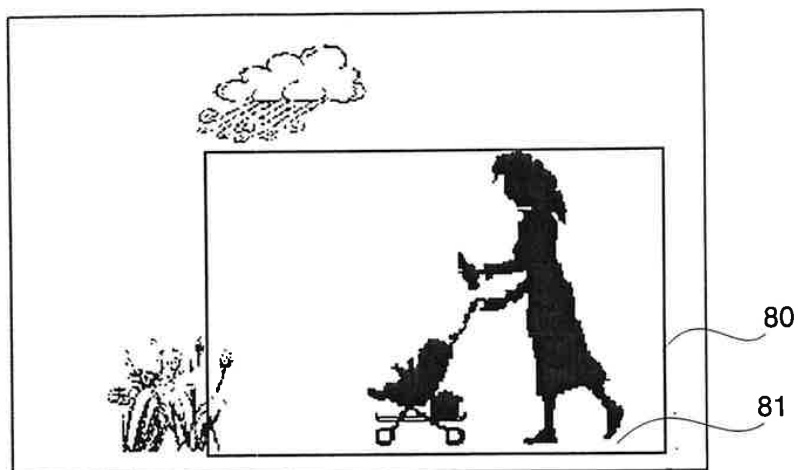


Fig. 9

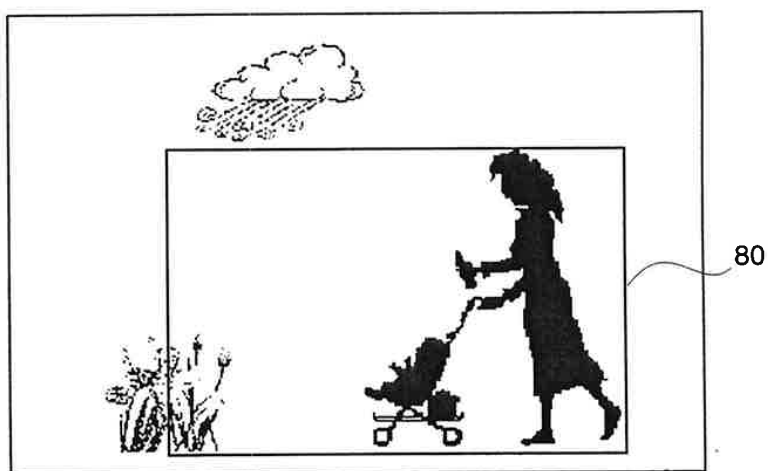


Fig. 10



Fig. 11



Fig. 12



Fig. 13

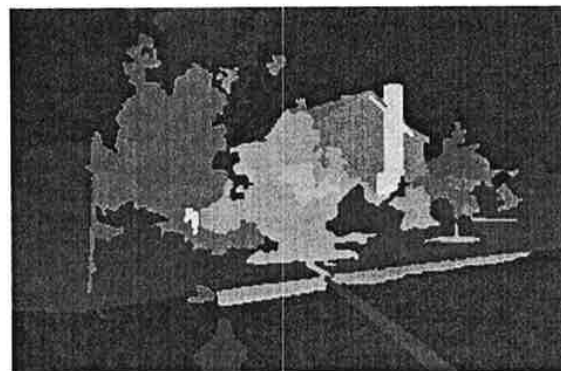


Fig. 14



Fig. 15

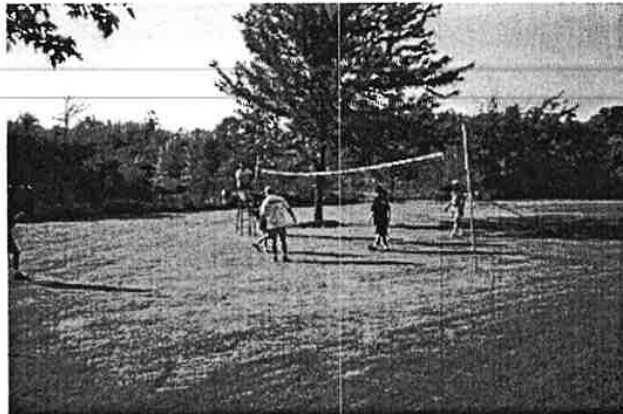


Fig. 16

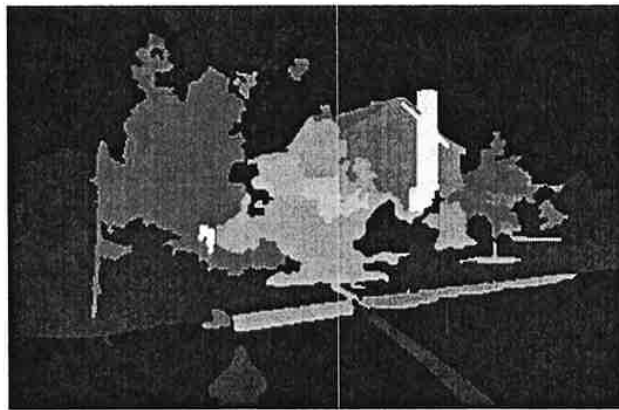


Fig. 17



Fig. 18



## UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS  
UNITED STATES PATENT AND TRADEMARK OFFICE  
WASHINGTON, D.C. 20231  
www.uspto.gov

#2

APPLICATION NUMBER	FILING/RECEIPT DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NUMBER
09/736,825	12/14/2000	Jiebo Luo	81595WFN

CONFIRMATION NO. 8670

## FORMALITIES LETTER



\*OC000000005740796\*

Patent Legal Staff  
Eastman Kodak Company  
343 State Street  
Rochester, NY 14650-2201

Date Mailed: 02/07/2001

## NOTICE TO FILE CORRECTED APPLICATION PAPERS

***Filing Date Granted***

This application has been accorded an Application Number and Filing Date. The application, however, is informal since it does not comply with the regulations for the reason(s) indicated below. Applicant is given **TWO MONTHS** from the date of this Notice within which to correct the informalities indicated below. Extensions of time may be obtained by filing a petition accompanied by the extension fee under the provisions of 37 CFR 1.136(a)

The required item(s) identified below must be timely submitted to avoid abandonment:

- Substitute drawings in compliance with 37 CFR 1.84 because:
  - drawings submitted to the Office are not electronically reproducible. Drawing sheets must be submitted on paper which is flexible, strong, white, smooth, non-shiny, and durable (see 37 CFR 1.84(e));

*A copy of this notice **MUST** be returned with the reply.*

AM

Customer Service Center  
Initial Patent Examination Division (703) 308-1202

PART 3 - OFFICE COPY



#3

81595WFN  
Customer No. 01333

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Jiebo Luo

AUTOMATICALLY PRODUCING  
AN IMAGE OF A PORTION OF A  
PHOTOGRAPHIC IMAGE

Serial No. 09/736,825

Filed December 14, 2000

Group Art Unit: 2621

Batch:

Allowed:

Examiner:

I hereby certify that this correspondence is being deposited today with the United States Postal Service as first class mail in an envelope addressed to Commissioner for Patents, Washington, D.C. 20231.

*Laurio A. Wurtz*  
Laurio A. Wurtz

*March 14, 2001*  
Date

Commissioner for Patents  
Washington, D.C. 20231

Sir:

LETTER TO THE OFFICIAL DRAFTSPERSON

Enclosed are 10 sheets of formal drawings depicting Figure(s) 1-18. Please substitute these drawings for those currently on file in the subject application. These drawings correct the informalities noted in the Notice to File Corrected Application Papers mailed February 7, 2001.

The Commissioner is hereby authorized to charge any fees in connection with this communication to Eastman Kodak Company Deposit Account No. 05-0225. **A duplicate copy of this letter is enclosed.**

Respectfully submitted,

*William F. Noval*

Attorney for Applicant  
Registration No. 22,049

William F. Noval/law  
Telephone: (716) 477-5272  
Facsimile: (716) 477-4646

E:\grp\THC\WFN\doctets\81595\drftprsn.doc

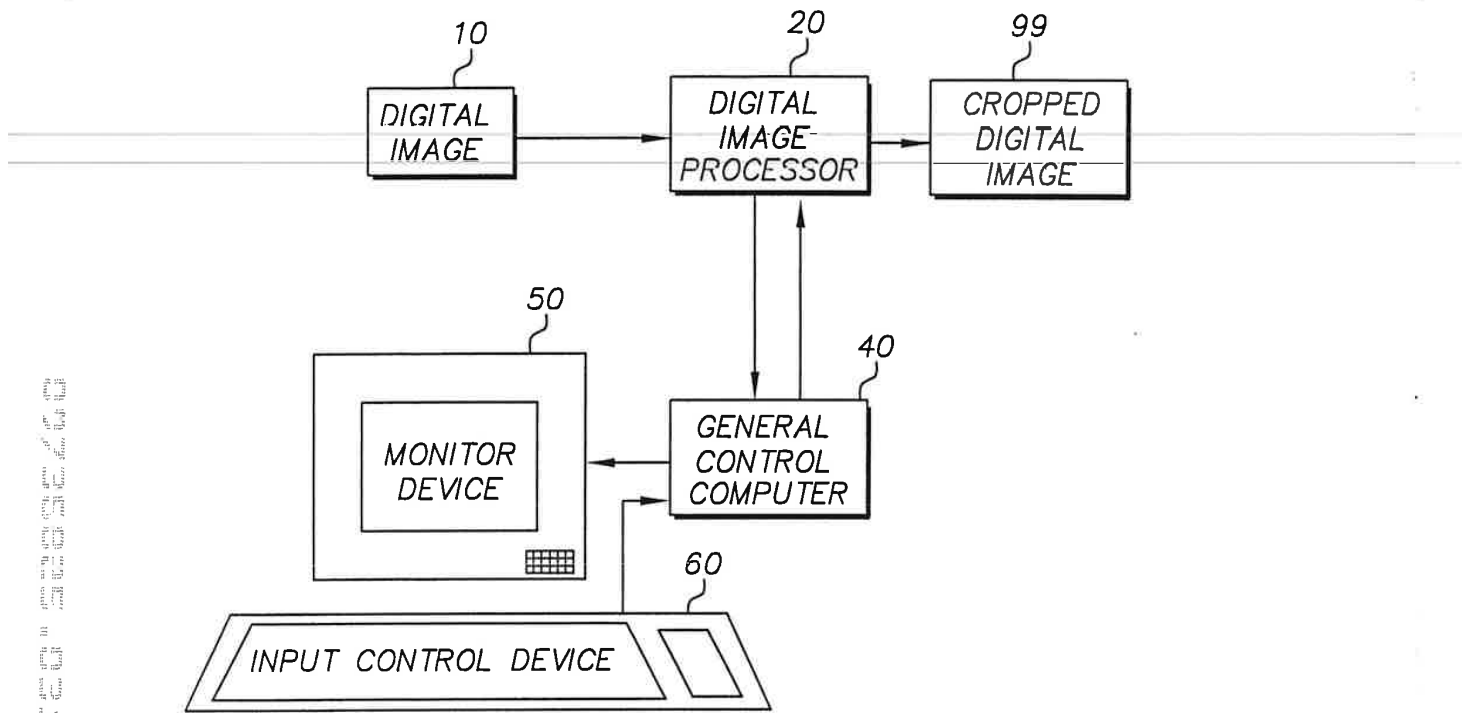


FIG. 1

PHOTOGRAPHIC FILM SAMPLE

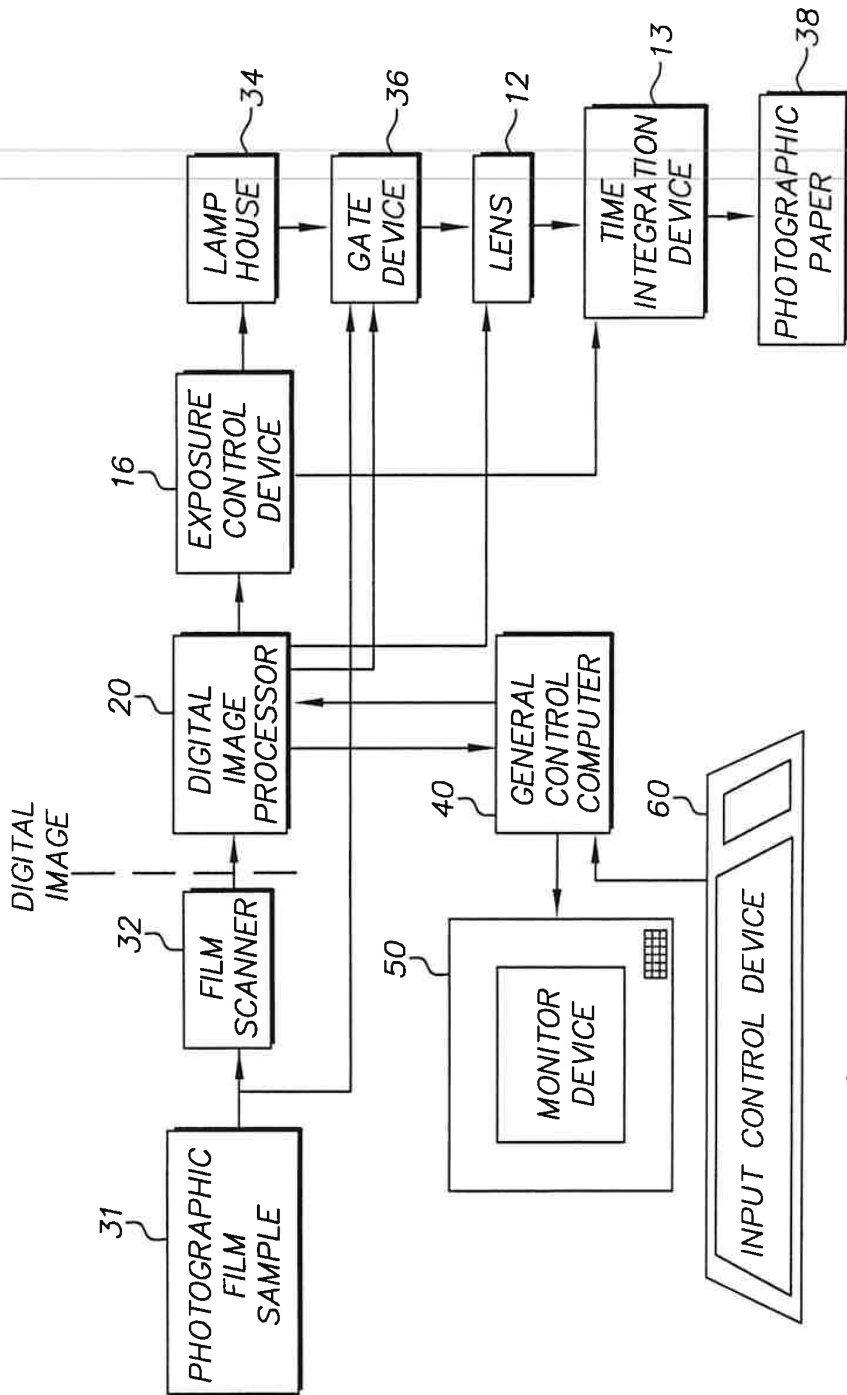


FIG. 1A

2



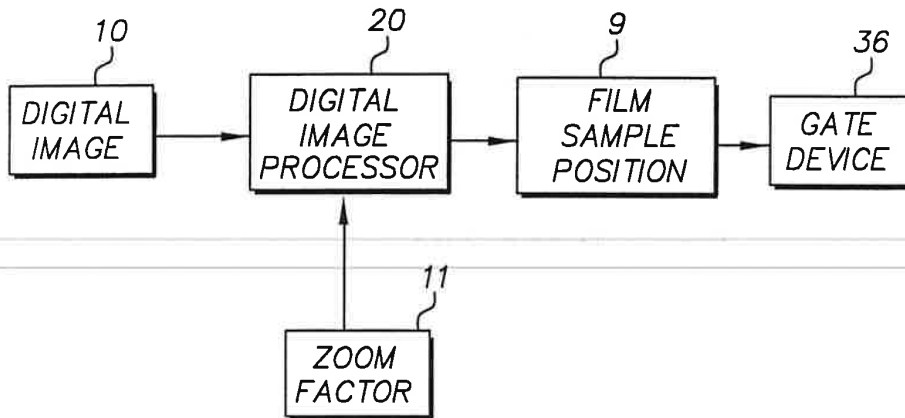


FIG. 2

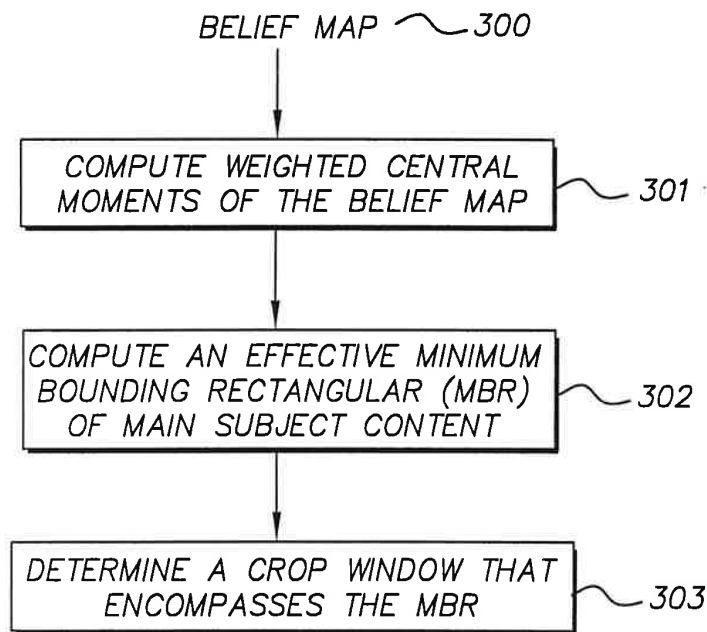


FIG. 4

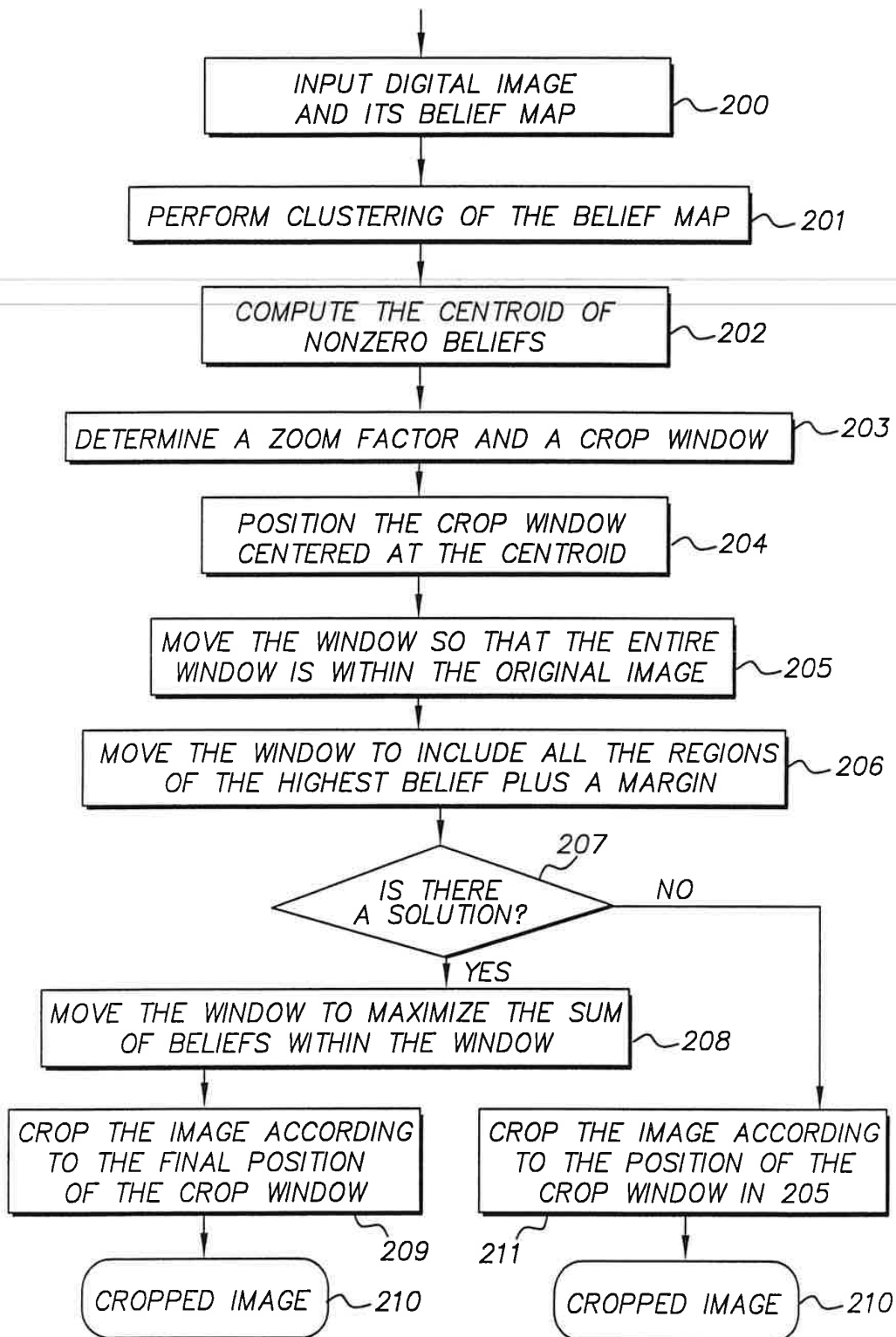


FIG. 3



Fig. 5

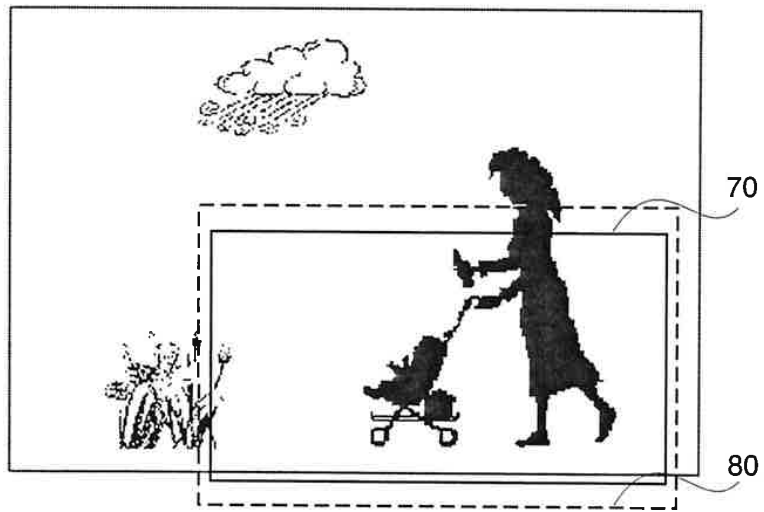


Fig. 6

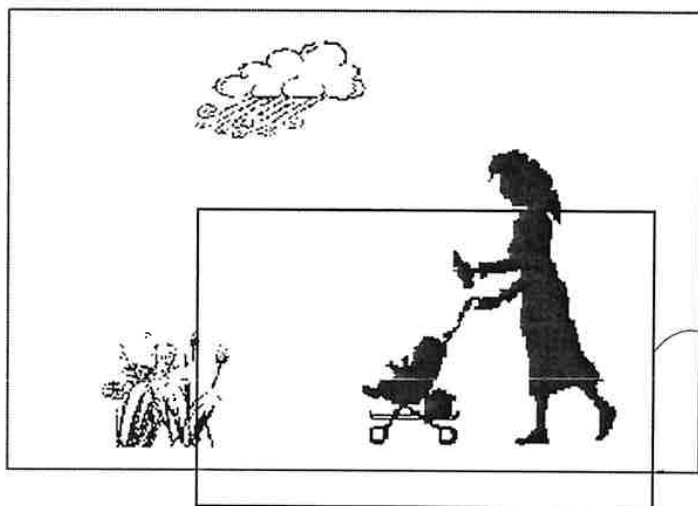


Fig. 7

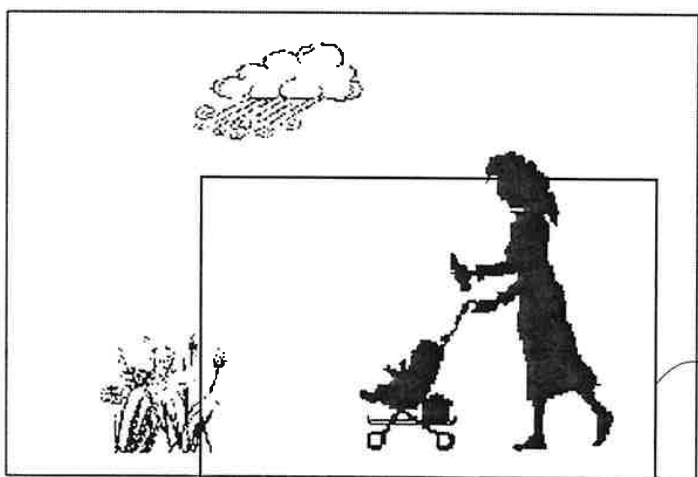


Fig. 8

6

FIG. 9 AND FIG. 10 ARE SIMILAR TO FIG. 8, BUT WITH THE FIG. 8 ELEMENTS 80 AND 81 IN THE FIG. 9 AND FIG. 10 SCENES.

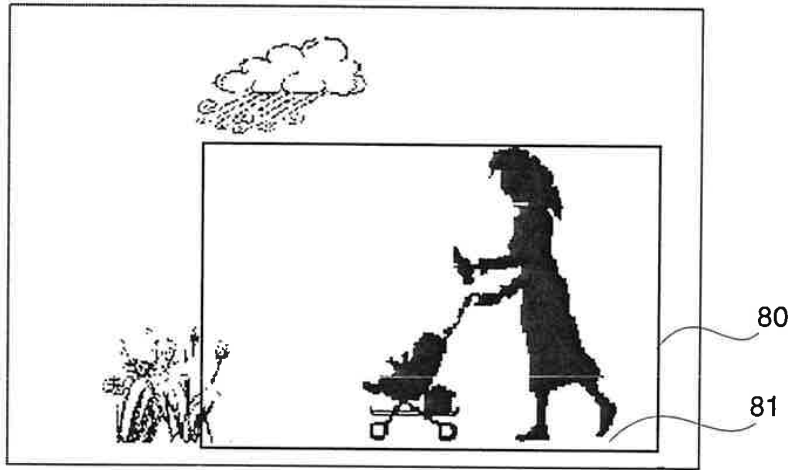


Fig. 9

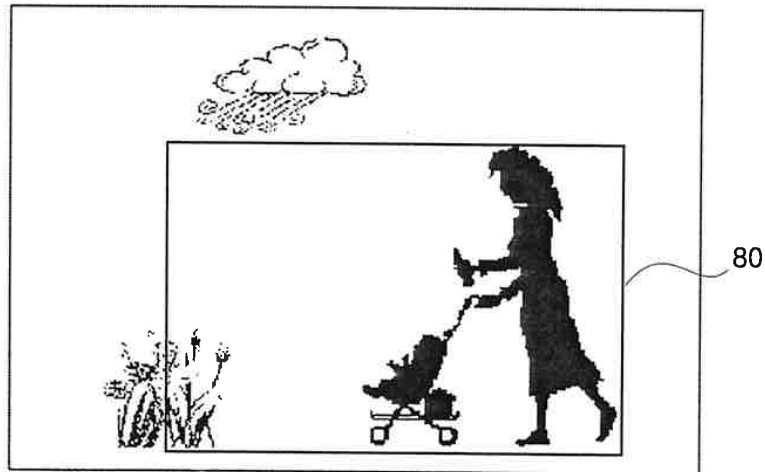


Fig. 10



Fig. 11

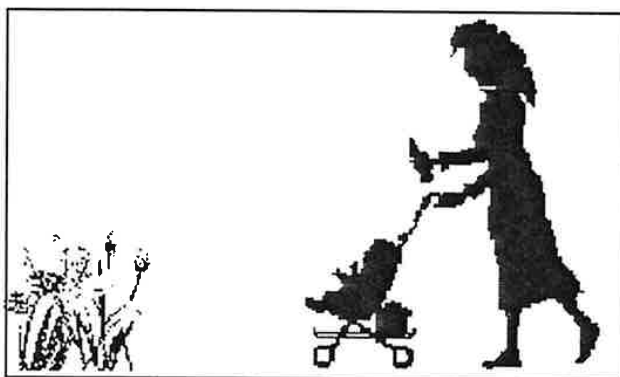


Fig. 12



Fig. 13

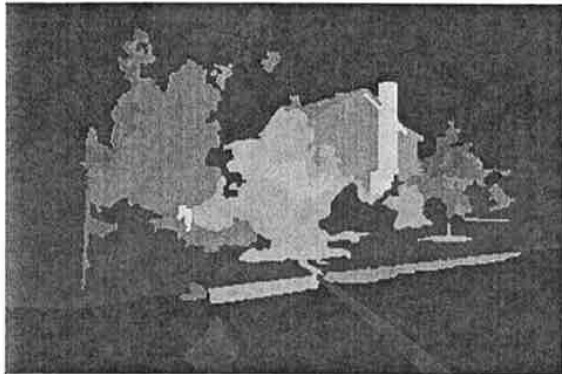


Fig. 14



Fig. 15

Figure 13, 14, and 15 are computer-generated images of the house in Figure 13. Figure 14 is a stylized, high-contrast representation of the house and surrounding landscape, possibly a computer-generated or processed image of the scene in Figure 13. Figure 15 is a black and white photograph of the same house and landscape as in Figure 13, but with a different lighting or processing effect, making the scene appear darker and more atmospheric.

THE UNIVERSITY OF MICHIGAN LIBRARY



Fig. 16

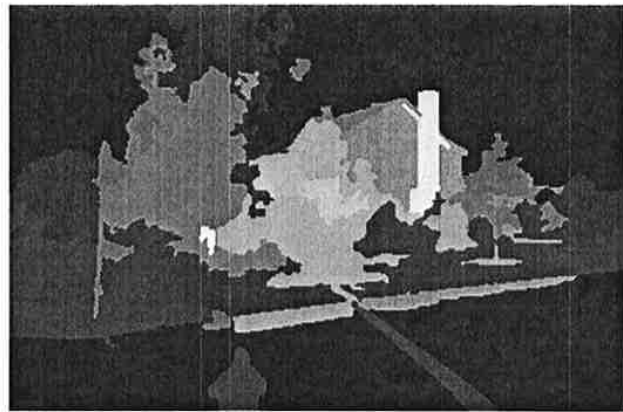


Fig. 17



Fig. 18



0360



UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS  
 UNITED STATES PATENT AND TRADEMARK OFFICE  
 WASHINGTON, D.C. 20231  
 www.uspto.gov



APPLICATION NUMBER	FILING/RECEIPT DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NUMBER
9/736,825	12/14/2000	Jiebo Luo	81595WFN

CONFIRMATION NO. 8670

Patent Legal Staff  
 Eastman Kodak Company  
 343 State Street  
 Rochester, NY 14650-2201

EASTMAN KODAK CO.  
 FEB 13 2001  
 PATENT LEGAL STAFF

FORMALITIES LETTER



\*OC00000005740796\*

Date Mailed: 02/07/2001

**NOTICE TO FILE CORRECTED APPLICATION PAPERS**

***Filing Date Granted***

This application has been accorded an Application Number and Filing Date. The application, however, is informal since it does not comply with the regulations for the reason(s) indicated below. Applicant is given **TWO MONTHS** from the date of this Notice within which to correct the informalities indicated below. Extensions of time may be obtained by filing a petition accompanied by the extension fee under the provisions of 37 CFR 1.136(a)

The required item(s) identified below must be timely submitted to avoid abandonment:

- Substitute drawings in compliance with 37 CFR 1.84 because:
  - drawings submitted to the Office are not electronically reproducible. Drawing sheets must be submitted on paper which is flexible, strong, white, smooth, non-shiny, and durable (see 37 CFR 1.84(e));

*A copy of this notice **MUST** be returned with the reply.*

Customer Service Center AM  
 Initial Patent Examination Division (703) 308-1202

PART 2 - COPY TO BE RETURNED WITH RESPONSE

*[Handwritten mark]*

Docket 81595WFN  
Customer No. 01333

3c784 U.S. PTO  
09/736825  
12/14/00

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of

Group Art Unit:

Jiebo Luo

Examiner:

AUTOMATICALLY PRODUCING  
AN IMAGE OF A PORTION OF A  
PHOTOGRAPHIC IMAGE

Express Mail Label No. EL267106180US

*December 14, 2000*

Date

Serial No.

Filed December 14, 2000

Commissioner for Patents  
Washington, D.C. 20231

Sir:

INFORMATION DISCLOSURE STATEMENT FOR CONSIDERATION  
BY THE OFFICE UNDER 37 C.F.R. 1.97-1.99

Enclosed herewith are patents and/or publications for consideration by the Patent and Trademark Office in regard to the invention claimed in the above-described application. In compliance with §1.56, such documents are listed in the enclosed Form PTO-1449.

Applicants request that the Patent and Trademark Office make of record the above-identified documents. Unless otherwise indicated, a full text copy of each document is attached. For documents not in English, an English translation or an equivalent English language patent or publication is attached. Where a translation is not available, a concise explanation of the relevance of each document not in English is included either here or in the specification.

This Information Disclosure Statement is submitted according to the following selected paragraph:

- I.  This Statement is being filed under §1.97(b) within three months of the filing date of the application (other than a CPA), or before the first Office action on the merits or before mailing of a first Office action after the filing of a request for continued examination.
- II.  This Statement is being filed under §1.97(c), with fee, **prior** to either a final action, a notice of allowance or an that otherwise closes prosecution in the application. Please charge the fee required by §1.17(p) to Eastman Kodak Company Deposit Order Account Number 05-0225. A duplicate copy of this Statement is enclosed.

III.  This Statement is being filed under §1.97(c), with a statement under, §1.97(e) **prior** to either a final action, a notice of allowance or an action that otherwise closes prosecution in the application. The undersigned hereby states that (check one):

each item of information contained in this Statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of this Statement.

no item of information in this Statement was cited in a communication from a foreign patent office in a counterpart foreign application and, to the knowledge of the person signing this certification statement under §1.97(e) after making reasonable inquiry, no item of information contained in this Statement was known to any individual designated in §1.56(c) more than three months prior to the filing of this Statement.

IV.  This Statement is being filed under §1.97(d), with petition and statement under §1.97(e), on or after the mailing date of either a final action, a notice of allowance (but prior to payment of the issue fee) or an action that otherwise closes prosecution in the application. The undersigned hereby petitions that this Statement be considered prior to issuance of the patent. Please charge the fee required by §1.17(p) to Eastman Kodak Company Deposit Order Account No. 05-0225. A duplicate copy of this Statement is enclosed. The undersigned hereby states that (check one):

each item of information in this Statement was cited in a communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of this Statement.

no item of information in this Statement was cited in a communication from a foreign patent office in a counterpart foreign application and, to the knowledge of the person signing this certification Statement under §1.97(e) after making reasonable inquiry, no item of information contained in this Statement was known to any individual designated in §1.56(c) more than three months prior to the filing of this Statement.

Respectfully submitted,



Attorney for Applicants  
Registration No. 22,049

William F. Noval/law  
Telephone: (716) 477-5272  
Facsimile: (716) 477-4646  
Enclosures

FORM PTO-1449      US DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE		Atty. Docket No. <b>81595WFN</b> Customer No. 01333	Serial No.			
If AFTER the later date of the first Office Action or 3 months from filing, use only with Rule 97(E) Certificate or Fee		Applicant: <b>Jiebo Luo</b>				
LIST OF ART CITED BY APPLICANT (Use several sheets if necessary)		Filing Date <b>December 14, 2000</b>	Group			
<b>U.S. PATENT DOCUMENTS</b>						
Examiner Initial*	DOCUMENT NUMBER	DATE	NAME	CLASS	SUBCLASS	FILING DATE IF APPROPRIATE
<i>AWC</i>	4,809,064	Feb. 28, 1989	<i>Amos, et al</i>	-	-	Nov. 19, 1987
<i>AWC</i>	5,872,619	Feb. 16, 1999	<i>Stephenson, et al.</i>	-	-	Sep. 19, 1996
<i>AWC</i>	5,872,643	Feb. 16, 1999	<i>Maeda, et al.</i>	-	-	Apr. 13, 1995
<i>AWC</i>	5,978,519	Nov. 2, 1999	<i>Bollman et al.</i>	-	-	Aug. 6, 1996
<i>AWC</i>	5,995,201	Nov. 30, 1999	<i>Sakaguchi</i>	-	-	Jan. 13, 1998
<b>FOREIGN PATENT DOCUMENTS</b>						
Examiner Initial*	DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUBCLASS	TRANSLATION YES   NO
<b>OTHER ART (Including Author, Title, Date, Pertinent Pages, Etc.)</b>						
EXAMINER <i>Alan Cant</i>	DATE CONSIDERED <i>1/8/03</i>					
<small>* EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609; Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.</small>						

J0784 U.S. PTO  
 09/736825  
 12/14/00

Rev. Fe. 12/14/00



US005978519A

# United States Patent [19]

**Bollman et al.**

[11] **Patent Number:** 5,978,519

[45] **Date of Patent:** \*Nov. 2, 1999

[54] **AUTOMATIC IMAGE CROPPING**

[75] **Inventors:** James E. Bollman, Williamson, N.Y.;  
Ramana L. Rao, Los Alamos, N.Mex.;  
Dennis L. Venable, Marion; Reiner  
Eschbach, Webster, both of N.Y.

5,115,271	5/1992	Iagopian	355/74
5,363,209	11/1994	Eschbach et al.	358/445
5,450,502	9/1995	Eschbach et al.	382/169
5,485,568	1/1996	Venable et al.	395/155
5,608,544	3/1997	Yamanishi	358/453
5,640,468	6/1997	Hsu	382/190
5,666,503	9/1997	Campanelli et al.	345/356

[73] **Assignee:** Xerox Corporation, Stamford, Conn.

[\*] **Notice:** This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

*Primary Examiner*—Yon J. Couso  
*Attorney, Agent, or Firm*—Oliff & Berridge, PLC

[57] **ABSTRACT**

The present invention describes a method for automatic cropping of images containing regions where intensity levels are uniform and other regions where intensity levels vary considerably. An image to be automatically cropped is scaled down to a grid and divided into non-overlapping blocks. The mean and variance of intensity levels are calculated for each block. Based on the distribution of variances in the blocks, a threshold is selected for the variance. All blocks with a variance higher than this threshold variance are selected as regions of interest. The regions of interest are then cropped to a bounding rectangle.

[21] **Appl. No.:** 08/692,559

[22] **Filed:** Aug. 6, 1996

[51] **Int. Cl.<sup>6</sup>** ..... G06K 9/20

[52] **U.S. Cl.** ..... 382/282; 382/270

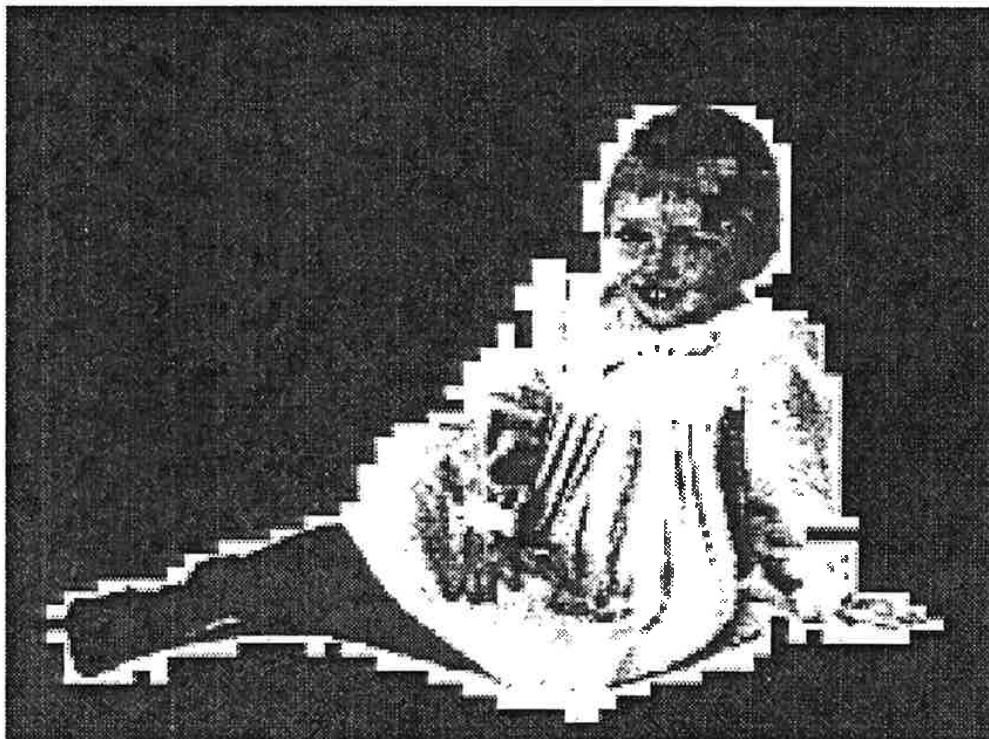
[58] **Field of Search** ..... 382/282, 283,  
382/171, 172, 270, 272; 358/538

[56] **References Cited**

**U.S. PATENT DOCUMENTS**

4,809,064 2/1989 Amos et al. .... 358/76

**21 Claims, 8 Drawing Sheets**



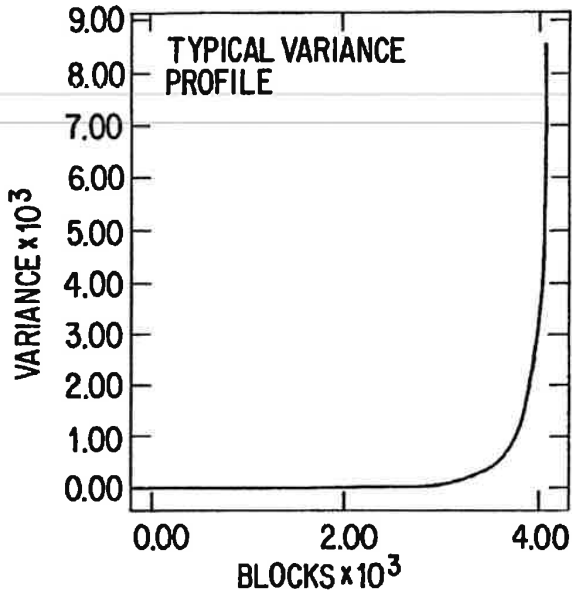


FIG. 1

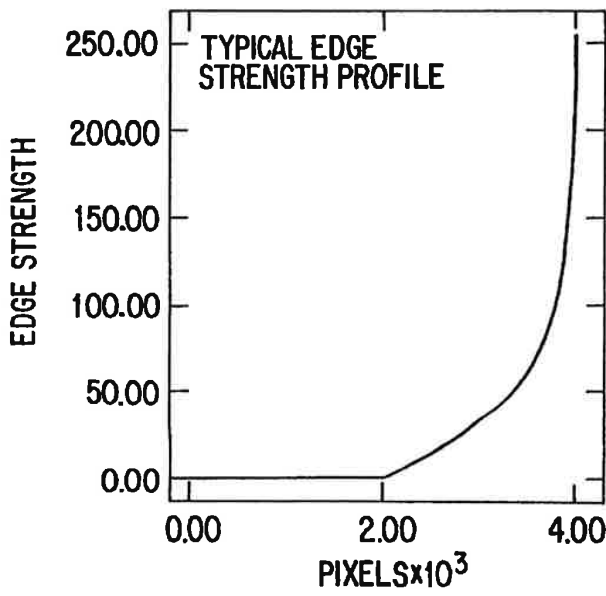


FIG. 2



FIG. 3A



FIG. 3B



FIG. 3C



FIG. 3D





FIG. 4A



FIG. 4B



FIG. 4C



FIG. 4D

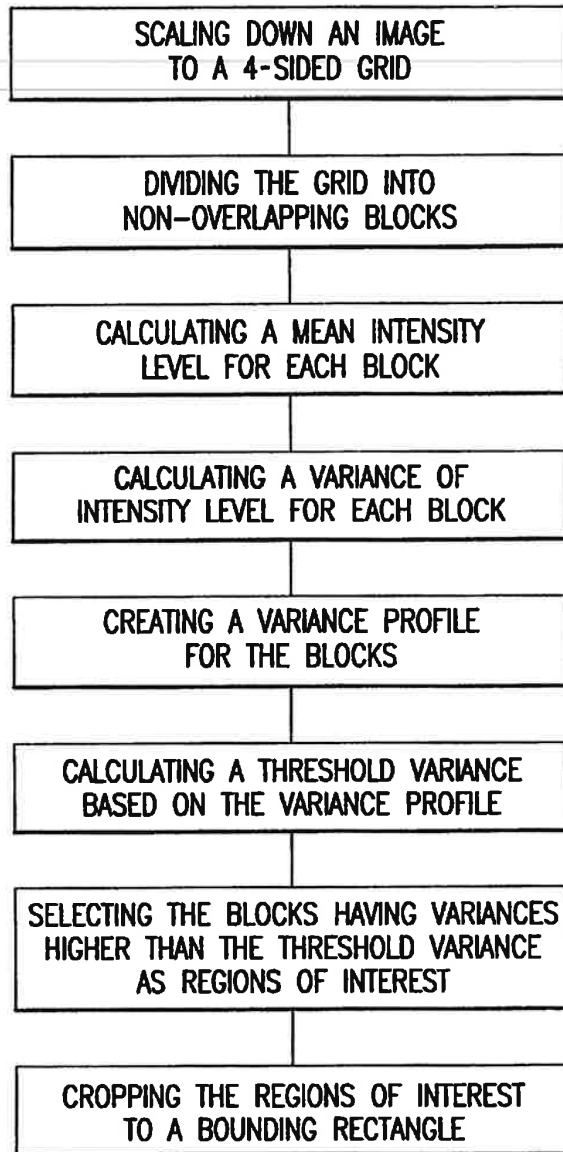


FIG.5

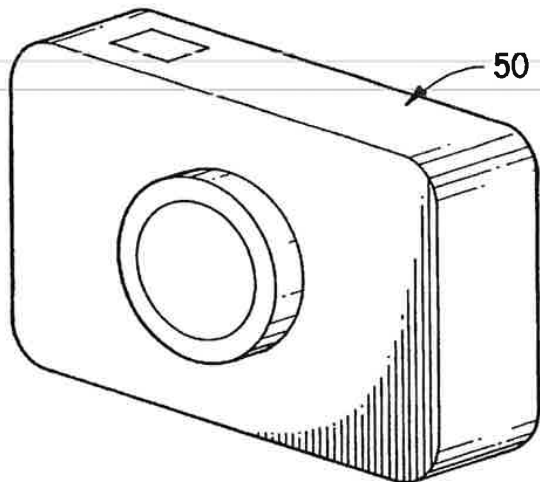


FIG. 6

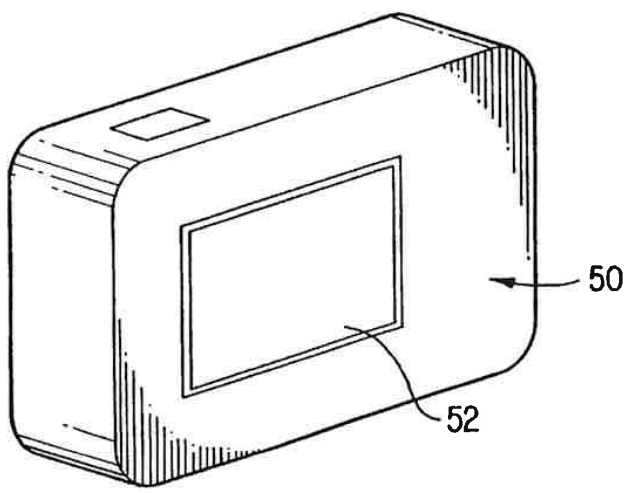


FIG. 7

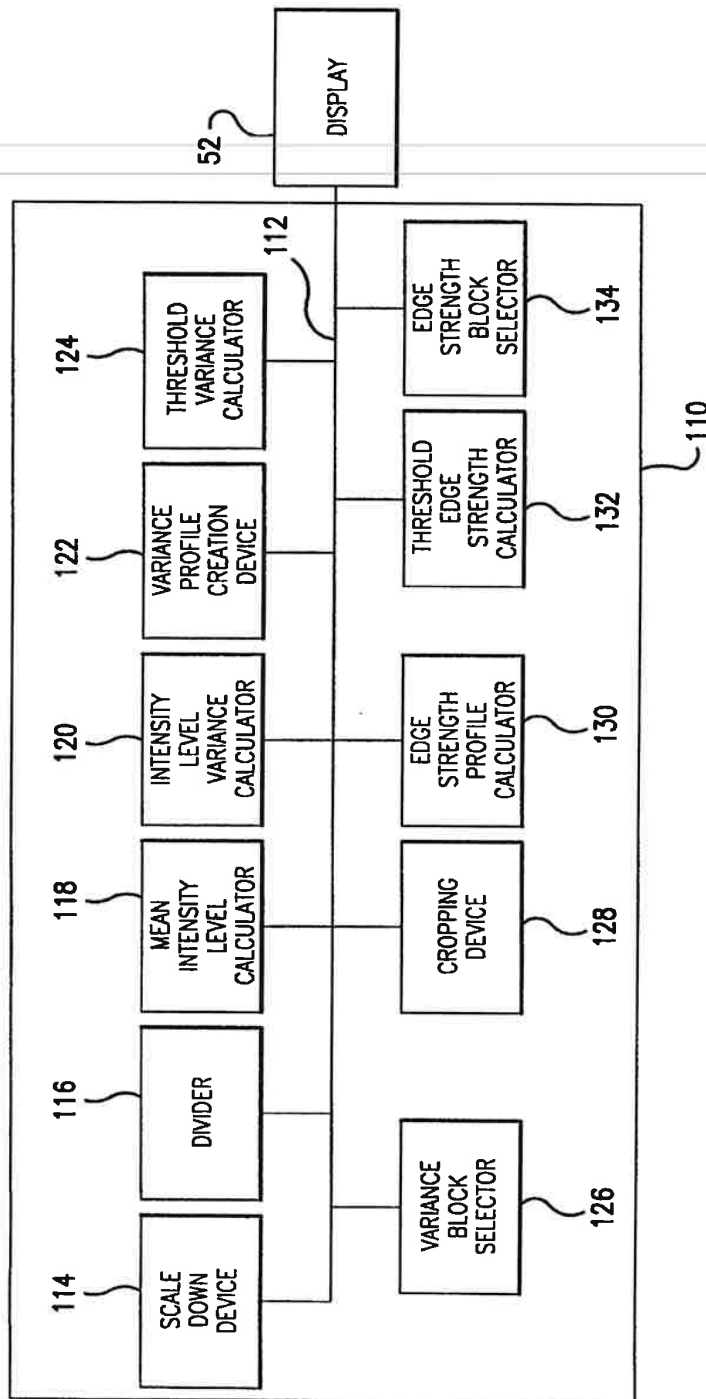


FIG.8

## AUTOMATIC IMAGE CROPPING

## BACKGROUND OF THE INVENTION

This invention is related to a method for the automatic cropping of images, and is particularly suitable to images that are texture-free, or relatively texture-free.

A typical image contains some regions where intensity level, and color, are uniform, and other regions where intensity level and color vary considerably. For instance, the "background" of an image may be uniform with a distinct "edge" separating the background from the "foreground." For example, a portrait typically comprises a subject set against a uniform backdrop or background, such that a sharp edge or boundary exists between the subject and the background.

Frequently, it is desirable to select only a particular region of an image, and to reproduce the selected region, thereby eliminating unwanted or excess background to give the image a more desirable composition. This selection process is referred to as cropping. Often, images are cropped to the foreground and most of the background is discarded.

Cropping is usually done by hand or requires operator interaction in order to properly select the subject and cropping dimensions. For example, U.S. Pat. No. 4,809,064 to Amos et al. discloses an apparatus for printing a selected portion of a photographic negative onto a photosensitive paper to form an enlarged and cropped photographic print. However, the apparatus requires human operation to determine the crop. Similarly, U.S. Pat. No. 5,115,271 to Hagopian discloses a variable photographic cropping device for maintaining multiple constant proportions of a visible area that includes a pair of masks situated in a housing having a central window. The apparatus also requires an operator.

In the field of automatic image enhancement, methods are known for improving the contrast in a natural scene image or altering the sharpness in a reproduction of an electronically encoded natural scene images. Such methods have been disclosed, for example, in U.S. Pat. Nos. 5,450,502 and 5,363,209 to Eschbach et al., the disclosures of which are incorporated herein by reference. However, such automatic image enhancement methods do not disclose automatic image cropping.

For high quality publication and printing, manual cropping may be preferred for artistic reasons. For large volume printing, including, but not limited to, passport photographs, yearbooks, catalogs, event books, portraits, other images with uniform backgrounds, and the like, it is desirable to have the option to use autocropping to enhance productivity and uniformity of the cropping process.

## SUMMARY OF INVENTION

The present invention relates to a method for the automatic cropping of images. Further, the present invention relates to a method for automatically cropping images that are texture-free, or relatively texture-free, to their regions of interest.

According to the present invention, an image to be automatically cropped is scaled down to a grid and divided into non-overlapping blocks. The mean and variance of intensity levels are calculated for each block. Based on the distribution of variances in the blocks, a threshold is selected for the variance. All blocks with a variance higher than the threshold variance are selected as regions of interest. The regions of interest are cropped to a bounding rectangle to provide an autocropped image with a tight fit. The

autocropped image may then be subjected to a post-processing image operation including, but not limited to, scaling the autocropped image to a larger or smaller dimension, image enhancement, annotating, transmitting, halftoning, and the like.

The present invention may optionally include an edge strength distribution analysis of an image. A threshold is chosen from a sorted list of edge strengths in order to select any block that contains a significant number of edge pixels and was not selected after the intensity variance analysis.

## BRIEF DESCRIPTION OF DRAWINGS

The file of this patent contains at least one drawing executed in color. Copies of this patent with color drawing(s) will be provided by the Patent and Trademark Office upon request and payment of the necessary fee.

FIG. 1 shows a typical intensity variance profile of an image.

FIG. 2 shows a typical edge strength profile of an image.

FIG. 3(a) shows a picture to be autocropped using the method of the present invention.

FIG. 3(b) shows the image after blocks with a luminance variance higher than a threshold variance are selected.

FIG. 3(c) shows the image after a post-processing cleanup pass is conducted.

FIG. 3(d) shows the autocropped image with a border scaled to the same horizontal dimension as the original picture.

FIG. 4(a) shows a second picture to be autocropped using the method of the present invention.

FIG. 4(b) shows the image after blocks with a luminance variance higher than a threshold variance are selected.

FIG. 4(c) shows the image after a post-processing cleanup pass is conducted.

FIG. 4(d) shows the autocropped image with a border scaled to the same vertical dimension as the original picture.

FIG. 5 is a flowchart depicting the steps of practicing the invention;

FIG. 6 is a front perspective view of an exemplary camera that practices the steps of the invention;

FIG. 7 is a rear perspective view of the exemplary camera shown in FIG. 6; and

FIG. 8 is a schematic diagram of an apparatus of the invention for automatically cropping an image.

## DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

The present invention relates to a method of automatic image cropping of images and the steps of practicing the invention are depicted in FIG. 5. Preferably, the present invention in embodiments relates to automatic image cropping of images that are texture-free or relatively texture-free.

The automatic image cropping method of the present invention is independent of the image input or acquisition method. Any image acquisition device that converts a picture into electronic or digital data, such as a computer data file, is acceptable for using the method of the present invention. Acquisition devices for images to be automatically cropped include, but are not limited to, a digital scan-to-print system, a digital camera 50 as shown in FIGS. 6 and 7, a digital scanner, a photo CD, or laser disc, or the like.

The images themselves are defined in terms of pixels, wherein each pixel is an electrical or electronic signal with

a digital gray value that varies between a white level and a black level. In a currently desirable system, in which calculations may be done on 8 bits of information or more, 256 levels of gray will be available for use. Pixels are also identified in terms of position. A pixel may define a unique location (m,n) within an image, identified by its m<sup>th</sup> pixel position in a line (or column), and its n<sup>th</sup> line position in a page (or row). Color is therefore represented by those gray values for red, blue and green. For example, in the RGB color space, a single color pixel is represented by three values, one for red, one for blue, and one for green. The color of the particular pixel can be defined as a combination of the red, blue and green color levels.

The automatic image cropping method of the present invention is also independent of the output method. The output methods for an image autocropped according to the present invention include, but are not limited to, a laser printer, ink jet ink printer, LCD display 52 on the digital camera 50 shown in FIG. 7, CRT display, magnetic tape or other media, dye sublimation printer, a photographic printer, or the like. These output devices may have many characteristics. However, they have as a common requirement the representation of gray or color pictorial images.

According to an embodiment of the present invention, images are initially defined in terms of the red, green, blue ("RGB") color space. Preferably, images defined in the RGB color space are directed to a color space converter and converted to a luminance color space. In embodiments, however, it is possible that the image will already be in luminance color space, as it is common to convert RGB values to luminance/chrominance space for other image processing. Whatever space is used, it must have a component that relates to the human visual perception of lightness or darkness.

In embodiments, the initial image data in RGB color space may be converted to a luminance color space using the luminance channel Y in Xerox YES color space of the "Xerox Color Encoding Standard," XNSS 289005, 1989. The RGB color space may also be converted to a luminance color space using the luminance channel Y in the known television encoding standard, YIQ. In an embodiment of the present invention, the luminance channel Y is calculated according to the following general formula:

$$Y=0.252939 \times (\text{Red Channel}) + 0.684458 \times (\text{Green Channel}) + 0.062603 \times (\text{Blue Channel}).$$

All statistical analysis and color conversion is preferably carried out on a scaled down version of the input image. This speeds up the automatic image cropping process and also provides a certain degree of robustness against noise.

According to embodiments of the present invention, an image to be autocropped is scaled down to a regular grid. The grid is then divided into non-overlapping square blocks, N×N, of a smaller size, including, but not limited to, 4×4, 8×8, 16×16, 64×64 pixels, or the like. The height and width of each block is indicated by N pixels. In an embodiment of the invention, the grid size is 256×256 pixels, and the block size is 4×4 pixels. However, different grid and block sizes can be used so long as the objectives of the present invention are achieved.

According to the present invention, the cue for selecting regions of interest in an image is the luminance profile of the intensity levels in the blocks. Alternatively, the RGB profile of the intensity levels in the blocks can be used. The mean and variance of the intensity level are calculated for each block consisting of N×N pixels.

The mean intensity level,  $\mu$ , in each block is calculated according to the following formula (1):

$$\mu = \frac{1}{N^2} \sum_{i=1}^{N^2} g_i \quad (1)$$

wherein  $g_i$  is the intensity level of the i<sup>th</sup> pixel in the block, and N is the size of each block in pixels. The variance,  $\sigma$ , in each block is calculated according to the following formula (2):

$$\sigma = \frac{1}{N^2 - 1} \sum_{i=1}^{N^2} (g_i - \mu)^2 \quad (2)$$

FIG. 1 shows the typical intensity variance profile of an image. From FIG. 1, it is apparent that most blocks in an image exhibit very low variances.

Based on statistical analysis and the distribution of the variances in the blocks, a threshold variance is selected. From FIG. 1, a generally optimal threshold variance is picked at the "knee" of the variance profile curve. According to the present invention, the threshold variance is preferably picked as a point on the curve furthest from a line joining the minimum and the maximum variance. All blocks with a variance higher than this threshold variance are selected as regions of interest (i.e., elements of the foreground) to remain in the autocropped image. All blocks with a variance less than the threshold are considered to be uninteresting (i.e., elements of the background) and are removed from the autocropped image. The threshold variance may also be adjusted higher or lower to include or exclude more blocks. For example, the threshold variance value may be reduced by an empirically determined selectivity factor to include more objects of interest. In an embodiment of the present invention, the threshold variance is reduced by about forty percent to include more blocks as regions of interest.

The blocks selected as regions of interest are then cropped to a bounding rectangle by finding the first selected block along the four sides of the grid, thereby giving a tight fit. All blocks within the bounding rectangle are included in the autocropped image. The tightness of the crop is application dependent and is fully adjustable.

The cropped image may be scaled to a larger (or smaller) dimension and a border selected for the scaled autocropped image. In an embodiment of the present invention, the automatic cropping of an image is set to a default border of about 0.01 (i.e., a 1% border) of the larger dimension.

An optional cleanup post-processing pass may be carried out to mark unselected blocks that are inside selected regions (i.e., typically the "interior" of an object) for further post-processing image operations. An embodiment of the present invention uses a seed fill algorithm to accomplish this purpose. Various seed fill algorithms are known in the art including, but not limited to, that recited in Paul S. Heckbert, *A Seed Fill Algorithm*, Graphics Gems, 1990, incorporated herein by reference. Selected regions smaller than the specified parameter for the smallest foreground image effect that is to be retained as interesting are eliminated. In an embodiment of the present invention, small details corresponding to "noise" in the background are examined, in blocks and pixels. These details are removed from the autocropped image. As a result, small glitches and spots are eliminated, thereby providing a better bounding rectangle, especially at the edges of the autocropped image.

After the intensity variance analysis, the image optionally may be further analyzed using edge strength information as

an additional cue. Some images display low contrast edges at the boundaries of the foreground and the background. Although the results of the intensity variance analysis described herein are satisfactory, results for some images may be improved if an edge strength distribution is analyzed.

Similar to the variance analysis, the edge strength distribution of the blocks is analyzed for a suitable threshold. A typical edge strength plot for an image is shown in FIG. 2. In embodiments, the edge strength computation may be carried out using a digital Laplacian operator, such as that recited in Rafael C. Gonzalez & Richard E. Woods, *Digital Image Processing*, 100, 420, 453 (1992), the disclosure of which is herein incorporated by reference. Using the Laplacian operator, a threshold is chosen from the edge strength distribution. Any blocks with significant edge information (i.e., a specified number of pixels greater than the threshold) and that are not selected as regions of interest from the variance analysis are marked as "interesting."

According to the present invention, automatic image cropping relies on several empirically determined parameters for its performance. The following parameters described below include, but are not limited to, those parameters that can be tuned to customize automatic image cropping to a particular image set being analyzed. One skilled in the art may alter any or all of these parameters based on the a priori information available about the images to be autocropped.

Grid Size is the size of the square grid on which the scaled down version of the input image is sampled. Increasing the size of this grid makes the program more sensitive to noise. Image effects that would be too small to be significant on a grid of size 256x256, for instance, may be significant on larger grids. The time of computation also increases with the increased grid size. For example, with a 512x512 sampling grid, the time is approximately four times that with a 256x256 grid if all other parameters are unchanged.

Block Size is the size of the local neighborhood over which variance analysis is carried out. Block Size corresponds to the height and width of the non-overlapping blocks, and is typically measured in pixels. A large block size results in a coarse analysis of the image, while a small block size results in a finer analysis. The block size controls the size of the local neighborhood that is included with an edge between the background and the foreground. The margin around the foreground is larger with a large block size.

Black Object Size is the size of the smallest background image effect that is retained as an uninteresting effect. This size is measured in blocks. All uninteresting regions whose size is less than the selected Block Size are marked as interesting regions. Typically these are completely covered by interesting regions implying that they are the interior of foreground effects.

White Object Size is the size of the smallest foreground image effect that is retained as an interesting effect. The default sizes for foreground and background effects are generally not identical, but can be independently selected. Increasing the value of this parameter has the effect of eliminating larger and larger connected regions of interest, and decreasing it has the opposite effect.

Additional parameters for an edge strength distribution analysis include, but are not limited to:

Filter Coefficients contains the filter coefficients to be used in the optional edge detection part of the automatic

image cropping. Filter coefficients are weights used in a sliding window (including, but not limited to, an odd number of rows and columns) used to compute the weighted average of the gray-scale values in the neighborhood of each pixel in an image. This operation can be accomplished through any edge detection filter and is not constrained to be Laplacian-like. This operation is called digital convolution and is known in the art of signal and image processing, for example in Rafael C. Gonzalez & Richard E. Woods, *Digital Image Processing*, (1992) and in J. Canny, *IEEE Transactions in Pattern Analysis and Machine Intelligence*, Vol. 8, pp. 697-98 (1986), incorporated herein by reference.

Edge Pixels is the number of pixels within a block that must have an edge strength greater than the threshold for the block to be selected as an interesting region. This parameter must not be disproportionately large when compared to the block size. Too small a value for this parameter makes it overly sensitive to noise. Conversely, the edge analysis phase doesn't add anything useful if the parameter has too high a value.

#### EXAMPLES

Two images autocropped according to the method of present invention, i.e., automatic image cropping, appear in FIGS. 3 and 4.

The parameters used for autocropping these images based on luminance intensity data are a Grid Size of 256x256 pixels; a Block Size of 4x4 pixels; a Black Object Size of 15 blocks; and a White Object Size of 4 blocks.

Example I:

FIG. 3 shows an image that has been subject to the automatic image cropping method of the present invention. FIG. 3(a) shows the original picture to be autocropped. FIG. 3(b) shows the picture after it is subsampled into a grid (256x256 pixels), divided into non-overlapping blocks (64x64 blocks of 4x4 pixels per block), and scaled to match the original. All blocks with a luminance variance higher than a threshold variance are selected as regions of interest. FIG. 3(c) shows the autocropped image in which a post-processing pass is made to toggle groups of blocks in large areas of the opposite type of blocks and to eliminate noise in the background. Also shown is a bounding box (i.e., red rectangle) that is calculated to give a tight fit to the autocropped image. FIG. 3(d) shows the autocropped image, the image within the bounding box of FIG. 3(c), scaled to the same horizontal dimension as the original and with a 5% border.

Example II:

FIG. 4 shows a second image that has been subject to the automatic image cropping method of the present invention. FIG. 4(a) shows the original picture to be autocropped. FIG. 4(b) shows the picture after it is subsampled into a grid (256x256 pixels), divided into non-overlapping blocks (64x64 of 4x4 pixels per block), and scaled to match the original. All blocks with a luminance variance higher than a threshold variance are selected as regions of interest. FIG. 4(c) shows the autocropped image in which a post-processing pass is made to toggle groups of blocks in large areas of the opposite type of blocks and to eliminate noise in the background. Also shown is a bounding box (i.e., red rectangle) that is calculated to give an autocropped image with a tight fit. FIG. 4(d) shows the autocropped image, the image within the bonding box of FIG. 4(c), scaled to the same vertical dimension as the original with the side and bottom borders corresponding to the original picture and with a 5% border on the top.



Thus, automatic image cropping according to the present invention handles most texture-free, or relatively texture-free, images in a predictable and productive fashion.

It will no doubt be appreciated that the present invention can be accomplished through application software accomplishing the functions described, to operate a digital computer or microprocessor, through a hardware circuit.

An apparatus 110 of the invention automatically crops an image is shown in FIG. 8. The apparatus 110 is connected to the display 52 by a bus 112. The apparatus includes a scale down device 114, a divider 116, a mean intensity level calculator 118, an intensity level variance calculator 120, a variance profile creation device 122, a threshold variance calculator 124, a variance block selector 126, a cropping device 128, an edge strength profile calculator 130, a threshold edge strength calculator 132 and an edge strength block selector 134. These components are interconnected by the bus 112.

The scale down device 114 scales down the image to a grid having four sides. The divider 116 divides the grid into a plurality of non-overlapping blocks. The mean intensity level calculator 118 calculates a mean intensity level for each of the blocks. The intensity level variance calculator 120 calculates a variance of an intensity level for each of the blocks. The variance profile creation device 122 creates a variance profile for the blocks. The threshold variance calculator 124 calculates a threshold variance based on the variance profile.

The variance block selector 126 selects the blocks having the variance higher than the threshold variance as regions of interest. The cropping device 128 crops the regions of interest to a bounding rectangle. The edge strength profile calculator 130 calculates a profile of edge strengths for the blocks. The threshold edge strength calculator 132 calculates a threshold edge strength from the profile. The edge strength block selector 134 selects the blocks that have an edge strength higher than the threshold edge strength and not selected as regions of interest.

The invention has been described with references to particular embodiments. Modifications and alterations will be apparent to those skilled in the art upon reading and understanding this specification. It is intended that all such modifications and alterations are included insofar as they come within the scope of the appended claims.

What is claimed is:

1. A method for automatically cropping an image, comprising:
  - scaling down said image to a grid having four sides;
  - dividing said grid into a plurality of non-overlapping blocks;
  - calculating a mean intensity level for each of said blocks;
  - calculating a variance of an intensity level for each of said blocks;
  - creating a variance profile for said blocks;
  - calculating a threshold variance based on said variance profile;
  - selecting said blocks having said variance higher than said threshold variance as regions of interest;
  - cropping said regions of interest to a bounding rectangle;
  - calculating a profile of edge strengths for said blocks;
  - calculating a threshold edge strength from said profile;
  - and

selecting said blocks having an edge strength higher than said threshold edge strength and not selected as regions of interest.

2. The automatic image cropping method of claim 1, wherein said intensity level is a luminance intensity.
3. The automatic image cropping method of claim 1, wherein said intensity level is a red, green, blue intensity.
4. The automatic image cropping method of claim 1, wherein said mean is calculated by the formula

$$\mu = \frac{1}{N^2} \sum_{i=1}^{N^2} g_i$$

wherein  $g_i$  is the intensity level of the  $i^{\text{th}}$  pixel in the block, and  $N$  is a height and a width of each block in pixels.

5. The automatic image cropping method of claim 4, wherein said variance is calculated by the formula

$$\sigma = \frac{1}{N^2 - 1} \sum_{i=1}^{N^2} (g_i - \mu)^2$$

6. The automatic image cropping method of claim 1, wherein said threshold variance is selected as a point on a curved portion of a variance profile.

7. The automatic image cropping method of claim 6, wherein said threshold variance is adjusted by a selectivity factor.

8. The automatic image cropping method of claim 7, wherein said threshold variance is reduced to select more regions of interest.

9. The automatic image cropping method of claim 7, wherein said selectivity factor is about forty percent.

10. The automatic image cropping method of claim 1, wherein said grid is 256 by 256 pixels.

11. The automatic image cropping method of claim 1, wherein said non-overlapping blocks are 4 by 4 pixels.

12. The automatic image cropping method of claim 1, wherein said bounding rectangle is defined by a first selected block along each of the four sides of the grid.

13. The automatic image cropping method of claim 1, further comprising subjecting an autocropped image to a post-processing image operation.

14. The automatic image cropping method of claim 1, further comprising scaling said bounding rectangle to a larger or smaller dimension having a border.

15. The automatic image cropping method of claim 14, wherein said border is a default border of about 1% of said dimension.

16. The automatic image cropping method of claim 1, further comprising marking unselected blocks inside selected regions for further post-processing image operations.

17. The automatic image cropping method of claim 1, further comprising removing details corresponding to noise in the background of the cropped image to provide a better bounding rectangle.

18. An apparatus for automatically cropping an image, comprising:

- means for scaling down said image to a grid having four sides;
- means for dividing said grid into a plurality of non-overlapping blocks;
- means for calculating a mean intensity level for each said blocks;

9

means for calculating a variance of an intensity level for each of said blocks;  
means for creating a variance profile for said blocks;  
means for calculating a threshold variance based on said variance profile;  
means for selecting said blocks having said variance higher than said threshold variance as regions of interest;  
means for cropping said regions of interest to a bounding rectangle;  
means for calculating a profile of edge strengths for said blocks;  
means for calculating a threshold edge strength from said profile; and

10

means for selecting said blocks having an edge strength higher than said threshold edge strength and not selected as regions of interest.  
19. The apparatus of claim 18, further comprising an input means for acquiring said image and an output means for storing said autocropped image.  
20. The apparatus of claim 19, wherein said input means is selected from the group consisting of a digital scanner and a digital camera.  
21. The apparatus of claim 20, wherein said output means is selected from the group consisting of a printer, a LCD display, a CRT display, a magnetic media, and a photographic printer.

\* \* \* \* \*



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER OF PATENTS AND TRADEMARKS  
Washington, D.C. 20231  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/736,825	12/14/2000	Jiebo Luo	81595WFN	8670

7590 01/14/2003

Patent Legal Staff  
Eastman Kodak Company  
343 State Street  
Rochester, NY 14650-2201

EXAMINER

CARTER, AARON W

ART UNIT PAPER NUMBER

2625

DATE MAILED: 01/14/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

9

<b>Office Action Summary</b>	<b>Application No.</b>	<b>Applicant(s)</b>	
	09/736,825	LUO, JIEBO	
	<b>Examiner</b>	<b>Art Unit</b>	
	Aaron W Carter	2625	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1)  Responsive to communication(s) filed on 16 March 2001.
- 2a)  This action is **FINAL**.                      2b)  This action is non-final.
- 3)  Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4)  Claim(s) 1-28 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5)  Claim(s) \_\_\_\_\_ is/are allowed.
- 6)  Claim(s) 1,3,8,12-15,17,22 and 26-28 is/are rejected.
- 7)  Claim(s) 2,4-7,9-11,16,18-21 and 23-25 is/are objected to.
- 8)  Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9)  The specification is objected to by the Examiner.
- 10)  The drawing(s) filed on \_\_\_\_\_ is/are: a)  accepted or b)  objected to by the Examiner.  
    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- 11)  The proposed drawing correction filed on 16 March 2001 is: a)  approved b)  disapproved by the Examiner.  
    If approved, corrected drawings are required in reply to this Office action.
- 12)  The oath or declaration is objected to by the Examiner.

**Priority under 35 U.S.C. §§ 119 and 120**

- 13)  Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).  
    a)  All    b)  Some \*    c)  None of:  
    1.  Certified copies of the priority documents have been received.  
    2.  Certified copies of the priority documents have been received in Application No. \_\_\_\_\_ .  
    3.  Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).  
    \* See the attached detailed Office action for a list of the certified copies not received.
- 14)  Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).  
    a)  The translation of the foreign language provisional application has been received.
- 15)  Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

**Attachment(s)**

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)                                  | 4) <input type="checkbox"/> Interview Summary (PTO-413) Paper No(s). _____  |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)                         | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449) Paper No(s) <u>4</u> . | 6) <input type="checkbox"/> Other:  |

## DETAILED ACTION

### *Claim Rejections - 35 USC § 112*

1. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

2. Claim 28 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

As to claim 28, this claim states the exact same limitations as claim 14. The examiner suggests changing the phrase "method of claim 1" to "method of claim 15." The claim will be treated as though that is what it says for the remainder of this office action.

### *Claim Rejections - 35 USC § 102*

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 1,3,8,12-15,17,22 and 26-28 are rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent 6,430,320 to Jia et al.

As to claim 1, Jia discloses a method of producing an image of at least a portion of a digital image (column 4, lines 27-28 and 42-44), the digital image comprising the steps of:

- a) providing a digital image having pixels (column 8, line 51):
- b) computing a belief map of the digital image, by using the pixels of the digital

---

image to determine a series of features (column 13, lines 25-35), and using such features to assign the probability (“crop statistics”, column 4, line 50) of the location of a main subject (column 4, lines 42-44) of the digital image in the belief map (Fig. 8, the belief map being indicated by the boundary pixels: 702, 710, 712, 714, 716, 718, and 720 located in both figures);

- c) determining a crop window having a shape and a zoom factor (Fig. 13, “Auto Crop to Requested Size”), the shape and zoom factor determining a size of the crop window (Fig. 14); and

- d) cropping the digital image to include a portion of the image of high subject content in response to the belief map and the crop window (column 4, 49-54).

As to claim 15, Jia discloses a method of producing an image of a portion of at least a portion of a photographic image onto a photographic receiver (column 4, lines 27-28 and 40-44), comprising the steps of:

- a) receiving a digital image corresponding to the photographic image, the digital image comprising pixels (column 8, line 51).

- b) computing a belief map of the digital image, by using the pixels of the digital image to determine a series of features (column 13, lines 25-35), and using such features to assign a probability (“crop statistics”, column 4, line 50, column 13, lines 38-39) of the location

of a main subject (column 4, lines 42-44) of the digital image in the belief map (Fig. 8, the belief map being indicated by the boundary pixels: 702, 710, 712, 714, 716, 718, and 720 located in both figures).

c) determining a crop window having a shape and a zoom factor (Fig. 13, "Auto Crop to Requested Size"), the shape and zoom factor determining a size of the crop window (Fig. 14); and=

d) locating the relative optical position of a photographic image, a lens assembly (column 5, lines 66-67), and a photographic receiver (column 5, lines 61-65) in response to the belief map and illuminating (column 5, lines 66-67) a portion of the photographic image of high subject content to produce an image of such portion onto the photographic receiver (column 4, 49-54).

As to claim 3 and 17, Jia discloses the method of claim 1 wherein step d) further comprises the steps of:

- i) selecting an initial position of the crop window at a location which includes the center of mass (Fig. 8);
- ii) using the belief values corresponding to the crop window to select the position of the crop window to include a portion of the image of high subject content in response to the belief map (Figs. 9B and 9C); and
- iii) cropping the digital image according to the position of the crop window (column 4, 49-54).

As to claim 8 and 22, Jia disclose the method of claim 1 wherein the crop window is completely within the digital image (Figs. 5-8).

As to claim 12 and 26, Jia discloses the method of claim 3 further comprising positioning said crop window such that said crop window includes all of said main subject cluster (Fig. 8).

As to claim 13 and 27, Jia discloses the method of claim 12 further comprising positioning said crop window to include a buffer around said main subject cluster (Fig. 8, the crop window includes buffering in the corners.

As to claim 14 and 28, Jia discloses a computer storage product having at least one computer storage medium having instructions stored therein causing one or more computers to perform the method of claim 1 (column 6, lines 33-49).

#### ***Allowable Subject Matter***

5. Claims 2,4-7,9-11,16,18-21 and 23-25 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

#### ***Double Patenting***

6. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed.



Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and, *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

7. Claim 1, 8, and 14 are rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claim 44-48 of copending Application No. 09/490915 ("App #2"). Although the conflicting claims are not identical, they are not patentably distinct from each other because they set forth subject matters which are obvious over each other and only differ in breadth of terminology used. For example, the limitation on lines 10-11 of claim 1, the phrase stating "cropping the digital image to include a portion of the image of high subject content in response to the belief map and the crop window." App #2 does not say this word for word but it is obvious that this is what is being claimed. On line 11-12 of claim 45 the statement "cropping the digital image to include main subjects indicated by the belief map to produce the cropped digital image by:" indicates that the following limitations in the claim are required for cropping the digital image. Therefore cropping the digital image to include a portion of the image of high subject content would be in response to the belief map, as stated in on lines 11-13, and in response to the crop window as stated on lines 9-10.

App #2 disclose a method of producing an image of at least a portion of a digital image comprising:

Digital image having pixels (claim 45, line 1).

Computing a belief map (claim 45, lines 3-5).

Determining a crop window having a shape and zoom (claim 45, lines 13-14)

Claim 8 corresponds to claim 46 of App #2.

As to claim 14, given the method, the computer storage product of claim 14 would have been at least obvious to one skilled in the art of image processing. In the image processing arts a reference, which anticipates or make obvious the invention of method claims will also at least make obvious the invention of computer storage product claims.

### *Conclusion*

8. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

U.S. Patent 6,282,317 to Luo discloses the inventor's method determining the main subject of a photographic image.

U.S. Patent 5,640,468 to Hsu discloses a method of identifying objects in an image.

U.S. Patent 6,434,271 to Christian et al. discloses method of locating objects in an image.

U.S. Patent 6,335,985 to Sambonsugi et al. discloses method of extracting an object from an image.

U.S. Patent 6,091,841 to Rogers et al. discloses a method that involves auto cropping and segmenting.

U.S. Patent 5,978,519 to Bollman et al. discloses a method of auto cropping an image.

U.S. Patent 6,456,732 to Kimbell et al. discloses automatic cropping and scaling of an image.

U.S. Patent 5,880,858 to Jin discloses an auto cropping method for use with scanners.

U.S. Patent 5,781,665 to Cullen et al. discloses a method for cropping an image.

***Contact Information***

---

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Aaron W. Carter whose telephone number is 703.306.4060. The examiner can normally be reached by telephone between 8am - 4:30pm (Mon. - Fri.).


If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Bhavesh Mehta can be reached on 703.308.5246. The fax phone number for the organization where the application or proceeding is assigned is 703.872.9314 for regular communications.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is 703.306.0377.

Aaron W. Carter  
Examiner  
Art Unit 2625

  
awc

January 9, 2003

  
**BHAVESH M. MEHTA**  
**SUPERVISORY PATENT EXAMINER**  
**TECHNOLOGY CENTER 2800**

**Notice of References Cited**

Application/Control No. 09/736,825	Applicant(s)/Patent Under Reexamination LUO, JIEBO	
Examiner Aaron W Carter	Art Unit 2625	Page 1 of 1

**U.S. PATENT DOCUMENTS**

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	Classification
	A	US-6,430,320	08-2002	Jia et al.	382/289
	B	US-6,434,271	08-2002	Christian et al.	382/194
	C	US-6,456,732	09-2002	Kimbell et al.	382/112
	D	US-6,282,317	08-2001	Luo et al.	382/203
	E	US-5,640,468	06-1997	Hsu, Shin-yi	382/190
	F	US-5,781,665	07-1998	Cullen et al.	382/254
	G	US-5,880,858	03-1999	Jin, Yuan-Chang	358/487
	H	US-6,335,985	01-2002	Sambonsugi et al.	382/190
	I	US-6,091,841	07-2000	Rogers et al.	382/132
	J	US-			
	K	US-			
	L	US-			
	M	US-			

**FOREIGN PATENT DOCUMENTS**

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

**NON-PATENT DOCUMENTS**

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

\*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)  
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.



US006430320B1

(12) **United States Patent**  
**Jia et al.**

(10) **Patent No.: US 6,430,320 B1**  
(45) **Date of Patent: Aug. 6, 2002**

(54) **IMAGE PROCESSING SYSTEM WITH  
AUTOMATIC IMAGE CROPPING AND  
SKEW CORRECTION**

(56) **References Cited**  
**U.S. PATENT DOCUMENTS**

(75) **Inventors:** Charles Chi Jia, San Diego;  
Anne-Marie Woodson, Lemon Grove;  
Cindy Sansom-Wai, San Diego; Daniel  
R Tretter, Palo Alto, all of CA (US)

5,058,185 A	*	10/1991	Morris et al.	382/199
5,093,653 A	*	3/1992	Ikehira	340/727
5,452,374 A	*	9/1995	Cullen et al.	382/293
5,781,666 A	*	7/1998	Ishizawa et al.	382/284
5,854,854 A	*	12/1998	Cullen et al.	382/176
5,940,544 A	*	8/1999	Nako	382/293
6,044,178 A	*	3/2000	Lin	382/260

(73) **Assignee:** Hewlett-Packard Company, Palo Alto,  
CA (US)

\* cited by examiner

(\*) **Notice:** Subject to any disclaimer, the term of this  
patent is extended or adjusted under 35  
U.S.C. 154(b) by 0 days.

*Primary Examiner*—Phuoc Tran  
*Assistant Examiner*—Amir Alavi

(21) **Appl. No.:** 09/546,110  
(22) **Filed:** Apr. 10, 2000

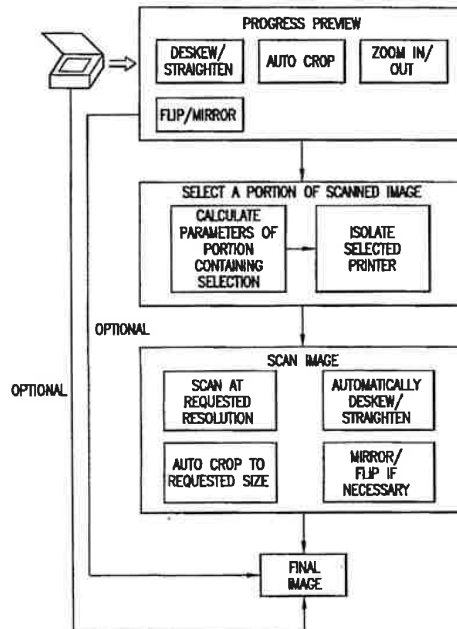
(57) **ABSTRACT**

**Related U.S. Application Data**

A system and method is described for automatically determining in a scanned document image the presence of unwanted extraneous information caused by an extraneous device and scanner background information. Once the presence of this information is determined, the system and method of the present invention can compute, for instance, skew and crop statistics. From this, the image can be automatically deskewed and cropped appropriately without the background and extraneous information. The system and method accomplishes this by first determining the presence of unwanted extraneous and background information and then appropriately processing the document image. The extraneous information is ignored during deskew and crop computations. Also, the scanner background and the extraneous information are prevented from being included in the final digital representation of the image.

- (63) Continuation of application No. 09/057,847, filed on Apr. 9, 1998.
- (51) **Int. Cl.<sup>7</sup>** ..... G06K 9/36
- (52) **U.S. Cl.** ..... 382/289; 382/299
- (58) **Field of Search** ..... 382/288, 289,  
382/290, 291, 292, 293, 294, 295, 199,  
282, 299, 300, 283, 205, 240, 190, 195,  
298, 275; 358/496, 497, 490, 486, 487,  
488, 474, 429

**12 Claims, 17 Drawing Sheets**



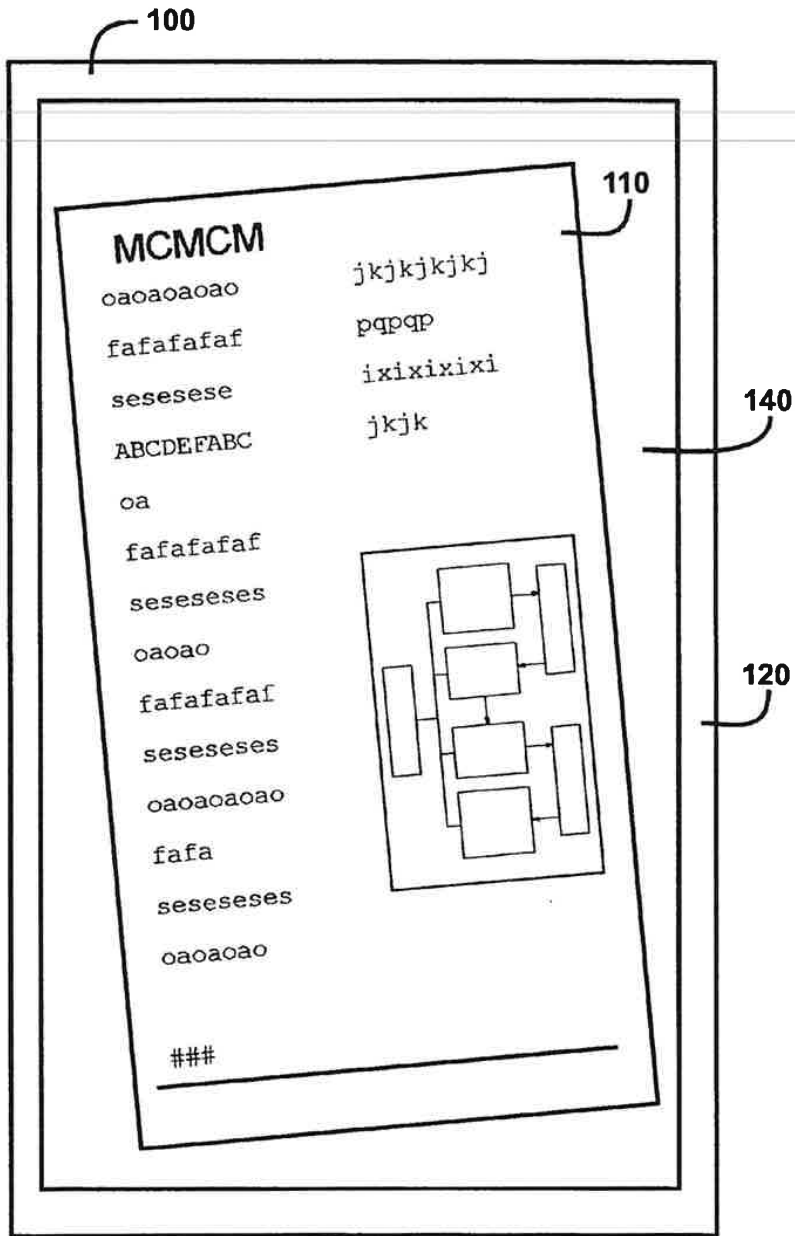


FIG. 1

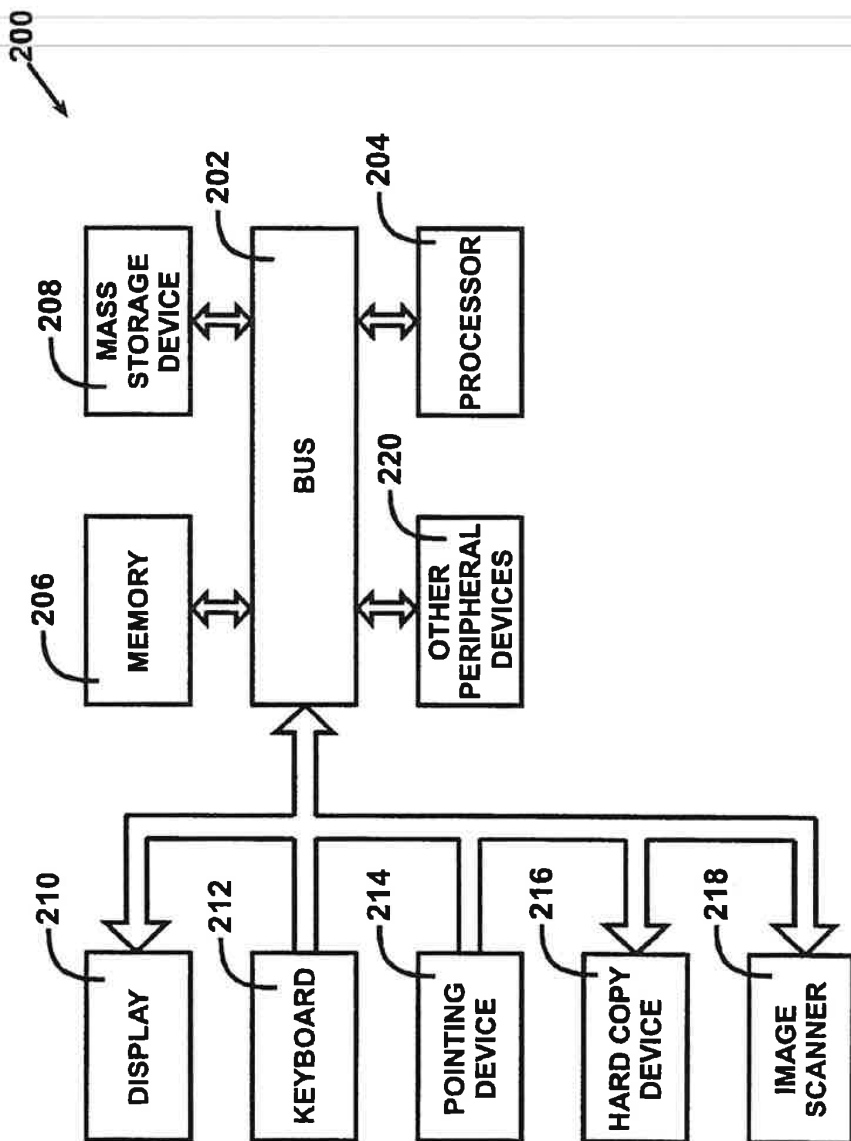


FIG. 2

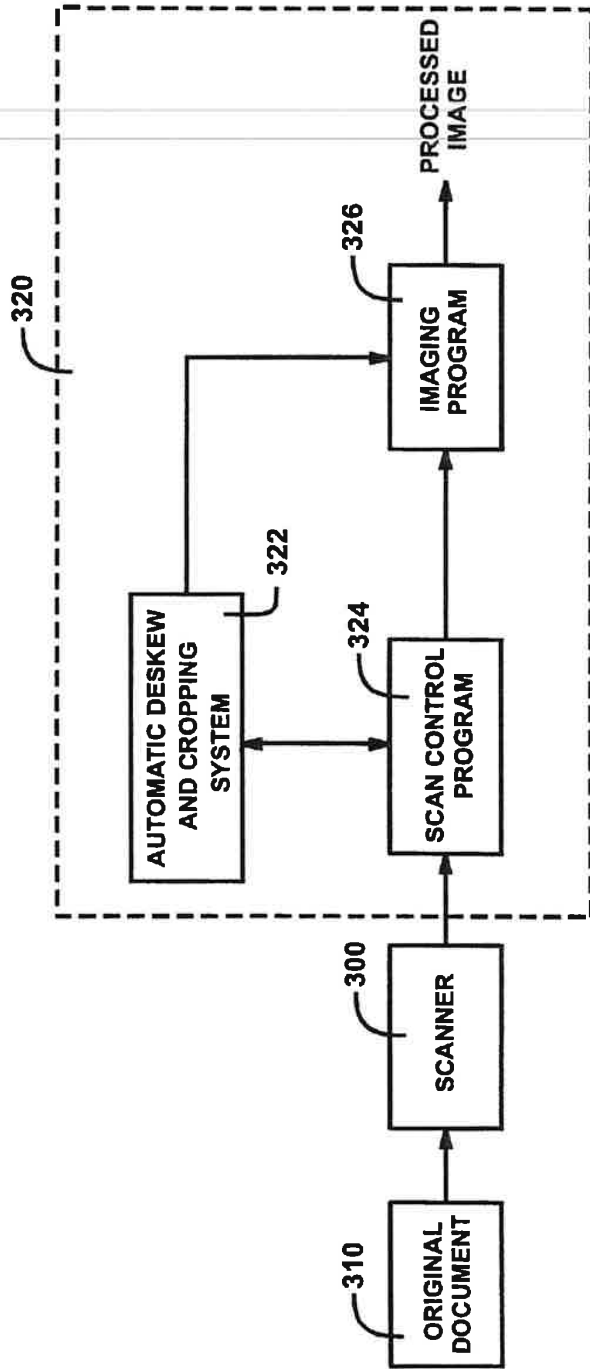


FIG. 3



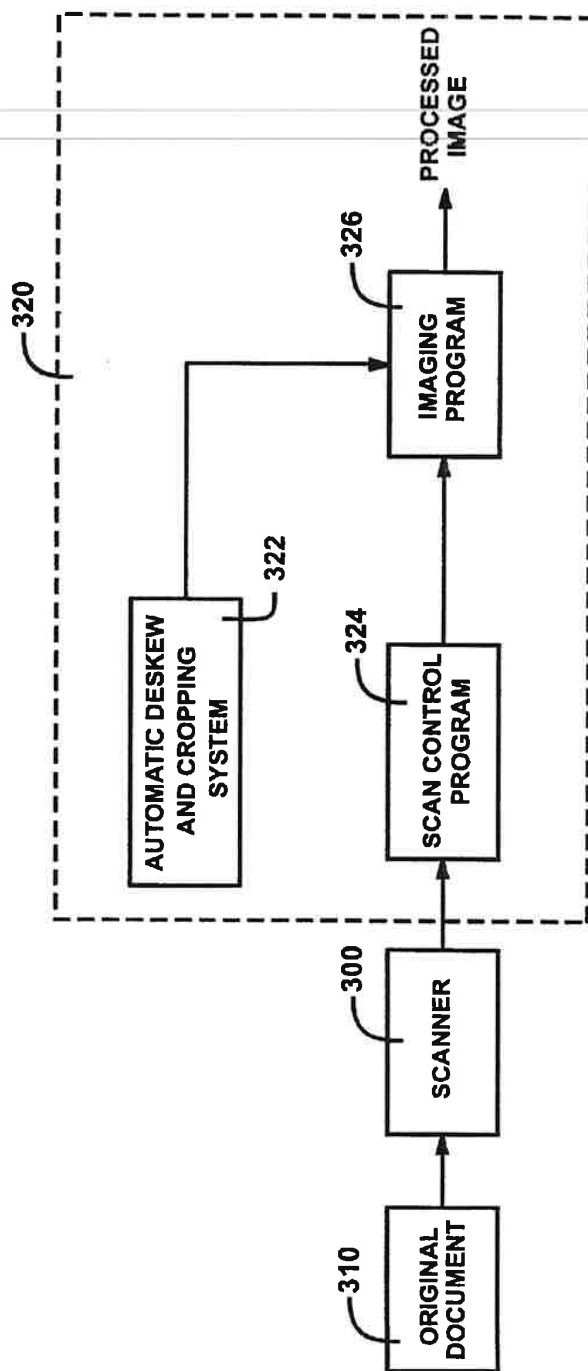


FIG. 4

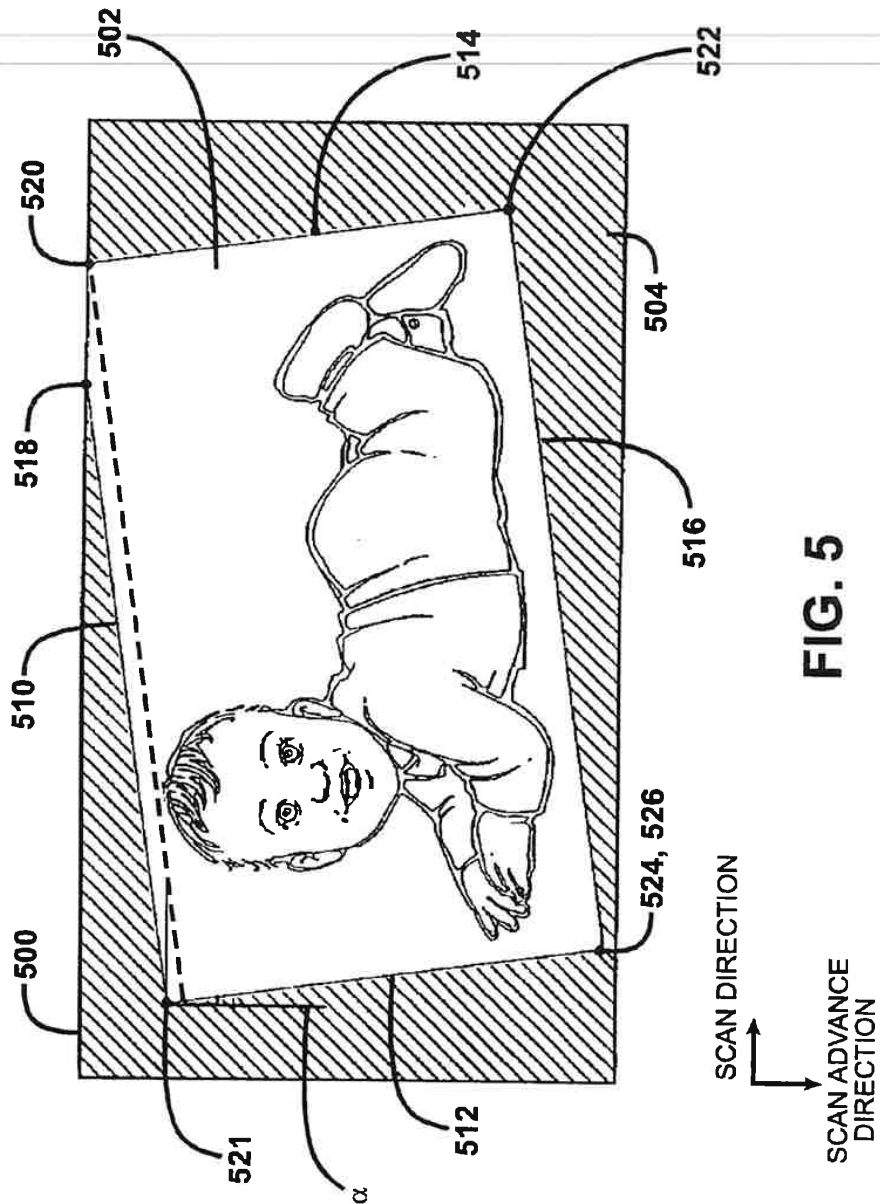
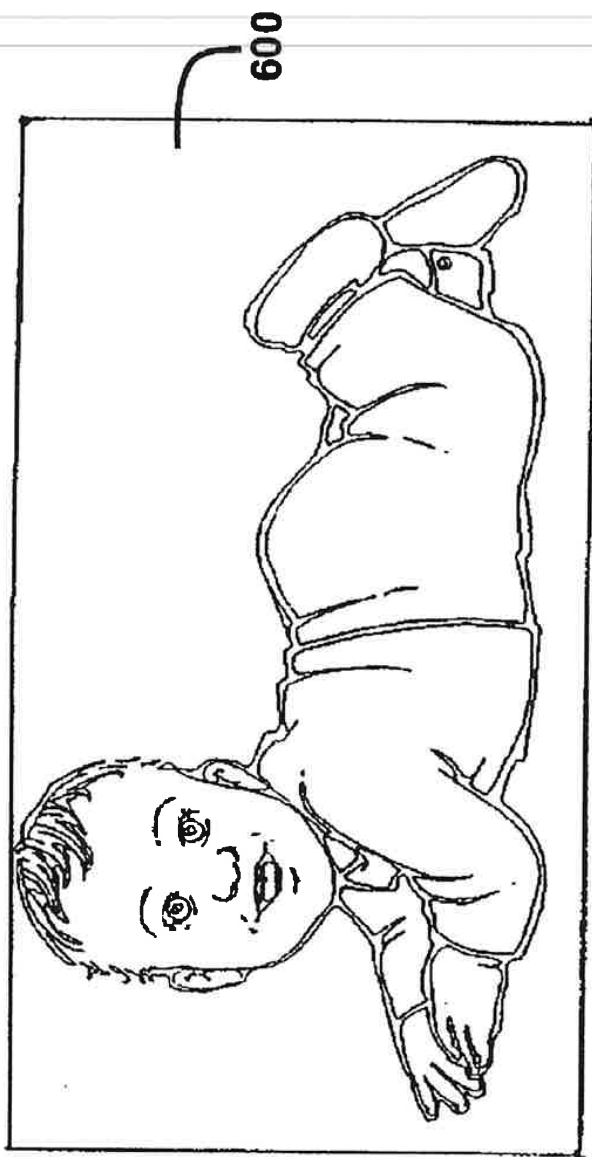


FIG. 5



**FIG. 6**

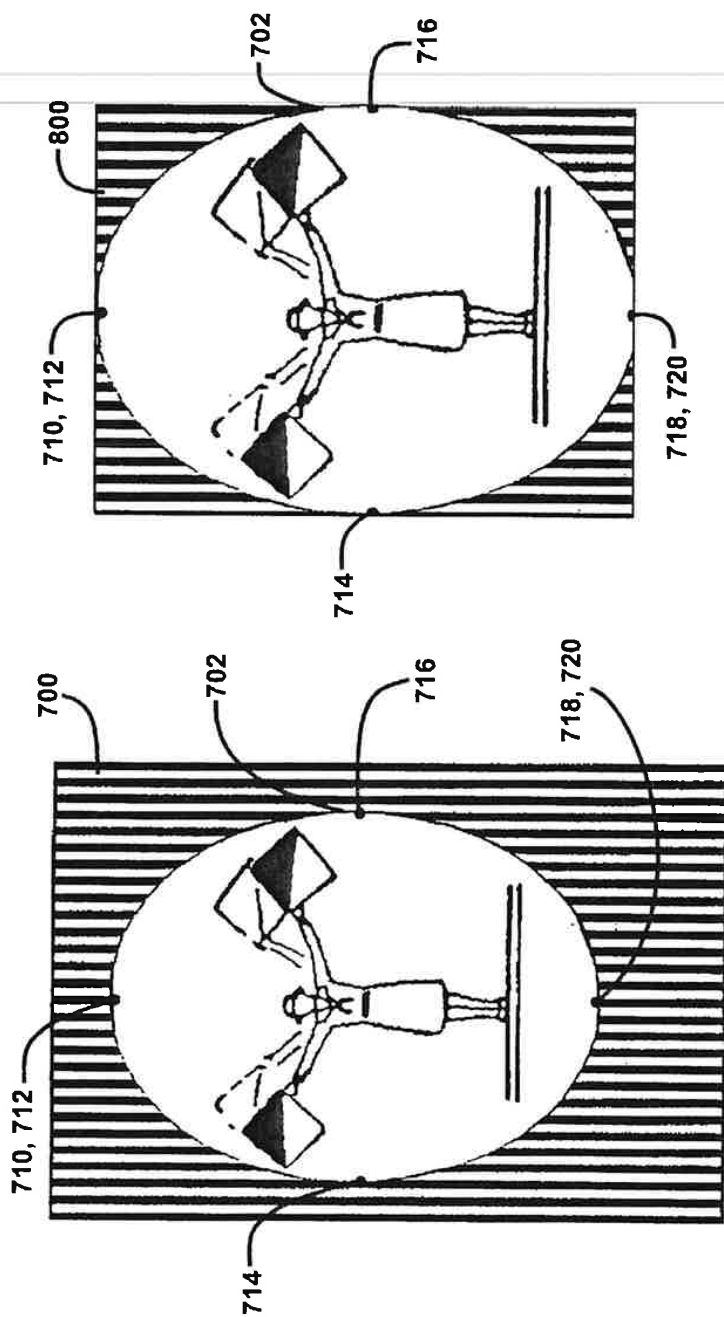


FIG. 8

FIG. 7

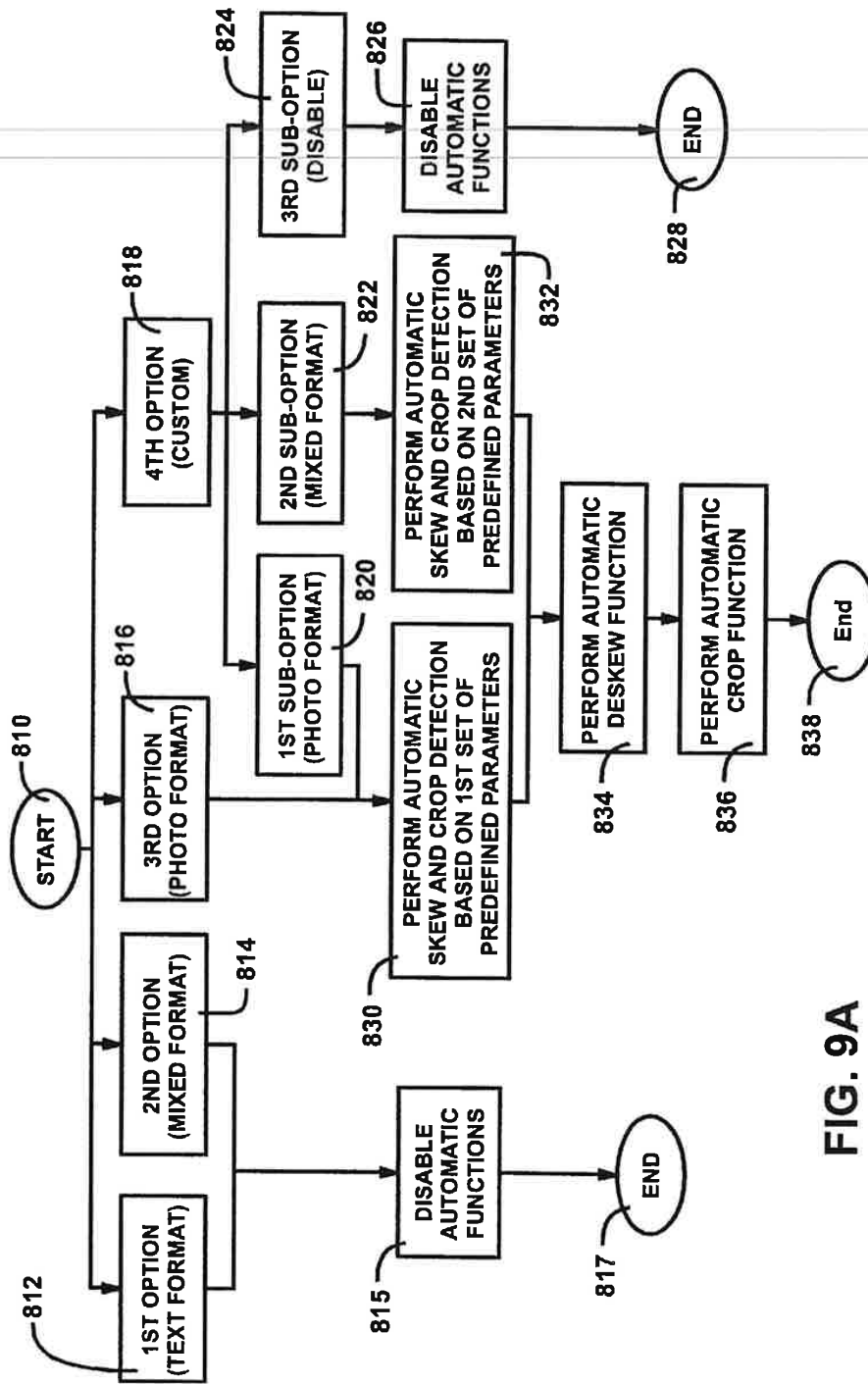


FIG. 9A

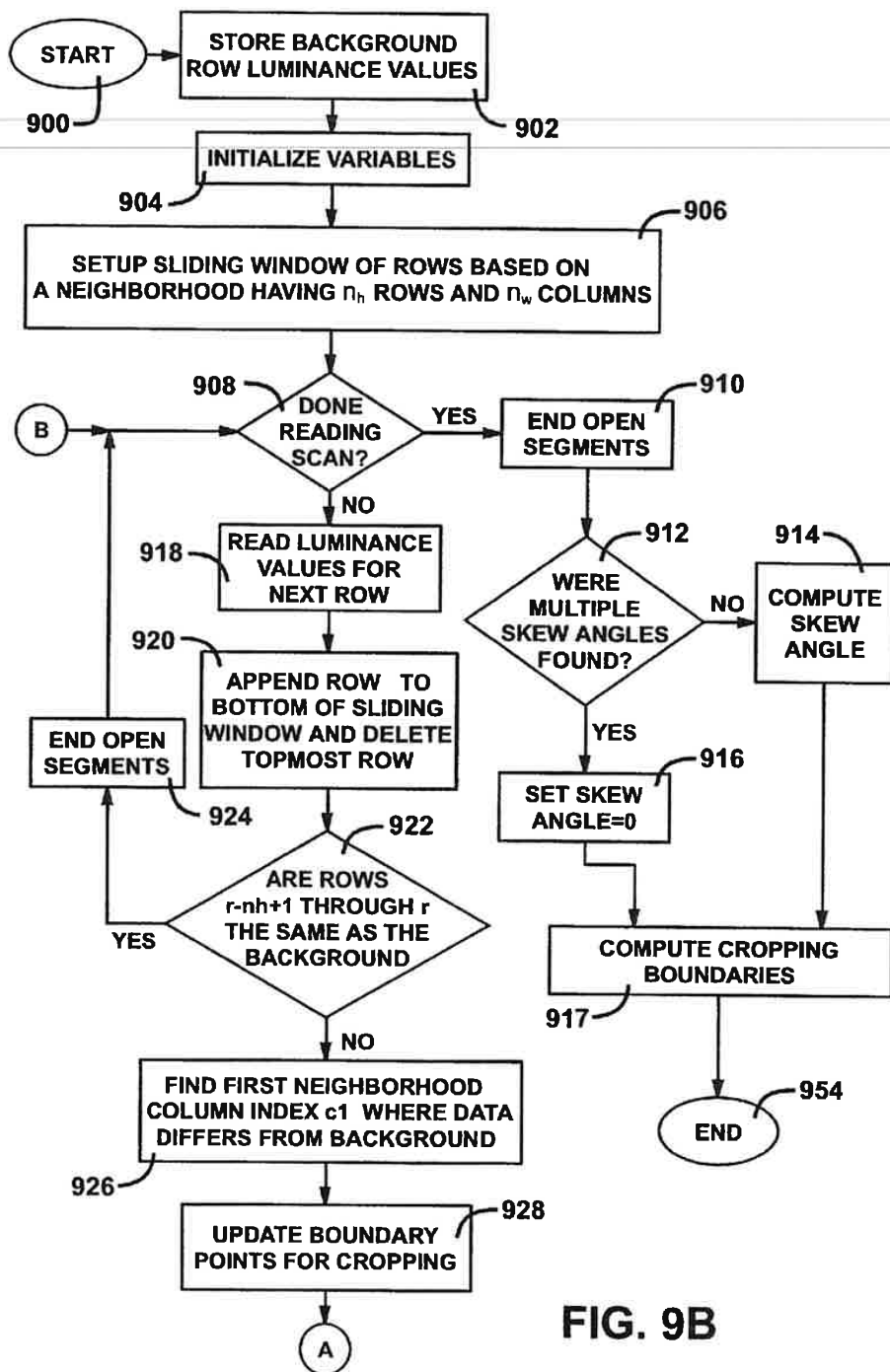


FIG. 9B

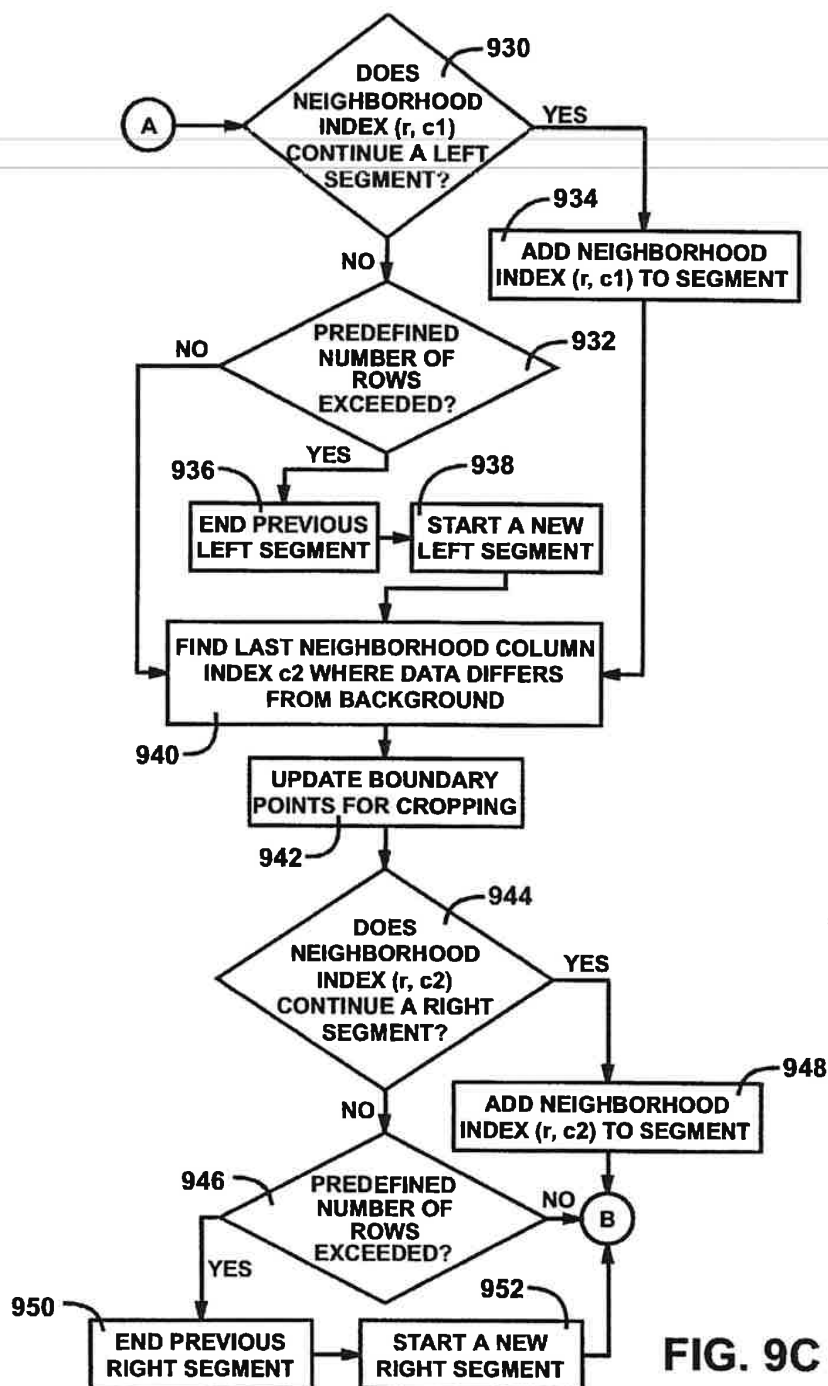


FIG. 9C

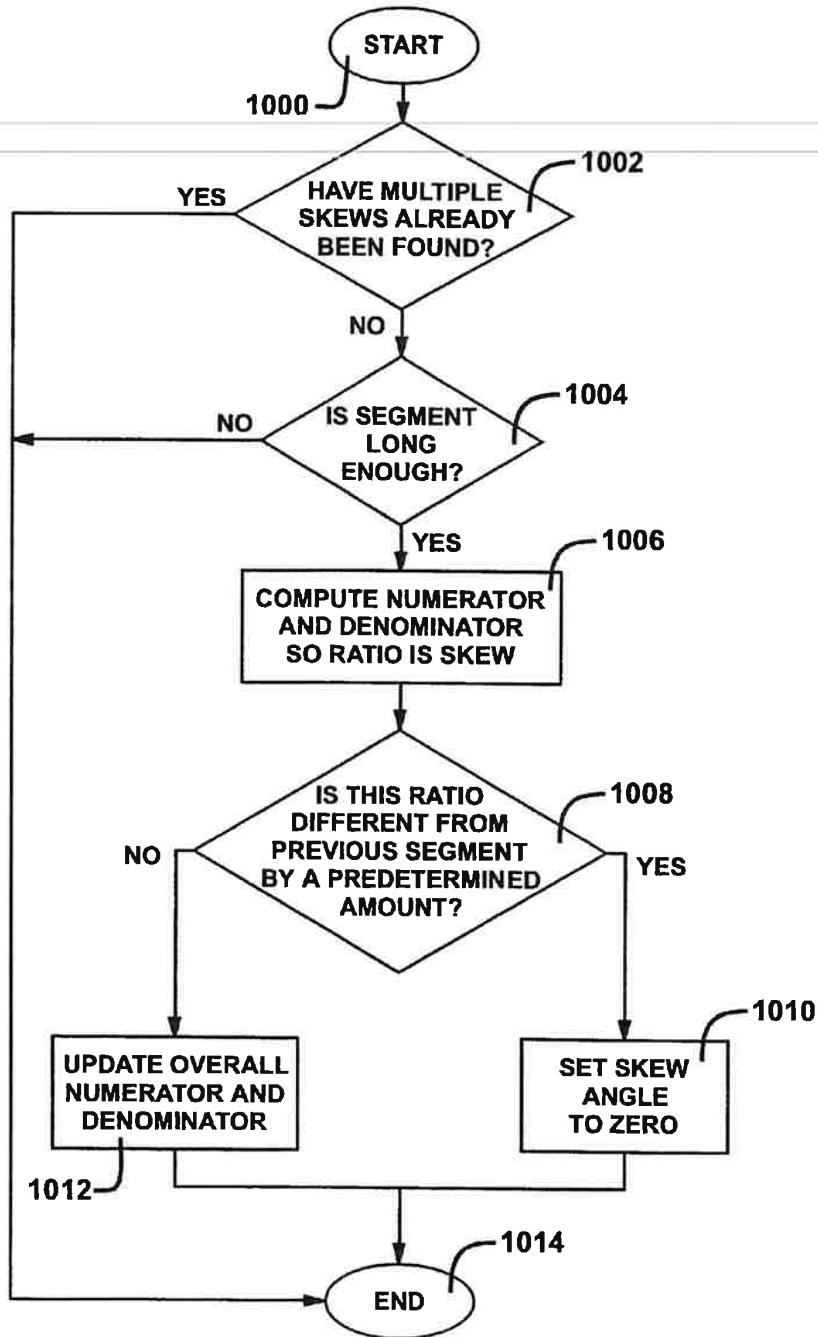
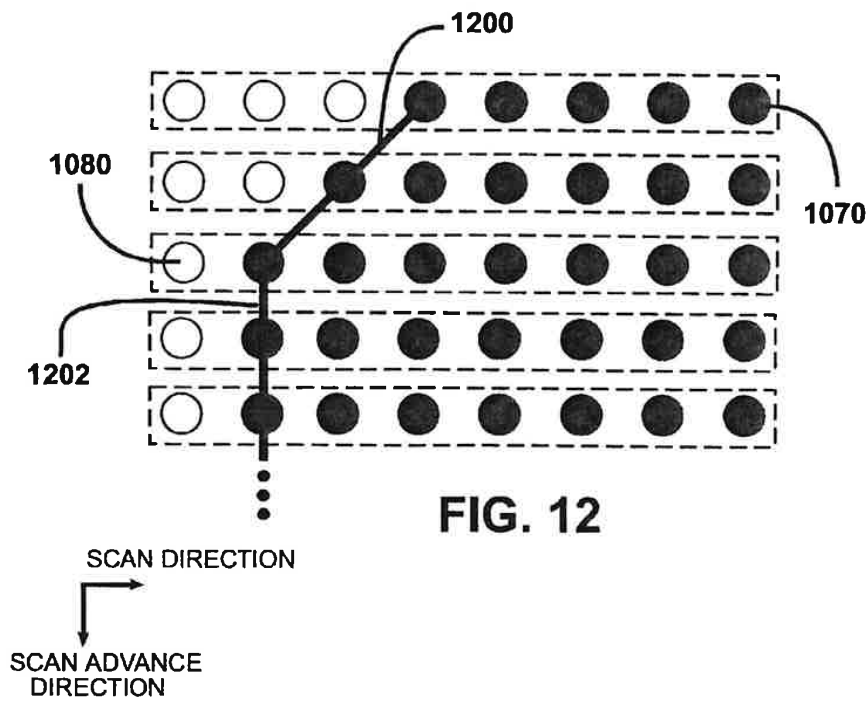
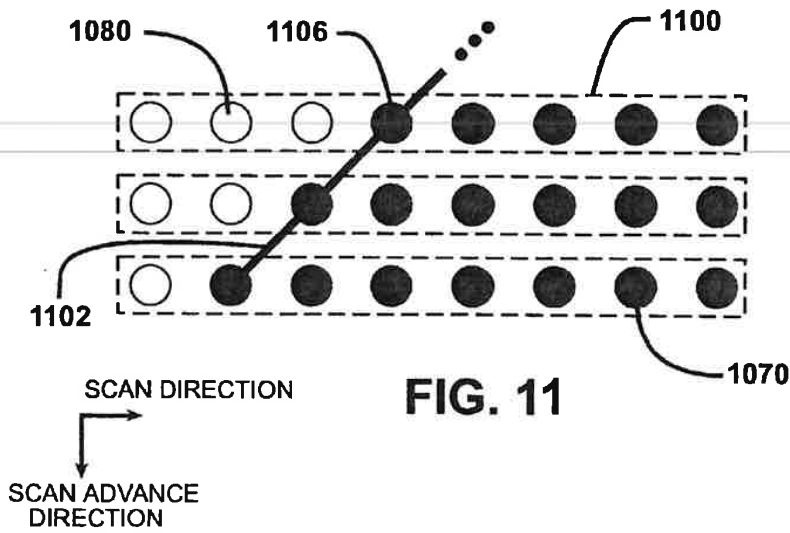


FIG. 10





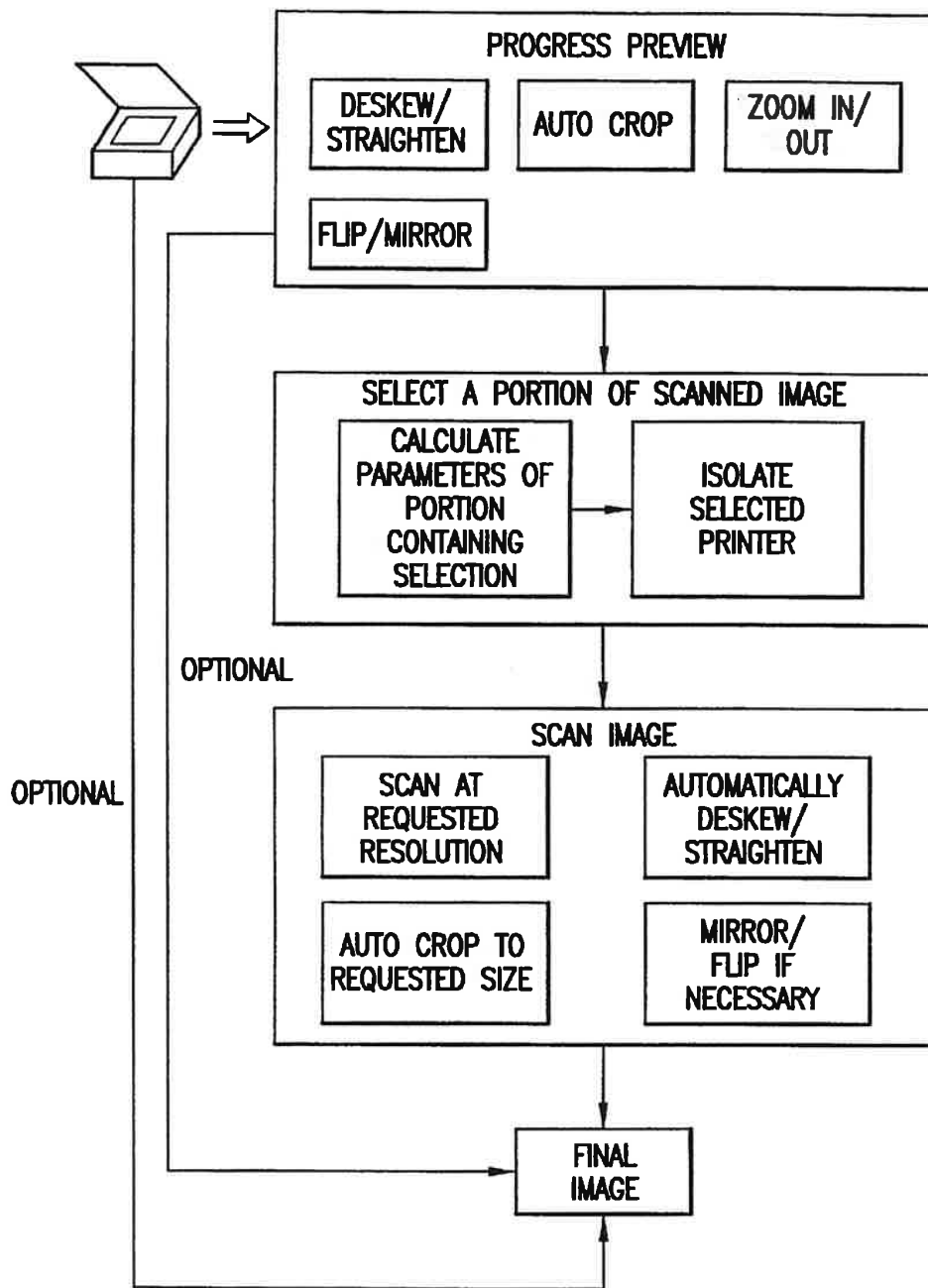


FIG. 13

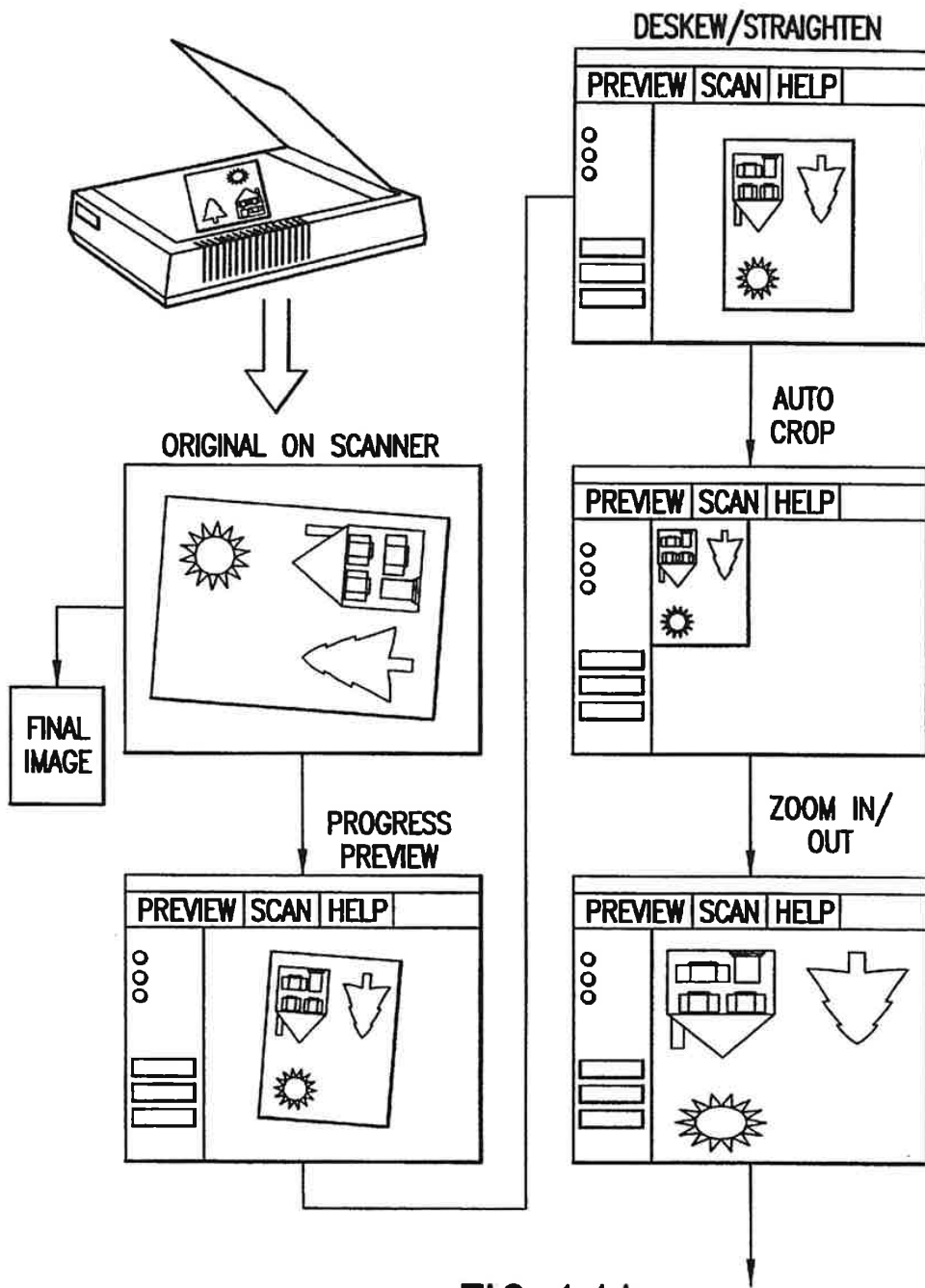


FIG. 14A

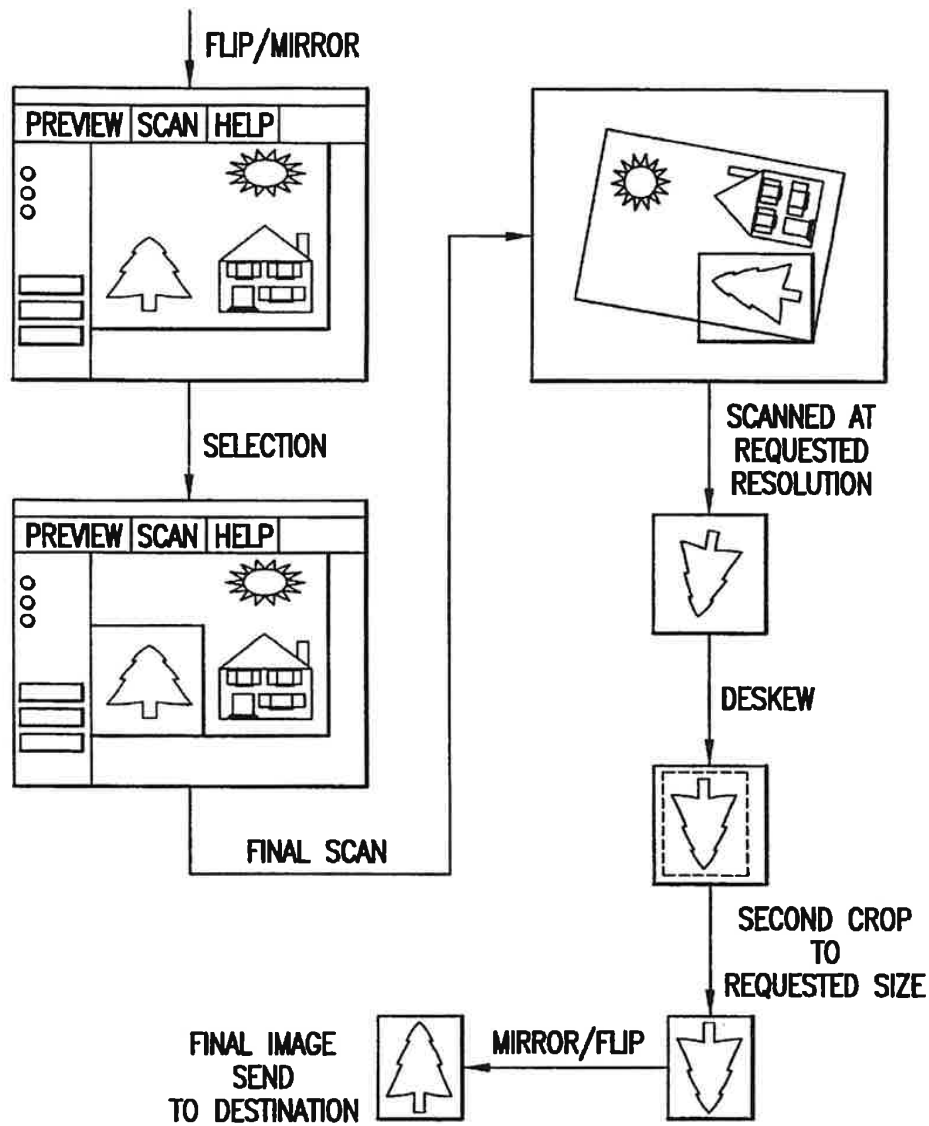


FIG. 14B

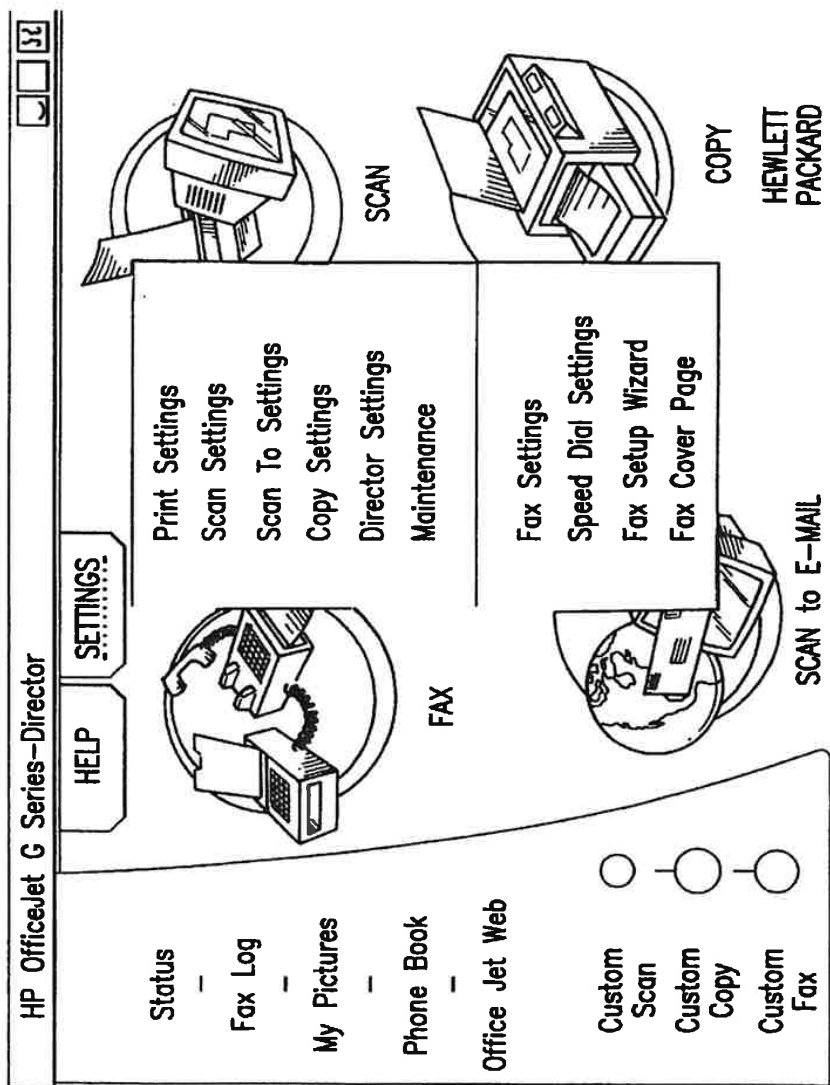


FIG. 15

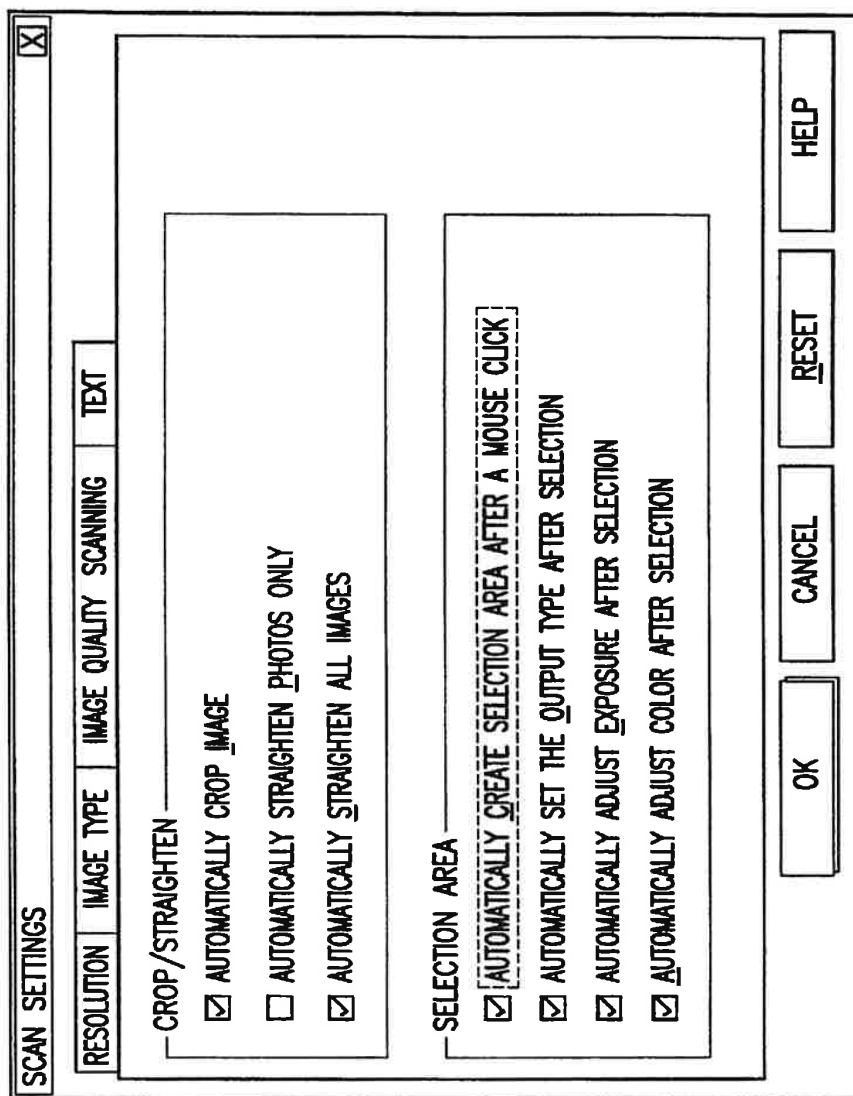


FIG. 16

## IMAGE PROCESSING SYSTEM WITH AUTOMATIC IMAGE CROPPING AND SKEW CORRECTION

### CROSS REFERENCE TO RELATED APPLICATIONS

This application is a continuation of U.S. patent application Ser. No. 09/057,847, filed on Apr. 9, 1998 by Tretter et al. and entitled "IMAGE PROCESSING SYSTEM WITH IMAGE CROPPING AND SKEW CORRECTION".

### BACKGROUND OF THE INVENTION

#### 1. Field of the Invention

The present invention pertains to image processing systems. More particularly, this invention relates to an image processing system with (1) skew correction that does not require human intervention or the presence of text or skew detection information on the original document, and (2) image cropping that is done regardless of the shape of the image.

#### 2. Description of the Related Art

It has been known that when a document (i.e., the original physical object, such as photo or text document) is scanned by a scanner, a digital image of the original document is typically generated. The digital image of the original document is, however, often found to be skewed (rotated) inside the entire scan image (i.e., inside the entire digital image obtained from the scanner). As is known, the scan image typically includes the image of the document as well as background information. A skew or inclination of the document image within the scan image is particularly likely to occur when the scanner uses an automatic document feed mechanism to feed the original document for scanning. In addition, when the size of the original document is relatively small in comparison to the scan region of the scanner, the scan image may contain considerable amount of background information.

For instance, some scanning devices are automatic sheet fed scanners with stationery charge coupled devices (CCD's). These scanning devices feed the document past the CCD for scanning. The document must be grabbed by a set of rollers for scanning. This mechanism can sometimes scratch the document. Also, small documents may not be securely grabbed or reliably sensed by the mechanism. In addition, only a single document at a time can be fed in the scanner. As a result, document carriers are used to overcome these problems. A document carrier is usually a transparent envelope having a white backdrop. The document or documents of interest are inserted within the envelope for scanning. The document carrier protects the scanned document from scratches and also provides the rollers with a larger width original to grab, thereby accomplishing successful feeding of the document through the scanner.

However, one disadvantage of using a document carrier is that the document carrier also becomes part of the scanned data. For example, if the carrier color does not exactly match the color of the scanner background, edges of the document carrier will be contained in the scanned data. This spurious data will cause the digital image to contain unwanted extraneous information. FIG. 1 illustrates a scan image 100 that exhibits these problems.

As can be seen from FIG. 1, the scan image 100 contains a document image 110 of an original document. The remaining area of the scan image 100 is background 120, which typically has a predetermined pixel pattern, and extraneous

information 140, which typically has known characteristics. The background 120 can be caused by the scanner background while the extraneous information 140 can be caused by a document carrier. The document image 110 is skewed inside the scan image 100 and the background 120 is a considerable fraction of the scan image 100. When the scan image 100 is displayed on a display or printed by a printer, the document image 110 typically has a relatively unpleasant and poor visual quality. In addition, the skewed image may also cause errors when the image data is further processed by other software programs, such as optical character recognition programs.

Techniques have been developed to try to detect and correct the skew problem. For example, U.S. Pat. No. 4,941,189, entitled OPTICAL CHARACTER READER WITH SKEW RECOGNITION and issued on Jul. 10, 1990, describes a skew correction technique that searches for text characters along a scan line. As another example, U.S. Pat. No. 5,452,374, entitled SKEW DETECTION AND CORRECTION OF A DOCUMENT IMAGE REPRESENTATION and issued on Sep. 19, 1995, describes another technique that segments the scan image into text and non-text regions and then determines the skew information based on the resulting segmentation.

These techniques, however, require the original document to contain at least some text. The techniques then rely on the detection of one or more lines of the text in the document. With the advent of inexpensive photo scanners and multimedia personal computers, scanners are nowadays used to scan not only text documents, but photographs and other image documents as well. The photographs, however, typically do not contain any text data. This thus causes the skew detection and correction techniques to be inapplicable to the scanned photo images. In addition, because photographs can have a variety of sizes and shapes, it is typically difficult to trim the background information from the scanned image of a photograph.

Another technique has been proposed that detects the skew information of a scanned image without requiring the presence of text in the scanned document. One such technique is described in U.S. Pat. No. 5,093,653, entitled IMAGE PROCESSING SYSTEM HAVING SKEW CORRECTION MEANS, and issued on Mar. 3, 1992. However, this technique requires human intervention.

### SUMMARY OF THE INVENTION

Described below is a system and method for automatically determining in a scanned document image the presence of unwanted extraneous information caused by an extraneous device, for example, a document carrier and scanner background information. Once the presence of this information is determined, the system and method of the present invention can compute, for instance, skew and crop statistics. From this, the image can be automatically deskewed and cropped appropriately without the background and extraneous information (such as marks from the document carrier). The system and method accomplishes this by first determining the presence of unwanted extraneous and background information and then appropriately processing the document image. The extraneous information is ignored during deskew and crop computations. Also, the scanner background and the extraneous information are prevented from being included with the final digital representation of the image.

Specifically, scanner background information and any extraneous information, such as edges created by the docu-

ment carrier, are ignored when processing information is computed, such as skew and crop statistics, while image edges are retained, such as document edges of an image or text pages. Thus, the system and method of the present invention optimizes automatic cropping and deskewing results of document images scanned by general purpose scanning devices that are used with or without document carriers.

Also, the system and method described below determines a skew angle of the document image without requiring text in the document or human intervention. This feature is accomplished by determining an edge of the document image within a scan image and using that edge to determine the skew angle of the document image. The edge can be determined by locating the first or last document image pixel of each scan line of pixels in the scan image that belongs to the document image (i.e., the edge pixel of the document image along that scan line). This is accomplished by comparing a scan line of pixels with a predetermined scan line of background pixels or alternatively by comparing a neighborhood around a scan line with predetermined background pixels. The skew angle of the document image is then determined by computing the slope of the detected edge in the scan image.

In addition, the system and method described below can determine the boundary of the document image. This feature is accomplished by locating (1) a first document image pixel and a last document image pixel for a first scan line of the document image in the scan image, (2) a first document image pixel and a last document image pixel of a last scan line of the document image in the scan image, (3) a leftmost document image pixel of the document image in the scan image, and (4) a rightmost document image pixel of the document image in the scan image.

**BRIEF DESCRIPTION OF THE DRAWINGS**

The present invention is illustrated by way of example and not by way of limitation in the Figures of the accompanying drawings, in which like references indicate similar elements, and in which:

FIG. 1 shows a skewed image of a document in a scan;

FIG. 2 shows a computer system that implements an image processing system;

FIG. 3 shows the image processing system implemented by the computer system of FIG. 2, wherein the image processing system includes an automatic deskew and image cropping system in accordance with one embodiment of the present invention;

FIG. 4 illustrates a different configuration of the automatic deskew and image cropping system in the image processing system of FIG. 3;

FIG. 5 shows a document image generated by the image processing system of FIG. 3 or 4 before being processed by the automatic deskew and image cropping system of FIGS. 3 and 4;

FIG. 6 shows the document image of FIG. 5 after being processed by the automatic deskew and image cropping system of FIGS. 3 and 4;

FIG. 7 shows another document image generated by the image processing system of FIG. 3 or 4 before being processed by the automatic deskew and image cropping system of FIGS. 3 and 4;

FIG. 8 shows the document image of FIG. 7 after being processed by the automatic deskew and image cropping system of FIGS. 3 and 4;

FIG. 9A illustrates a sample user interface for implementing the automatic deskew and image cropping system of FIGS. 3 and 4;

FIGS. 9B-9C and 10 show flow chart diagrams of the automatic deskew and image cropping system of FIGS. 3 and 4;

FIGS. 11 and 12 illustrate calculation of the skew angle and boundary information of a document image by the automatic deskew and image cropping system of FIGS. 3 and 4 when the document image has rectangular and non-rectangular shapes.

FIG. 13 illustrates a high level block diagram/flowchart of the present invention suitable for use in a specific embodiment.

FIGS. 14A-14B illustrate a pictorial block diagram of a working example of a specific embodiment depicted in FIG. 13.

FIGS. 15-16 illustrate sample user interfaces of the working example of FIGS. 14A-14B operating in a computer environment.

**DETAILED DESCRIPTION OF THE INVENTION**

The present invention is a system and method for automatically determining scanner background information and extraneous information within a digital representation of a scanned document image. The scanner background information is caused by the scanner's background and the extraneous information is caused by an extraneous device, such as a document carrier. For instance, due to the physical appearance of the document carrier, it can leave marks within the digital representation of the scanned document image. Once the presence of this information is determined, the system and method of the present invention can compute, for instance, skew and crop statistics. From this, the image can be automatically deskewed and cropped appropriately without the background and extraneous information.

The present invention can be used with general purpose scanning devices for scanning an image as scanned data input. The image can be a photograph, multiple photographs in one scan, text only or mixed documents containing photographs, text, graphics, etc. The present invention parses the scanned data input for determining the presence of scanner background information and extraneous information, which, for example, can be caused by a document carrier. Also, the scanned data input is parsed for determining edges and a skew angle of the image. The parsed data is used to compute skew and crop statistics of the scanned data for cropping and deskewing the image. This ultimately provides an aligned digital representation of the scanned image without unwanted scanner background information and extraneous information. Specifically, the scanner background and any indicia of an extraneous device, such as a document carrier, are ignored when the skew and crop statistics are computed, while image edges are retained, such as document edges of text pages. Thus, the present invention properly crops and deskews images scanned by general purpose scanning devices that are used with or without document carriers.

One of the features of the present invention is to provide skew correction for a scanned image without requiring the presence of text. Another feature of the present invention is to provide skew correction for a scanned image without requiring human intervention. A further feature of the present invention is to provide image cropping for a scanned



image regardless of the size and/or shape of the original. A still further feature of the present invention is to provide skew correction and image cropping for a scanned image in a cost effective manner.

FIG. 2 illustrates a computer system 200 that implements an image processing system 320 (shown in FIGS. 3 and 4) within which an automatic deskew and image cropping system 322 (shown in FIGS. 3 and 4) in accordance with one embodiment of the present invention may be implemented. Although FIG. 2 shows some of the basic components of the computer system 200, it is neither meant to be limiting nor to exclude other components or combinations of components in the system. The image processing system 320 and the automatic deskew and image cropping system 322 in accordance with the present invention will be described in more detail below, also in conjunction with FIGS. 3 through 12.

In one embodiment, the computer system 200 can be a personal computer having a scanner, a notebook computer having a scanner, a palmtop computer having a scanner, a workstation having a scanner, or a mainframe computer having a scanner. In another embodiment, the computer system 100 can be a scan system that also has some or all of the components of a computer system.

As can be seen from FIG. 2, the computer system 200 includes a bus 202 for transferring data and other information. The computer system 200 also includes a processor 204 coupled to the bus 202 for processing data and instructions. The processor 204 can be any known and commercially available processor or microprocessor. A memory 206 is also provided in the computer system 200. The memory 206 is connected to the bus 202 and typically stores information and instructions to be executed by the processor 204. The memory 206 may also include a frame buffer (not shown in FIG. 2) that stores a frame of bitmap image to be displayed on a display 210 of the computer system 200.

The memory 206 can be implemented by various types of memories. For example, the memory 206 can be implemented by a RAM (Random Access Memory) and/or a nonvolatile memory. In addition, the memory 206 can be implemented by a combination of a RAM, a ROM (Read Only Memory), and/or an electrically erasable and programmable nonvolatile memory.

The computer system 200 also includes a mass storage device 208 connected to the bus 202. The mass storage device 208 stores data and other information. In addition, the mass storage device 208 stores system and application programs. The programs are executed by the processor 204 and need to be downloaded to the memory 206 before being executed by the processor 204.

The display 210 is coupled to the bus 202 for displaying information to a user of the computer system 200. A keyboard or keypad input device 212 is also provided that is connected to the bus 202. An additional input device of the computer system 200 is a cursor control device 214, such as a mouse, a trackball, a trackpad, or a cursor direction key. The cursor control device 214 is also connected to the bus 202 for communicating direction information and command selections to the processor 326, and for controlling cursor movement on the display 210. Another device which may also be included in the computer system 200 is a hard copy device 216. The hard copy device 216 is used in the computer system 200 to print text and/or image information on a medium such as paper, film, or similar types of media.

In addition, the computer system 200 includes an image scanner 218. The image scanner 218 is used to convert an

original document (i.e., the original physical document, such as photo or text document) into a digitized image which can be further processed by the computer system 200. In one embodiment, the image scanner 218 is a fax machine-type image scanner that has a scan region of one scan line wide. The length of the scan region is the width of the scan line. In this case, the scan head of the image scanner 218 simultaneously images the entire scan line. A document feed mechanism is provided to advance the original document after each scan. In another embodiment, the image scanner 218 is a copier-type image scanner that has a relatively large scan region. For this type of scanner, the original document is placed against the scan window of the scanner and the scan head of the scanner moves in one direction after each scan.

The computer system 200 also includes other peripheral devices 220. These other devices 220 may include a digital signal processor, a MODEM (modulation/demodulation), and/or a CD-ROM drive. In addition, the computer system 200 may function without some of the above described components. For example, the computer system 200 may function without the hard copy device 216.

As described above, the computer system 200 includes the image processing system 320 (shown in FIGS. 3 and 4) which includes the automatic deskew and image cropping system 322 of the present invention (also shown in FIGS. 3 and 4). In one embodiment, the image processing system 320 is implemented as a series of software programs that are run by the processor 326, which interacts with scan data received from the scanner 218. It will, however, be appreciated that the image processing system 320 can also be implemented in discrete hardware or firmware.

Similarly, the automatic deskew and image cropping system 322 alone can be implemented either as a software program run by the processor 326 or in the form of discrete hardware or firmware within the image processing system 320. The image processing system 320, as well as the automatic deskew and image cropping system 322, will be described in more detail below, in the form of software programs.

As can be seen from FIG. 3, the image processing system 320 includes a scan control program 324 and an imaging program 326, in addition to the automatic deskew and image cropping system 322. All of the programs 322 through 326 are typically stored in the mass storage device 208 of the computer system 200 (FIG. 2). These programs are loaded into the memory 206 from the mass storage device 208 before they are executed by the processor 204.

The scan control program 324 interfaces with the scanner 218 and the imaging program 326. The function of scan control program 324 is to control the scanning operation of the scanner 218 and to receive the scan image of an original document 310 from the scanner 218. As is known, the scan image of a document typically includes the digital image of the document (i.e., the document image) and some background image and extraneous information if an extraneous device, such as a document carrier, is used to aid in scanning the document. The scan control program 324 can be, for example, a scanner driver program for the scanner 218. Alternatively, the scan control program 324 can be any known scanner program for interfacing the scanner 218 with a user.

As described above, the scan control program 324 controls the scanner 218 to scan the document 310. The original document 310 can be of different shapes and sizes. For example, the document 310 can be of a rectangular shape, a

5 polygon shape, or a circular or oval shape. FIG. 5 shows one example of a scan image 500 of the document 310 obtained by the scan control program 324. As can be seen from FIG. 5, the document image 502 of document 310 is skewed inside the scan image 500 and has a skew angle  $\alpha$ . As can be seen from the scan image 500, the scanned document 310 has a rectangular shape. FIG. 7 shows another scan image 700 of the document 310 obtained by the scan control program 324 when the document 310 has an oval shape. Both FIGS. 5 and 7 show considerable background within scan images 500 and 700, respectively.

As shown in FIG. 3, the imaging program 326 is used in the image processing system 320 to process the scan image (e.g., the scan image 500 or 700 of FIG. 5 or 7, respectively) of the original document 310 received from the scan control program 324. The imaging program 326 typically processes the scan image of the original document 310 so that the scan image can be displayed on the display 210 or printed by the hard copy device 216. The processing functions of the imaging program 326 typically include resampling and interpolation of the scan image. The imaging program 326 typically includes a device-specific image driver program. For example, the imaging program 326 can include a known display driver program or a known printer driver program. The imaging program 326 can be any image processing application.

As can be seen from FIG. 3, the automatic deskew and image cropping system 322 of the image processing system 320 interfaces with the scan control program 324 and the imaging program 326. The automatic deskew and image cropping system 322 receives digital data representing the scan image of the document 310 from the scan control program 324 and automatically determines the presence of scanner background information and extraneous information caused by an extraneous device, such as a document carrier. For instance, due to the physical appearance of the document carrier, it can leave marks within the digital data representing the scanned document image 310. The automatic deskew and image cropping system 322 ignores the scanner background information and extraneous information and detects the skew angle and boundary of the document image of the document 310 within the scan image. This provides correction of the skew of the document image (i.e., deskewed) so that much or all of the scanner background information and the extraneous information of the image can be eliminated.

In the case where a document carrier is used, the document carrier can cause unwanted extraneous information because it becomes part of the scanned data. For example, if the carrier color does not exactly match the color of the scanner background, edges of the document carrier will be contained in the scanned data. The present invention detects and deliberately ignores this spurious data and it is deemed as invalid image data. As a result, the document carrier information does not influence the results of other functions and operations of the automatic deskew and image cropping system 322, such as the automatic crop and deskew functions (discussed below in detail).

Many different document carrier sizes exist, and the present invention is not limited to any particular size. For illustrative purposes only, two such sizes of document carriers are a full page carrier, which can be approximately 8.5"x11" (usually for text or mixed documents), and a half page carrier, which can be approximately 8.5"x5.75" (usually for photos). Typically, document carriers have some known physical characteristic or characteristics or some form of indicia that can be used as a basis to form boundaries within the scanned data. This allows unwanted document

carrier information to be distinguished from wanted image data. For instance, the bottom of some document carriers contain a semi-circular notch, which is a known physical characteristic on all document carriers in that class. The semi-circular notch allows a user to more easily insert a document into the document carrier.

The automatic deskew and image cropping system 322 is preprogrammed with known physical characteristics of certain extraneous devices of certain classes. Namely, if a particular class of document carriers are known to have semi-circular notches, the automatic deskew and image cropping system 322 is preprogrammed to indicate that the particular class is associated with semi-circular notches as a known physical characteristic. If the known physical characteristic is found after scanning the document image 310, scanned data representing edges of the document carrier are located so that the entire unwanted extraneous information caused by the document carrier is cropped out and discarded.

Also, because the full size document carrier is too long to be fed sideways, only one orientation for scanning exists if the full size document carrier is used. As such, the known physical characteristic, such as the semi-circle, can only be at the bottom or top edges and cannot be at the left or right edges. Hence, the automatic deskew and image cropping system 322 searches for these known physical characteristics of document carriers, such as semi-circles, and crops out unwanted information appropriately. By discarding the edges of the document carrier, additional functions and operations of the automatic deskew and image cropping system 322 can be performed more accurately.

The automatic deskew and image cropping system 322 detects the skew angle of the document image (e.g., the document image 502 of FIG. 5) inside the scan image (e.g., the scan image 500 of FIG. 5) by first detecting an edge of the document image and then determining the slope of the edge. This allows the skew angle detection of the document image to be done without requiring the presence of text or special skew detection marks on the document image. This also allows the imaging program 326 to correct the skew of the document image without human intervention.

In addition, the automatic deskew and image cropping system 322 detects the boundary of the document image (e.g., the document image 502 of FIG. 5). There are several ways that the automatic deskew and image cropping system 322 detects the boundary of the document image. Two sample techniques are discussed in detail below for illustrative purposes only. Each technique can be custom configured for specific implementations. The first sample technique detects the boundary by locating a first and a last document image pixel for the first scan line of the document image, a first and a last document image pixel for the last scan line of the document image, a leftmost document image pixel of the document image, and a rightmost document image pixel of the document image within the scan image. The positioned information of these six pixels is then used to compute the extent (i.e., boundary) of the document image in the scan image after skew correction. This information is then provided to the imaging program 326, allowing the imaging program 326 to trim or crop the scan image to obtain the document image without much or all of the background information.

The automatic deskew and image cropping system 322 detects the skew angle and boundary information of a document image within a scan image by locating the first and last pixels of each scan line of the document image inside the scan image. The automatic deskew and image

cropping system 322 can accomplish this by comparing each scan line of pixels in the scan image with a predetermined scan line of background pixels to locate the first and last document image pixels. This can alternatively, and preferably, be accomplished by comparing a neighborhood around each scan line of pixels in the scan image with predetermined background pixels to locate the first and last document image pixels. This allows boundary edge segments of the document image to be developed. The automatic deskew and image cropping system 322 then determines the length of each edge segment of the document image and calculates the skew of the edge segment. If the automatic deskew and image cropping system 322 determines that an edge segment is not long enough, the program 322 does not calculate the skew of that edge segment.

In addition, if the automatic deskew and image cropping system 322 determines that the document image has multiple skew angles (i.e., the skew of an edge segment in the document image is not equal to that of another edge segment of the document image), the program 322 determines that the document image has a non-rectangular shape. When this occurs, the automatic deskew and image cropping system 322 sets the skew angle of the document image to  $\theta$ , which is preferably zero, whether the document image is skewed or not. In other words, if the automatic deskew and image cropping system 322 determines that the document image has a non-rectangular (e.g., circular, oval, or polygonal) shape, the program 322 preferably does not detect the skew angle of the document image. Instead, the program 322 provides the boundary information of the document image so that much or all of the background can be trimmed or cropped away from the scan image.

Moreover, when the automatic deskew and image cropping system 322 determines that the detected document image is not of a rectangular shape, the program 322 preferably defines the smallest rectangle that contains all of the six boundary pixels and informs the imaging program 326 to take the entire interior of this rectangle as the cropped document image (see, for example, FIG. 8). In this case, not all background information is trimmed off. The operation of automatic deskew and image cropping system 322 is now described in more detail below, also in conjunction with FIGS. 5-6 when the document 310 has a rectangular shape or FIGS. 7-8 when the document 310 has a non-rectangular shape.

As can be seen from FIGS. 3 and 5-6, the skew detection and image cropping program 322 checks the scan image 500 to locate the first and last document image pixels of the first scan line of the document image 502. As can be seen from FIG. 5, the program 322 learns that the first scan line of the scan image 500 is the first scan line of the document image 502. The program 322 then locates the first document image pixel 518 and the last document image pixel 520 of the first scan line of the document image 502. As the automatic deskew and cropping system 322 continues checking the first and last document image pixels of other scan lines of the document image 502, edge segments 510, 512, 514, 516 are developed. In addition, the leftmost document image pixel 521 and rightmost document image pixel 522 are located. The first and last document image pixels (i.e., 524 and 526) of the last scan line of the document image 502 are also located. As can be seen from FIG. 5, the first document image pixel 524 of the last scan line of the document image 502 overlaps the last document image pixel 526 of that scan line.

After the edge segments 510, 512, 514, 516 of the document image 502 are developed, the automatic deskew

and cropping system 322 calculates the skew angle  $\alpha$  which is then sent to the imaging program 326 (FIG. 3), along with cropping boundaries computed from the skew angle  $\alpha$  and the pixels 518, 520, 522, 524, 526.

As described above, the automatic deskew and cropping system 322 of FIG. 3 also detects if the document image is of a rectangular shape when the program calculates the skew angle  $\alpha$  of the document image. If the program 322 detects that the document image (e.g., the document image 702 of FIG. 7) is not of a rectangular shape, then the program 322 preferably does not calculate the skew angle of the document image and preferably sets the skew angle to zero. The automatic deskew and cropping system 322 detects whether a document image is rectangular or not by determining if the document image has multiple skew angles. When this occurs, the document image has a non-rectangular shape (e.g., the polygonal shape). In addition, the program 322 also detects if the document image has a rectangular shape by detecting if the edge segments of the document image are longer than a predetermined length. Those edge segments shorter than the predetermined length are discarded, and no skew angle is computed for such segments. If all detected segments are discarded, the program 322 determines that the document image has a non-rectangular shape (e.g., oval or circular shape) and again does not calculate the skew angle of the document image. When this occurs, the program 322 preferably locates those six boundary pixels of the document image. FIGS. 9A through 10 show in flow chart diagram form the automatic deskew and cropping system 322, which will be described in more detail below.

As can be seen from FIGS. 3 and 7-8, when the document 310 has a document image 702 that is of an oval shape, the program 322 of FIG. 3 detects multiple edges that are of different skew angles and/or shorter than the predetermined edge length. In one embodiment, the predetermined edge length contains approximately twenty five pixels. In alternative embodiments, the predetermined edge length can be longer or shorter than twenty five pixels.

When the program 322 detects that the document image 702 is not rectangular, the program 322 preferably locates the six boundary pixels (i.e., the first and last document image pixels 710 and 712 of the first scan line of the document image 702, the leftmost document image pixel 714, the rightmost document image pixel 716, and the first and last document image pixels 718 and 720 of the last scan line of the last scan line of the document image 702. As can be seen from FIG. 7, the first and last document image pixels 710 and 712 of the first scan line of the document image 702 overlap each other and the first and last document image pixels of the last scan line of the document image 702 overlap each other.

As can be seen in FIGS. 3 and 5-6, the imaging program 326 then corrects the skew of the document image 502 in accordance with the skew angle  $\alpha$  received from the automatic deskew and cropping system 322 and eliminates all of the background 504 in the scan image 500 in accordance with the six document image pixels 518-526. The imaging program 326 does this in a known way, which will not be described in more detail below. The processed document image 600 is shown in FIG. 6.

As can be seen from FIGS. 5 and 6, the processed document image 600 of FIG. 6 is identical to the unprocessed document image 502 of FIG. 5 except that no background information of the scan image 500 is displayed in FIG. 6. In addition, the processed document image 600 is not skewed. Moreover, the processed document image 600 of

FIG. 6 does not have the cut-off edge. This is due to the fact that the imaging program 326 further trims the document image 502 of FIG. 5 based on the document image pixels 518-526.

When processing the document image 702 of FIG. 7, the automatic deskew and cropping system 322 (FIG. 3) only sends the pixel information of the six boundary pixels 710 through 720 to imaging program 326 (FIG. 3). Based on these six pixels 710-720, the imaging program 326 creates a smallest rectangle 800 that contains all of these pixels and the document image 702. The imaging program 326 then trims away everything in the scan image 700 of FIG. 7 that is outside of the rectangle 800 to obtain the cropped document image 702.

As can be seen from FIG. 3, because the automatic deskew and cropping system 322 interfaces with the scan control program 324, the automatic deskew and cropping system 322 receives one scan line of pixels from the scan control program 324 as soon as the scan control program 324 controls the scanner 218 to finish scanning one such scan line. This causes the automatic deskew and cropping system 322 to operate in parallel with the operation of the scan control program 324. As a result, the automatic deskew and cropping system 322 can determine the skew angle and boundary information of the document image of the document 310 as soon as the scan control program 324 finishes scanning the document 310.

It is, however, appreciated that the automatic deskew and cropping system 322 is not limited to the above described configuration. FIG. 4 shows another embodiment of the image processing system 320 in which the automatic deskew and cropping system 322 only interfaces with the imaging program 326. This allows the automatic deskew and cropping system 322 to detect the skew angle and boundary information of the document image of the document 310 after the entire document 310 has been scanned and its scan image has been sent to the imaging program 326 from the scan control program 324.

FIG. 9A illustrates a sample user interface for implementing the automatic deskew and image cropping system of FIGS. 3 and 4. The present invention increases user ease by automatically deskewing and cropping scanner background information and extraneous information (although automatic functions can be disabled, if desired). For automatic operation, the system starts 810 a user is given options for specifying a type of document to be scanned, such as text only, mixed format, photo only, custom options, etc., and the automatic deskew and cropping system 322 finds the best crop and deskew operation. The options can be presented in two tiers. The first tier allows novice users to simply specify the kind of document they are scanning (photo only, mixed document, etc.). The second tier allows more sophisticated users to further customize processing.

Namely, several options can be presented to a user. These options increase processing flexibility for the user. First, second and third options 812, 814, 816 can be for novice users and a fourth option 818 can be for advanced users with customization functions. The first option 812 can be for images that contain text only and the second option 814 can be for mixed formats (for example, images that contain a combination of photographs, text, graphics, etc). For the first and second options 812, 814, automatic deskew and cropping functions are preferably disabled 815 and the routine ends 817. The third option 816 can be for images that contain only photos. If the user chooses the fourth option 818, the user can be presented with three customization

sub-options. A first sub-option 820 for images that contain only photos, a second sub-option 822 for mixed formats and a third sub-option 824 for manually disabling the automatic functions 826 after which, the routine ends 828.

If the third option 824 and the first sub-option 820 are chosen, an automatic skew and crop detection step 830 is performed based on a first set of predefined parameters (discussed below in detail). If the second sub-option 822 is chosen, an automatic skew and crop detection step 832 performed based on a second set of predefined parameters (discussed below in detail). The automatic deskew and cropping system 322 determines the boundaries and location of the scanner background and extraneous information, if it exists. As discussed above, the extraneous information can be caused by a document carrier. The document carrier information is found based on the first and second set of predefined parameters (discussed below in detail). Next, the automatic deskew and cropping system 322 performs an automatic deskew and cropping (crop out portions of the scanned data that are not part of the photo) function as steps 834 and 836, the routing then ends 838. For example, during cropping, unwanted scanner background or document carrier information will be automatically cropped out. In addition, the automatic functions provide cropping for multiple photos being scanned as a single page. In this case, regions outside of the multiple photos are cropped out.

The following description is for illustrative purposes only. The extraneous device can be any extraneous device and does not have to be a document carrier. Specifically, if a document carrier is the extraneous device causing the extraneous information, depending on the option chosen by the user, the automatic deskew and cropping system 322 searches for the known physical characteristics of the particular document carrier. For instance, if the user chooses the third option or the first sub-option, for example, for photos only, the automatic deskew and cropping system 322 searches for either a half or full size document carrier. This is because a user could utilize either the half or full size document carrier for a photo. Similarly, if the user chooses the second sub-option, for example for mixed formats, the automatic deskew and cropping system 322 preferably searches for a full size document carrier. This is because a mixed document typically is too large for the half size document carrier. Therefore, a search is preferably performed for either the half or full size document carrier if the third option or the first sub-option (photo only) is chosen while a search is preferably performed for the full size document carrier if the second sub-option (mixed format) is chosen.

For the half size document carrier, an initial search is performed for known physical characteristics, such as a semicircle at the bottom, top, right, or left edges. This is because some document carriers, such as the half size document carrier, can be fed into the scanner device in any orientation. As a result, the known physical characteristic, such as the semicircle, can appear at the bottom, top, left or right edges of the scan. For the full size document carrier, an initial search is performed for known physical characteristics, such as a semicircle at the bottom or top edges. This is because some document carriers, such as the full size document carrier, can be fed into the scanner device in only two orientations. As such, the known physical characteristic, such as the semicircle, can appear only at the bottom or top edges of the scan.

If the known physical characteristic is found, scanned data representing edges of the document carrier are ignored during computation of skew and crop statistics, and are

eventually cropped out and discarded as unwanted information of the scan. Also, because the full size document carrier is too long to be fed sideways, only one orientation for scanning exists if the full size document carrier is used. As such, the known physical characteristic, such as the semi-circle, can only be at the bottom or top edges and cannot be at the left or right edges. Hence, the present invention searches for these known physical characteristics of document carriers, such as semi-circles, and crops out unwanted information appropriately. By ignoring the edges of the document carrier, more accurate automatic deskewing and cropping of the information of interest can be performed.

FIGS. 9B and 9C show the process of the automatic deskew and cropping system 322 (FIGS. 3 and 4) in developing the edge segments and the six boundary pixels of the document image. FIG. 10 shows the process of the system 322 of FIGS. 3 and 4 in detecting the skew angle of the document image based on the edge segments developed by the process of FIGS. 9B and 9C. FIG. 11 shows how edge segments are developed in a rectangular document image. FIG. 12 shows how edge segments are developed in a circular or oval document image. FIGS. 9B, 9C and 10 will be described in more detail below, also in connection with FIGS. 11 and 12.

In one embodiment, an edge of the document image is determined within a scan image and that edge is used to determine the skew angle of the document image. The edge can be determined by locating the first or last document image pixel of each scan line of pixels in the scan image that belongs to the document image (i.e., the edge pixel of the document image along that scan line). This is accomplished by comparing each scan line of pixels with a predetermined scan line of background pixels. The skew angle of the document image is then determined by computing the slope of the detected edge in the scan image.

In another embodiment, a pixel of a scan line is regarded as an image pixel when its color is different from the color of the corresponding reference background pixel by more than the predetermined threshold value and the color of its adjacent pixel is also different from the color of the corresponding reference background pixel by more than the predetermined threshold value. In other words, small groups of pixels are analyzed together, such as a neighborhood of pixels. This can be accomplished by using a sliding window of pixels. This increases accuracy and more readily distinguishes actual wanted document data from unwanted extraneous information and background noise. This embodiment is more robust in the presence of scanner noise.

Specifically, as can be seen from FIGS. 9B and 9C, the process starts at step 900. At step 902 color values of background pixels are set. At step 904 variables are initialized and a neighborhood size is defined. The neighborhood size can be defined with a pixel size having a neighborhood height of pixels and a neighborhood width of pixels ( $n^h$  rows,  $n^w$  columns). The values are set as the reference values for comparing with the colors of the pixels of a neighborhood around each scan line of the scan image to locate the first and last image pixels (i.e., edge pixels) of each scan line. In another embodiment, only the luminance value of each pixel is used, where luminance is computed as approximately one-fourth red, one-half green, and one-eighth blue. In one embodiment, a pixel is regarded as an image pixel when its color (or luminance) is different from the color (or luminance) of the corresponding reference background pixel by more than a predetermined threshold value. The term color will be used hereinafter interchangeably to mean color and/or luminance. The threshold value is typically a con-

stant that is determined based on the expected variability of the scanner background.

At step 906, a sliding window can be set up as a neighborhood of (Air) pixels comprised of several rows, such as two, three, four, etc. rows. The size of the sliding window or neighborhood of pixels can be adjusted to suit certain conditions. For example, a larger neighborhood of pixels can be used when a photograph is to be scanned. In contrast, a smaller neighborhood of pixels can be used when a mixed document is to be scanned. It should be noted that the neighborhood of pixels for a mixed document should not exceed a maximum predetermined value. This is because text data could be mistaken as background noise if a neighborhood of pixels that is too large is used. The neighborhood of pixels can be defined with a size having a neighborhood height and a neighborhood width ( $n^h$  rows,  $n^w$  columns).

At step 908, it is determined if all of the scan lines of the scan image have been processed. If so, steps 910-914 are performed to calculate the skew angle of the document image inside the scan image. As can be seen from FIG. 9B, step 912 is employed to determine if the document image is of non-rectangular shape. The program 322 (FIGS. 3 and 4) does this at step 912 by determining if different skew angles are found for the edge segments of the document image. If so, the program 322 does not calculate the skew angle of the document image. Instead, the skew angle is set to zero in step 916. If, at step 912, it is determined that these are not multiple skew angles, then step 914 is performed to calculate the skew angle of the document image. In either case, the program 322 finishes by computing the cropping boundaries in step 917 and ending at step 954.

When, at step 908, if it is determined that the scan image has not been completely checked, step 918 is then performed to obtain the neighborhood around the next unchecked scan line of pixels (e.g., scan row  $r$ ). Next, although the sliding window is initially set at some number, at step 920 the sliding window is incremented every time a scan line is checked so that row  $r$  is appended to the bottom of the sliding window and the topmost row is deleted. A color of a neighborhood around each of the pixels of the scan row  $r$  is then compared with a color of predetermined background pixels at step 922 to determine if they match. In other words, for a neighborhood of three rows, rows  $r$ ,  $r-1$ ,  $r-2$  are compared to predetermined background pixels. If they match, (i.e., rows  $r-n^h+1$  through  $r$  contains substantially background pixel values), then the program 322 returns to step 908 via step 924. If not, step 926 is performed, at which the first document image pixel (i.e., pixel  $c1$ ) where a neighborhood index, such as row  $r$  and column  $c1$  having a color different from that of the corresponding background pixel is located. In this case, row  $r$  and column  $c1$  indicates a lower corner. However, this row is arbitrary and any row could be used for the neighborhood index, as long as it is the same all of time.

The process then moves to step 928, at which the boundary pixel storage is updated. This is done by comparing the current first and last pixels with the stored six boundary pixels to determine if these six pixels need to be updated. The positioned values of these six pixels are initially set at zero. If, for example, the positional value of the current first pixel is less than that of the stored leftmost pixel, then the stored leftmost pixel is replaced with the current first pixel. This allows the six boundary pixels of the document image to be finally determined.

Then step 930 is performed, at which it is determined if neighborhood index ( $r$ ,  $c1$ ) continue a left edge segment. If

so, step 934 is performed to continue the edge segment by adding neighborhood index (r, c1) to segment the edge segment. For example, as can be seen from FIG. 11, with image pixels 1070 and background pixels 1080, if scan line 1100 is currently checked and pixel 1106 is determined to be the first pixel of the scan line 1100. Step 930 of FIG. 9C then determines if the pixel 1106 continues the edge segment 1102 and causes the edge segment 1102 to extend from the pixel 1106. However, edge segments are preferably allowed to skip a predefined number of rows if subsequent rows are not aligned. This is because random noise can cause one or several rows to temporarily misalign or diverge for only a few rows. In this case, the edge segment should continue. FIG. 12 shows the development of edge segments 1200 and 1202 of a circular or oval document image. Similarly, edge segments are preferably allowed to skip a predefined number of rows if subsequent rows are not aligned.

Thus, as can be seen from FIG. 9C, when the answer is no at step 930, it is determined in step 932 whether a predefined number of rows has been exceeded. If so, step 936 is then performed to end that left edge segment. Step 938 is then performed to start a new left edge segment from this first pixel. If a predefined number of rows has not been exceeded, then steps 940 through 952 are performed so that a last neighborhood column index c2 is located where a color differs from that of the corresponding background pixel. As can be seen from FIGS. 9B-9C, steps 940-952 are basically the same steps as steps 926-938, except that steps 940-952 are employed to locate and process the last pixel of the scan line while steps 926-938 are employed to locate and process the first pixel of the scan line. Also, steps 926-938 can be performed in parallel with steps 940-952. In other words, steps 940-952 do not have to be performed sequentially after steps 926-938.

FIG. 10 shows the process of updating the skew information based on a detected edge segment. This process is undertaken when a segment is ended, as in steps 910, 924, 936, and 950 of FIGS. 9B and 9C. The routine starts 1000 and it is determined in step 1002 whether multiple skews have already been found. If so, the routine ends at step 1014. If not, whenever the segment is too short, it is discarded in step 1004. If the segment is long enough, a numerator and denominator ratio are determined at step 1006. Next, if the ratio is too different from that of a previous segment, or in other words, if the document image is determined to have a non-rectangular shape in step 1008, the skew angle is set to zero, and subsequent segments are discarded in step 1010. Otherwise, the slope of the detected segment is used to update the skew angle estimate in step 1012 and the routine then ends in step 1014.

In addition, in typical scanner devices, the user is permitted to change brightness settings, which alters the luminance values of the scanned data. Since the automatic deskew and cropping system 322 can use luminance values to perform edge detection, the automatic deskew and cropping system 322 performs dynamic adjustment of background threshold values to match changes in brightness settings. Moreover, the user is usually permitted to change color/grayscale mode settings (such as 24 bit color or 8 bit grayscale scans), which alters the luminance values of the scanned data since the luminance values of grayscale images are different from the color images. The automatic deskew and cropping system 322 performs dynamic adjustment of threshold values to match changes in color/grayscale mode settings.

FIG. 13 illustrates a high level block diagram/flowchart of the present invention suitable for use in a specific embodiment. In this embodiment, a document 1306, which can be

represented by images, is converted into a digital format. This can be accomplished with an optical scanning device 1308, such as a flatbed scanner, as shown in FIGS. 13-14, preferably coupled to a computer system. The computer system is preferably physically connected to the scanner 1308 and can be any suitable electronic computer system that allows the scanner 1308 to interface with a user in a software environment, for example, with software driven modules and graphical user interfaces.

FIGS. 14A-14B illustrate a pictorial block diagram of a working example of a specific embodiment depicted in FIG. 13. Referring to FIGS. 14A-14B along with FIG. 13, in general, the user can have the scanner 1308 either perform a final scan for producing a final image 1310 of the document 1306 (which can have images 1410, 1412, 1414) without a preview or have the scanner first initially preview the information contained on the document 1306. The user preferably controls the scanner 1308 via graphical user interfaces of the software environment of the computer system.

Namely, to convert the document 1306 into a digital format, a user (not shown) can place the document 1306 on the scanner 1308 and initiate scanning of the document 1306, as shown in FIGS. 13 and 14A. This can be done by either activating buttons located on the scanner 1308 itself or by accessing the scanner 1308 through a software interface (not shown). An initial preview of the document 1306 can be accomplished by having the scanner activate a software module that performs a progress preview scan 1312 of the document 1306. If a progress preview 1312 is performed, the user can select a portion of the scanned image 1324 for further processing 1330 before the final image 1310 is produced. The progress preview function 1312 allows processing of the document 1306 before a final scan. This is convenient to a user because it reduces processing of the document 1306 by a digital editing software program before the document is ready for digital use.

In particular, the progress preview 1312 allows an initial preview of the document 1306 that is to be converted into a digital format. A graphical user interface 1416 can be used to interface the user with the progress preview mode 1312. Also, it should be noted that, preferably, in the progress preview 1312, all actions are simulations. As such, all actions designated in the preview 1312 will be performed and applied during the final scan for the final image 1310. Moreover, some of the calculations performed for the preview mode 1312 are reversed.

The graphical user interface 1416 can be any suitable interface that allows a user to digitally view the document 1306 and also to perform and view digital processing that is to be performed on the document 1306 digitally. For example, if images 1410, 1412, 1414 of the document 1306 were skewed 1420 after progress preview 1312 (for instance, from skewed placement of the document 1306 on the scanner 1308 before scanning), the images can be automatically deskewed/straightened 1314 by the software module or manually deskewed/straightened 1314 by the user via the user interface 1416. In addition, the images 1410, 1412, 1414 can be automatically cropped 1316, zoomed in 1318 or flipped (to produce a mirror image of the original image) 1320 by the software module for preparation of the document 1306 before a final scan. These operations can also be performed manually by the user via the user interface 1416.

Next, a portion of the document 1306 can be manually selected 1324 by the user or automatically selected by the progress preview 1312. Automatic selection can be accom-

plished by any suitable background and pixel separation method for distinguishing the images 1410, 1412, 1414. Manual selection 1324 of a portion of the document can be performed by allowing the user to select a portion of a digital representation of the document. For example, the user defined a "rough" selection by drawing a perimeter around the desired portion or desired image 1414 via the user interface.

Optionally, after the desired image 1414 or selected portion is selected, parameters of the image 1414, such as its location relative to other images in the document, similarities between neighboring pixels, etc. are calculated to isolate 1328 the desired image 1414 and make a more accurate selection of the "rough" selection made by the user. Also, the area selected by the user can be modified so that it is an area that is larger than an area selected by the user to ensure that the result of additional processing, such as deskew (discussed below) yields the correct user selected area. Once the desired image 1414 is isolated, the document 1306 can be scanned 1330 into its final digital format.

During final scanning 1330 of the document 1306, numerous functions can be incorporated into the final scan 1330. For instance, the user can change the resolution that the image 1414 is to be scanned, the image 1414 can be manually or automatically deskewed 1334, the image 1414 can be manually or automatically cropped to an appropriate or requested size 1336 (to eliminate or reduce non-desired background data or non-desired overlapping images), or the image 1414 can be manually or automatically flipped or mirrored 1338, if necessary. Last, the final image 1310 is produced by the scan 1330.

FIGS. 15-16 illustrate sample user interfaces of the working example of FIGS. 14A-14B operating in a computer environment. The user interface of FIG. 15 can display the main functions for operating the scanner 1308 of FIG. 13. For example, basic functions for assisting a user, such as help files and scanner settings. Other sample functions are shown graphically in FIG. 15. The user interface of FIG. 16 can display specific functions for operating the scanner 1308 of FIG. 13 with the scan settings. For example, specific functions for controlling the image quality, resolution of the scan, etc. Sample functions are shown graphically in FIG. 16.

In the foregoing specification, the invention has been described with reference to specific embodiments thereof. It will, however, be evident to those skilled in the art that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention. The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense.

What is claimed is:

1. A method of processing a document image inside a scan image having a plurality of scan lines of pixels, comprising:

- (a) applying at least one scan processing simulation to a portion of the scan image to produce a first processed simulation in a first resolution mode;
- (b) providing a visual preview of the first processed simulation for approval;
- (c) reversing the applied simulation to return the scan image to its original state and then either returning to step (a) to allow application of another scan processing simulation if the preview is disapproved or continuing to step (d) if the preview is approved; and
- (d) secondarily applying the approved scan processing simulation to the portion of the scan image in a second resolution mode that is higher than the first resolution mode.

2. The method of claim 1 wherein the simulation includes automatically comparing a neighborhood of pixels located around a group of successively received scan lines from the scan image with predetermined background pixels to define image boundaries for each group of scan lines, forming an edge segment during receipt of the scan lines by extending an image boundary between successive groups of scan lines and determining a skew angle by calculating an aggregate slope of all edge segments longer than a predetermined length value, and further comprising searching for predefined known characteristics within the document image and ignoring the predefined known characteristics if found.

3. The method of claim 1, wherein the scan image includes document image pixels of the document image and background pixels of the scan image, wherein comparing a neighborhood of pixels further comprises:

receiving a neighborhood of pixels located around a group of scan lines; and

comparing the group of scan lines of the scan image with corresponding background pixels to define left and right image boundaries for each group of scan lines.

4. The method of claim 3, further comprising:

comparing color of the group of scan lines with color of the corresponding background pixels;

comparing color of adjacent pixels of the group of scan lines with color of the corresponding background pixels; and

confirming the location of an image boundary when the color of the group of scan lines is different from that of the corresponding background pixels and the color of the adjacent pixels are different from that of the corresponding background pixels.

5. The method of claim 4, wherein the simulation includes at least one of cropping, skewing and rotating the scan image.

6. The method of claim 1, further comprising ending the edge segment and generating a new edge segment that extends from an end scan line of the group of scan lines if the group of scan lines do not continue the edge segment.

7. A computer apparatus, comprising:

a computer executable program stored on a storage medium, the computer executable program, when executed, processes a document image inside a scan image having a plurality of scan lines of pixels by:

sending a portion of the scan image in a first resolution mode to a digital processor, applying and storing at least one scan processing simulation to a portion of the scan image, allowing approval or disapproval of the simulation, reversing the applied simulation, if the simulation is approved, sending a portion of the scan image in a second resolution mode higher than the first resolution mode to the digital processor without the simulation and applying the stored simulation to the portion of the scan image in the second resolution mode.

8. The apparatus of claim 7, wherein the scan image includes image pixels of the document image and background pixels of the scan image, further comprising,

receiving a neighborhood of pixels located around a group of scan lines;

comparing color of the group of scan lines with color of corresponding background pixels;

comparing color of adjacent pixels of the group of scan lines with color of the corresponding background pixels; and

confirming the location of an image boundary when the color of the group of scan lines is different from that of

19

the corresponding background pixels and the color of the adjacent pixels are different from that of the corresponding background pixels.

9. The apparatus of claim 8, further comprising a first set of instructions that receives and examines a neighborhood of pixels located around a group of scan lines of pixels of the scan image, a second set of instructions that compares a neighborhood of pixels located around a group of scan lines with predetermined background pixels to define image boundaries for each group of scan lines, a third set of instructions that forms an edge segment by extending an image boundary between continuous groups of scan lines and a fourth set of instructions that determines the skew angle by calculating the slope of all edge segments longer than a predetermined length value.

10. The apparatus of claim 7, further comprising a first subset of the third set of instructions that ends the edge segment and generates a new edge segment that extends

20

from an end scan line of the group of scan lines if the group of scan lines do not continue the edge segment.

11. The apparatus of claim 9, further comprising

a first subset of the fourth set of instructions that determines if the document image has a rectangular shape by determining the geometrical relationship to previous edge segments and if the edge segment is longer than a predetermined length value; and

a second subset of the fourth set of instructions that sets the skew angle to zero if the document image is not within predefined limits of the rectangular shape.

12. The apparatus of claim 7, further comprising a fifth set of instructions that searches for predefined known characteristics within the document image and ignores the known characteristics if found.

\* \* \* \* \*





US006434271B1

(12) **United States Patent**  
**Christian et al.**

(10) **Patent No.:** **US 6,434,271 B1**  
(45) **Date of Patent:** **Aug. 13, 2002**

(54) **TECHNIQUE FOR LOCATING OBJECTS WITHIN AN IMAGE**  
(75) **Inventors:** **Andrew Dean Christian, Lincoln;**  
**Brian Lyndall Avery, Lexington, both**  
**of MA (US)**  
(73) **Assignee:** **Compaq Computer Corporation,**  
**Houston, TX (US)**  
(\* ) **Notice:** **Subject to any disclaimer, the term of this**  
**patent is extended or adjusted under 35**  
**U.S.C. 154(b) by 0 days.**

5,657,426 A 8/1997 Waters et al. .... 395/2.85  
5,748,804 A \* 5/1998 Surka ..... 382/291  
5,764,283 A \* 6/1998 Pingali et al. .... 348/169  
5,845,007 A \* 12/1998 Ohashi et al. .... 382/199  
5,947,169 A \* 10/1999 Bachelder ..... 382/151  
6,335,985 B1 \* 1/2002 Sambonsugi et al. .... 382/190

**OTHER PUBLICATIONS**

Describing Motion for Recognition, Little, et al., 1995  
IEEE, pp. 235-240.  
Compact Representations of Videos Through Dominant and  
Multiple Motion Estimation, Sawhney, et al. IEEE 1996, pp.  
814-830.  
Registration of Images with Geometric Distortions. Ardeshir  
Goshtasby, vol. 26, Jan. 1988, pp. 60-64. The Integration of  
Optical Flow and Deformable Models with Applications to  
Human Face Shape and Motion Estimation, DeCarlo, et al.  
IEEE 1996, pp. 231-238.  
The Integration of Optical Flow and Deformable Models  
with Applications to Human Face Shape and Motion Esti-  
mation, decarlo, et al. IEEE 1996, pp. 231-238.  
A Vision System for Observing and Extracting Facial Action  
Parameters, Essa, et al. IEEE 1994, pp. 76-83.  
Realistic Modeling for Facila Animation, Lee, et al, *Com-  
puter Graphics Proceedings, Annual Confeence Series,*  
1995, pp. 55-62

(21) **Appl. No.:** **09/020,043**  
(22) **Filed:** **Feb. 6, 1998**  
(51) **Int. Cl.<sup>7</sup>** ..... **G06K 9/46**  
(52) **U.S. Cl.** ..... **382/194; 382/103; 382/199;**  
**382/203; 382/266; 382/291; 348/169; 348/415.1**  
(58) **Field of Search** ..... **382/174, 187,**  
**382/190-199, 103, 168, 203, 266, 286,**  
**291; 348/169, 415.1; 707/20**

(56) **References Cited**  
**U.S. PATENT DOCUMENTS**

4,644,582 A	2/1987	Morishita et al. ....	382/6
4,821,029 A	4/1989	Logan et al. ....	340/712
4,851,616 A	7/1989	Wales et al. ....	178/18
5,048,103 A	9/1991	Leclerc ..... 382/44	
5,067,015 A	11/1991	Combridge et al. ....	358/133
5,105,186 A	4/1992	May ..... 340/784	
5,280,610 A	1/1994	Travis, Jr. et al. ....	395/600
5,376,947 A	12/1994	Kurode ..... 345/173	
5,440,744 A	8/1995	Jacobson et al. ....	395/650
5,459,636 A	* 10/1995	Gee et al. .... 707/20	
5,487,115 A	* 1/1996	Surka ..... 382/296	
5,551,027 A	8/1996	Choy et al. .... 395/600	
5,581,758 A	12/1996	Burnett et al. .... 395/614	
5,630,017 A	5/1997	Gasper et al. .... 395/2.85	
5,640,468 A	* 6/1997	Hsu ..... 382/190	
5,640,558 A	6/1997	Li ..... 395/612	
5,652,880 A	7/1997	Seagraves ..... 395/614	
5,652,882 A	7/1997	Doktor ..... 395/617	

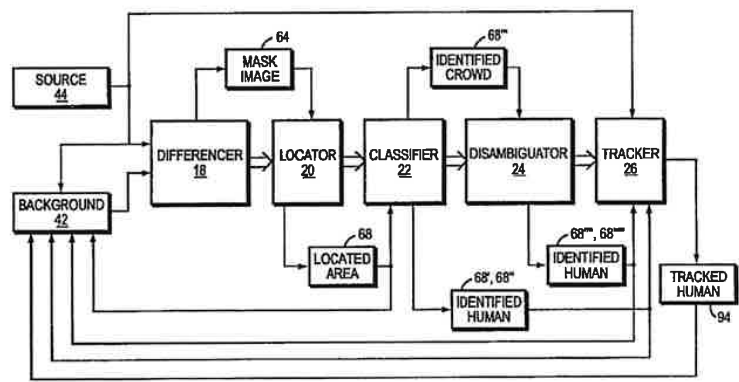
(List continued on next page.)

*Primary Examiner*—Jose L. Couso  
*Assistant Examiner*—Gregory Desire  
(74) *Attorney, Agent, or Firm*—Cesari and McKenna, LLP,  
Edwin H. Paul

(57) **ABSTRACT**

A technique for locating objects within an image is dis-  
closed. In one embodiment, the technique is realized by  
obtaining an image and then identifying an object within the  
image based upon an orientation of the object within the  
image. The image can be a representation of a plurality of  
pixels, wherein the plurality of pixels are arranged in a  
plurality of columns and rows, and wherein at least some of  
the plurality of pixels are enabled to represent the object.

**54 Claims, 14 Drawing Sheets**



OTHER PUBLICATIONS

Facial Feature Localization and Adaptation of a Generic Face Model for Model-Based Coding, Reinders, et al., *Signal Processing: Image Communication* vol. 7, pp. 57-74, 1995.

Real-time Recognition of Activity Using Temporal Templates, Aaron F Bobick, et al. *The Workshop on Applications of Computer Vision* Dec. 1996., pp. 1-5.

3D Human Body Model Acquisition from Multiple Views, Kakadiaris et al., IEEE, 1995, pp. 618-623.

Analyzing Articulated Motion Using Expectation-Maximization, Rowley, et al. *Computer Vision and Pattern Recognition* San Juan, PR, Jun. 1997, Total of 7 pages.

Mixture Models for Optical Flow Computation. Jepson, et al., University of Toronto. *Department of Computer Science*, Apr. 1993, pp. 1-16.

Analyzing and Recognizing Walking Figures in XYT, Niyogi, et al. IEEE 1994. pp. 469-474.

Nonparametric Recognition of Nonrigid Motion. Polana, et al, *Department of Computer Science*, pp. 1-29.

Model-Based Tracking of Self-Occluding Articulated Objects. Rehg, et al., *5th Intl. Conf. On Computer Vision* Cambridge, Ma, Jun. 1995 total of 6 pages.

A Unified Mixture Framework For Motion Segmentation: Incorporating Spatial Coherence and Estimating The Number of Models, Weiss, et al., IEEE 1996, pp. 321-326.

Learning visual Behavior for Gesture Analysis, Wilson, et al. IEEE 1995, pp. 229-234.

\* cited by examiner

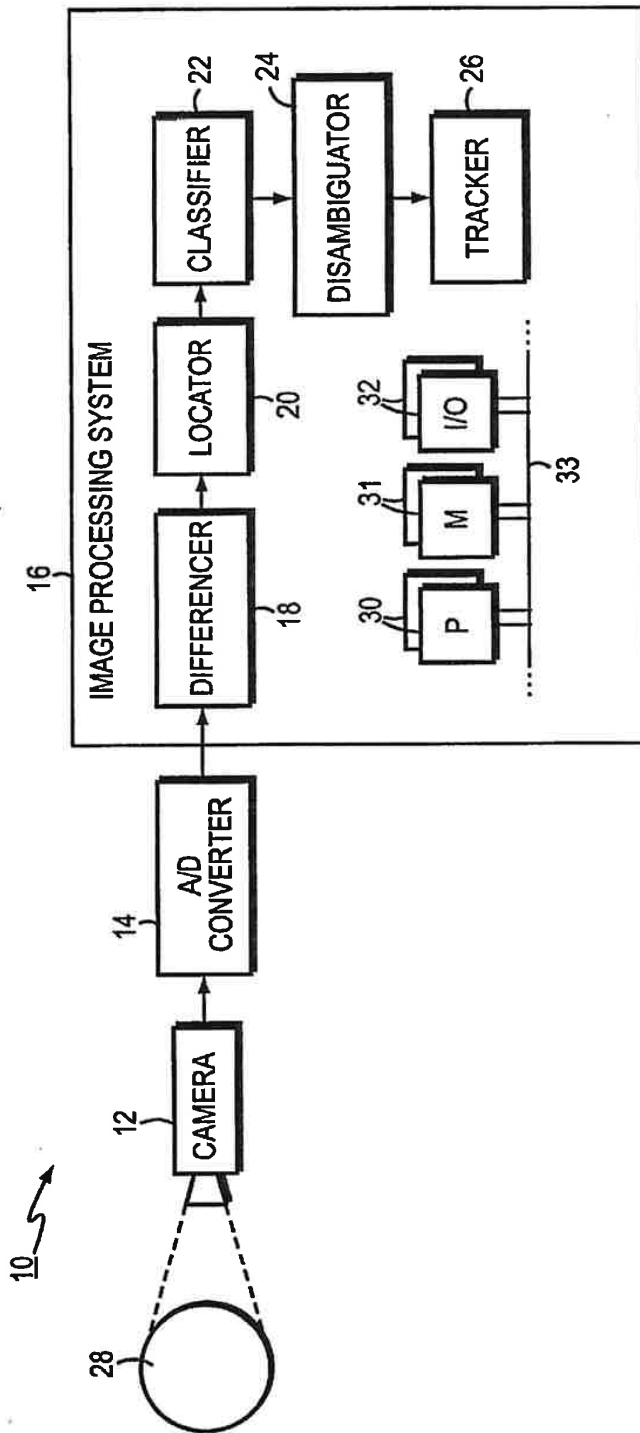


FIG. 1

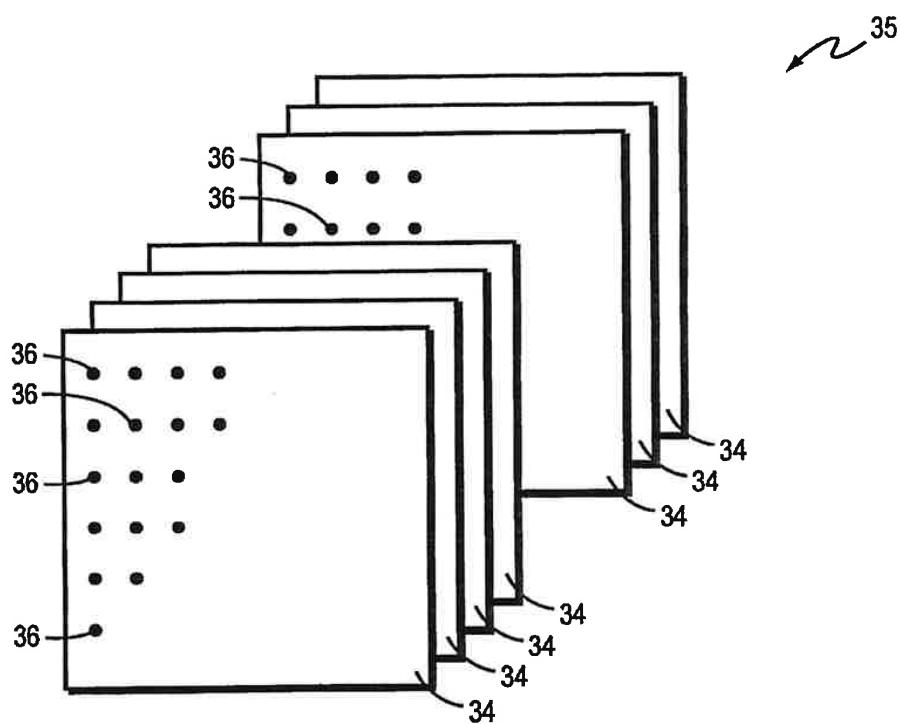


FIG. 2

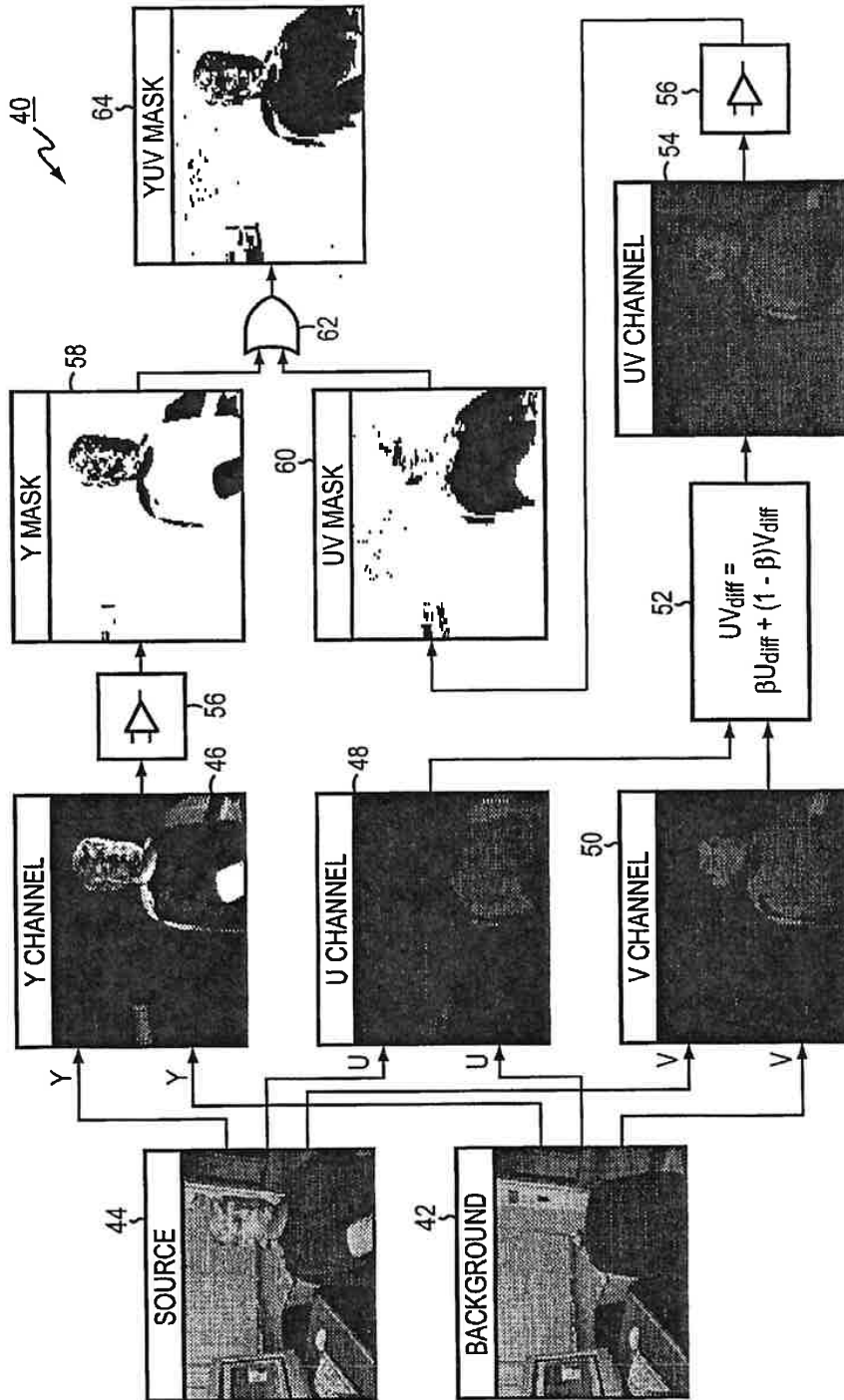


FIG. 3

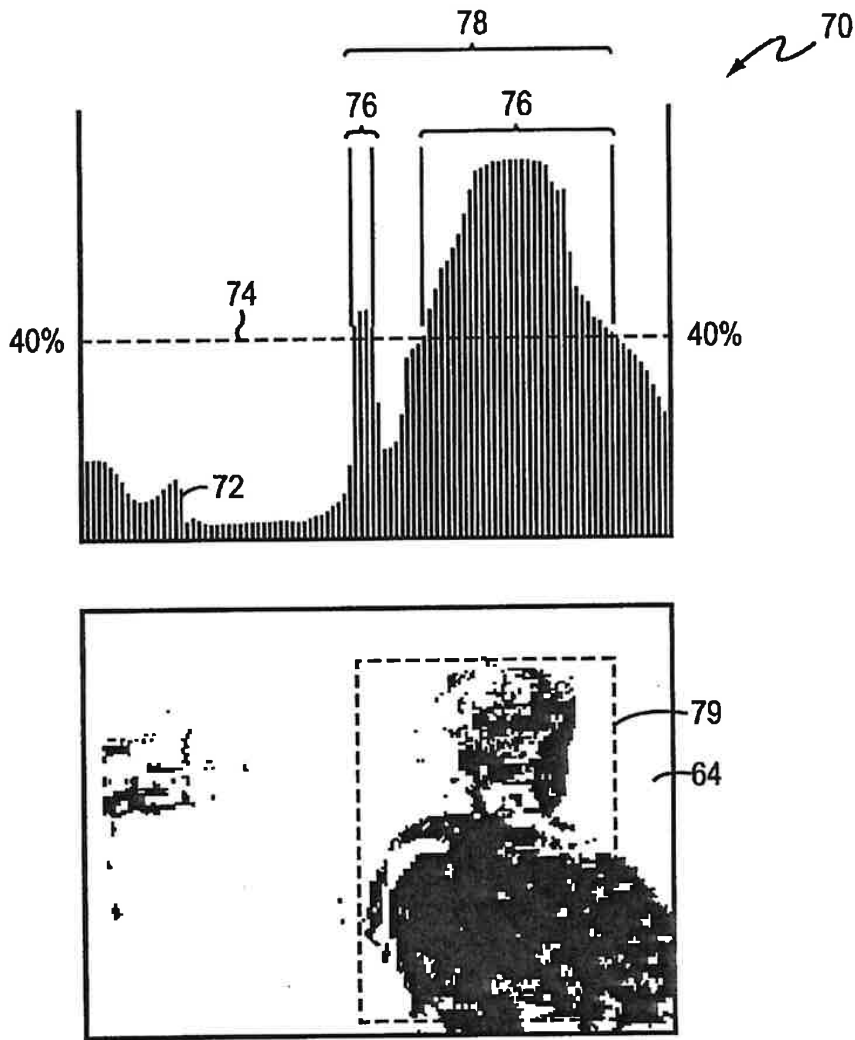


FIG. 4.

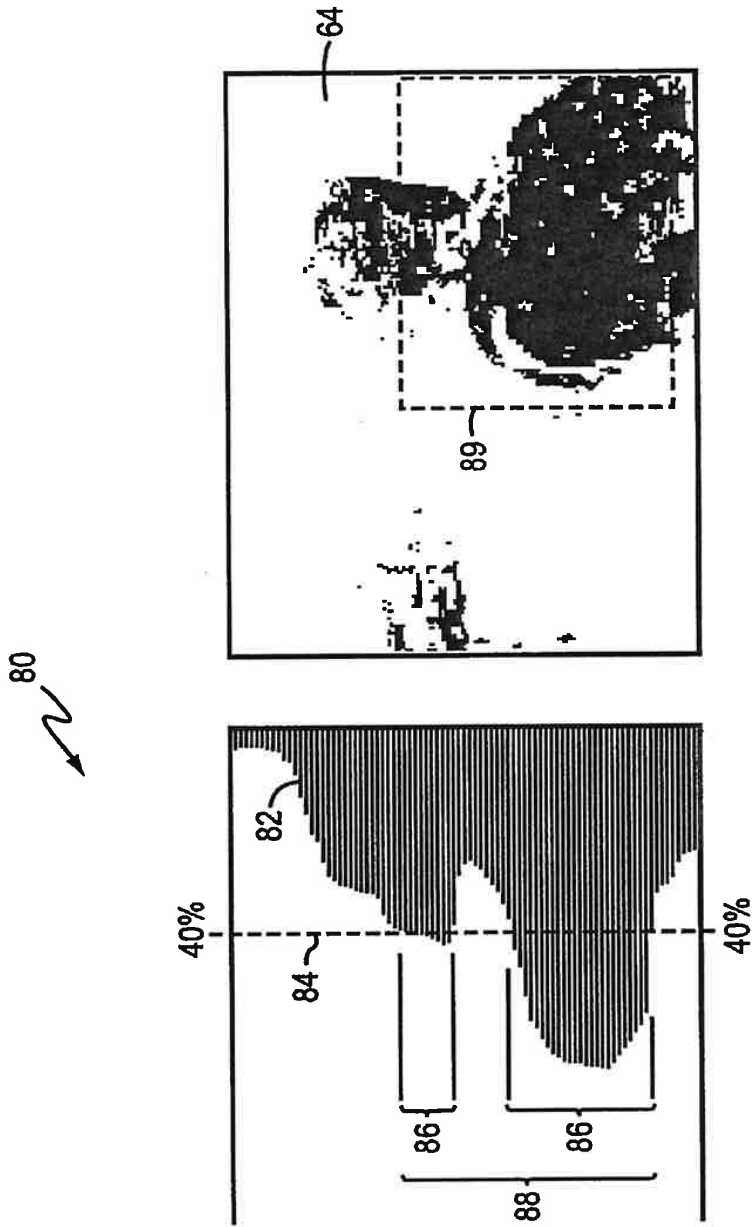


FIG. 5

80 ↘

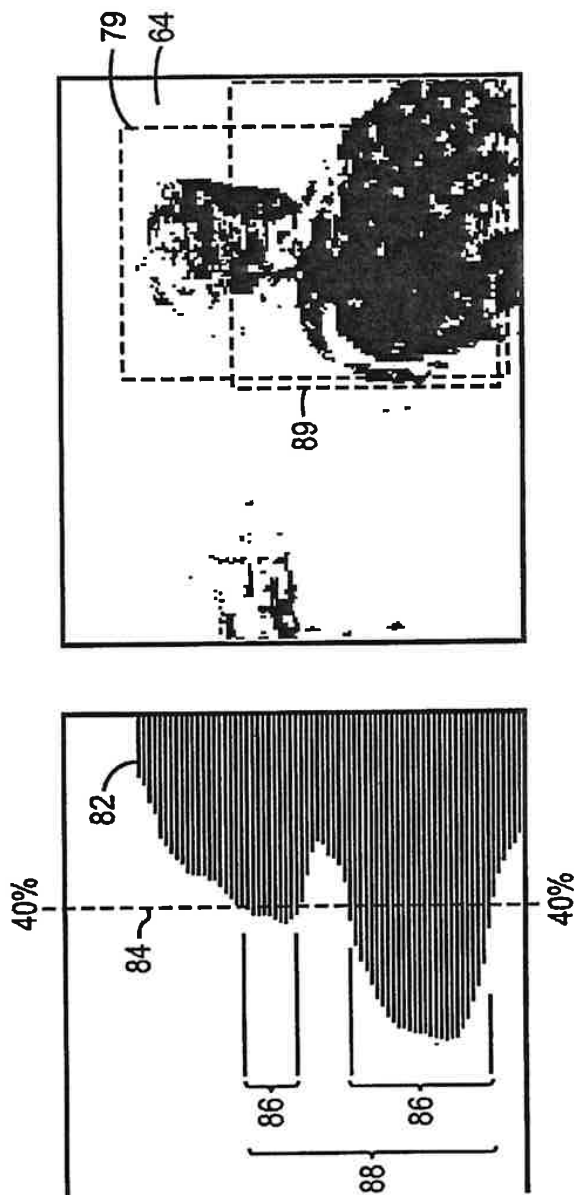


FIG. 6



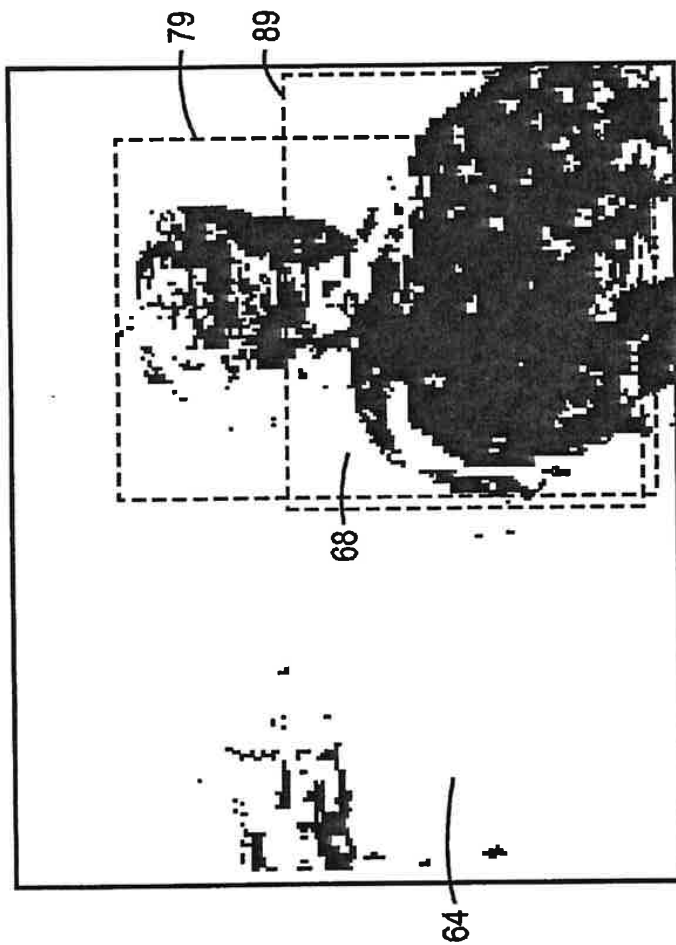


FIG. 7

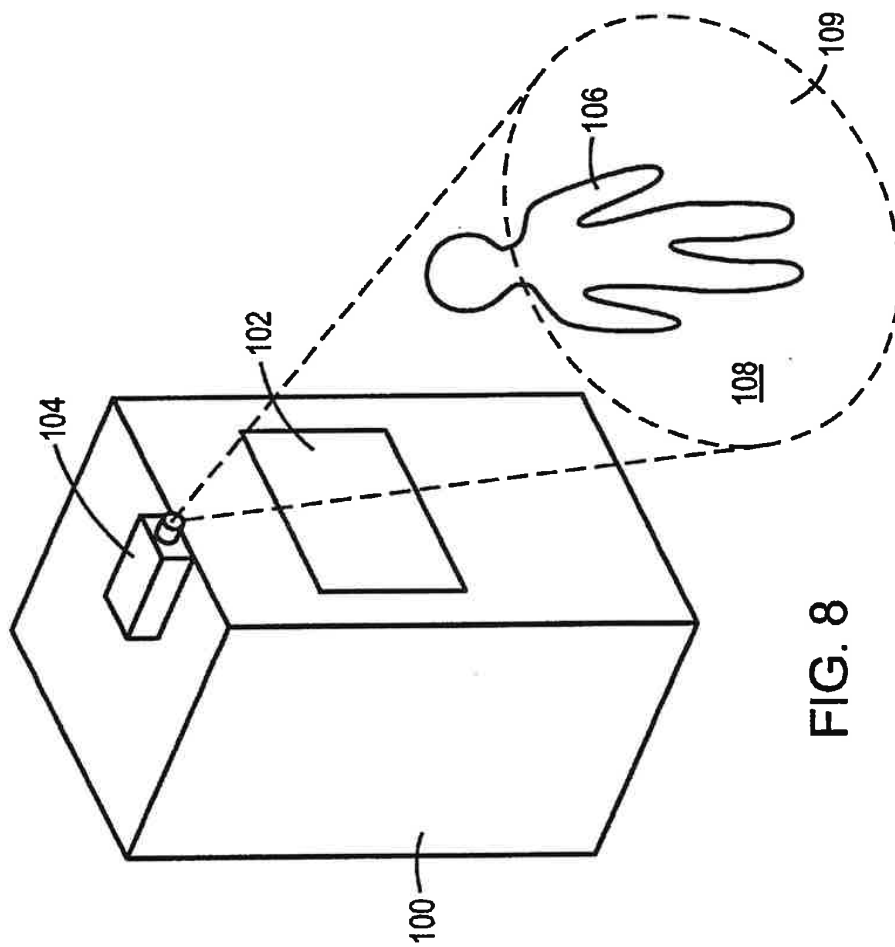


FIG. 8

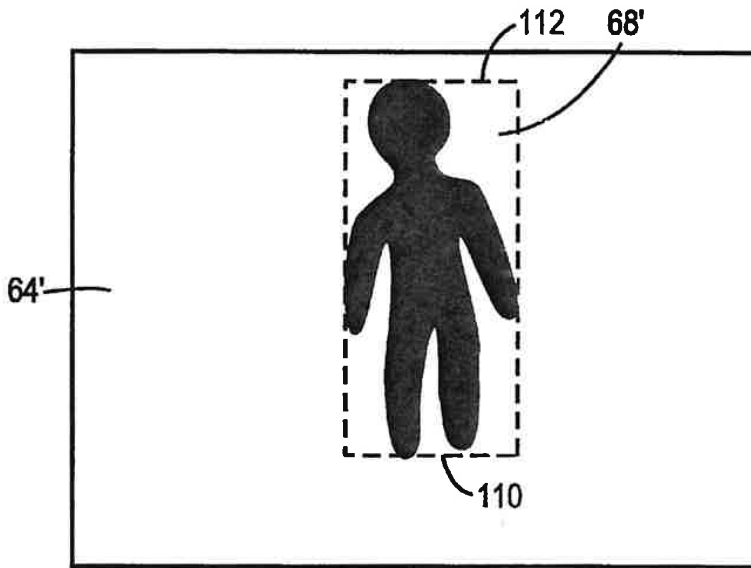


FIG. 9

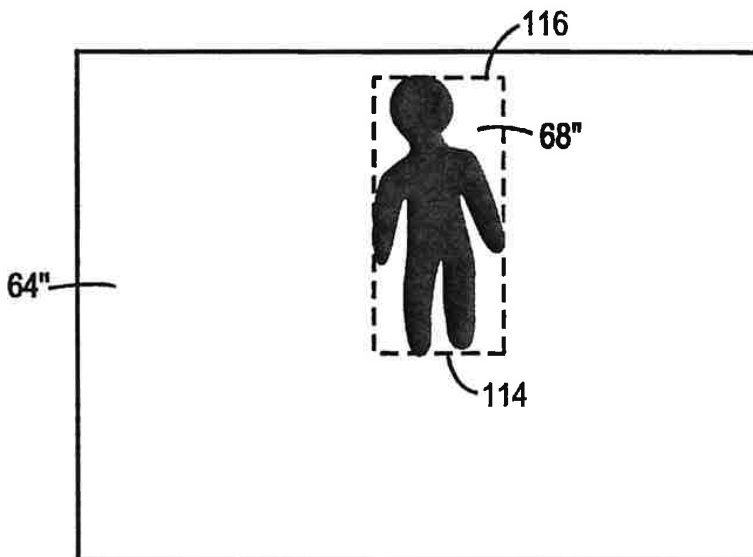


FIG. 10

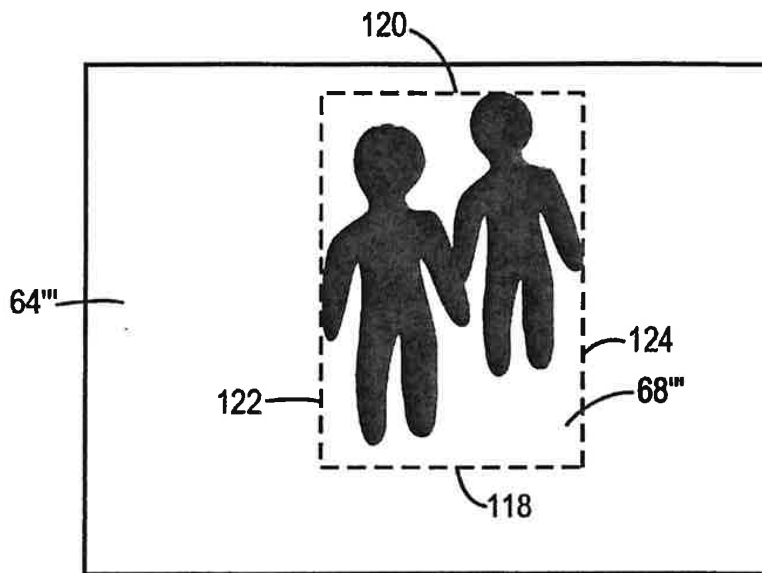


FIG. 11

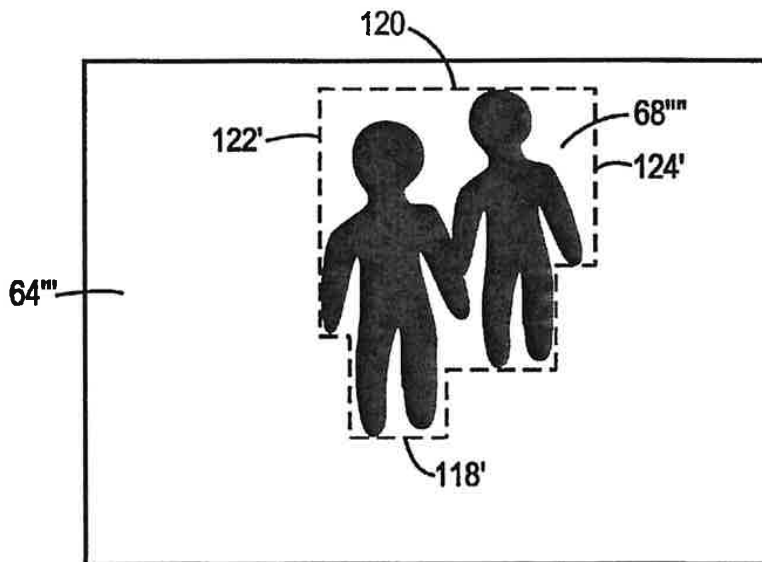


FIG. 12A

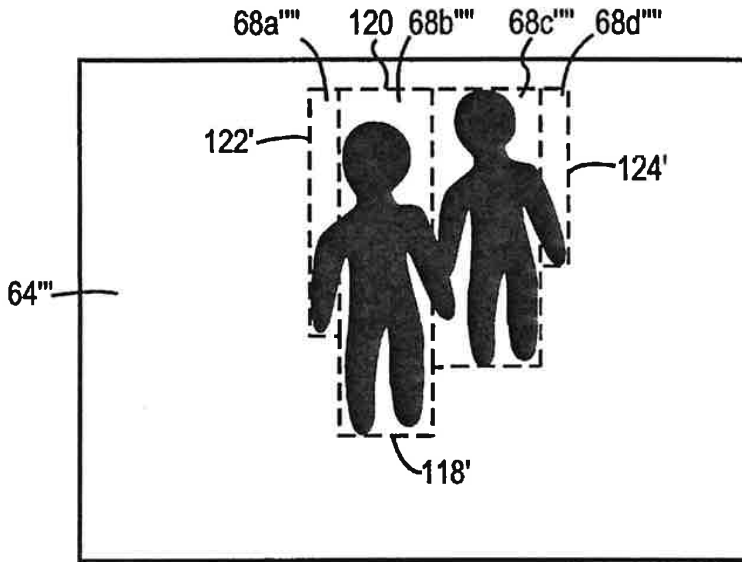


FIG. 12B

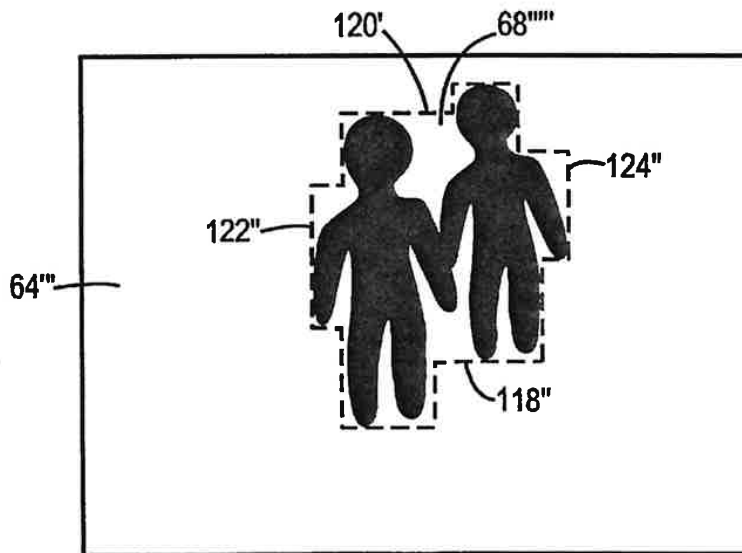


FIG. 13

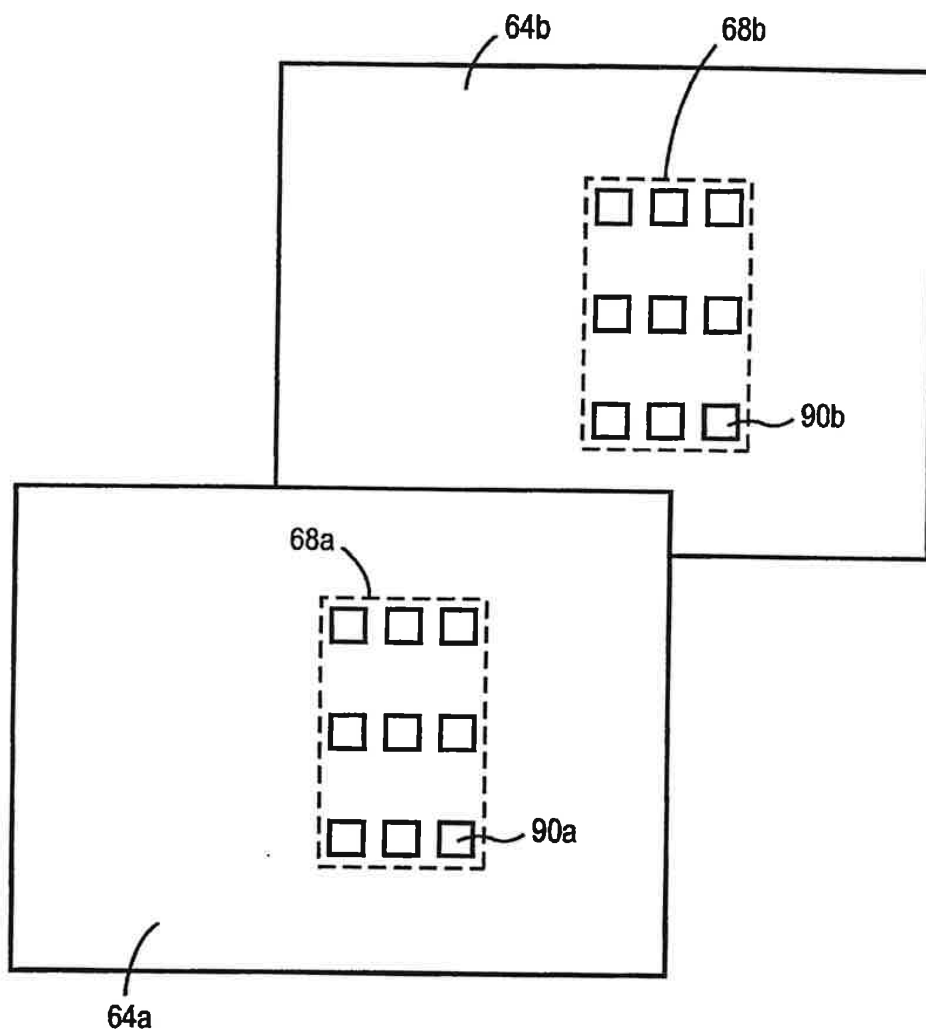


FIG. 14

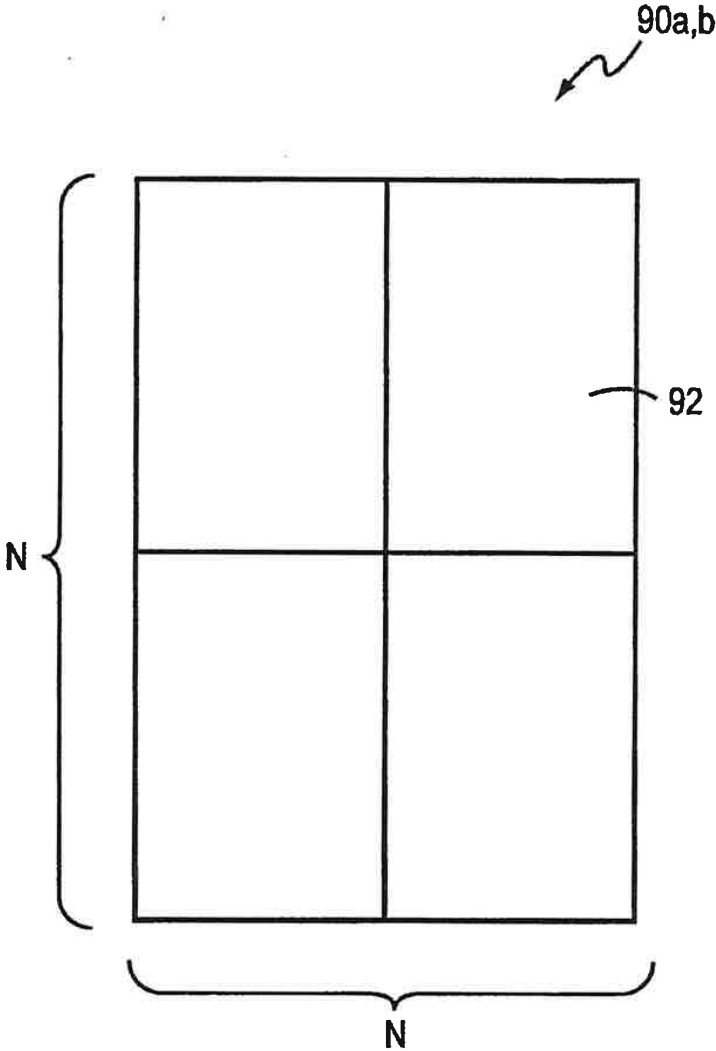


FIG. 15

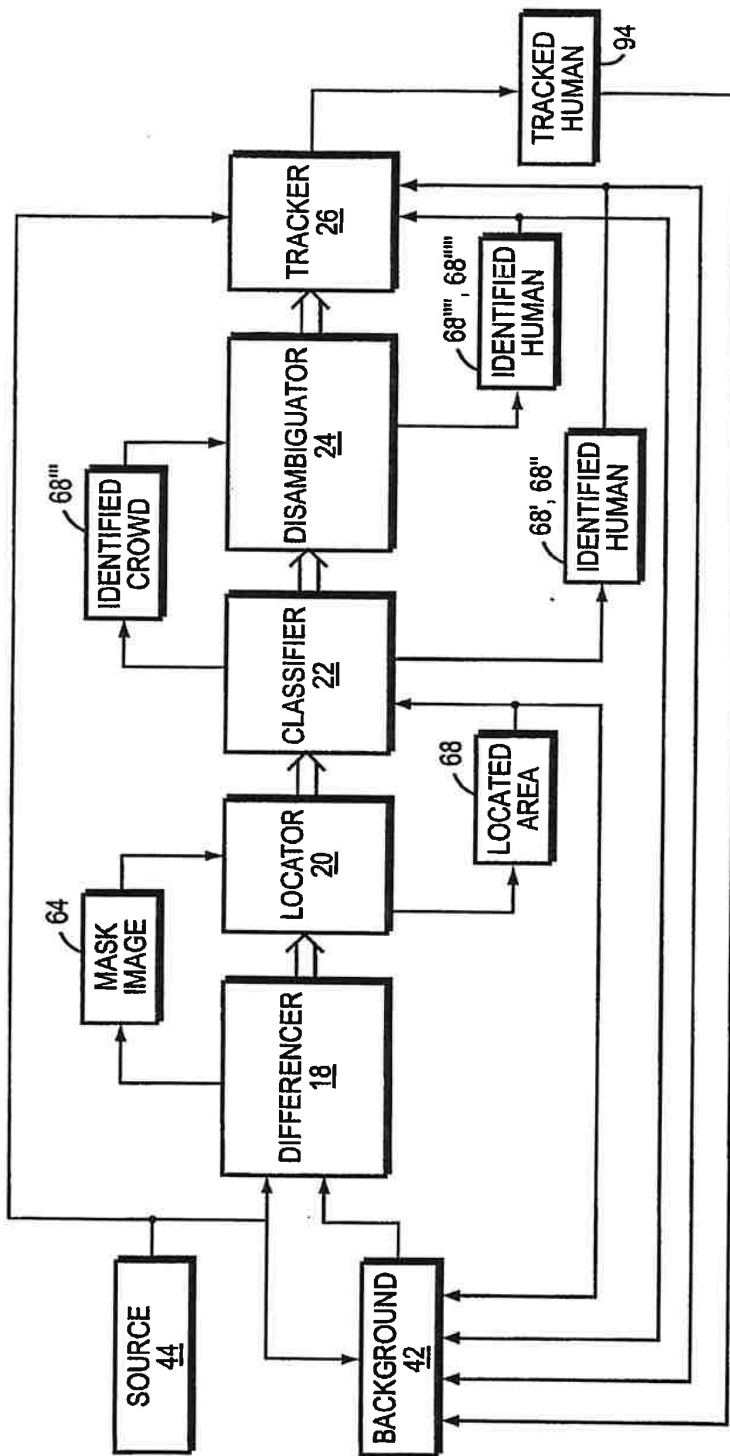


FIG. 16



## TECHNIQUE FOR LOCATING OBJECTS WITHIN AN IMAGE

### FIELD OF THE INVENTION

The present invention relates generally to visual recognition systems and, more particularly, to a technique for locating objects within an image.

### BACKGROUND OF THE INVENTION

An interface to an automated information dispensing kiosk represents a computing paradigm that differs from the conventional desktop environment. That is, an interface to an automated information dispensing kiosk differs from the traditional Window, Icon, Mouse and Pointer (WIMP) interface in that such a kiosk typically must detect and communicate with one or more users in a public setting. An automated information dispensing kiosk therefore requires a public multi-user computer interface.

Prior attempts have been made to provide a public multi-user computer interface and/or the constituent elements thereof. For example, a proposed technique for sensing users is described in "Pfinder: Real-time Tracking of the Human Body", Christopher Wren, Ali Azarbayejani, Trevor Darrell, and Alex Pentland, IEEE 1996. This technique senses only a single user, and addresses only a constrained virtual world environment. Because the user is immersed in a virtual world, the context for the interaction is straight-forward, and simple vision and graphics techniques are employed. Sensing multiple users in an unconstrained real-world environment, and providing behavior-driven output in the context of that environment present more complex vision and graphics problems which are not addressed by this technique.

Another proposed technique is described in "Real-time Self-calibrating Stereo Person Tracking Using 3-D Shape Estimation from Blob Features", Ali Azarbayejani and Alex Pentland, ICPR January 1996. The implementing system uses a self-calibrating blob stereo approach based on a Gaussian color blob model. The use of a Gaussian color blob model has a disadvantage of being inflexible. Also, the self-calibrating aspect of this system may be applicable to a desktop setting, where a single user can tolerate the delay associated with self-calibration. However, in an automated information dispensing kiosk setting, some form of advance calibration would be preferable so as to allow a system to function immediately for each new user.

Other proposed techniques have been directed toward the detection of users in video sequences. The implementing systems are generally based on the detection of some type of human motion in a sequence of video images. These systems are considered viable because very few objects move exactly the way a human does. One such system addresses the special case where people are walking parallel to the image plane of a camera. In this scenario, the distinctive pendulum-like motion of human legs can be discerned by examining selected scan-lines in a sequence of video images. Unfortunately, this approach does not generalize well to arbitrary body motions and different camera angles.

Another system uses Fourier analysis to detect periodic body motions which correspond to certain human activities (e.g., walking or swimming). A small set of these activities can be recognized when a video sequence contains several instances of distinctive periodic body motions that are associated with these activities. However, many body motions, such as hand gestures, are non-periodic, and in practice, even periodic motions may not always be visible to identify the periodicity.

Another system uses action recognition to identify specific body motions such as sitting down, waving a hand, etc. In this approach, a set of models for the actions to be recognized are stored and an image sequence is filtered using the models to identify the specific body motions. The filtered image sequence is thresholded to determine whether a specific action has occurred or not. A drawback of this system is that a stored model for each action to be recognized is required. This approach also does not generalize well to the case of detecting arbitrary human body motions.

Recently, an expectation-maximization (EM) technique has been proposed to model pixel movement using simple affine flow models. In this technique, the optical flow of images is segmented into one or more independent rigid body motion models of individual body parts. However, for the human body, movement of one body part tends to be highly dependent on the movement of other body parts. Treating the parts independently leads to a loss in detection accuracy.

The above-described proposed techniques either do not allow users to be detected in a real-world environment in an efficient and reliable manner, or do not allow users to be detected without some form of clearly defined user-related motion. These shortcomings present significant obstacles to providing a fully functional public multi-user computer interface. Accordingly, it would be desirable to overcome these shortcomings and provide a technique for allowing a public multi-user computer interface to detect users.

### OBJECTS OF THE INVENTION

The primary object of the present invention is to provide a technique for locating objects within an image.

The above-stated primary object, as well as other objects, features, and advantages, of the present invention will become readily apparent from the following detailed description which is to be read in conjunction with the appended drawings.

### SUMMARY OF THE INVENTION

According to the present invention, a technique for locating objects within an image is provided. The technique can be realized by having a processing device such as, for example, a digital computer, obtain an image. The processing device then identifies an object within the image based upon an orientation of the object within the image.

The orientation of the object within the image can be such that the object has a first orientation within the image. For example, if the object is a upright standing human, the first orientation is a vertical orientation.

The image can be, for example, a representation of a plurality of pixels, wherein at least some of the plurality of pixels are enabled to represent the object. The plurality of pixels can be configured to have a second orientation. For example, if the plurality of pixels are configured in a plurality of columns, the second orientation is a vertical orientation.

It should be noted that the first and second orientations do not have to be identical. For example, the first orientation could be a diagonal orientation, and the second orientation could be a horizontal orientation, or vice versa.

Regardless of the direction of orientation, the processing device can identify an object within the image by first counting each enabled pixel along the second orientation. The processing device can then identify portions of the representation having a quantity of enabled pixels exceeding a threshold value.

3

The processing device can further thereby identify an object within the image by first grouping together substantially adjacent identified portions of the representation. The processing device can then identify areas of the representation corresponding to each group of substantially adjacent identified portions of the representation.

The processing device can further thereby identify an object within the image by first recording the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of the representation. The processing device can then frame areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of the representation.

The plurality of pixels can also be configured to have a third orientation. For example, if the plurality of pixels are also configured in a plurality of rows, the third orientation is a horizontal orientation.

It should be noted that the second and third orientations should not be identical. For example, the second orientation and the third orientation could be orthogonal.

The processing device can further thereby identify an object within the image by first counting each enabled pixel in each framed area along the third orientation. The processing device can then identify portions of each framed area having a quantity of enabled pixels exceeding a threshold value.

The processing device can further thereby identify an object within the image by first grouping together substantially adjacent identified portions of each framed area. The processing device can then identify areas of each framed area corresponding to each group of substantially adjacent identified portions of each framed area.

The processing device can further thereby identify an object within the image by first recording the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of each framed area. The processing device can then frame areas of each framed area coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of each framed area.

In a more specific embodiment, the plurality of pixels can be arranged in a plurality of columns and rows. If such is the case, the processing device can thereby identify an object within the image by first counting each enabled pixel in each of the plurality of columns and rows. The processing device can then identify each of the plurality of columns having a quantity of enabled pixels exceeding a column threshold value, and identify each of the plurality of rows having a quantity of enabled pixels exceeding a row threshold value.

The processing device can further thereby identify an object within the image by first grouping together substantially adjacent identified columns, and grouping together substantially adjacent identified rows. The processing device can then identify areas of the representation corresponding to each group of substantially adjacent identified columns, and identify areas of the representation corresponding to each group of substantially adjacent identified rows.

The processing device can further thereby identify an object within the image by first recording the locations of the outermost enabled pixels within each group of substantially adjacent identified columns, and recording the locations of the outermost enabled pixels within each group of substantially adjacent identified rows. The processing device can then frame areas of the representation coinciding with the locations of the outermost enabled pixels within each group

4

of substantially adjacent identified columns, and frame areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified rows.

The processing device can further thereby identify an object within the image by first overlaying the areas of the representation that were framed to coincide with the locations of the outermost enabled pixels within each group of substantially adjacent identified columns with the areas of the representation that were framed to coincide with the locations of the outermost enabled pixels within each group of substantially adjacent identified rows. The processing device can then identify common overlaid areas as areas of the representation that contain a significant number of enabled pixels.

The image can be a first representation of a plurality of first pixels representing a difference between a second representation of a plurality of second pixels and a third representation of a plurality of third pixels, wherein each of the plurality of first pixels is enabled to represent a difference between a corresponding one of the plurality of second pixels and a corresponding one of the plurality of third pixels, wherein the object is represented by at least some of the enabled first pixels.

The first representation can be, for example, a first electrical representation of a mask image that indicates the difference between corresponding pixels in the second and third plurality of pixels. The first electrical representation can be stored, for example, as digital data on a tape, disk, or other memory device for manipulation by the processing device.

The second representation can be, for example, a second electrical representation of an image of a scene that is captured by a camera at a first point in time and then digitized to form the plurality of second pixels. The second electrical representation can be stored on the same or another memory device for manipulation by the processing device.

The third representation can be, for example, a third electrical representation of an image of the scene that is captured by a camera at a second point in time and then digitized to form the plurality of third pixels. The third electrical representation can be stored on the same or another memory device for manipulation by the processing device.

Thus, the first representation typically represents a difference in the scene at the first point in time as compared to is the scene at the second point in time.

#### BRIEF DESCRIPTION OF THE DRAWINGS

In order to facilitate a fuller understanding of the present invention, reference is now made to the appended drawings. These drawings should not be construed as limiting the present invention, but are intended to be exemplary only.

FIG. 1 is a schematic diagram of a vision system in accordance with the present invention.

FIG. 2 shows a video sequence of temporally ordered frames which are organized as arrays of pixels.

FIG. 3 is a flowchart diagram of a differencing algorithm in accordance with the present invention.

FIG. 4 shows a vertical histogram for a YUV-mask image in accordance with the present invention.

FIG. 5 shows a first embodiment of a horizontal histogram for a YUV-mask image in accordance with the present invention.

5

FIG. 6 shows a second embodiment of a horizontal histogram for a YUV-mask image in accordance with the present invention.

FIG. 7 shows overlaid frames on a YUV-mask image in accordance with the present invention.

FIG. 8 shows a public kiosk having an interactive touch-screen monitor and a video camera in accordance with the present invention.

FIG. 9 shows a first area in a YUV-mask image in accordance with the present invention.

FIG. 10 shows a second area in a YUV-mask image in accordance with the present invention.

FIG. 11 shows a YUV-mask image having an area that was classified as an area containing more than one human in accordance with the present invention.

FIG. 12A shows a YUV-mask image having a first redefined area in accordance with the present invention.

FIG. 12B shows a YUV-mask image having a divided first redefined area in accordance with the present invention.

FIG. 13 shows a YUV-mask image having a second redefined area in accordance with the present invention.

FIG. 14 shows a sampled area in a current YUV-mask image and a prior YUV-mask image in accordance with the present invention.

FIG. 15 shows an NxN color sample in accordance with the present invention.

FIG. 16 is a data flow diagram for a vision system in accordance with the present invention.

#### DETAILED DESCRIPTION OF THE INVENTION

Referring to FIG. 1, there is shown a schematic diagram of a vision system 10 in accordance with the present invention. The vision system 10 comprises a camera 12 which is coupled to an optional analog-to-digital (A/D) converter 14. The optional A/D converter 14 is coupled to an image processing system 16. The image processing system 16 comprises a differencer 18, a locator 20, a classifier 22, a disambiguator 24, and a tracker 26.

The camera 12 may be of a conventional analog variety, or it may be of a digital type. If the camera 12 is a digital type of camera, then the optional A/D converter 14 is not required. In either case, the camera 12 operates by capturing an image of a scene 28. A digitized version of the captured image of the scene 28 is then provided to the image processing system 16.

The differencer 18, the locator 20, the classifier 22, the disambiguator 24, and the tracker 26 are preferably implemented as software programs in the image processing system 16. Thus, the image processing system 16 also preferably comprises at least one processor (P) 30, memory (M) 31, and input/output (I/O) interface 32, which are connected to each other by a bus 33, for implementing the functions of the differencer 18, the locator 20, the classifier 22, the disambiguator 24, and the tracker 26.

As previously mentioned, the camera 12 captures an image of the scene 28 and a digitized version of the captured image is provided to the image processing system 16. Referring to FIG. 2, the digitized version of each captured image takes the form of a frame 34 in a video sequence of temporally ordered frames 35. The video sequence of temporally ordered frames 35 may be produced, for example, at a rate of thirty per second. Of course, other rates may alternatively be used.

6

Each frame 34 is organized as an array of pixels 36. Each pixel 36 has a light intensity value for a corresponding portion of the captured image of the scene 28. The pixels 36 may have color values, although the present invention may also be practiced with the pixels 36 not having color values. Typically, the value of each pixel 36 is stored as digital data on a tape, disk, or other memory device, such as the memory 31, for manipulation by the image processing system 16.

The video sequence of temporally ordered frames 35 is presented to the image processing system 16 via the I/O interface 32. The digital data representing the value of each pixel 36 may be stored in the memory 31 at an address that corresponds to the location of each pixel 36 in a corresponding frame 34. Machine executable instructions of operating system and application software programs, which may also be stored in the memory 31, are executed by the processor 30 to manipulate the digital data representing the value of each pixel 36. Thus, in the preferred embodiment described herein, the functions of the differencer 18, the locator 20, the classifier 22, the disambiguator 24, and the tracker 26 are implemented by the processor 30 through the execution of machine executable instructions, as described in detail below.

In the preferred embodiment described herein, the vision system 10 is used to identify a person in a single digitized image, and then track the person through a succession of digitized images. It should be noted, however, that the vision system 10 can be used to identify essentially any type of object in a single digitized image, and then track the object through a succession of digitized images. The vision system 10 accomplishes these tasks in part through the use of a background-differencing algorithm which uses luminance and chrominance information, as described in detail below.

The differencer 18 operates by storing a "background" image and then comparing each subsequently stored "source" image to the background image. The background image and the source images are digitized versions of images of the scene 28 that are captured by the camera 12. Thus, the background image and the source images make up the frames 34 that make up the video sequence of temporally ordered frames 35.

The background image forms a default or base image to which all of the source images are compared. In its simplest form, the background image can be an image that is captured when it is known that no extraneous objects (e.g., a person) are within the field of view of the camera 12. However, the background image is more typically formed by averaging together a number of source images (e.g., the last ten captured source images). This allows the background image to be continuously updated every time a new source image is captured (e.g., every 5 seconds), which allows environmental changes, such as subtle changes in lighting conditions, to be gradually incorporated into the background image.

The above-described time-averaged background image updating scheme also allows more prominent changes to be gradually incorporated, or not incorporated, into the background image. That is, if the vision system 10 determines, through a differencing algorithm that is described in detail below, that there are extraneous objects (e.g., a person or a potted plant) within the field of view of the camera 12, and hence within one or more captured source images, then the background image can be selectively updated to gradually incorporate, or not incorporate, these extraneous objects into the background image. For example, if the vision system 10 determines, through the differencing algorithm that is

described in detail below, that there is an extraneous object (e.g., a person or a potted plant) within the field of view of the camera 12, and hence within one or more captured source images, then the background image is updated without using the area in each captured source image that corresponds to the extraneous object. That is, the background image is selectively updated to not incorporate the extraneous object into the background image.

If at some later time the vision system 10 determines, through a classifying, a disambiguating, or a tracking algorithm that is described in detail below, that the extraneous object is not an object of interest (e.g., a potted plant), then the background image is updated using the area in each captured source image that corresponds to the extraneous object to gradually incorporate the extraneous object into the background image. That is, the background image is selectively updated to gradually incorporate the extraneous object into the background image.

On the other hand, if at some later time the vision system 10 determines, through the classifying, the disambiguating, or the tracking algorithms that are described in detail below, that the extraneous object is an object of interest (e.g., a person), then the background image continues to be updated without using the area in each captured source image that corresponds to the extraneous object. That is, the background image continues to be selectively updated so as to not incorporate the extraneous object into the background image. For example, an object may be considered an object of interest if the object has moved within a preselected amount of time.

At this point it should be noted that in all of the above-described time-averaged background image updating scenarios, the background image is always updated using the areas in each captured source image that do not correspond to the extraneous object. Also, the above-described time-averaged background image updating scheme allows certain objects to "fade" from within the background image. For example, if an object was present within one or more prior captured source images, but is no longer present within more recent captured source images, then as the number of more recent captured source images within which the object is no longer present increases with time, the object will fade from within the background image as more of the more recent captured source images are averaged together to form the background image.

Source images can be captured by the camera 12 at literally any time, but are typically captured by the camera 12 subsequent to the capturing, or forming, of the background image. Source images often contain extraneous objects (e.g., a person) which are to be identified and tracked.

As previously mentioned, the differencer 18 operates by comparing each source image to the background image. Each frame 34 in the video sequence of temporally ordered frames 35, including the background image and all of the source images, is in YUV color space. YUV color space is a standard used by, for example, television cameras. The Y-component corresponds to the brightness or luminance of an image, the U-component corresponds to the relative amount of blue light that is in an image, and the V-component corresponds to the relative amount of red light that is in an image. Together, the U and V components specify the chrominance of an image.

Referring to FIG. 3, there is shown a flowchart diagram of a differencing algorithm 40 in accordance with the present invention. A background image 42 and a source image 44 are

both provided in YUV format. The individual Y, U, and V components are extracted from both the background image 42 and the source image 44. The individual Y, U, and V components from the background image 42 and the source image 44 are then differenced to form corresponding Y, U, and V difference images. That is, a Y-difference image 46 is formed by subtracting the Y-component value for each pixel in the background image 42 from the Y-component value for a corresponding pixel in the source image 44, a U-difference image 48 is formed by subtracting the U-component value for each pixel in the background image 42 from the U-component value for a corresponding pixel in the source image 44, and a V-difference image 50 is formed by subtracting the V-component value for each pixel in the background image 42 from the V-component value for a corresponding pixel in the source image 44. The value of each resulting pixel in the Y, U, and V difference images may be negative or positive.

Next, a weighting operation 52 is performed on corresponding pixels in the U-difference image 48 and the V-difference image 50. That is, a weighted average is computed between corresponding pixels in the U-difference image 48 and the V-difference image 50. This results in a UV-difference image 54. The formula used for each pixel is as follows:

$$UV_{diff} = \beta U_{diff} + (1 - \beta) V_{diff} \quad (1)$$

wherein the value of  $\beta$  is between 0 and 1. Typically, a  $\beta$ -value of approximately 0.25 is used, resulting in a greater weight being given to the V-component than the U-component. This is done for two reasons. First, human skin contains a fair amount of red pigment, so humans show up well in V color space. Second, the blue light component of most cameras is noisy and, consequently, does not provide very clean data.

Next, a thresholding operation 56 is performed on each pixel in the Y-difference image 46 and the UV-difference image 54. That is, the value of each pixel in the Y-difference image 46 and the UV-difference image 54 is thresholded to convert each pixel to a boolean value corresponding to either "on" or "off". A separate threshold value may be selected for both the Y-difference image 46 and the UV-difference image 54. Each threshold value may be selected according to the particular object (e.g., a person) to be identified by the vision system 10. For example, a high threshold value may be selected for the Y-difference image 46 if the object (e.g., a person) to be identified is known to have high luminance characteristics.

The result of thresholding each pixel in the Y-difference image 46 and the UV-difference image 54 is a Y-mask image 58 and a UV-mask image 60, respectively. Literally, the Y-mask image 58 represents where the source image 44 differs substantially from the background image 42 in luminance, and the UV-mask image 60 represents where the source image 44 differs substantially from the background image 42 in chrominance.

Next, a boolean "OR" operation 62 is performed on corresponding pixels in the Y-mask image 58 and the UV-mask image 60. That is, each pixel in the Y-mask image 58 is boolean "OR" functioned together with a corresponding pixel in the UV-mask image 60. This results in a combined YUV-mask image 64. The YUV-mask image 64 represents where the source image 44 differs substantially in luminance and chrominance from the background image 42. More practically, the YUV-mask image 64 shows where the source image 44 has changed from the background image 42. This change can be due to lighting changes in the scene

28 (e.g., due to a passing cloud), objects entering or exiting the scene 28 (e.g., people, frisbees, etc.), or objects in the scene 28 that change visually (e.g., a computer monitor running a screen saver). In the preferred embodiment described herein, the change corresponds to the presence of a human.

The locator 20 operates by framing areas in the YUV-mask image 64 using a thresholding scheme, and then overlaying the framed areas to locate specific areas in the YUV-mask image 64 that represent where the source image 44 differs substantially in luminance and chrominance from the background image 42, as determined by the differencer 18. The specific areas are located, or identified, based upon an orientation of each area in the YUV-mask image 64.

Referring to FIG. 4, the locator 20 first divides the YUV-mask image 64 into vertical columns (not shown for purposes of figure clarity) and then counts the number of pixels that are turned "on" in each column of the YUV-mask image 64. The locator 20 uses this information to form a vertical histogram 70 having vertical columns 72 which correspond to the vertical columns of the YUV-mask image 64. The height of each column 72 in the vertical histogram 70 corresponds to the number of pixels that are turned "on" in each corresponding column of the YUV-mask image 64.

Next, the locator 20 thresholds each column 72 in the vertical histogram 70 against a selected threshold level 74. That is, the height of each column 72 in the vertical histogram 70 is compared to the threshold level 74, which in this example is shown to be 40%. Thus, if more than 40% of the pixels in a column of the YUV-mask image 64 are turned "on", then the height of the corresponding column 72 in the vertical histogram 70 exceeds the 40% threshold level 74. In contrast, if less than 40% of the pixels in a column of the YUV-mask image 64 are turned "on", then the height of the corresponding column 72 in the vertical histogram 70 does not exceed the 40% threshold level 74.

Next, the locator 20 groups adjacent columns in the vertical histogram 70 that exceed the threshold level into column sets 76. The locator 20 then joins column sets that are separated from each other by only a small gap to form merged column sets 78. The locator 20 then records the vertical limits of each remaining column set. That is, the location of the highest pixel that is turned "on" in a column of the YUV-mask image 64 that corresponds to a column 72 in a column set of the vertical histogram 70 is recorded. Similarly, the location of the lowest pixel that is turned "on" in a column of the YUV-mask image 64 that corresponds to a column 72 in a column set of the vertical histogram 70 is recorded.

Next, the locator 20 places a frame 79 around each area in the YUV-mask image 64 that is defined by the outermost columns that are contained in each column set of the vertical histogram 70, and by the highest and lowest pixels that are turned "on" in each column set of the vertical histogram 70. Each frame 79 therefore defines an area in the YUV-mask image 64 that contains a significant number of pixels that are turned "on", as determined in reference to the threshold level 74.

Referring to FIG. 5, the locator 20 repeats the above-described operations, but in the horizontal direction. That is, the locator 20 first divides the YUV-mask image 64 into horizontal rows (not shown for purposes of figure clarity) and then counts the number of pixels that are turned "on" in each row of the YUV-mask image 64. The locator 20 uses this information to form a horizontal histogram 80 having horizontal rows 82 which correspond to the horizontal rows of the YUV-mask image 64. The length of each row 82 in the

horizontal histogram 80 corresponds to the number of pixels that are turned "on" in each corresponding row of the YUV-mask image 64.

Next, the locator 20 thresholds each row 82 in the horizontal histogram 80 against a selected threshold level 84. That is, the length of each row 82 in the horizontal histogram 80 is compared to the threshold level 84, which in this example is shown to be 40%. Thus, if more than 40% of the pixels in a row of the YUV-mask image 64 are turned "on", then the length of the corresponding row 82 in the horizontal histogram 80 exceeds the 40% threshold level 84. In contrast, if less than 40% of the pixels in a row of the YUV-mask image 64 are turned "on", then the length of the corresponding row 82 in the horizontal histogram 80 does not exceed the 40% threshold level 84.

Next, the locator 20 groups adjacent rows in the horizontal histogram 80 that exceed the threshold level into row sets 86. The locator 20 then joins row sets that are separated from each other by only a small gap to form merged row sets 88. The locator 20 then records the horizontal limits of each remaining row set. That is, the location of the leftmost pixel that is turned "on" in a row of the YUV-mask image 64 that corresponds to a row 82 in a row set of the horizontal histogram 80 is recorded. Similarly, the location of the rightmost pixel that is turned "on" in a row of the YUV-mask image 64 that corresponds to a row 82 in a row set of the horizontal histogram 80 is recorded.

Next, the locator 20 places a frame 89 around each area in the YUV-mask image 64 that is defined by the outermost rows that are contained in each row set of the horizontal histogram 80, and by the leftmost and rightmost pixels that are turned "on" in each row set of the horizontal histogram 80. Each frame 89 therefore defines an area in the YUV-mask image 64 that contains a significant number of pixels that are turned "on", as determined in reference to the threshold level 84.

At this point it should be noted that the locator 20 may alternatively perform the horizontal histogramming operation described above on only those areas in the YUV-mask image 64 that have been framed by the locator 20 during the vertical histogramming operation. For example, referring to FIG. 6, the locator 20 can divide the YUV-mask image 64 into horizontal rows (not shown for purposes of figure clarity) in only the area defined by the frame 79 that was obtained using the vertical histogram 70. The locator 20 can then proceed as before to count the number of pixels that are turned "on" in each row of the YUV-mask image 64, to form the horizontal histogram 80 having horizontal rows 82 which correspond to the horizontal rows of the YUV-mask image 64, to threshold each row 82 in the horizontal histogram 80 against a selected threshold level 84, to group adjacent rows in the horizontal histogram 80 that exceed the threshold level into row sets 86 and merged row sets 88, and to place a frame 89 around each area in the YUV-mask image 64 that is defined by the outermost rows that are contained in each row set of the horizontal histogram 80, and by the leftmost and rightmost pixels that are turned "on" in each row set of the horizontal histogram 80. By performing the horizontal histogramming operation on only those areas in the YUV-mask image 64 that have been framed by the locator 20 during the vertical histogramming operation, the locator 20 eliminates unnecessary processing of the YUV-mask image 64.

Referring to FIG. 7, the locator 20 next overlays the frames 79 and 89 that were obtained using the vertical histogram 70 and the horizontal histogram 80, respectively, to locate areas 68 that are common to the areas defined by

the frames 69 and 70. The locations of these common areas 68, of which only one is shown in this example, are the locations of areas in the YUV-mask image 64 that represent where the source image 44 differs substantially in luminance and chrominance from the background image 42, as determined by the differencer 18. In the preferred embodiment described herein, these areas 68 are likely to contain a human.

It should be noted that although the locator 20, as described above, divides the YUV-mask image 64 into vertical columns and horizontal rows, it is within the scope of the present invention to have the locator 20 divide the YUV-mask image 64 in any number of manners. For example, the locator 20 can divide the YUV-mask image 64 into diagonal sections, and then count the number of pixels that are turned "on" in each diagonal section of the YUV-mask image 64. Thus, it is within the scope of the present invention that the above described columns and rows can be oriented in any number of directions besides just the vertical and horizontal directions described above.

The classifier 22 operates by filtering each area 68 in the YUV-mask image 64 that was located by the locator 20 for human characteristics. More specifically, the classifier 22 operates by filtering each area 68 in the YUV-mask image 64 for size, location, and aspect ratio. In order for the classifier 22 to perform the filtering operation, the position and the orientation of the camera 12 must be known. For example, referring to FIG. 8, there is shown a public kiosk 100 having an interactive touchscreen monitor 102 mounted therein and a video camera 104 mounted thereon. The interactive touchscreen monitor 102 provides an attraction for a passing client 106, while the video camera 104 allows the passing client 106 to be detected in accordance with the present invention. The video camera 104 is mounted at an angle on top of the public kiosk 100 such that the field of view of the video camera 104 encompasses a region 108 in front of the public kiosk 100. The region 108 includes the terrain 109 upon which the passing client 106 is standing or walking. The terrain 109 provides a reference for determining the size and location of the passing client 106, as described in detail below.

Referring to FIG. 9, if the passing client 106 is a six-foot tall human standing approximately three feet away from the public kiosk 100, then the passing client 106 will show up as an area 68" in a YUV-mask image 64" having a bottom edge 110 located at the bottom of the YUV-mask image 64" and a top edge 112 located at the top of the YUV-mask image 64". On the other hand, referring to FIG. 10, if the passing client 106 is a six-foot tall human standing approximately twenty feet away from the public kiosk 100, then the passing client 106 will show up as an area 68" in a YUV-mask image 64" having a bottom edge 114 located in the middle of the YUV-mask image 64" and a top edge 116 located at the top of the YUV-mask image 64".

With the position and the orientation of the video camera 104 known, as well as the size and the location of an area 68 within a YUV-mask image 64, calculations can be made to determine the relative size and location (e.g., relative to the public kiosk 100) of an object (e.g., the client 106) that was located by the locator 20 and is represented by an area 68 in a YUV-mask image 64. That is, given the position and the orientation of the video camera 104 and the location of the bottom edge of an area 68 in a YUV-mask image 64, a first calculation can be made to obtain the distance (e.g., in feet and inches) between the public kiosk 100 and the object (e.g., the client 106) that was located by the locator 20 and is represented by the area 68 in the YUV-mask image 64.

Given the distance between the public kiosk 100 and the object, as well as the size of the area 68 in a YUV-mask image 64, a second calculation can be made to obtain the actual size of the object (e.g., in feet and inches). At this point, three useful characteristics are known about the object: the distance between the public kiosk 100 and the object (in feet and inches), the height of the object (in feet and inches), and the width of the object (in feet and inches).

The classifier 22 can now filter each area 68 in the YUV-mask image 64 for size, location, and aspect ratio. For example, assuming that there is only an interest in identifying humans over the height of four feet, the classifier 22 will filter out those objects that are shorter than four feet in height. Also, assuming that there is only an interest in identifying humans who come within ten feet of the public kiosk 100, the classifier 22 will filter out those objects that are further than ten feet away from the public kiosk 100. Furthermore, assuming that there is only an interest in identifying a single human, the classifier 22 will filter out those objects that are taller than seven feet in height (e.g., the typical maximum height of a human) and larger than three feet in width (e.g., the typical maximum width of a human).

If an area 68 in a YUV-mask image 64 that was located by the locator 20 is large enough to contain more than one human (e.g., a crowd of humans), then the classifier 22 typically only filters the area 68 in the YUV-mask image 64 for size (i.e., to eliminate small objects) and location (i.e., to eliminate objects too far away from the public kiosk 100). The area 68 in the YUV-mask image 64 is then passed on to the disambiguator 24 for further processing, as described in detail below.

It should be noted that the classifier 22 can also filter areas of a YUV mask image according to other characteristics such as, for example, texture and color.

In view of the foregoing, it will be recognized that the classifier 22 can be used to identify large humans (e.g., adults), small humans (e.g., children), or other objects having associated sizes. Thus, the vision system 10 can be used to identify objects having specific sizes.

The disambiguator 24 operates by further processing each area 68 in a YUV-mask image 64 that was classified by the classifier 22 as containing more than one human (e.g., a crowd of humans). More specifically, the disambiguator 24 operates by identifying discontinuities in each area 68 in the YUV-mask image 64 that was classified by the classifier 22 as containing more than one human (e.g., a crowd of humans). The identified discontinuities are then used by the disambiguator 24 to divide each area 68 in the YUV-mask image 64 that was classified by the classifier 22 as containing more than one human (e.g., a crowd of humans). The disambiguator 24 then filters each divided area in the YUV-mask image 64 for size, location, and aspect ratio so that each individual human can be identified within the crowd of humans. Thus, the disambiguator 24 operates to disambiguate each individual human from the crowd of humans.

Referring to FIG. 11, there is shown a YUV-mask image 64" having an area 68" that was classified by the classifier 22 as an area containing more than one human. The area 68" has a bottom edge 118, a top edge 120, a left edge 122, and a right edge 124. In a public kiosk application, the disambiguator 24 is most beneficially used to identify the human (i.e., the client) that is closest to the public kiosk. The disambiguator 24 accomplishes this task by identifying discontinuities along the bottom edge 118 of the area 68", and then using the identified discontinuities to divide the area 68". Referring to FIG. 12A, the YUV-mask image 64"

is shown having a redefined area 68<sup>'''</sup> that is defined by a bottom edge 118', the top edge 120, a left edge 122', and a right edge 124'. The discontinuities that are shown along the bottom edge 118' of the redefined area 68<sup>'''</sup> are identified by identifying the location of the lowest pixel that is turned "on" in each column (see FIG. 4) that passes through the area 68<sup>'''</sup> in the YUV-mask image 64<sup>'''</sup>. The bottom edge 118' of the redefined area 68<sup>'''</sup> coincides with the locations of the lowest pixels that are turned "on" in groups of some minimum number of columns that pass through the area 68<sup>'''</sup> in the YUV-mask image 64<sup>'''</sup>. It should be noted that the left edge 122' and the right edge 124' of the area 68<sup>'''</sup> in the YUV-mask image 64<sup>'''</sup> are shortened because of the identified discontinuities that are shown along the bottom edge 118' of the redefined area 68<sup>'''</sup>.

Next, the disambiguator 24 divides the redefined area 68<sup>'''</sup> in the YUV-mask image 64<sup>'''</sup> according to the identified discontinuities. For example, referring to FIG. 12B, the redefined area 68<sup>'''</sup> is divided into four subareas 68a<sup>'''</sup>, 68b<sup>'''</sup>, 68c<sup>'''</sup>, and 68d<sup>'''</sup> according to the discontinuities that were identified as described above.

After the redefined area 68<sup>'''</sup> has been divided into the four subareas 68a<sup>'''</sup>, 68b<sup>'''</sup>, 68c<sup>'''</sup>, and 68d<sup>'''</sup>, the disambiguator 24 filters each of the four subareas 68a<sup>'''</sup>, 68b<sup>'''</sup>, 68c<sup>'''</sup>, and 68d<sup>'''</sup> for size, location, and aspect ratio so that each individual human can be identified within the crowd of humans. For example, subareas 68a<sup>'''</sup> and 68d<sup>'''</sup> can be filtered out since they are too small to contain a human. The remaining two subareas, however, subareas 68b<sup>'''</sup> and 68c<sup>'''</sup>, pass through the filter of the disambiguator 24 since each of these areas is large enough to contain a human, is shaped so as to contain a human (i.e., has a suitable aspect ratio), and is located at a suitable location within in the YUV-mask image 64<sup>'''</sup>. The disambiguator 24 can thereby identify these remaining two subareas as each containing a human. Thus, the disambiguator 24 can disambiguate individual humans from a crowd of humans.

It should be noted that, similar to the filtering operation of the classifier 22, the filtering operation of the disambiguator 24 requires that the position and orientation of the camera 12 be known in order to correctly filter for size, location, and aspect ratio.

At this point it should be noted that the disambiguator 24 can also identify discontinuities along the top edge 120, the left edge 122, and the right edge 124 of the area 68<sup>'''</sup> in the YUV-mask image 64<sup>'''</sup>. For example, the disambiguator 24 can identify discontinuities along both the bottom edge 118 and the top edge 120 of the area 68<sup>'''</sup> in the YUV-mask image 64<sup>'''</sup>. Referring to FIG. 13, the YUV-mask image 64<sup>'''</sup> is shown having a redefined area 68<sup>'''</sup> that is defined by a bottom edge 118'', a top edge 120'', a left edge 122'', and a right edge 12''. The bottom edge 118'' of the redefined area 68<sup>'''</sup> coincides with the locations of the lowest pixels that are turned "on" in groups of some minimum number of columns that pass through the area 68<sup>'''</sup> in the YUV-mask image 64<sup>'''</sup>, while the top edge 120'' of the redefined area 68<sup>'''</sup> coincides with the locations of the highest pixels that are turned "on" in groups of some minimum number of columns that pass through the area 68<sup>'''</sup> in the YUV-mask image 64<sup>'''</sup>. The minimum number of columns in each group of columns can be the same or different for the bottom edge 118'' and the top edge 120''. Again, it should be noted that the left edge 122'' and the right edge 124'' of the area 68<sup>'''</sup> in the YUV-mask image 64<sup>'''</sup> are shortened because of the identified discontinuities that are shown along the bottom edge 118'' and the top edge 120'' of the redefined area 68<sup>'''</sup>. By identifying discontinuities along more than one edge of the

area 68<sup>'''</sup>, a more accurate representation of each identified area is obtained.

The disambiguator 24 can divide the redefined area 68<sup>'''</sup> shown in FIG. 13 in a similar manner to that described with respect to FIG. 12B. The disambiguator 24 can then filter the divided areas for size, location, and aspect ratio so that each individual human can be identified within the crowd of humans. Thus, the disambiguator 24 can disambiguate an individual human from a crowd of humans so that each individual human can be identified within the crowd of humans.

It should be noted that the disambiguator 24 can also filter areas of a YUV mask image according to other characteristics such as, for example, texture and color.

In view of the foregoing, it will be recognized that the disambiguator 24 can be used to disambiguate an individual object from a plurality of objects so that each individual object can be identified within the plurality of objects.

Once an individual object has been identified by either the classifier 22 or the disambiguator 24, the tracker 26 can track the object through a succession of digitized images. The tracker 26 operates by matching areas in a "current" YUV-mask image that were identified by either the classifier 22 or the disambiguator 24 as areas containing a human with areas in "prior" YUV-mask images that were also identified by either the classifier 22 or the disambiguator 24 as areas containing a human. A current YUV-mask image is typically a YUV-mask image 64 that is formed from a background image and a recently captured source image. A prior YUV-mask image is typically a YUV-mask image 64 that is formed from a background image and a source image that is captured prior to the recently captured source image. Prior YUV-mask images are typically stored in the memory 31.

The tracker 26 first compares each area in the current YUV-mask image that was identified by either the classifier 22 or the disambiguator 24 as an area containing a human with each area in the prior YUV-mask images that was identified by either the classifier 22 or the disambiguator 24 as an area containing a human. A score is then established for each pair of compared areas. The score may be calculated as a weighted sum of the differences in size between the compared areas, the differences in location between the compared areas, the differences in aspect ratio between the compared areas, the differences in texture between the compared areas, and the differences in color, or the color accuracy, between the compared areas.

The differences in size, location, and aspect ratio between the compared areas can be calculated using the size, location, and aspect ratio information that was utilized by the classifier 22 as described above. Color accuracy is measured by taking small samples of color from selected corresponding locations in each pair of compared areas. The color samples are actually taken from the source images from which the current and prior YUV-mask images were formed since the YUV-mask images themselves do not contain color characteristics, only difference characteristics. That is, color samples are taken from an area in a source image which corresponds to an area in a current or prior YUV-mask image which is formed from the source image. For example, a color sample may be taken from an area in a "current" source image which corresponds to an area in an associated current YUV-mask image. Likewise, a color sample may be taken from an area in a "prior" source image which corresponds to an area in an associated prior YUV-mask image. The color samples are therefore taken in selected corresponding locations in source images from which current and prior YUV-mask images which are



formed, wherein the selected corresponding locations in the source images correspond to selected corresponding locations in areas in the current and prior YUV-mask images which are to be compared.

Referring to FIG. 14, there is shown a current YUV-mask image 64a and a prior YUV-mask image 64b. The current and prior YUV-mask images 64a and 64b each have an area 68a and 68b, respectively, that has been identified by either the classifier 22 or the disambiguator 24 as an area containing a human. Color samples 90a and 90b are taken from selected corresponding locations in the areas 68a and 68b in the current and prior YUV-mask images 64a and 64b, respectively.

There are several methods that can be used to select the corresponding locations in the areas 68a and 68b in the current and prior YUV-mask images 64a and 64b, respectively. One method is to select corresponding locations arranged in a grid pattern within each of the YUV-mask image areas 68a and 68b. Typically, each grid pattern is distributed uniformly within each of the YUV-mask image areas 68a and 68b. For example, a grid pattern may consist of nine uniformly spaced patches arranged in three columns and three rows, as shown in FIG. 14. The color samples 90a and 90b are taken from the nine selected corresponding locations in the areas 68a and 68b in the current and prior YUV-mask images 64a and 64b, respectively.

A second method is to select corresponding locations arranged in a grid pattern within each of the YUV-mask image areas 68a and 68b wherein a corresponding location is used only if the color samples 90a and 90b each contain more than a given threshold of enabled pixels.

Referring to FIG. 15, each color sample 90a or 90b may consist of an N×N sample square of pixels 92. For example, N may equal two. The color values of the pixels 92 within each sample square are averaged. To compare two areas, a subset of the best color matches between corresponding color samples from each compared area are combined to provide a measure of color accuracy between the compared areas. For example, the best five color matches from nine color samples taken from each area 68a and 68b from the corresponding current and prior YUV-mask images 64a and 64b may be used to determine color accuracy. The use of a subset of the color matches is beneficial because it can enable tracking in the presence of partial occlusions. This measure of color accuracy is combined with the differences in size, location, aspect ratio, and texture of the compared areas to establish a score for each pair of compared areas.

The scores that are established for each pair of compared areas are sorted and placed in an ordered list (L) from highest score to lowest score. Scores below a threshold value are removed from the list and discarded. The match with the highest score is recorded by the tracker as a valid match. That is, the compared area in the prior YUV-mask image is considered to be a match with the compared area in the current YUV-mask image. This match and any other match involving either of these two compared areas is removed from the ordered list of scores. This results in a new ordered list (L). The operation of selecting the highest score, recording a valid match, and removing elements from the ordered list is repeated until no matches remain.

The tracker 26 works reliably and quickly. It can accurately track a single object (e.g., a human) moving through the frames 34 in the video sequence of temporally ordered frames 35, as well as multiple objects (e.g., several humans) which may temporarily obstruct or cross each others paths.

Because the age of each frame 34 is known, the tracker 26 can also determine the velocity of a matched area. The

velocity of a matched area can be determined by differencing the centroid position of a matched area (i.e., the center of mass of the matched area) in a current YUV-mask image with the centroid position of a corresponding matched area in a prior YUV-mask image. The differencing operation is performed in both the X and Y coordinates. The differencing operation provides a difference value that corresponds to a distance that the matched area in the current YUV-mask image has traveled in relation to the corresponding matched area in the prior YUV-mask image. The difference value is divided by the amount of time that has elapsed between the "current" and "prior" frames to obtain the velocity of the matched area.

It should be noted that the velocity of a matched area can be used as a filtering mechanism since it is often known how fast an object (e.g., a human) can travel. In this case, however, the filtering would be performed by the tracker 26 rather than the classifier 22 or the disambiguator 24.

In view of the foregoing, it will be recognized that the vision system 10 can be used to identify an object in each of a succession of digitized images. The object can be animate, inanimate, real, or virtual. Once the object is identified, the object can be tracked through the succession of digitized images.

Referring to FIG. 16, there is shown a data flow diagram for the vision system 10. Background image data 42 is provided to the differencer 18. Source image data 44 is provided to the differencer 18 and to the tracker 26. The differencer 18 provides mask image data 64 to the locator 20. The locator 20 provides located area data 68 to the classifier 22. The classifier 22 provides identified human data 68' and 68" to the tracker 26, and identified crowd data 68''' to the disambiguator 24. The disambiguator 24 provides identified human data 68'''' and 68''''' to the tracker 26. As previously described, background image data 42 is typically formed with source image data 44, located area data 68 from the locator 20, identified human data 68' and 68" from the classifier 22, identified human data 68'''' and 68''''' from the disambiguator 24, and tracked human data 94 from the tracker 26.

The present invention is not to be limited in scope by the specific embodiment described herein. Indeed, various modifications of the present invention, in addition to those described herein, will be apparent to those of skill in the art from the foregoing description and accompanying drawings. Thus, such modifications are intended to fall within the scope of the appended claims.

What is claimed is:

1. A method for locating objects within an image, wherein an object is represented by a plurality of enabled pixels within the image, the method comprising the steps of:

defining a first representation of the image, the first representation having a plurality of first pixels,

obtaining a second and a third representation of the image; the second representation having a plurality of second pixels, and the third representation having a plurality of third pixels,

forming the plurality of first pixels by taking the difference between the plurality of second pixels and the plurality of third pixels, wherein at least some of the plurality of first pixels are enabled and represent the object,

defining at least two orientations of the object;

counting the number of enabled plurality of first pixels along the each of the two orientations;

defining a threshold as a quantity of the plurality of first pixels counted;



identifying portions of the plurality of first pixels exceeding the threshold; and

identifying the object within the image based upon the orientation and the identified portions of the plurality of first pixels exceeding the threshold of the object within the image.

2. The method as defined in claim 1, wherein the step of identifying an object within the image based upon an orientation of the object within the image includes the steps of:

grouping together substantially adjacent identified portions of the representation; and

identifying areas of the representation corresponding to each group of substantially adjacent identified portions of the representation.

3. The method as defined in claim 2, wherein the step of identifying an object within the image based upon an orientation of the object within the image further includes the steps of:

recording the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of the representation; and

framing areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of the representation.

4. The method as defined in claim 3, wherein the plurality of pixels are also configured having a third orientation, wherein the step of identifying an object within the image based upon an orientation of the object within the image further includes the steps of:

counting each enabled pixel in each framed area along the third orientation; and

identifying portions of each framed area having a quantity of enabled pixels exceeding a threshold value.

5. The method as defined in claim 4, wherein the step of identifying an object within the image based upon an orientation of the object within the image further includes the steps of:

grouping together substantially adjacent identified portions of each framed area; and

identifying areas of each framed area corresponding to each group of substantially adjacent identified portions of each framed area.

6. The method as defined in claim 5, wherein the step of identifying an object within the image based upon an orientation of the object within the image further includes the steps of:

recording the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of each framed area; and

framing areas of each framed area coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of each framed area.

7. The method as defined in claim 4, wherein the second orientation and the first orientation are orthogonal.

8. The method as defined in claim 1, wherein the image is a representation of a plurality of pixels, wherein the plurality of pixels are arranged in a plurality of columns, wherein at least some of the plurality of pixels are enabled to represent the object, wherein the step of identifying an object within the image based upon an orientation of the object within the image includes the steps of:

counting each enabled pixel in each of the plurality of columns; and

identifying each of the plurality of columns having a quantity of enabled pixels exceeding a threshold value.

9. The method as defined in claim 8, wherein the step of identifying an object within the image based upon an orientation of the object within the image further includes the steps of:

grouping together substantially adjacent identified columns; and

identifying areas of the representation corresponding to each group of substantially adjacent identified columns.

10. The method as defined in claim 9, wherein the step of identifying an object within the image based upon an orientation of the object within the image further includes the steps of:

recording the locations of the outermost enabled pixels within each group of substantially adjacent identified columns; and

framing areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified columns.

11. The method as defined in claim 1, wherein the image is a representation of a plurality of pixels, wherein the plurality of pixels are arranged in a plurality of rows, wherein at least some of the plurality of pixels are enabled to represent the object, wherein the step of identifying an object within the image based upon an orientation of the object within the image includes the steps of:

counting each enabled pixel in each of the plurality of rows; and

identifying each of the plurality of rows having a quantity of enabled first pixels exceeding a threshold value.

12. The method as defined in claim 11, wherein the step of identifying an object within the image based upon an orientation of the object within the image further includes the steps of:

grouping together substantially adjacent identified rows; and

identifying areas of the representation corresponding to each group of substantially adjacent identified rows.

13. The method as defined in claim 12, wherein the step of identifying an object within the image based upon an orientation of the object within the image further includes the steps of:

recording the locations of the outermost enabled pixels within each group of substantially adjacent identified rows; and

framing areas of the representation coinciding with the locations of the outermost enabled first pixels within each group of substantially adjacent identified rows.

14. The method as defined in claim 1, wherein the image is a representation of a plurality of pixels, wherein the plurality of pixels are arranged in a plurality of columns and rows, wherein at least some of the plurality of pixels are enabled to represent the object, wherein the step of identifying an object within the image based upon an orientation of the object within the image includes the steps of:

counting each enabled pixel in each of the plurality of columns and rows;

identifying each of the plurality of columns having a quantity of enabled pixels exceeding a column threshold value; and

identifying each of the plurality of rows having a quantity of enabled pixels exceeding a row threshold value.

15. The method as defined in claim 14, wherein the step of identifying an object within the image based upon an orientation of the object within the image further includes the steps of:

19

grouping together substantially adjacent identified columns;

grouping together substantially adjacent identified rows; identifying areas of the representation corresponding to each group of substantially adjacent identified columns; and

identifying areas of the representation corresponding to each group of substantially adjacent identified rows.

16. The method as defined in claim 15, wherein the step of identifying an object within the image based upon an orientation of the object within the image further includes the steps of:

recording the locations of the outermost enabled pixels within each group of substantially adjacent identified columns;

recording the locations of the outermost enabled pixels within each group of substantially adjacent identified rows;

framing areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified columns; and framing areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified rows.

17. The method as defined in claim 16, wherein the step of identifying an object within the image based upon an orientation of the object within the image further includes the steps of:

overlying the areas of the representation that were framed to coincide with the locations of the outermost enabled pixels within each group of substantially adjacent identified columns with the areas of the representation that were framed to coincide with the locations of the outermost enabled pixels within each group of substantially adjacent identified rows; and

identifying common overlaid areas as areas of the representation that contain a significant number of enabled pixels.

18. The method as defined in claim 1, wherein the image is a first representation of a plurality of first pixels representing a difference between a second representation of a plurality of second pixels and a third representation of a plurality of third pixels, wherein each of the plurality of first pixels is enabled to represent a difference between a corresponding one of the plurality of second pixels and a corresponding one of the plurality of third pixels, wherein the object is represented by at least some of the enabled first pixels.

19. An apparatus for locating objects within an image, wherein the object is represented by a plurality of enabled pixels within the image, the apparatus comprising:

a first representation of the image, the first representation having a plurality of first pixels,

an obtainer for obtaining a second and a third representation of the image; the second representation having a plurality of second pixels, and the third representation having a plurality of third pixels,

the plurality of first pixels formed by taking the difference between the plurality of second pixels and the plurality of third pixels, wherein at least some of the plurality of first pixels are enabled and represent the object,

means for defining at least two orientations of the object; a counter of the number of enabled plurality of first pixels along the each of the two orientations;

means for defining a threshold as a quantity of the plurality of first pixels counted;

20

a first identifier of portions of the plurality of first pixels exceeding the threshold; and

a second identifier for identifying the object within the image based upon the orientation and the identified portions of the plurality of first pixels exceeding the threshold of the object within the image.

20. The apparatus as defined in claim 19, wherein the first identifier further includes:

a first grouper for grouping together substantially adjacent identified portions of the representation; and

a third identifier for identifying areas of the representation corresponding to each group of substantially adjacent identified portions of the representation.

21. The apparatus as defined in claim 20, wherein the first identifier further includes:

a first recorder for recording the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of the representation; and

a first framer for framing areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of the representation.

22. The apparatus as defined in claim 21, wherein the plurality of pixels are also configured having a third orientation, wherein the first identifier further includes:

a second counter for counting each enabled pixel in each framed area along the third orientation; and

a fourth identifier for identifying portions of each framed area having a quantity of enabled pixels exceeding a threshold value.

23. The apparatus as defined in claim 22, wherein the first identifier further includes:

a second grouper for grouping together substantially adjacent identified portions of each framed area; and

a fifth identifier for identifying areas of each framed area corresponding to each group of substantially adjacent identified portions of each framed area.

24. The apparatus as defined in claim 23, wherein the first identifier further includes:

a second recorder for recording the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of each framed area; and

a second framer for framing areas of each framed area coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of each framed area.

25. The apparatus as defined in claim 22, wherein the second orientation and the first orientation are orthogonal.

26. The apparatus as defined in claim 19, wherein the image is a representation of a plurality of pixels, wherein the plurality of pixels are arranged in a plurality of columns, wherein at least some of the plurality of pixels are enabled to represent the object, wherein the first identifier includes:

a counter for counting each enabled pixel in each of the plurality of columns; and

a second identifier for identifying each of the plurality of columns having a quantity of enabled pixels exceeding a threshold value.

27. The apparatus as defined in claim 26, wherein the first identifier further includes:

a grouper for grouping together substantially adjacent identified columns; and

a third identifier for identifying areas of the representation corresponding to each group of substantially adjacent identified columns.

21

28. The apparatus as defined in claim 27, wherein the first identifier further includes:

a recorder for recording the locations of the outermost enabled pixels within each group of substantially adjacent identified columns; and

a framer for framing areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified columns.

29. The apparatus as defined in claim 19, wherein the image is a representation of a plurality of pixels, wherein the plurality of pixels are arranged in a plurality of rows, wherein at least some of the plurality of pixels are enabled to represent the object, wherein the first identifier includes:

a counter for counting each enabled pixel in each of the plurality of rows; and

a second identifier for identifying each of the plurality of rows having a quantity of enabled first pixels exceeding a threshold value.

30. The apparatus as defined in claim 29, wherein the first identifier further includes:

a grouper for grouping together substantially adjacent identified rows; and

a third identifier for identifying areas of the representation corresponding to each group of substantially adjacent identified rows.

31. The apparatus as defined in claim 30, wherein the first identifier further includes:

a recorder for recording the locations of the outermost enabled pixels within each group of substantially adjacent identified rows; and

a framer for framing areas of the representation coinciding with the locations of the outermost enabled first pixels within each group of substantially adjacent identified rows.

32. The apparatus as defined in claim 19, wherein the image is a representation of a plurality of pixels, wherein the plurality of pixels are arranged in a plurality of columns and rows, wherein at least some of the plurality of pixels are enabled to represent the object, wherein the first identifier includes:

a counter for counting each enabled pixel in each of the plurality of columns and rows;

a second identifier for identifying each of the plurality of columns having a quantity of enabled pixels exceeding a column threshold value; and

a third identifier for identifying each of the plurality of rows having a quantity of enabled pixels exceeding a row threshold value.

33. The apparatus as defined in claim 32, wherein the first identifier further includes:

a first grouper for grouping together substantially adjacent identified columns;

a second grouper for grouping together substantially adjacent identified rows;

a fourth identifier for identifying areas of the representation corresponding to each group of substantially adjacent identified columns; and

a fifth identifier for identifying areas of the representation corresponding to each group of substantially adjacent identified rows.

34. The apparatus as defined in claim 33, wherein the first identifier further includes:

a first recorder for recording the locations of the outermost enabled pixels within each group of substantially adjacent identified columns;

22

a second recorder for recording the locations of the outermost enabled pixels within each group of substantially adjacent identified rows;

a first framer for framing areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified columns; and

a second framer for framing areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified rows.

35. The apparatus as defined in claim 34, wherein the first identifier further includes:

an overlayer for overlaying the areas of the representation that were framed to coincide with the locations of the outermost enabled pixels within each group of substantially adjacent identified columns with the areas of the representation that were framed to coincide with the locations of the outermost enabled pixels within each group of substantially adjacent identified rows; and

a sixth identifier for identifying common overlaid areas as areas of the representation that contain a significant number of enabled pixels.

36. The apparatus as defined in claim 19, wherein the image is a first representation of a plurality of first pixels representing a difference between a second representation of a plurality of second pixels and a third representation of a plurality of third pixels, wherein each of the plurality of first pixels is enabled to represent a difference between a corresponding one of the plurality of second pixels and a corresponding one of the plurality of third pixels, wherein the object is represented by at least some of the enabled first pixels.

37. An article of manufacture for locating objects within an image, wherein the object is represented by a plurality of enabled pixels within the image, the article of manufacture comprising:

a computer readable storage medium; and  
computer programming stored on the storage medium; wherein the stored computer programming is configured to be readable from the computer readable storage medium by a computer and thereby cause the computer to operate so as to:

define a first representation of the image, the first representation having a plurality of first pixels, obtain a second and a third representation of the image; the second representation having a plurality of second pixels, and the third representation having a plurality of third pixels,

the plurality of first pixels formed by taking the difference between the plurality of second pixels and the plurality of third pixels, wherein at least some of the plurality of first pixels are enabled and represent the object,

define at least two orientations of the object; count the number of enabled plurality of first pixels along the each of the two orientations;

define a threshold as a quantity of the plurality of first pixels counted; identify of portions of the plurality of of first pixels exceeding the threshold; and

identify the object within the image based upon the orientation and the identified portions of the plurality of first pixels exceeding the threshold of the object within the image.

38. The article of manufacture as defined in claim 37, further causing the computer to operate so as to:

group together substantially adjacent identified portions of the representation; and

identify areas of the representation corresponding to each group of substantially adjacent identified portions of the representation.

39. The article of manufacture as defined in claim 38, further causing the computer to operate so as to:

record the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of the representation; and

frame areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of the representation.

40. The article of manufacture as defined in claim 39, wherein the plurality of pixels are also configured having a third orientation, further causing the computer to operate so as to:

count each enabled pixel in each framed area along the third orientation; and

identify portions of each framed area having a quantity of enabled pixels exceeding a threshold value.

41. The article of manufacture as defined in claim 40, further causing the computer to operate so as to:

group together substantially adjacent identified portions of each framed area; and

identify areas of each framed area corresponding to each group of substantially adjacent identified portions of each framed area.

42. The article of manufacture as defined in claim 41, further causing the computer to operate so as to:

record the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of each framed area; and

frame areas of each framed area coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified portions of each framed area.

43. The article of manufacture as defined in claim 40, wherein the second orientation and the first orientation are orthogonal.

44. The article of manufacture as defined in claim 37, wherein the image is a representation of a plurality of pixels, wherein the plurality of pixels are arranged in a plurality of columns, wherein at least some of the plurality of pixels are enabled to represent the object, further causing the computer to operate so as to:

count each enabled pixel in each of the plurality of columns; and

identify each of the plurality of columns having a quantity of enabled pixels exceeding a threshold value.

45. The article of manufacture as defined in claim 44, further causing the computer to operate so as to:

group together substantially adjacent identified columns; and

identify areas of the representation corresponding to each group of substantially adjacent identified columns.

46. The article of manufacture as defined in claim 45, further causing the computer to operate so as to:

record the locations of the outermost enabled pixels within each group of substantially adjacent identified columns; and

frame areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified columns.

47. The article of manufacture as defined in claim 37, wherein the image is a representation of a plurality of pixels, wherein the plurality of pixels are arranged in a plurality of rows, wherein at least some of the plurality of pixels are enabled to represent the object, further causing the computer to operate so as to:

count each enabled pixel in each of the plurality of rows; and

identify each of the plurality of rows having a quantity of enabled first pixels exceeding a threshold value.

48. The article of manufacture as defined in claim 47, further causing the computer to operate so as to:

group together substantially adjacent identified rows; and identify areas of the representation corresponding to each group of substantially adjacent identified rows.

49. The article of manufacture as defined in claim 48, further causing the computer to operate so as to:

record the locations of the outermost enabled pixels within each group of substantially adjacent identified rows; and

frame areas of the representation coinciding with the locations of the outermost enabled first pixels within each group of substantially adjacent identified rows.

50. The article of manufacture as defined in claim 37, wherein the image is a representation of a plurality of pixels, wherein the plurality of pixels are arranged in a plurality of columns and rows, wherein at least some of the plurality of pixels are enabled to represent the object, further causing the computer to operate so as to:

count each enabled pixel in each of the plurality of columns and rows;

identify each of the plurality of columns having a quantity of enabled pixels exceeding a column threshold value; and

identify each of the plurality of rows having a quantity of enabled pixels exceeding a row threshold value.

51. The article of manufacture as defined in claim 50, further causing the computer to operate so as to:

group together substantially adjacent identified columns; group together substantially adjacent identified rows;

identify areas of the representation corresponding to each group of substantially adjacent identified columns; and

identify areas of the representation corresponding to each group of substantially adjacent identified rows.

52. The article of manufacture as defined in claim 51, further causing the computer to operate so as to:

record the locations of the outermost enabled pixels within each group of substantially adjacent identified columns;

record the locations of the outermost enabled pixels within each group of substantially adjacent identified rows;

frame areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified columns; and

frame areas of the representation coinciding with the locations of the outermost enabled pixels within each group of substantially adjacent identified rows.

53. The article of manufacture as defined in claim 52, further causing the computer to operate so as to:

overlay the areas of the representation that were framed to coincide with the locations of the outermost enabled pixels within each group of substantially adjacent identified columns with the areas of the representation that

25

were framed to coincide with the locations of the outermost enabled pixels within each group of substantially adjacent identified rows; and  
identify common overlaid areas as areas of the representation that contain a significant number of enabled pixels.  
54. The article of manufacture as defined in claim 37, wherein the image is a first representation of a plurality of first pixels representing a difference between a second

26

representation of a plurality of second pixels and a third representation of a plurality of third pixels, wherein each of the plurality of first pixels is enabled to represent a difference between a corresponding one of the plurality of second pixels and a corresponding one of the plurality of third pixels, wherein the object is represented by at least some of the enabled first pixels.

\* \* \* \* \*



US006456732B1

(12) **United States Patent**  
**Kimbell et al.**

(10) **Patent No.:** **US 6,456,732 B1**  
(45) **Date of Patent:** **Sep. 24, 2002**

- (54) **AUTOMATIC ROTATION, CROPPING AND SCALING OF IMAGES FOR PRINTING**
- (75) **Inventors:** Benjamin D. Kimbell, Boulder; Dan L. Dalton, Greeley; Michael L. Rudd, Ft. Collins, all of CO (US)
- (73) **Assignee:** Hewlett-Packard Company, Palo Alto, CA (US)
- (\*) **Notice:** Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

5,596,655 A	1/1997	Lopez	382/173
5,623,681 A	* 4/1997	Rivette et al.	707/530
5,713,070 A	1/1998	Ohkubo	399/363
5,784,487 A	* 7/1998	Copperman	382/175
5,838,836 A	* 11/1998	Omvik	382/276
5,880,858 A	* 3/1999	Jin	358/487
5,943,679 A	* 8/1999	Niles et al.	707/526
5,960,448 A	* 9/1999	Reichek et al.	707/526
5,974,199 A	* 10/1999	Lee et al.	382/289
5,978,519 A	* 11/1999	Bollman et al.	382/282
6,011,635 A	* 1/2000	Bungo et al.	358/488
6,043,823 A	* 3/2000	Kodaira et al.	345/433
6,240,204 B1	* 5/2001	Hidaka et al.	382/167
6,389,434 B1	* 5/2002	Rivette et al.	707/530

- (21) **Appl. No.:** 09/151,892
- (22) **Filed:** Sep. 11, 1998
- (51) **Int. Cl.7** ..... G06K 9/00
- (52) **U.S. Cl.** ..... 382/112; 358/449; 382/180; 399/187; 707/521
- (58) **Field of Search** ..... 382/112, 175, 382/176, 297, 298, 299, 276, 289; 358/537, 538, 401, 448, 449, 453, 464, 474, 488, 451, 1.2, 1.9, 1.18; 399/133, 155, 156, 160, 182, 183, 188, 190, 196, 86, 130; 707/526, 521; 427/153

\* cited by examiner

*Primary Examiner*—Jayanti K. Patel  
(74) *Attorney, Agent, or Firm*—Augustus W. Winfield

(57) **ABSTRACT**

A method for automatically cropping, rotating, and scaling a scanned image to ensure that a printed copy of the scanned image is the same size as the original, when possible. The method attempts to honor the default or operator designated orientation of the printed image, but will automatically rotate the image if that will eliminate unnecessary image reduction. Optimal orientation and scaling factors are automatically determined based on the target page size and the size and shape of the information of interest in the original image (not the boundaries of the original document). The operator selects a desired printed orientation (or accepts a default orientation) and selects a desired printed paper size (or accepts a default printed paper size). If an image will fit within the printable margins without rotation or cropping, the image is simply printed without modification. If the image will fit without rotation by cropping white space, then white space is cropped. If the image with all white space cropped will still not fit, the image is oriented so that long sides on the cropped image align with long sides on the printed paper. If the cropped and rotated image still does not fit, the cropped image is scaled to fit within the printable margins and the image is oriented so that long sides on the cropped image align with long sides on the printed paper.

(56) **References Cited**

**U.S. PATENT DOCUMENTS**

3,618,032 A	* 11/1971	Goldsberry et al.	707/517
4,047,811 A	* 9/1977	Allis et al.	399/191
4,593,989 A	* 6/1986	Fujiwara et al.	399/190
4,907,033 A	* 3/1990	Maruta et al.	355/320
5,020,115 A	* 5/1991	Black	382/298
5,053,885 A	10/1991	Telle	358/449
5,191,429 A	* 3/1993	Rourke	358/296
5,212,568 A	* 5/1993	Graves et al.	358/474
5,220,649 A	* 6/1993	Forcier	707/541
5,260,805 A	* 11/1993	Barrett	358/449
5,280,367 A	1/1994	Zuniga	358/462
5,438,430 A	* 8/1995	Mackinlay et al.	358/450
5,483,606 A	1/1996	Deaber	382/294
5,546,474 A	8/1996	Zuniga	382/176
5,557,728 A	9/1996	Garrett et al.	395/157

**3 Claims, 3 Drawing Sheets**

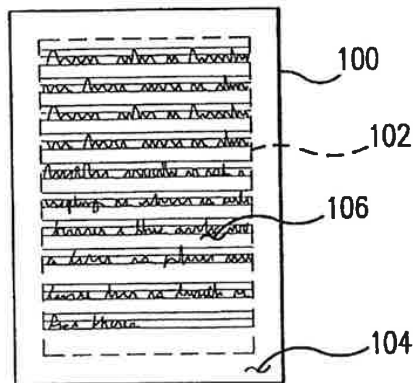


FIG.1A

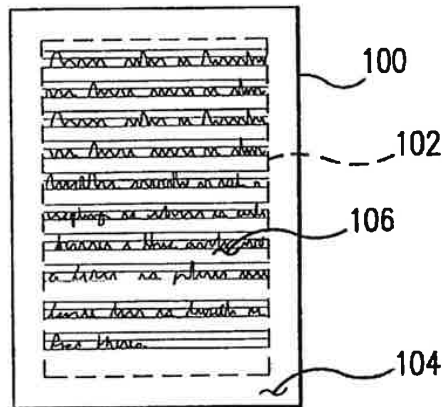


FIG.1B

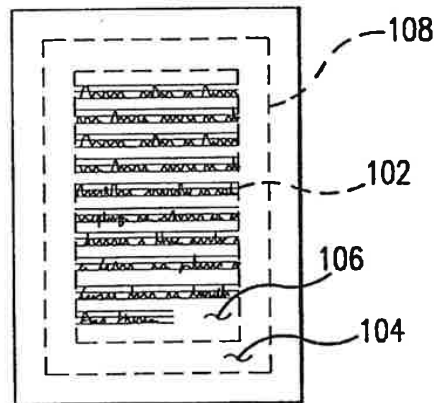
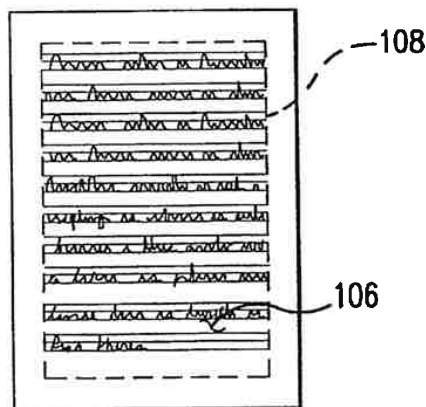


FIG.1C



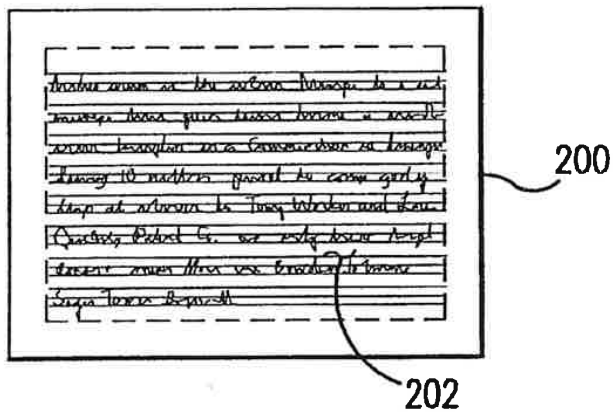


FIG. 2A

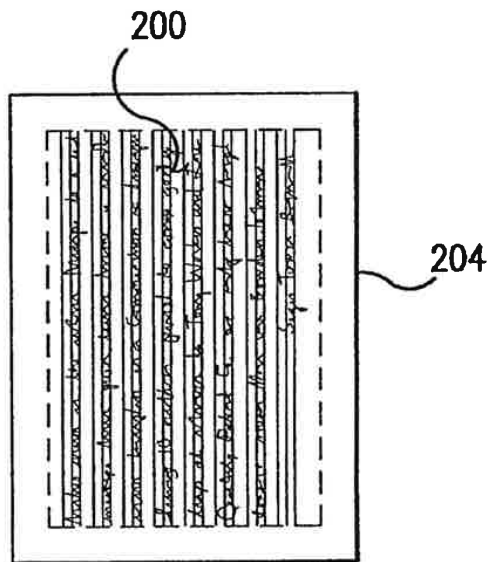


FIG. 2B



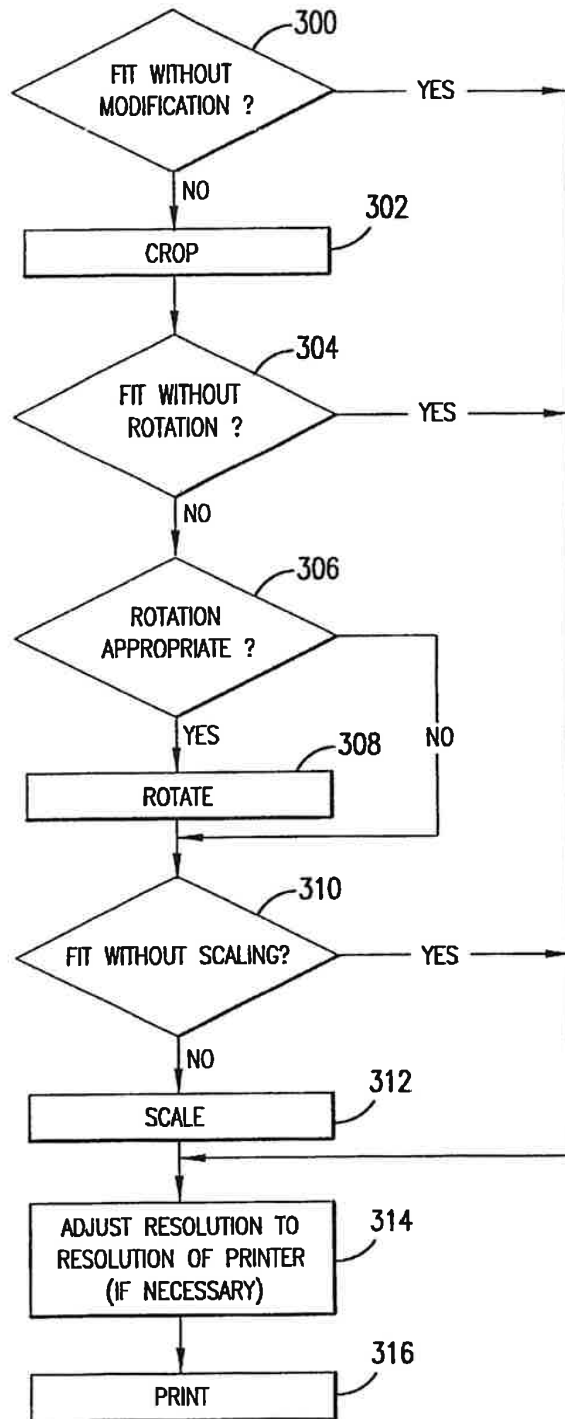


FIG.3

1

## AUTOMATIC ROTATION, CROPPING AND SCALING OF IMAGES FOR PRINTING

### FIELD OF INVENTION

This invention relates generally to printing copies of documents that have been scanned by optical scanners, digital cameras, facsimile machines, digital photocopiers, or other digital imaging devices, and more specifically to an automatic method of rotating, cropping and scaling of images for printing.

### BACKGROUND OF THE INVENTION

Photocopiers typically require the operator to properly orient the original document and may require the operator to select the proper paper bin. For example, for landscape mode, the operator typically must orient the original document in a landscape orientation and select a paper bin having paper in a landscape orientation. Similarly, when printing a scanned image from a computer, the operator typically must specify orientation of the image on the page. If an operator makes an inappropriate choice, resources such as toner, paper, and time may be wasted if the photocopier prints pages that are not useful or not what was expected.

Photocopiers also typically require an operator to input or choose a scale factor to reduce oversized images to fit onto the output page or to magnify small images to fit onto the output page. Some photocopiers may provide an automatic scaling feature, in which the printed document size is automatically scaled based on the dimensions of the edges of the original document.

Photocopiers typically can print to the edges of the output page. In contrast, many computer software applications, for example word processing software, force unprintable margins around the edges of a page. In addition, many computer software applications automatically scale an image to fit inside the unprintable margins. Consider an image, including printed text, that is scanned by a document scanner. Assume that the scanned image includes white space around the edges. If that scanned image is imported into a word processor and reprinted, word processing software will typically reduce the image, including the scanned margins, to fit within the printable area of a page. The net result is that the printed text is reduced in size.

In some situations, the primary goal is a printed page with an image of interest that is as large as possible. There is a need for additional automation in optimizing the printed size of a scanned image.

### SUMMARY OF THE INVENTION

One goal of the present invention is make the printed image the same size as the original image (or slightly larger) when possible. The method attempts to honor the default or operator designated orientation of the printed image, but will automatically rotate the image if that will eliminate unnecessary image reduction. Optimal orientation and scaling factors are automatically determined based on the target page size and the size and shape of the original image (not the boundaries of the original document). The operator selects a desired printed orientation (or accepts a default orientation) and selects a desired printed paper size (or accepts a default printed paper size). If an image will fit within the printable margins without rotation or cropping, the image is simply printed without modification. If there is white space that can be cropped, and if the image will fit without rotation by cropping white space, then white space

2

is cropped. If the image with all white space cropped will still not fit, and if the image is not oriented so that long sides on the cropped image align with long sides on the printed paper, then the image is rotated. If the cropped and rotated image still does not fit, the cropped and rotated image is scaled to fit within the printable margins and the image is oriented so that long sides on the cropped image align with long sides on the printed paper.

### BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1A is a plan view of a scanned document with white margins.

FIG. 1B is a plan view of the scanned document of FIG. 1A reduced within printable margins of a printed page.

FIG. 1C is a plan view of the scanned document of FIG. 1A, printed within printable margins of a printed page, without reduction but with cropped margins.

FIG. 2A is a plan view of a scanned document.

FIG. 2B is a plan view of the document of FIG. 2A cropped and rotated and printed within printable margins of a printed page.

FIG. 3 is flow chart of a method of cropping, rotating and scaling in accordance with the invention.

### DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT OF THE INVENTION

FIG. 1A illustrates an image 100 resulting from scanning a text document. Dashed line 102 is not part of the image, but instead depicts a rectangular boundary between a margin 104 containing only "white space" and non-white information of interest 106, which in this example image is text.

FIG. 1B illustrates the typical situation when image 100 is printed using computer software. Dashed line 108 depicts the printable area on the page as determined by printer hardware, and known by the computer software. Often, computer software will avoid clipping an image, and instead will scale an image to fit within the printable area of a page. In FIG. 1B, image 100 has been slightly reduced by the computer software so that the entire scanned image 100, including the margins 104, is printed within the printable area depicted by line 108. Specifically, for text as illustrated in FIG. 1A, the text in FIG. 1B is smaller than the text in FIG. 1A.

FIG. 1C illustrates a first aspect of a method in accordance with the present invention, which is to digitally crop the image before sending the image to a printer or to software for printing, so that the resulting information of interest 106 (text in the example) in FIG. 1C is the same size as (or larger than) the corresponding information of interest in FIG. 1A. One approach is to simply always crop (delete margin data) the entire margin 104 of FIG. 1A so that the information of interest 106 in FIG. 1C extends to the edges of the printable area depicted by line 108. If a large margin is entirely cropped, the image of interest may be enlarged if sent to software for printing. Alternatively, if the goal is to always keep the printed image the same size as the original image of interest, sufficient margin may be cropped to keep the printed image the same size as the original even if sent to software for printing. If the cropped image is to be sent direct to a printer, printers typically do not scale, so that full cropping will still result in the printed image of interest having the same size as the original image of interest. If margins are completely cropped, the image of interest is then preferably centered horizontally and vertically within the

printed page. One reason for centering is to avoid printing the image of interest in the upper left corner where a staple may interfere. Note that some images may not have any white space to be cropped, and some images may have a lot of white space, only some of which needs to be cropped. In general, if white space is present, sufficient cropping is performed to make the resulting cropped image fit, if possible.

FIG. 2A illustrates a scanned image 200 that includes non-white image area of interest (for example, scanned text) 202, where the information of interest 202 is wider than the shortest dimension of the printable area depicted by line 108 in FIGS. 1B and 1C. That is, if image 200 is printed onto a page in portrait orientation, the image must be reduced or some of the information of interest 202 will be cropped during printing. In the present application, automatic rotation is preferable to scaling or cropping of information of interest in the image. Accordingly, if image 200 will not fit, even after cropping, within the printable area in its original orientation, and if the longest dimension of the rectangular non-white image 200 is not aligned with the longest dimension of a rectangular printable area, then the image is rotated. If the longest dimension of the rectangular non-white image 200 is already aligned with the longest dimension of a rectangular printable area, then no rotation is performed.

In FIG. 2B, image 200 has been cropped (to remove white space margins only) and rotated so that the longest dimension of the non-white image 202 is aligned with the longest dimension of the printable area on printed page 204. The non-white image 202 in FIG. 2B is the same size as non-white image 202 within the original image 200 in FIG. 2A. The non-white image 202 is scaled only if it will not fit into the printable area after cropping and rotation (if appropriate).

FIG. 3 is a flow chart of a method in accordance with the present invention. Before entry into the method of FIG. 3, a preferred printed orientation is selected (or a default orientation is accepted). In addition, a printed page size is selected (or a default page size is accepted). Before entry into the method of FIG. 3, the image is oriented according to the selected orientation. At decision 300, if a scanned image will fit, within the printable area, in the selected orientation, without cropping, rotating, or scaling, then the image is saved or transferred without modification. Otherwise, at step 302, white-space margins (if available) are identified and cropped (margin data is deleted), entirely or just to the extent necessary to make the image fit. For examples of methods for automatic classification of various areas of an image (to determine a rectangle defining the information of interest), see U.S. Pat. Nos. 5,280,367; 5,546,474; and 5,596,655. At decision 304, if the scanned image will fit, within the printable area in the selected orientation, after cropping but without rotation or scaling, then the image is saved or transferred without rotation or scaling. Otherwise, at decision 306, if the longest dimension on the cropped image does not align with the longest dimension of the printable area on the printed paper, then the image is rotated ninety degrees (step 308). At decision 310, if the scanned image will fit, within the printable area, after cropping and after rotation (if appropriate) but without scaling, then the image is saved or transferred without scaling. Otherwise, the image is scaled to fit at step 312.

One alternative goal of the present invention is make the non-white portion of the printed image the same size as the non-white information of interest portion of the original

image, when possible. If one pixel in the scanned image is printed as one pixel on the printed page, then the scanned image and the printed image need to have the same resolution. Accordingly, at step 314, given the printer resolution, if the resolution of the scanned image is different than the resolution of the printer, the image resolution is adjusted (by interpolation, decimation, or both) to match the resolution of the printer. For example, if the resolution of the image is 300 pixels per inch and the resolution of the printer is 600 pixels per inch, the image pixels may be interpolated to double the effective resolution. Alternatively, if resolution of the image is 600 pixels per inch and the resolution of the printer is 300 pixels per inch then the image pixels are decimated to halve the effective resolution. For non-integral ratios, image pixels may be interpolated and then decimated to provide the proper resolution.

Some document scanners may be interfaced to a computer that in turn is interfaced to a printer. In a configuration including a computer, the automated cropping, rotation, and scaling may be performed by firmware within the scanner or by software in the computer, or by firmware within the printer. Some document scanners may be directly interfaced to a printer, so that a copy mode may be performed without an intermediate computer. In a direct connection, the automated cropping, rotation, and scaling may be performed by firmware within the scanner, or by firmware within the printer.

The foregoing description of the present invention has been presented for purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed, and other modifications and variations may be possible in light of the above teachings. The embodiment was chosen and described in order to best explain the principles of the invention and its practical application to thereby enable others skilled in the art to best utilize the invention in various embodiments and various modifications as are suited to the particular use contemplated. It is intended that the appended claims be construed to include other alternative embodiments of the invention except insofar as limited by the prior art.

What is claimed is:

1. A method of automatically modifying a scanned image by a system, the method comprising the following steps:
  - (a) determining, by the system, that the scanned image will not fit within a printable area of a printed page;
  - (b) determining, by the system, that the scanned image data includes margins that do not need to be printed; and
  - (c) deleting, by the system, data corresponding to the margins of the scanned image.
2. The method of claim 1 further comprising:
  - (d) determining, by the system, that the scanned image after deleting margin data in step (c) will not fit within the printable area of the printed page; and
  - (e) rotating, by the system, the scanned image ninety-degrees.
3. The method of claim 2 further comprising:
  - (f) determining, by the system, that the scanned image will not fit within the printable area of the printed page even with cropping and rotating; and
  - (g) scaling, by the system, the scanned image to fit within the printable area of the printed page.

\* \* \* \* \*



US006282317B1

(12) **United States Patent**  
**Luo et al.**

(10) **Patent No.:** **US 6,282,317 B1**  
(45) **Date of Patent:** **Aug. 28, 2001**

- (54) **METHOD FOR AUTOMATIC DETERMINATION OF MAIN SUBJECTS IN PHOTOGRAPHIC IMAGES**
- (75) **Inventors:** **Jiebo Luo**, Pittsford; **Stephen Etz**, Fairport; **Amit Singhal**, Rochester, all of NY (US)
- (73) **Assignee:** **Eastman Kodak Company**, Rochester, NY (US)
- (\*) **Notice:** Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

- (21) **Appl. No.:** **09/223,860**
- (22) **Filed:** **Dec. 31, 1998**
- (51) **Int. Cl.<sup>7</sup>** ..... **G06K 9/46**
- (52) **U.S. Cl.** ..... **382/203**
- (58) **Field of Search** ..... **382/203, 204, 382/205, 206, 207, 168, 169, 173, 218, 195, 155, 156, 159, 160, 190, 202, 227**

(56) **References Cited**

**U.S. PATENT DOCUMENTS**

4,975,975	*	12/1990	Filipski	382/38
5,724,456		3/1998	Boyack et al.	382/274
5,809,322	*	9/1998	Akerib	395/800.14
5,850,352	*	12/1998	Moezzi et al.	364/514
5,862,260	*	1/1999	Rhoads	382/232
5,983,218	*	11/1999	Syeda-Mahmood	707/104
6,014,461	*	1/2000	Hennessey et al.	382/195

**OTHER PUBLICATIONS**

V. D. Gesu, et al., "Local Operators to detect regions of interest," *Pattern Recognition Letters*, vol. 18, pp. 1077-1081, 1977.

R. Milanese, *Detecting salient regions in an image: From biological evidence to computer implementations*, PhD thesis, University of Geneva, Switzerland, 1993.

X. Marichal, et al., "Automatic detection of interest areas of an image or of a sequence of images," in *Proc. IEEE Int. Conf. Image Process.*, 1996.

W. Osberger, et al., "Automatic identification of perpetually important regions in an image," in *Proc. IEEE Int. Conf. Pattern Recognition*, 1998.

Q. Huang, et al., "Foreground/background segmentation of color images by integration of multiple cues," in *Proc. IEEE Int. Conf. Image Process.*, 1995.

T. F. Syeda-Mahmood, "Data and model-driven selection using color regions," *Int. J. Comput. Vision*, vol. 21, No. 1, pp. 9-36, 1997.

Luo, et al., "Towards physics-based segmentation of photographic color images," *Proceedings of the IEEE International Conference on Image Processing*, 1997.

L. R. Williams, "Perceptual organization of occluding contours," in *Proc. IEEE Int. Conf. Computer Vision*, 1990.

J. August, et al., "Fragment grouping via the principle of perceptual occlusion," in *Proc. IEEE Int. Conf. Pattern Recognition*, 1996.

Lee, "Color image quantization based on physics and psychophysics," *Journal of Society of Photographic Science and Technology of Japan*, vol. 59, No. 1, pp. 212-225, 1996.

J. Pearl, *Probabilistic Reasoning in Intelligent Systems*, San Francisco, CA: Morgan Kaufmann, 1988.

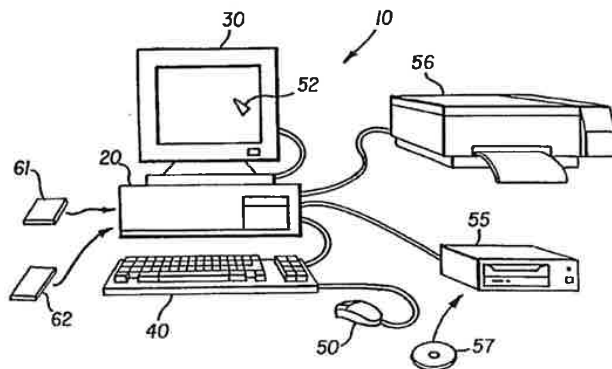
\* cited by examiner

**Primary Examiner**—Andrew W. Johns  
**Assistant Examiner**—Seyed Azarian  
(74) **Attorney, Agent, or Firm**—David M. Woods

(57) **ABSTRACT**

A method for detecting a main subject in an image, the method comprises: receiving a digital image; extracting regions of arbitrary shape and size defined by actual objects from the digital image; grouping the regions into larger segments corresponding to physically coherent objects; extracting for each of the regions at least one structural saliency feature and at least one semantic saliency feature; and integrating saliency features using a probabilistic reasoning engine into an estimate of a belief that each region is the main subject.

**37 Claims, 9 Drawing Sheets**



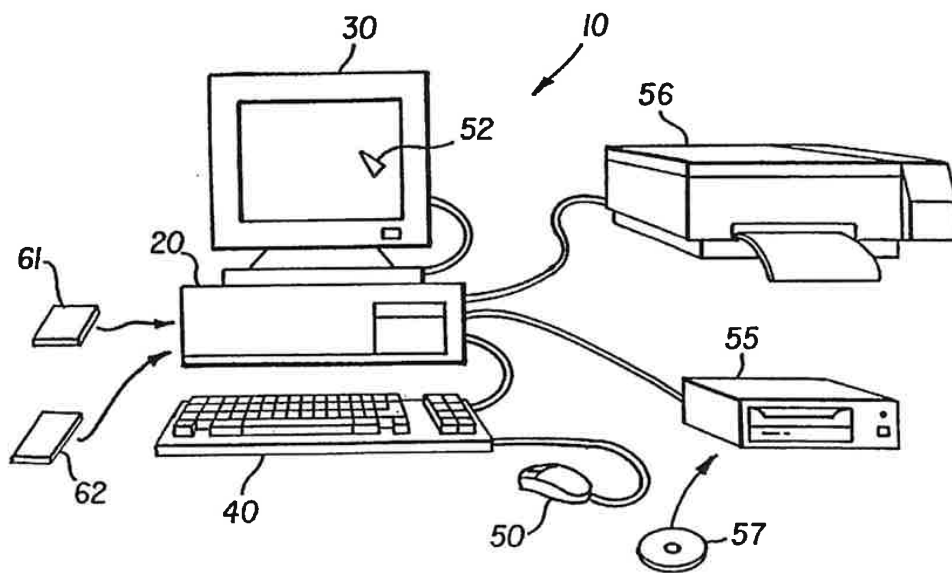


FIG. 1

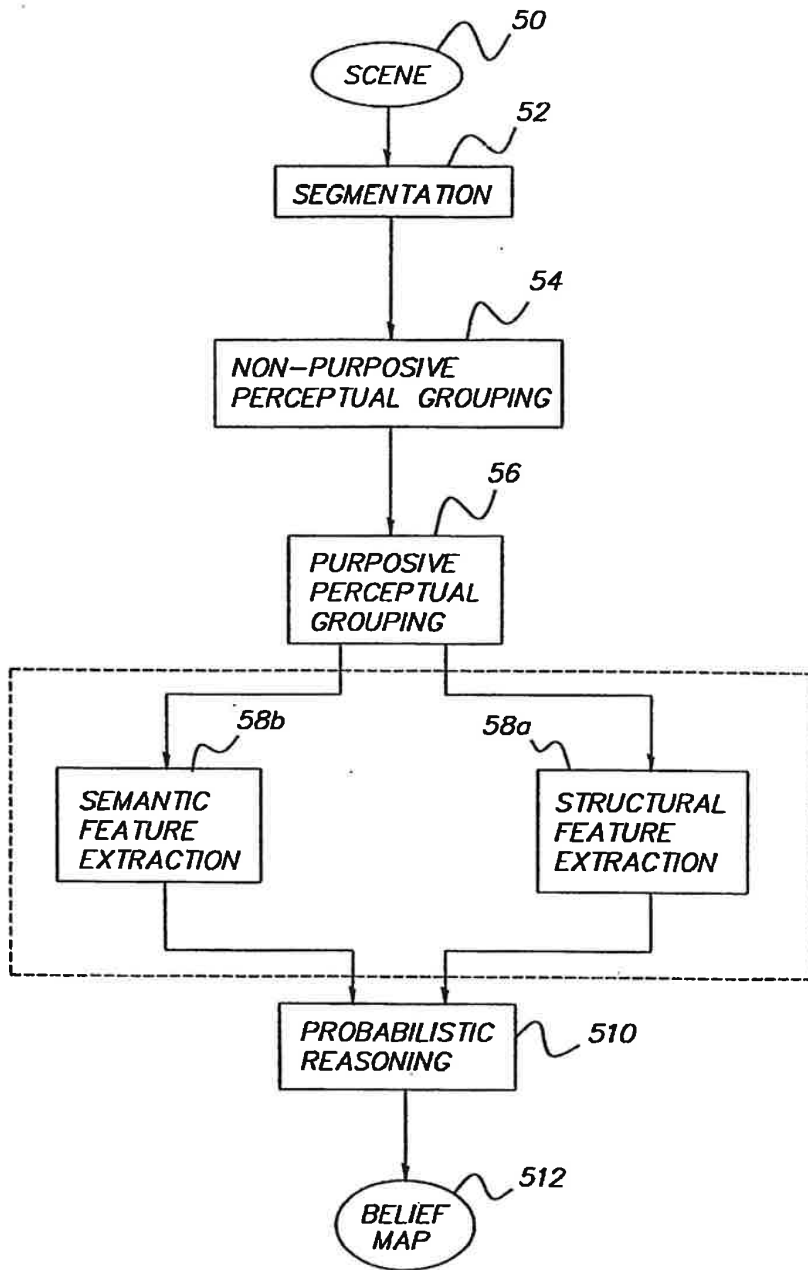


FIG. 2

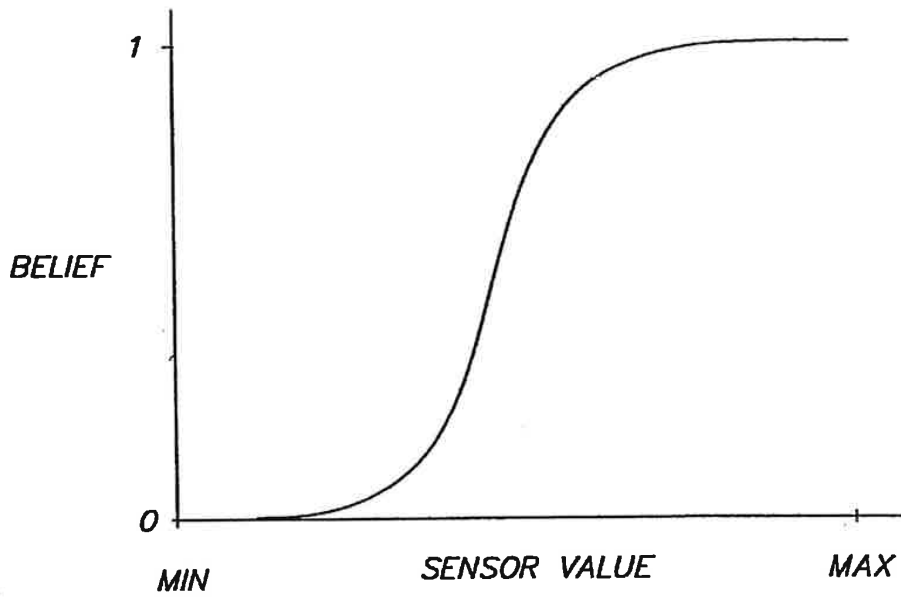


FIG. 3

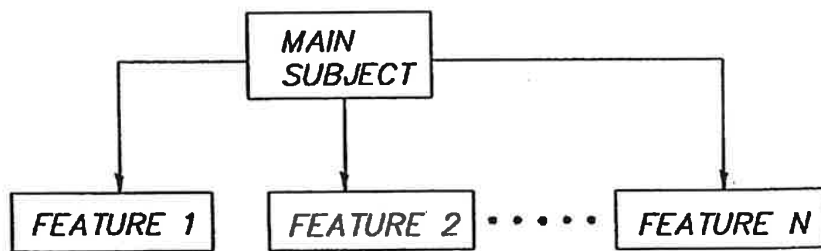


FIG. 7

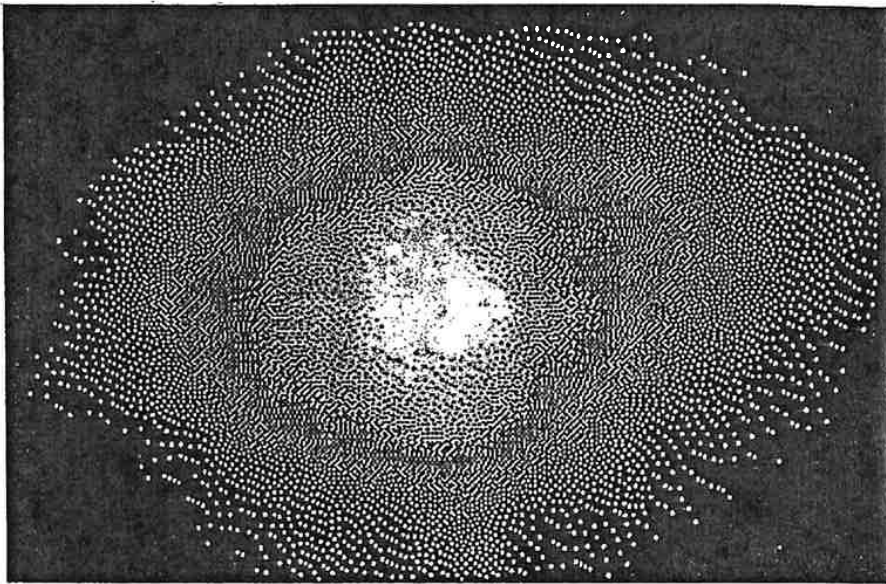
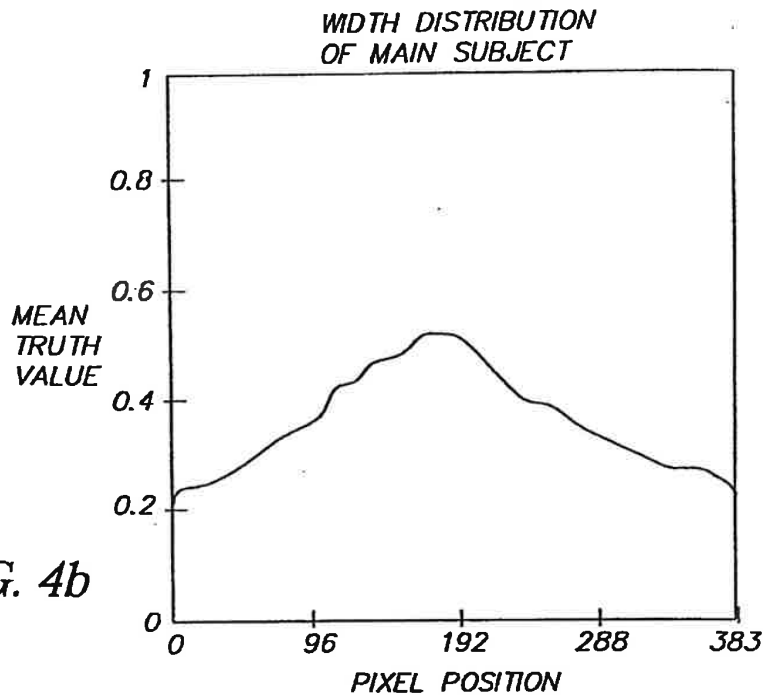
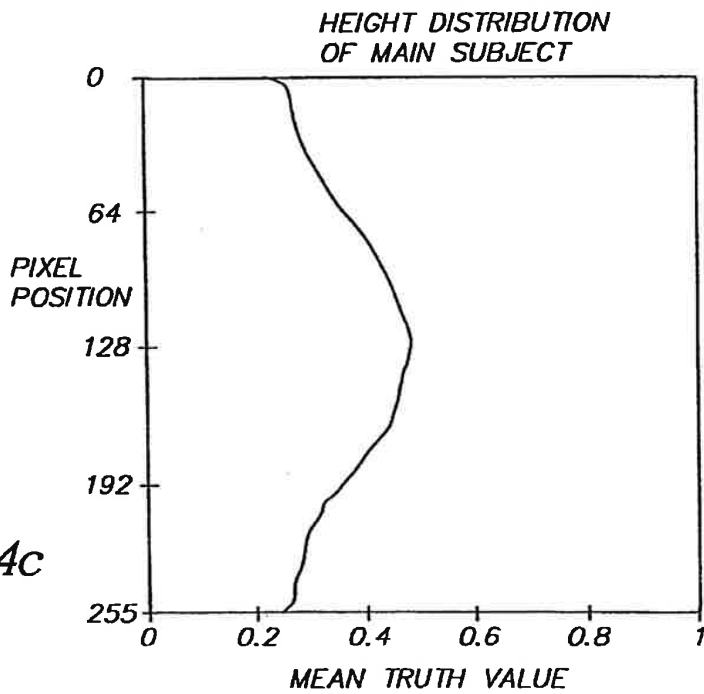


Fig. 4a





*FIG. 4b*



*FIG. 4c*

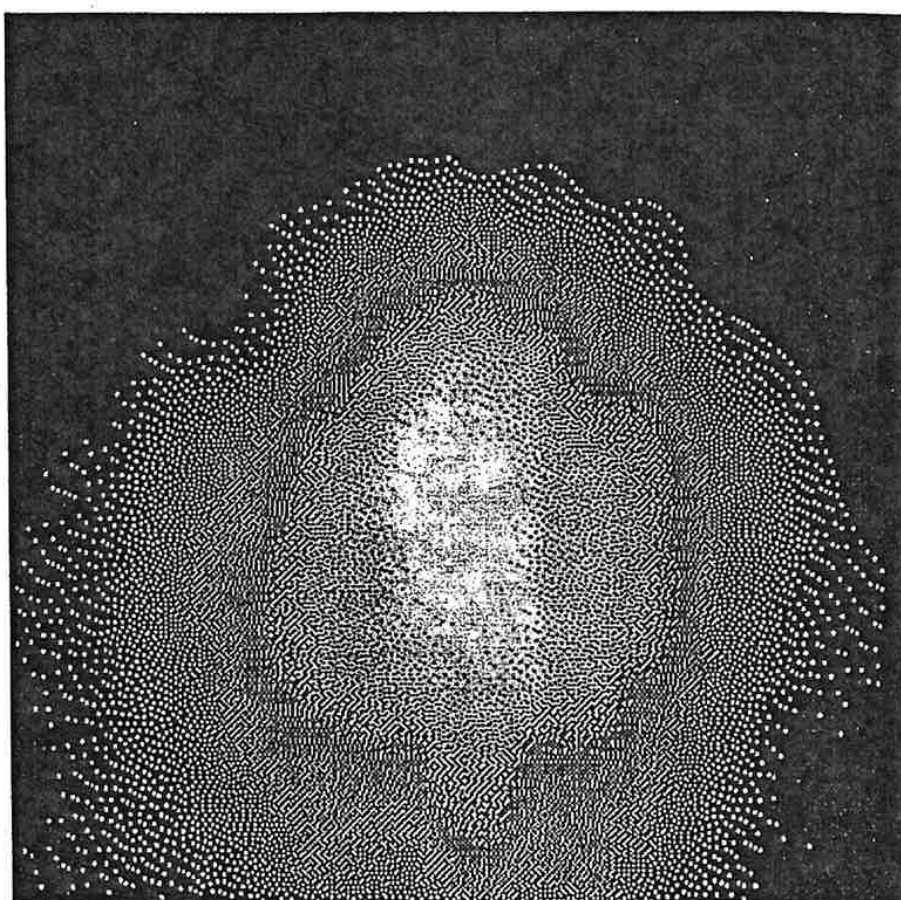
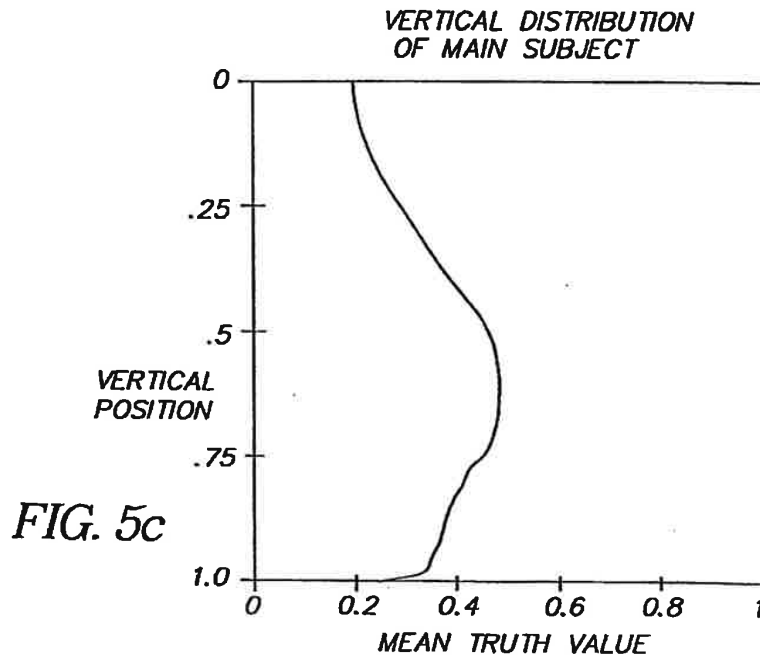
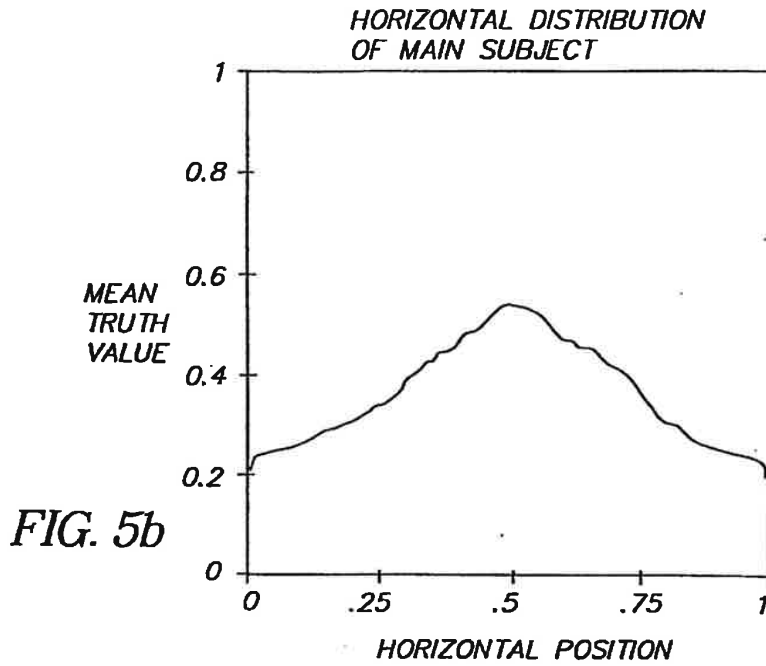


Fig. 5a



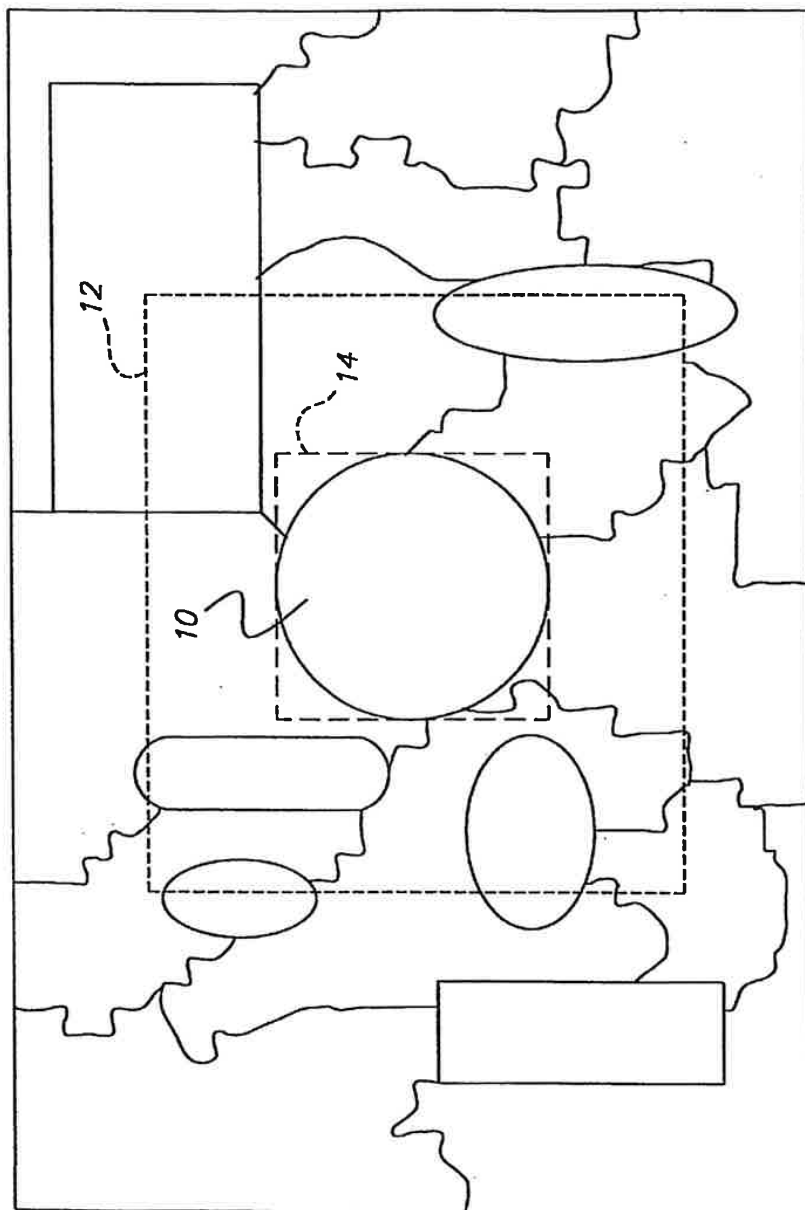


FIG. 6

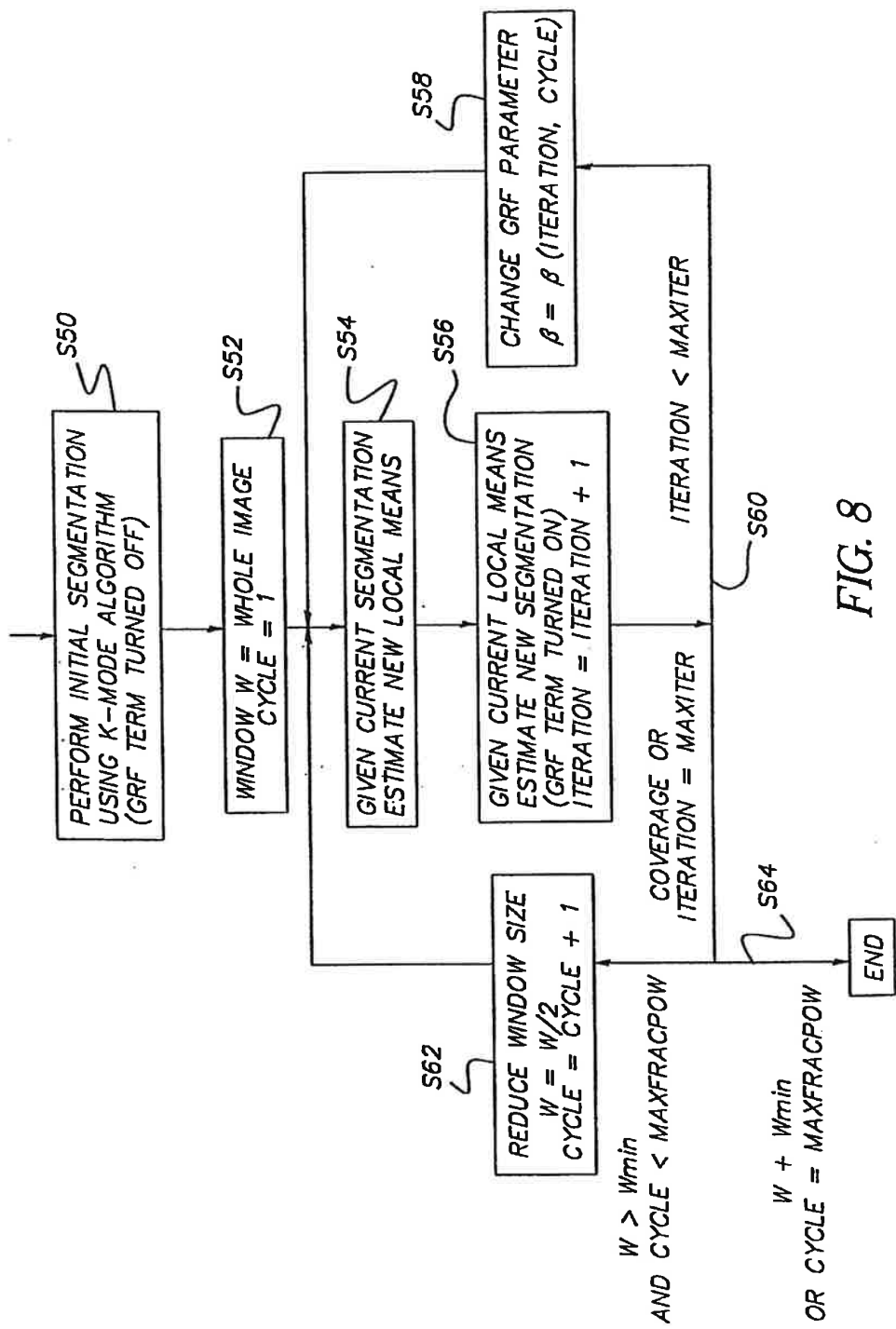


FIG. 8

## METHOD FOR AUTOMATIC DETERMINATION OF MAIN SUBJECTS IN PHOTOGRAPHIC IMAGES

### FIELD OF THE INVENTION

The invention relates generally to the field of digital image processing and, more particularly, to locating main subjects, or equivalently, regions of photographic interest in a digital image.

### BACKGROUND OF THE INVENTION

In photographic pictures, a main subject is defined as what the photographer tries to capture in the scene. The first-party truth is defined as the opinion of the photographer and the third-party truth is defined as the opinion from an observer other than the photographer and the subject (if applicable). In general, the first-party truth typically is not available due to the lack of specific knowledge that the photographer may have about the people, setting, event, and the like. On the other hand, there is, in general, good agreement among third-party observers if the photographer has successfully used the picture to communicate his or her interest in the main subject to the viewers. Therefore, it is possible to design a method to automatically perform the task of detecting main subjects in images.

Main subject detection provides a measure of saliency or relative importance for different regions that are associated with different subjects in an image. It enables a discriminative treatment of the scene contents for a number of applications. The output of the overall system can be modified versions of the image, semantic information, and action.

The methods disclosed by the prior art can be put in two major categories. The first category is considered "pixel-based" because such methods were designed to locate interesting pixels or "spots" or "blocks", which usually do not correspond to entities of objects or subjects in an image. The second category is considered "region-based" because such methods were designed to locate interesting regions, which correspond to entities of objects or subjects in an image.

Most pixel-based approaches to region-of-interest detection are essentially edge detectors. V. D. Gesu, et al., "Local operators to detect regions of interest," *Pattern Recognition Letters*, vol. 18, pp. 1077-1081, 1997, used two local operators based on the computation of local moments and symmetries to derive the selection. Arguing that the performance of a visual system is strongly influenced by information processing done at early vision stage, two transforms named the discrete moment transform (DMT) and discrete symmetry transform (DST) are computed to measure local central moments about each pixel and local radial symmetry. In order to exclude trivial symmetry cases, nonuniform region selection is needed. The specific DMT operator acts like a detector of prominent edges (occlusion boundaries) and the DST operator acts like a detector of symmetric blobs. The results from the two operators are combined via logic "AND" operation. Some morphological operations are needed to dilate the edge-like raw output map generated by the DMT operator.

R. Milanese, Detecting salient regions in an image: From biology to implementation, PhD thesis, University of Geneva, Switzerland, 1993, developed a computational model of visual attention, which combines knowledge about the human visual system with computer vision techniques. The model is structured into three major stages. First, multiple feature maps are extracted from the input image (for examples, orientation, curvature, color contrast and the

like). Second, a corresponding number of "conspicuity" maps are computed using a derivative of Gaussian model, which enhance regions of interest in each feature map. Finally, a nonlinear relaxation process is used to integrate the conspicuity maps into a single representation by finding a compromise among inter-map and intra-map inconsistencies. The effectiveness of the approach was demonstrated using a few relatively simple images with remarkable regions of interest.

To determine an optimal tonal reproduction, J. R. Boyack, et al., U.S. Pat. No. 5,724,456, developed a system that partitions the image into blocks, combines certain blocks into sectors, and then determines a difference between the maximum and minimum average block values for each sector. A sector is labeled an active sector if the difference exceeds a pre-determined threshold value. All weighted counts of active sectors are plotted versus the average luminance sector values in a histogram, which is then shifted via some predetermined criterion so that the average luminance sector value of interest will fall within a destination window corresponding to the tonal reproduction capability of a destination application.

In summary, this type of pixel-based approach does not explicitly detect region of interest corresponding to semantically meaningful subjects in the scene. Rather, these methods attempt to detect regions where certain changes occur in order to direct attention or gather statistics about the scene.

X. Marichal, et al., "Automatic detection of interest areas of an image or of a sequence of images," in *Proc. IEEE Int. Conf. Image Process.*, 1996, developed a fuzzy logic-based system to detect interesting areas in a video sequence. A number of subjective knowledge-based interest criteria were evaluated for segmented regions in an image. These criteria include: (1) an interaction criterion (a window predefined by a human operator); (2) a border criterion (rejecting of regions having large number of pixels along the picture borders); (3) a face texture criterion (de-emphasizing regions whose texture does not correspond to skin samples); (4) a motion criterion (rejecting regions with no motion and low gradient or regions with very large motion and high gradient); and (5) a continuity criterion (temporal stability in motion). The main application of this method is for directing the resources in video coding, in particular for videophone or videoconference. It is clear that motion is the most effective criterion for this technique targeted at video instead of still images. Moreover, the fuzzy logic functions were designed in an ad hoc fashion. Lastly, this method requires a window predefined by a human operator, and therefore is not fully automatic.

W. Osberger, et al., "Automatic identification of perceptually important regions in an image," in *Proc. IEEE Int. Conf. Pattern Recognition*, 1998, evaluated several features known to influence human visual attention for each region of a segmented image to produce an importance value for each feature in each region. The features mentioned include low-level factors (contrast, size, shape, color, motion) and higher level factors (location, foreground/background, people, context), but only contrast, size, shape, location and foreground/background (determining background by determining the proportion of total image border that is contained in each region) were implemented. Moreover, this method chose to treat each factor as being of equal importance by arguing that (1) there is little quantitative data which indicates the relative importance of these different factors and (2) the relative importance is likely to change from one image to another. Note that segmentation was obtained using the split-and-merge method based on 8x8 image blocks and

this segmentation method often results in over-segmentation and blotchiness around actual objects.

Q. Huang, et al., "Foreground/background segmentation of color images by integration of multiple cues," in *Proc. IEEE Int. Conf. Image Process.*, 1995, addressed automatic segmentation of color images into foreground and background with the assumption that background regions are relatively smooth but may have gradually varying colors or be lightly textured. A multi-level segmentation scheme was devised that included color clustering, unsupervised segmentation based on MDL (Minimum Description Length) principle, edge-based foreground/background separation, and integration of both region and edge-based segmentation. In particular, the MDL-based segmentation algorithm was used to further group the regions from the initial color clustering, and the four corners of the image were used to adaptively determine an estimate of the background gradient magnitude. The method was tested on around 100 well-composed images with prominent main subject centered in the image against large area of the assumed type of uncluttered background.

T. F. Syeda-Mahmood, "Data and model-driven selection using color regions," *Int. J. Comput. Vision*, vol. 21, no. 1, pp. 9-36, 1997, proposed a data-driven region selection method using color region segmentation and region-based saliency measurement. A collection of 220 primary color categories was pre-defined in the form of a color LUT (look-up-table). Pixels are mapped to one of the color categories, grouped together through connected component analysis, and further merged according to compatible color categories. Two types of saliency measures, namely self-saliency and relative saliency, are linearly combined using heuristic weighting factors to determine the overall saliency. In particular, self-saliency included color saturation, brightness and size while relative saliency included color contrast (defined by CIE distance) and size contrast between the concerned region and the surrounding region that is ranked highest among neighbors by size, extent and contrast in successive order.

In summary, almost all of these reported methods have been developed for targeted types of images: video-conferencing or TV news broadcasting images, where the main subject is a talking person against a relatively simple static background (Osberg, Marichal); museum images, where there is a prominent main subject centered in the image against large area of relatively clean background (Huang); and toy-world images, where the main subject are a few distinctively colored and shaped objects (Milanese, Syeda). These methods were either not designed for unconstrained photographic images, or even if designed with generic principles were only demonstrated for their effectiveness on rather simple images. The criteria and reasoning processes used were somewhat inadequate for less constrained images, such as photographic images.

#### SUMMARY OF THE INVENTION

It is an object of this invention to provide a method for detecting the location of main subjects within a digitally captured image and thereby overcoming one or more problems set forth above.

It is also an object of this invention to provide a measure of belief for the location of main subjects within a digitally captured image and thereby capturing the intrinsic degree of uncertainty in determining the relative importance of different subjects in an image. The output of the algorithm is in the form of a list of segmented regions ranked in a descending

order of their likelihood as potential main subjects for a generic or specific application. Furthermore, this list can be converted into a map in which the brightness of a region is proportional to the main subject belief of the region.

It is also an object of this invention to use ground truth data. Ground truth, defined as human outlined main subjects, is used to feature selection and training the reasoning engine.

It is also an object of this invention to provide a method of finding main subjects in an image in an automatic manner.

It is also an object of this invention to provide a method of finding main subjects in an image with no constraints or assumptions on scene contents.

It is further an object of the invention to use the main subject location and main subject belief to obtain estimates of the scene characteristics.

The present invention comprises the steps of:

- a) receiving a digital image;
- b) extracting regions of arbitrary shape and size defined by actual objects from the digital image;
- c) grouping the regions into larger segments corresponding to physically coherent objects;
- d) extracting for each of the regions at least one structural saliency feature and at least one semantic saliency feature; and,
- e) integrating saliency features using a probabilistic reasoning engine into an estimate of a belief that each region is the main subject.

The above and other objects of the present invention will become more apparent when taken in conjunction with the following description and drawings wherein identical reference numerals have been used, where possible, to designate identical elements that are common to the figures.

#### ADVANTAGEOUS EFFECT OF THE INVENTION

The present invention has the following advantages of: a robust image segmentation method capable of identifying object regions of arbitrary shapes and sizes, based on physics-motivated adaptive Bayesian clustering and non-purposive grouping;

emphasis on perceptual grouping capable of organizing regions corresponding to different parts of physically coherent subjects;

utilization of a non-binary representation of the ground truth, which capture the inherent uncertainty in determining the belief of main subject, to guide the design of the system;

a rigorous, systematic statistical training mechanism to determine the relative importance of different features through ground truth collection and contingency table building;

extensive, robust feature extraction and evidence collection;

combination of structural saliency and semantic saliency, the latter facilitated by explicit identification of key foreground- and background- subject matters;

combination of self and relative saliency measures for structural saliency features; and,

a robust Bayes net-based probabilistic inference engine suitable for integrating incomplete information.

#### BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a perspective view of a computer system for implementing the present invention;

FIG. 2 is a block diagram illustrating a software program of the present invention;

FIG. 3 is an illustration of the sensitivity characteristic of a belief sensor with sigmoidal shape used in the present invention;

FIG. 4 is an illustration of the location PDF with unknown-orientation, FIG. 4(a) is an illustration of the PDF in the form of a 2D function, FIG. 4(b) is an illustration of the PDF in the form of its projection along the width direction, and FIG. 4(c) is an illustration of the PDF in the form of its projection along the height direction;

FIG. 5 is an illustration of the location PDF with known-orientation, FIG. 5(a) is an illustration of the PDF in the form of a 2D function, FIG. 5(b) is an illustration of the PDF in the form of its projection along the width direction, and FIG. 5(c) is an illustration of the PDF in the form of its projection along the height direction;

FIG. 6 is an illustration of the computation of relative saliency for the central circular region using an extended neighborhood as marked by the box of dotted line;

FIG. 7 is an illustration of a two level Bayes net used in the present invention; and,

FIG. 8 is block diagram of a preferred segmentation method.

#### DETAILED DESCRIPTION OF THE INVENTION

In the following description, the present invention will be described in the preferred embodiment as a software program. Those skilled in the art will readily recognize that the equivalent of such software may also be constructed in hardware.

Still further, as used herein, computer readable storage medium may comprise, for example; magnetic storage media such as a magnetic disk (such as a floppy disk) or magnetic tape; optical storage media such as an optical disc, optical tape, or machine readable bar code; solid state electronic storage devices such as random access memory (RAM), or read only memory (ROM); or any other physical device or medium employed to store a computer program.

Referring to FIG. 1, there is illustrated a computer system 10 for implementing the present invention. Although the computer system 10 is shown for the purpose of illustrating a preferred embodiment, the present invention is not limited to the computer system 10 shown, but may be used on any electronic processing system. The computer system 10 includes a microprocessor based unit 20 for receiving and processing software programs and for performing other processing functions. A touch screen display 30 is electrically connected to the microprocessor based unit 20 for displaying user related information associated with the software, and for receiving user input via touching the screen. A keyboard 40 is also connected to the microprocessor based unit 20 for permitting a user to input information to the software. As an alternative to using the keyboard 40 for input, a mouse 50 may be used for moving a selector 52 on the display 30 and for selecting an item on which the selector 52 overlays, as is well known in the art.

A compact disk-read only memory (CD-ROM) 55 is connected to the microprocessor based unit 20 for receiving software programs and for providing a means of inputting the software programs and other information to the microprocessor based unit 20 via a compact disk 57, which typically includes a software program. In addition, a floppy disk 61 may also include a software program, and is inserted

into the microprocessor based unit 20 for inputting the software program. Still further, the microprocessor based unit 20 may be programmed, as is well known in the art, for storing the software program internally. A printer 56 is connected to the microprocessor based unit 20 for printing a hardcopy of the output of the computer system 10.

Images may also be displayed on the display 30 via a personal computer card (PC card) 62 or, as it was formerly known, a personal computer memory card international association card (PCMCIA card) which contains digitized images electronically embodied on the card 62. The PC card 62 is ultimately inserted into the microprocessor based unit 20 for permitting visual display of the image on the display 30.

Referring to FIG. 2, there is shown a block diagram of an overview of the present invention. First, an input image of a natural scene is acquired and stored S0 in a digital form. Then, the image is segmented S2 into a few regions of homogeneous properties. Next, the region segments are grouped into larger regions based on similarity measures S4 through non-purposive perceptual grouping, and further grouped into larger regions corresponding to perceptually coherent objects S6 through purposive grouping (purposive grouping concerns specific objects). The regions are evaluated for their saliency S8 using two independent yet complementary types of saliency features—structural saliency features and semantic saliency features. The structural saliency features, including a set of low-level early vision features and a set of geometric features, are extracted S8a, which are further processed to generate a set of self-saliency features and a set of relative saliency features. Semantic saliency features in the forms of key subject matters, which are likely to be part of either foreground (for example, people) or background (for example, sky, grass), are detected S8b to provide semantic cues as well as scene context cues. The evidences of both types are integrated S10 using a reasoning engine based on a Bayes net to yield the final belief map of the main subject S12.

To the end of semantic interpretation of images, a single criterion is clearly insufficient. The human brain, furnished with its a priori knowledge and enormous memory of real world subjects and scenarios, combines different subjective criteria in order to give an assessment of the interesting or primary subject(s) in a scene. The following extensive list of features are believed to have influences on the human brain in performing such a somewhat intangible task as main subject detection: location, size, brightness, colorfulness, texturefulness, key subject matter, shape, symmetry, spatial relationship (surroundedness/occlusion), borderness, indoor/outdoor, orientation, depth (when applicable), and motion (when applicable for video sequence).

In the present invention, the low-level early vision features include color, brightness, and texture. The geometric features include location (centrality), spatial relationship (borderness, adjacency, surroundedness, and occlusion), size, shape, and symmetry. The semantic features include flesh, face, sky, grass, and other green vegetation. Those skilled in the art can define more features without departing from the scope of the present invention.

S2: Region Segmentation

The adaptive Bayesian color segmentation algorithm (Luo et al., "Towards physics-based segmentation of photographic color images," Proceedings of the IEEE International Conference on Image Processing, 1997) is used to generate a tractable number of physically coherent regions of arbitrary shape. Although this segmentation method is preferred, it will be appreciated that a person of ordinary skill in the art can use a different segmentation method to



obtain object regions of arbitrary shape without departing from the scope of the present invention. Segmentation of arbitrarily shaped regions provides the advantages of: (1) accurate measure of the size, shape, location of and spatial relationship among objects; (2) accurate measure of the color and texture of objects; and (3) accurate classification of key subject matters.

Referring to FIG. 8, there is shown a block diagram of the preferred segmentation algorithm. First, an initial segmentation of the image into regions is obtained S50. A color histogram of the image is computed and then partitioned into a plurality of clusters that correspond to distinctive, prominent colors in the image. Each pixel of the image is classified to the closest cluster in the color space according to a preferred physics-based color distance metric with respect to the mean values of the color clusters (Luo et al., "Towards physics-based segmentation of photographic color images," Proceedings of the IEEE International Conference on Image Processing, 1997). This classification process results in an initial segmentation of the image. A neighborhood window is placed at each pixel in order to determine what neighborhood pixels are used to compute the local color histogram for this pixel. The window size is initially set at the size of the entire image S52, so that the local color histogram is the same as the one for the entire image and does not need to be recomputed. Next, an iterative procedure is performed between two alternating processes: re-computing S54 the local mean values of each color class based on the current segmentation, and re-classifying the pixels according to the updated local mean values of color classes S56. This iterative procedure is performed until a convergence is reached S60. During this iterative procedure, the strength of the spatial constraints can be adjusted in a gradual manner S58 (for example, the value of  $\beta$ , which indicates the strength of the spatial constraints, is increased linearly with each iteration). After the convergence is reached for a particular window size, the window used to estimate the local mean values for color classes is reduced by half in size S62. The iterative procedure is repeated for the reduced window size to allow more accurate estimation of the local mean values for color classes. This mechanism introduces spatial adaptivity into the segmentation process. Finally, segmentation of the image is obtained when the iterative procedure reaches convergence for the minimum window size S64. S4 & S6: Perceptual Grouping

The segmented regions may be grouped into larger segments that consist of regions that belong to the same object. Perceptual grouping can be non-purposive and purposive. Referring to FIG. 2, non-purposive perceptual grouping S4 can eliminate over-segmentation due to large illumination differences, for example, a table or wall with remarkable illumination falloff over a distance. Purposive perceptual grouping S6 is generally based on smooth, noncoincidental connection of joints between parts of the same object, and in certain cases models of typical objects (for example, a person has head, torso and limbs).

Perceptual grouping facilitates the recognition of high-level vision features. Without proper perceptual grouping, it is difficult to perform object recognition and proper assessment of such properties as size and shape. Perceptual grouping includes: merging small regions into large regions based on similarity in properties and compactness of the would-be merged region (non-purposive grouping); and grouping parts that belong to the same object based on commonly shared background, compactness of the would-be merged region, smoothness in contour connection between regions, and model of specific object (purposive grouping).

#### S8: Feature Extraction

For each region, an extensive set of features, which are shown to contribute to visual attention, are extracted and associated evidences are then computed. The list of features consists of three categories—low-level vision features, geometric features, and semantic features. For each feature, either or both of a self-saliency feature and a relative saliency feature are computed. The self-saliency is used to capture subjects that stand out by themselves (for example, in color, texture, location and the like), while the relative saliency is used to capture subjects that are in high contrast to their surrounding (for example, shape). Furthermore, raw measurements of features, self-salient or relatively salient, are converted into evidences, whose values are normalized to be within  $[0, 1.0]$ , by belief sensor functions with appropriate nonlinearity characteristics. Referring to FIG. 3, there is shown a sigmoid-shaped belief sensor function used in the present invention. A raw feature measurement that has a value between a minimum value and a maximum value is mapped to a belief value within  $[0, 1]$ . A Gaussian-shaped belief sensor function (not shown) is also used for some features, as will be described hereinbelow.

#### Structural Saliency Features

Structural saliency features include individually or in combination self saliency features and relative saliency features.

Referring to FIG. 6, an extended neighborhood is used to compute relative saliency features. First, a minimum bounding rectangle (MBR) 14 of a region of concern 10 (shown by the central circular region) is determined. Next, this MBR is extended in all four directions (stopping at the image borders wherever applicable) of the region using an appropriate factor (for example, 2). All regions intersecting this stretched MBR 12, which is indicated by the dotted lines, are considered neighbors of the region. This extended neighborhood ensures adequate context as well natural scalability for computing the relative saliency features.

The following structural saliency features are computed:

contrast in hue (a relative saliency feature)

In terms of color, the contrast in hue between an object and its surrounding is a good indication of the saliency in color.

$$\text{contrast}_{\text{color}} = \sum_{\text{neighborhood}} \frac{||\text{hue} - \text{hue}_{\text{surrounding}}||}{\text{hue}_{\text{surrounding}}} \quad (1)$$

where the neighborhood refers to the context previously defined and henceforth.

colorfulness (a self-saliency feature) and contrast in colorfulness (a relative saliency feature)

In terms of colorfulness, the contrast between a colorful object and a dull surrounding is almost as good an indicator as the contrast between a dull object and a colorful surrounding. Therefore, the contrast in colorfulness should always be positive. In general, it is advantageous to treat a self saliency and the corresponding relative saliency as separate features rather than combining them using certain heuristics. The influence of each feature will be determined separately by the training process, which will be described later.

$$\text{colorfulness} = \text{saturation} \quad (2)$$

$$\text{contrast}_{\text{colorfulness}} = \frac{\| \text{saturation} - \text{saturation}_{\text{surrounding}} \|}{\text{saturation}_{\text{surrounding}}} \quad (3)$$

brightness (a self-saliency feature) and contrast in brightness (a relative saliency feature)

In terms of brightness, the contrast between a bright object and a dark surrounding is almost as good as the contrast between a dark object and a bright surrounding. In particular, the main subject tends to be lit up in flash scenes.

$$\text{brightness} = \text{luminance} \quad (4)$$

$$\text{contrast}_{\text{brightness}} = \frac{|\text{brightness} - \text{brightness}_{\text{surrounding}}|}{\text{brightness}_{\text{surrounding}}} \quad (5)$$

texturefulness (a self-saliency feature) and contrast in texturefulness (a relative saliency feature)

In terms of texturefulness, in general, a large uniform region with very little texture tends to be the background. On the other hand, the contrast between a highly textured object and a nontextured or less textured surrounding is a good indication of main subjects. The same holds for a nontextured or less textured object and a highly textured surrounding.

$$\text{texturefulness} = \text{texture\_energy} \quad (6)$$

$$\text{contrast}_{\text{texturefulness}} = \frac{|\text{texturefulness} - \text{texturefulness}_{\text{surrounding}}|}{\text{texturefulness}_{\text{surrounding}}} \quad (7)$$

location (a self-saliency feature)

In terms of location, the main subject tends to be located near the center instead of the peripheral of the image, though not necessarily right in the center of the image. In fact, professional photographers tend to position the main subject at the horizontal gold partition positions.

The centroid of a region alone is usually not sufficient to indicate the location of the region without any indication of its size and shape. A centrality measure is defined by computing the integral of a probability density function (PDF) over the area of a given region. The PDF is derived from a set of training images, in which the main subject regions are manually outlined, by summing up the ground truth maps over the entire training set. In other words, the PDF represents the distribution of main subjects in terms of location. A more important advantage of this centrality measure is that every pixel of a given region, not just the centroid, contributes to the centrality measure of the region to a varying degree depending on its location.

$$\text{centrality} = \frac{1}{N_R} \sum_{(x,y) \in R} \text{PDF}_{\text{MSD\_location}}(x, y) \quad (8)$$

where  $(x,y)$  denotes a pixel in the region  $R$ ,  $N_R$  is the number of pixels in region  $R$ , and  $\text{PDF}_{\text{MSD\_location}}$  denotes a 2D probability density function (PDF) of main subject location. If the orientation is unknown, the PDF is symmetric about the center of the image in both vertical and horizontal directions, which results in an orientation-independent centrality measure. An orientation-unaware PDF is shown in FIG. 4(a) and the projection in the width and height directions are also shown in FIG. 4(b) and FIG. 4(c), respectively. If the orientation is known, the PDF is symmetric about the

center of the image in the horizontal direction but not in the vertical direction, which results in an orientation-aware centrality measure. An orientation-aware PDF is shown in FIG. 5(a) and the projection in the horizontal and vertical directions are also shown in FIG. 5(b) and FIG. 5(c), respectively.

size (a self-saliency feature)

Main subjects should have considerable but reasonable sizes. However, in most cases, very large regions or regions that span at least one spatial direction (for example, the horizontal direction) are most likely to be background regions, such as sky, grass, wall, snow, or water. In general, both very small and very large regions should be discounted.

$$\text{size} = \begin{cases} 0 & \text{if } s > s4 \\ 1 - \frac{s-s2}{s3-s2} & \text{if } s > s3 \text{ and } s < s4 \\ 1 & \text{if } s > s2 \text{ and } s < s3 \\ \frac{s-s1}{s2-s1} & \text{if } s > s1 \text{ and } s < s2 \\ 0 & \text{if } s < s1 \end{cases} \quad (9)$$

where  $s1, s2, s3,$  and  $s4$  are predefined threshold ( $s1 < s2 < s3 < s4$ ).

In practice, the size of a region is measured as a fraction of the entire image size to achieve invariance to scaling.

$$\text{size} = \frac{\text{region\_pixels}}{\text{image\_pixels}} \quad (10)$$

In this invention, the region size is classified into one of three bins, labeled "small," "medium" and "large" using two thresholds  $s2$  and  $s3$ , where  $s2 < s3$ .

shape (a self-saliency feature) and contrast in shape (a relative saliency feature)

In general, objects that have distinctive geometry and smooth contour tend to be man-made and thus have high likelihood to be main subjects. For example, square, round, elliptic, or triangle shaped objects. In some cases, the contrast in shape indicates conspicuity (for example, a child among a pool of bubble balls).

The shape features are divided into two categories, self salient and relatively salient. Self salient features characterize the shape properties of the regions themselves and relatively salient features characterize the shape properties of the regions in comparison to those of neighboring regions.

The aspect ratio of a region is the major axis/minor axis of the region. A Gaussian belief function maps the aspect ratio to a belief value. This feature detector is used to discount long narrow shapes from being part of the main subject.

Three different measures are used to characterize the convexity of a region: (1) perimeter-based—perimeter of the convex hull divided by the perimeter of region; (2) area-based—area of region divided by the area of the convex hull; and (3) hyperconvexity—the ratio of the perimeter-based convexity and area-based convexity. In general, an object of complicated shape has a hyperconvexity greater than 1.0. The three convexity features measure the compactness of the region. Sigmoid belief functions are used to map the convexity measures to beliefs.

The rectangularity is the area of the MBR of a region divided by the area of the region. A sigmoid belief function maps the rectangularity to a belief value. The circularity is the square of the perimeter of the region divided by the area of region. A sigmoid belief function maps the circularity to a belief value.

Relative shape-saliency features include relative rectangularity, relative circularity and relative convexity. In particular, each of these relative shape features is defined as the average difference between the corresponding self salient shape feature of the region and those of the neighborhood regions, respectively. Finally, a Gaussian function is used to map the relative measures to beliefs.

symmetry (a self-saliency feature)

Objects of striking symmetry, natural or artificial, are also likely to be of great interest. Local symmetry can be computed using the method described by V. D. Gesu, et al., "Local operators to detect regions of interest," *Pattern Recognition Letters*, vol. 18, pp. 1077-1081, 1997.

spatial relationship (a relative saliency feature)

In general, main subjects tend to be in the foreground. Consequently, main subjects tend to share boundaries with a lot of background regions (background clutter), or be enclosed by large background regions such as sky, grass, snow, wall and water, or occlude other regions. These characteristics in terms of spatial relationship may reveal the region of attention. Adjacency, surroundedness and occlusion are the main features in terms of spatial relationship. In many cases, occlusion can be inferred from T-junctions (L. R. Williams, "Perceptual organization of occluding contours," in *Proc. IEEE Int. Conf Computer Vision*, 1990) and fragments can be grouped based on the principle of perceptual occlusion (J. August, et al., "Fragment grouping via the principle of perceptual occlusion," in *Proc. IEEE Int. Conf Pattern Recognition*, 1996).

In particular, a region that is nearly completely surrounded by a single other region is more likely to be the main subject. Surroundedness is measured as the maximum fraction of the region's perimeter that is shared with any one neighboring region. A region that is totally surrounded by a single other region has the highest possible surroundedness value of 1.0.

$$\text{surroundedness} = \frac{\max_{\text{neighbors}} \text{length\_of\_common\_border}}{\text{region\_perimeter}} \quad (11)$$

borderness (a self-saliency feature)

Many background regions tend to contact one or more of the image borders. In other words, a region that has significant amount of its contour on the image borders tends to belong to the background. The percentage of the contour points on the image borders and the number of image borders shared (at most four) can be good indications of the background.

In the case where the orientation is unknown, one borderness feature places each region in one of six categories determined by the number and configuration of image borders the region is "in contact" with. A region is "in contact" with a border when at least one pixel in the region falls within a fixed distance of the border of the image. Distance is expressed as a fraction of the shorter dimension of the image. The six categories for borderness\_a are defined in Table 1.

TABLE 1

Categories for orientation-independent borderness_a.	
Category	The region is in contact with . . .
0	none of the image borders
1	exactly one of the image borders
2	exactly two of the image borders, adjacent to one another
3	exactly two of the image borders, opposite to one another

TABLE 1-continued

Categories for orientation-independent borderness_a.	
Category	The region is in contact with . . .
4	exactly three of the image borders
5	exactly four (all) of the image borders

Knowing the proper orientation of the image allows us to refine the borderness feature to account for the fact that regions in contact with the top border are much more likely to be background than regions in contact with the bottom. This feature places each region in one of 12 categories determined by the number and configuration of image borders the region is "in contact" with, using the definition of "in contact with" from above. The four borders of the image are labeled as "Top", "Bottom", "Left", and "Right" according to their position when the image is oriented with objects in the scene standing upright. In this case, the twelve categories for borderness\_b are defined in Table 2, which lists each possible combination of borders a region may be in contact with, and gives the category assignment for that combination.

TABLE 2

Categories for orientation-dependent borderness_a.				
The region is in contact with . . .				Category
Top	Bottom	Left	Right	Category
N	N	N	N	0
N	Y	N	N	1
Y	N	N	N	2
N	N	Y	N	3
N	N	N	Y	3
N	Y	Y	N	4
N	Y	N	Y	4
Y	N	N	N	5
Y	N	N	N	5
Y	Y	N	N	6
N	N	Y	Y	7
N	Y	Y	Y	8
Y	Y	Y	N	9
Y	Y	N	Y	9
Y	N	Y	Y	10
Y	Y	Y	Y	11

Regions that include a large fraction of the image border are also likely to be background regions. This feature indicates what fraction of the image border is in contact with the given region.

$$\text{borderness}_b = \frac{\text{perimeter\_pixels\_in\_this\_region}}{2 * (\text{image\_height} + \text{image\_width} - 2)} \quad (12)$$

When a large fraction of the region perimeter is on the image border, a region is also likely to be background. Such a ratio is unlikely to exceed 0.5, so a value in the range [0,1] is obtained by scaling the ratio by a factor of 2 and saturating the ratio at the value of 1.0.

$$\text{borderness}_c = \frac{\min(1, 2 * \text{num\_region\_perimeter\_pixels\_on\_border})}{\text{region\_perimeter}} \quad (13)$$

Again, note that instead of a composite borderness measure based on heuristics, all the above three borderness measures are separately trained and used in the main subject detection.

Semantic Saliency Features

flesh/face/people (foreground, self saliency features)

A majority of photographic images have people and about the same number of images have sizable faces in them. In conjunction with certain shape analysis and pattern analysis, some detected flesh regions can be identified as faces. Subsequently, using models of human figures, flesh detection and face a detection can lead to clothing detection and eventually people detection.

The current flesh detection algorithm utilizes color image segmentation and a pre-determined flesh distribution in a chrominance space (Lee, "Color image quantization based on physics and psychophysics," Journal of Society of Photographic Science and Technology of Japan, Vol. 59, No. 1, pp. 212-225, 1996). The flesh region classification is based on Maximum Likelihood Estimation (MLE) according to the average color of a segmented region. The conditional probabilities are mapped to a belief value via a sigmoid belief function.

A primitive face detection algorithm is used in the present invention. It combines the flesh map output by the flesh detection algorithm with other face heuristics to output a belief in the location of faces in an image. Each region in an image that is identified as a flesh region is fitted with an ellipse. The major and minor axes of the ellipse are calculated as also the number of pixels in the region outside the ellipse and the number of pixels in the ellipse not part of the region. The aspect ratio is computed as a ratio of the major axis to the minor axis. The belief for the face is a function of the aspect ratio of the fitted ellipse, the area of the region outside the ellipse, and the area of the ellipse not part of the region. A Gaussian belief sensor function is used to scale the raw function outputs to beliefs.

It will be appreciated that a person of ordinary skill in the art can use a different face detection method without departing from the present invention.

key background subject matters (self saliency features)

There are a number of objects that frequently appear in photographic images, such as sky, cloud, grass, tree, foliage, vegetation, water body (river, lake, pond), wood, metal, and the like. Most of them have high likelihood to be background objects. Therefore, such objects can be ruled out while they also serve as precursors for main subjects as well as scene types.

Among these background subject matters, sky and grass (may include other green vegetation) are detected with relatively high confidence due to the amount of constancy in terms of their color, texture, spatial extent, and spatial location.

Probabilistic Reasoning

All the saliency features are integrated by a Bayes net to yield the likelihood of main subjects. On one hand, different evidences may compete with or contradict each other. On the other hand, different evidences may mutually reinforce each other according to prior models or knowledge of typical photographic scenes. Both competition and reinforcement are resolved by the Bayes net-based inference engine.

A Bayes net (J. Pearl, *Probabilistic Reasoning in Intelligent Systems*, San Francisco, Calif.: Morgan Kaufmann, 1988) is a directed acyclic graph that represents causality relationships between various entities in the graph. The direction of links represents causality. It is an evaluation means knowing joint Probability Distribution Function (PDF) among various entities. Its advantages include explicit uncertainty characterization, fast and efficient computation, quick training, high adaptivity and ease of building, and representing contextual knowledge in human reasoning framework. A Bayes net consists of four components:

1. Priors: The initial beliefs about various nodes in the Bayes net
2. Conditional Probability Matrices (CPMs): the statistical relationship between two connected nodes in the Bayes net
3. Evidences: Observations from feature detectors that are input to the Bayes net
4. Posteriors: The final computed beliefs after the evidences have been propagated through the Bayes net.

Referring to FIG. 7, a two-level Bayesian net is used in the present invention that assumes conditional independence between various feature detectors. The main subject is determined at the root node 20 and all the feature detectors are at the leaf nodes 22. There is one Bayes net active for each region (identified by the segmentation algorithm) in the image. The root node gives the posterior belief in that region being part of the main subject. It is to be understood that the present invention can be used with a Bayes net that has more than two levels without departing from the scope of the present invention.

Training Bayes Nets

One advantage of Bayes nets is each link is assumed to be independent of links at the same level. Therefore, it is convenient for training the entire net by training each link separately, i.e., deriving the CPM for a given link independent of others. In general, two methods are used for obtaining CPM for each root-feature node pair:

1. Using Expert Knowledge

This is an ad-hoc method. An expert is consulted to obtain the conditional probabilities of each feature detector observing the main subject given the main subject.

2. Using Contingency Tables

This is a sampling and correlation method. Multiple observations of each feature detector are recorded along with information about the main subject. These observations are then compiled together to create contingency tables which, when normalized, can then be used as the CPM. This method is similar to neural network type of training (learning). This method is preferred in the present invention.

Consider the CPM for centrality as an example. This matrix was generated using contingency tables derived from the ground truth and the feature detector. Since the feature detector in general does not supply a binary decision (referring to Table 3), fractional frequency count is used in deriving the CPM. The entries in the CPM are determined by

$$CPM = \left( \sum_{i \in I} \sum_{r \in R_i} n_i F_r^T T_r \right) P \tag{14}$$

$$F_r = [f_0^r \ f_1^r \ \dots \ f_M^r]$$

$$T_r = [t_0^r \ t_1^r \ \dots \ t_L^r]$$

$$P = \text{diag}\{p_j\}$$

$$p_j = \left( \sum_{i \in I} \sum_{r \in R_i} n_i t_r^j \right)^{-1}$$

where I is the set of all training images,  $R_i$  is the set of all regions in image i,  $n_i$  is the number of observations (observers) for image i. Moreover,  $F_r$  represents an M-label feature vector for region r,  $T_r$  represents an L-level ground-truth vector, and P denotes an LxL diagonal matrix of normalization constant factors. For example, in Table 3, regions 1, 4, 5 and 7 contribute to boxes 00, 11, 10 and 01 in Table 4, respectively. Note that all the belief values have been normalized by the proper belief sensors. As an intuitive interpretation of the first column of the CPM for centrality,

a "central" region is about twice as likely to be the main subject than not a main subject.

TABLE 3

An example of training the CPM.			
Region Number	Ground Truth	Feature Detector Output	Contribution
1	0	0.017	00
2	0	0.211	00
3	0	0.011	00
4	0.933	0.953	11
5	0	0.673	10
6	1	0.891	11
7	0.93	0.072	01
8	1	0.091	01

TABLE 4

The trained CPM.		
	Feature = 1	feature = 0
Main subject = 1	0.35 (11)	0.65 (01)
Main subject = 0	0.17 (10)	0.83 (00)

The output of the algorithm is in the form of a list of segmented regions ranked in a descending order of their likelihood as potential main subjects for a generic or specific application. Furthermore, this list can be converted into a map in which the brightness of a region is proportional to the main subject belief of the region. This "belief" map is more than a binary map that only indicates location of the determined main subject. The associated likelihood is also attached to each region so that the regions with large brightness values correspond to regions with high confidence or belief being part of the main subject. This reflects the inherent uncertainty for humans to perform such a task. However, a binary decision, when desired, can be readily obtained by applying an appropriate threshold to the belief map. Moreover, the belief information may be very useful for downstream applications. For example, different weighting factors can be assigned to different regions in determining bit allocation for image coding.

What is claimed is:

1. A method for detecting a main subject in an image, the method comprising the steps of:
  - a) receiving a digital image;
  - b) extracting regions of arbitrary shape and size defined by actual objects from the digital image;
  - c) extracting for each of the regions at least one structural saliency feature and at least one semantic saliency feature; and,
  - d) integrating the structural saliency feature and the semantic feature using a probabilistic reasoning engine into an estimate of a belief that each region is the main subject.
2. The method as in claim 1, wherein step (b) includes using a color distance metric defined in a color space, a spatial homogeneity constraint, and a mechanism for permitting spatial adaptivity.
3. The method as in claim 1, wherein the structural saliency feature of step (c) includes at least one of a low-level vision feature and a geometric feature.
4. The method as in claim 1, wherein step (c) includes using either individually or in combination a color, brightness and/or texture as a low-level vision feature; a location,

size, shape, convexity, aspect ratio, symmetry, borderness, surroundedness and/or occlusion as a geometric feature; and a flesh, face, sky, grass and/or other green vegetation as the semantic saliency feature.

5. The method as in claim 1, wherein step (d) includes using a collection of human opinions to train the reasoning engine to recognize the relative importance of the saliency features.

6. The method as in claim 1, wherein step (c) includes using either individually or in combination a self-saliency feature and a relative saliency feature as the structural saliency feature.

7. The method as in claim 6, wherein step (c) includes using an extended neighborhood window to compute a plurality of the relative saliency features, wherein the extended neighborhood window is determined by the steps of:

- (c1) finding a minimum bounding rectangle of a region;
- (c2) stretching the minimum bounding rectangle in all four directions proportionally; and
- (c3) defining all regions intersecting the stretched minimum bounding rectangle as neighbors of the region.

8. The method as in claim 4, wherein step (c) includes using a centrality as the location feature, wherein the centrality feature is computed by the steps of:

- (c1) determining a probability density function of main subject locations using a collection of training data;
- (c2) computing an integral of the probability density function over an area of a region; and,
- (c3) obtaining a value of the centrality feature by normalizing the integral by the area of the region.

9. The method as in claim 4, wherein step (c) includes using a hyperconvexity as the convexity feature, wherein the hyperconvexity feature is computed as a ratio of a perimeter-based convexity measure and an area-based convexity measure.

10. The method as in claim 4, wherein step (c) includes computing a maximum fraction of a region perimeter shared with a neighboring region as the surroundedness feature.

11. The method as in claim 4, wherein step (c) includes using an orientation-unaware borderness feature as the borderness feature, wherein the orientation-unaware borderness feature is categorized by the number and configuration of image borders a region is in contact with, and all image borders are treated equally.

12. The method as in claim 4, wherein step (c) includes using an orientation-aware borderness feature as the borderness feature, wherein the orientation-aware borderness feature is categorized by the number and configuration of image borders a region is in contact with, and each image border is treated differently.

13. The method as in claim 4, wherein step (c) includes using the borderness feature that is determined by what fraction of an image border is in contact with a region.

14. The method as in claim 4, wherein step (c) includes using the borderness feature that is determined by what fraction of a region border is in contact with an image border.

15. The method as in claim 1, wherein step (d) includes using a Bayes net as the reasoning engine.

16. The method as in claim 1, wherein step (d) includes using a conditional probability matrix that is determined by using fractional frequency counting according to a collection of training data.

17. The method as in claim 1, wherein step (d) includes using a belief sensor function to convert a measurement of a feature into evidence, which is an input to a Bayes net.

18. The method as in claim 1, wherein step (d) includes outputting a belief map, which indicates a location of and a belief in the main subject.

19. A method for detecting a main subject in an image, the method comprising the steps of:

- a) receiving a digital image;
- b) extracting regions of arbitrary shape and size defined by actual objects from the digital image;
- c) grouping the regions into larger segments corresponding to physically coherent objects;
- d) extracting for each of the regions at least one structural saliency feature and at least one semantic saliency feature; and,
- e) integrating the structural saliency feature and the semantic feature using a probabilistic reasoning engine into an estimate of a belief that each region is the main subject.

20. The method as in claim 19, wherein step (b) includes using a color distance metric defined in a color space, a spatial homogeneity constraint, and a mechanism for permitting spatial adaptivity.

21. The method as in claim 19, wherein step (c) includes using either individually or in combination non-purposive grouping and purposive grouping.

22. The method as in claim 19, wherein step (d) includes using either individually or in combination at least one low-level vision feature and at least one geometric feature as the structural saliency feature.

23. The method as in claim 19, wherein step (d) includes using either individually or in combination a color, brightness and/or texture as a low-level vision feature; a location, size, shape, convexity, aspect ratio, symmetry, borderness, surroundedness and/or occlusion as a geometric feature; and a flesh, face, sky, grass and/or other green vegetation as the semantic saliency feature.

24. The method as in claim 19, wherein step (e) includes using a collection of human opinions to train the reasoning engine to recognize the relative importance of the saliency features.

25. The method as in claim 19, wherein step (d) includes using either individually or in combination a self-saliency feature and a relative saliency feature as the structural saliency feature.

26. The method as in claim 25, wherein step (d) includes using an extended neighborhood window to compute a plurality of the relative saliency features, wherein the extended neighborhood window is determined by the steps of:

- (c1) finding a minimum bounding rectangle of a region;
- (c2) stretching the minimum bounding rectangle in all four directions proportionally; and,

(c3) defining all regions intersecting the stretched minimum bounding rectangle as neighbors of the region.

27. The method as in claim 23, wherein step (d) includes using a centrality as the location feature, wherein the centrality feature is computed by the steps of:

- (c1) determining a probability density function of main subject locations using a collection of training data;
- (c2) computing an integral of the probability density function over an area of a region; and,
- (c3) obtaining a value of the centrality feature by normalizing the integral by the area of the region.

28. The method as in claim 23, wherein step (d) includes using a hyperconvexity as the convexity feature, wherein the hyperconvexity feature is computed as a ratio of a perimeter-based convexity measure and an area-based convexity measure.

29. The method as in claim 23, wherein step (d) includes computing a maximum fraction of a region perimeter shared with a neighboring region as the surroundedness feature.

30. The method as in claim 23, wherein step (d) includes using an orientation-unaware borderness feature as the borderness feature, wherein the orientation-unaware borderness feature is categorized by the number and configuration of image borders a region is in contact with, and all image borders are treated equally.

31. The method as in claim 23, wherein step (d) includes using an orientation-aware borderness feature as the borderness feature, wherein the orientation-aware borderness feature is categorized by the number and configuration of image borders a region is in contact with, and each image border is treated differently.

32. The method as in claim 23, wherein step (d) includes using the borderness feature that is determined by what fraction of an image border is in contact with a region.

33. The method as in claim 23, wherein step (d) includes using the borderness feature that is determined by what fraction of a region border is in contact with an image border.

34. The method as in claim 19, wherein step (e) includes using a Bayes net as the reasoning engine.

35. The method as in claim 19, wherein step (c) includes using a conditional probability matrix that is determined by using fractional frequency counting according to a collection of training data.

36. The method as in claim 19, wherein step (e) includes using a belief sensor function to convert a measurement of a feature into evidence, which is an input to a Bayes net.

37. The method as in claim 19, wherein step (c) includes outputting a belief map, which indicates a location of and a belief in the main subject.

\* \* \* \* \*



US005640468A

**United States Patent** [19]  
**Hsu**

[11] **Patent Number:** 5,640,468  
[45] **Date of Patent:** Jun. 17, 1997

[54] **METHOD FOR IDENTIFYING OBJECTS AND FEATURES IN AN IMAGE**

[76] **Inventor:** Shin-yi Hsu, 2312 Hemlock La., Vestal, N.Y. 13850

[21] **Appl. No.:** 234,767

[22] **Filed:** Apr. 28, 1994

[51] **Int. Cl.<sup>6</sup>** ..... G06K 9/46

[52] **U.S. Cl.** ..... 382/190

[58] **Field of Search** ..... 382/103, 181, 382/190, 199, 240, 263, 284, 302

[56] **References Cited**

**U.S. PATENT DOCUMENTS**

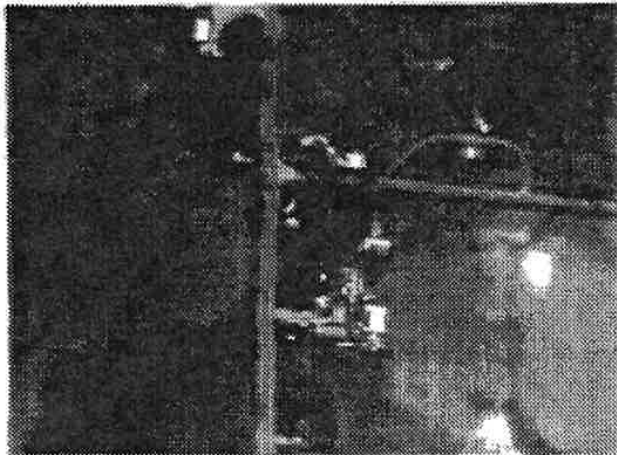
4,384,336	5/1983	Frankle et al. ....	382/302
4,754,488	6/1988	Lyke .....	382/199
4,809,347	2/1989	Nash et al. ....	382/240
4,835,532	5/1989	Fant .....	382/284
5,187,754	2/1993	Curran et al. ....	382/263
5,325,449	6/1994	Burt et al. ....	382/240

*Primary Examiner*—Jose L. Couso  
*Attorney, Agent, or Firm*—Salzman & Levy

**16 Claims, 10 Drawing Sheets**  
**(4 of 10 Drawing(s) in Color)**

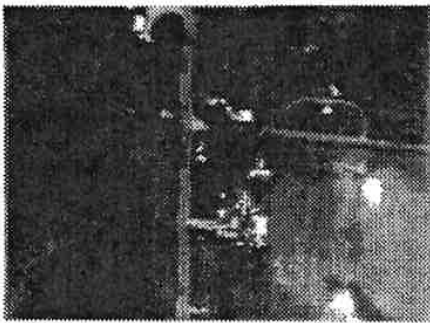
[57] **ABSTRACT**

The present invention features the use of the fundamental concept of color perception and multi-level resolution to perform scene segmentation and object/feature extraction in the context of self-determining and self-calibration modes. The technique uses only a single image, instead of multiple images as the input to generate segmented images. Moreover, a flexible and arbitrary scheme is incorporated, rather than a fixed scheme of segmentation analysis. The process allows users to perform digital analysis using any appropriate means for object extraction after an image is segmented. First, an image is retrieved. The image is then transformed into at least two distinct bands. Each transformed image is then projected into a color domain or a multi-level resolution setting. A segmented image is then created from all of the transformed images. The segmented image is analyzed to identify objects. Object identification is achieved by matching a segmented region against an image library. A featureless library contains full shape, partial shape and real-world images in a dual library system. The depth contours and height-above-ground structural components constitute a dual library. Also provided is a mathematical model called a Parzen window-based statistical/neural network classifier, which forms an integral part of this featureless dual library object identification system. All images are considered three-dimensional. Laser radar based 3-D images represent a special case.



M704J3 Original Scene  
Mono input 1





M704J3 Original Scene  
Mono Input 1

*Figure 1*

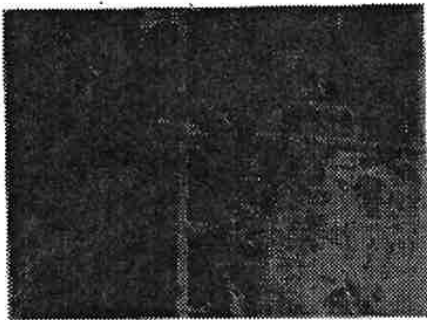


Mono  
M704J3 → F 3  
M704J3 → G 3  
M704J3 → B 2

A diagram consisting of three arrows pointing from the text above to a rectangular box containing the letters 'fc'. The arrows are arranged vertically and converge towards the box.

*Figure 2*

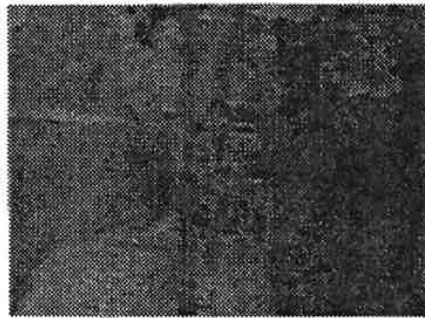




M704J3 → R 2  
M705J3 → G 3  
M706J3 → B 3

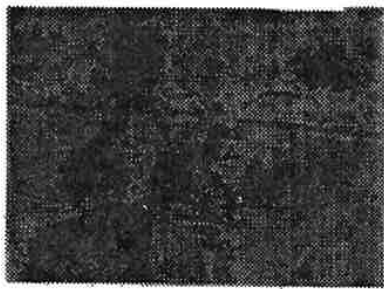
→ M7J3FC

*Figure 3*



M104J3 Compressed 25 (3) R  
50 (3) G  
75 (2) B

*Figure 4*



M7J3FC Treat as Mono

Compression -5 R (3)  
-10 G (3)  
-25 B (3) → Output 2

Figure 5

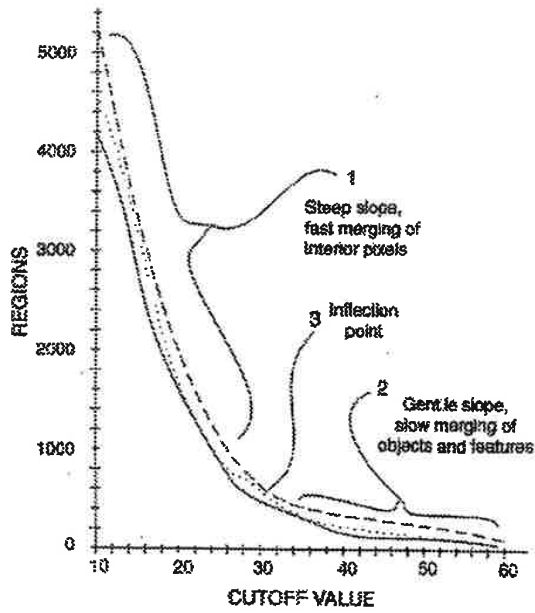


Figure 9

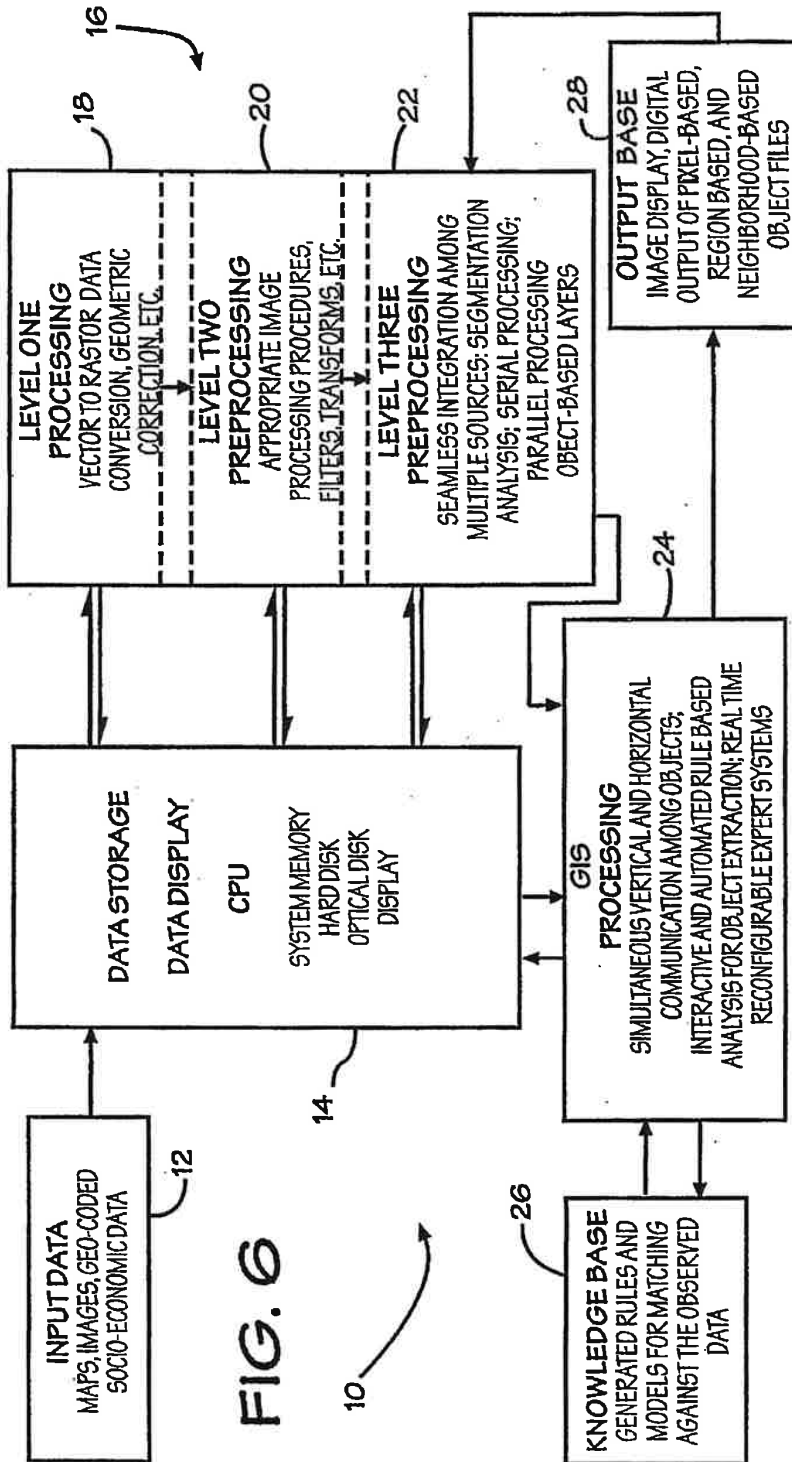


FIG. 6

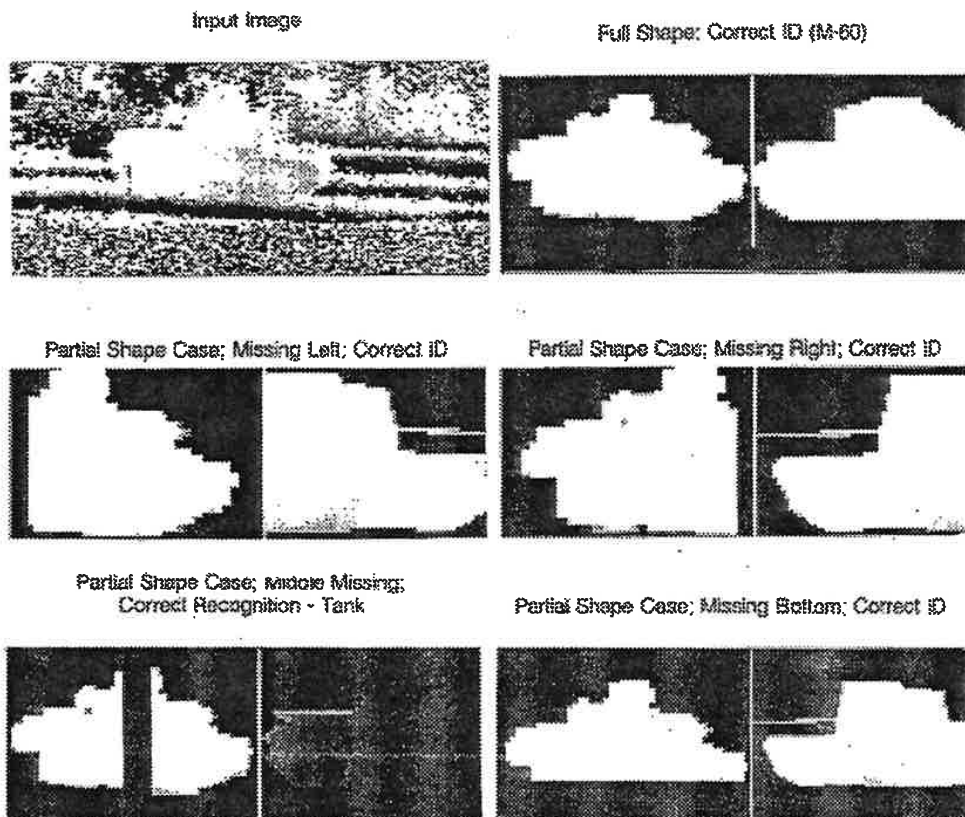


Figure 7

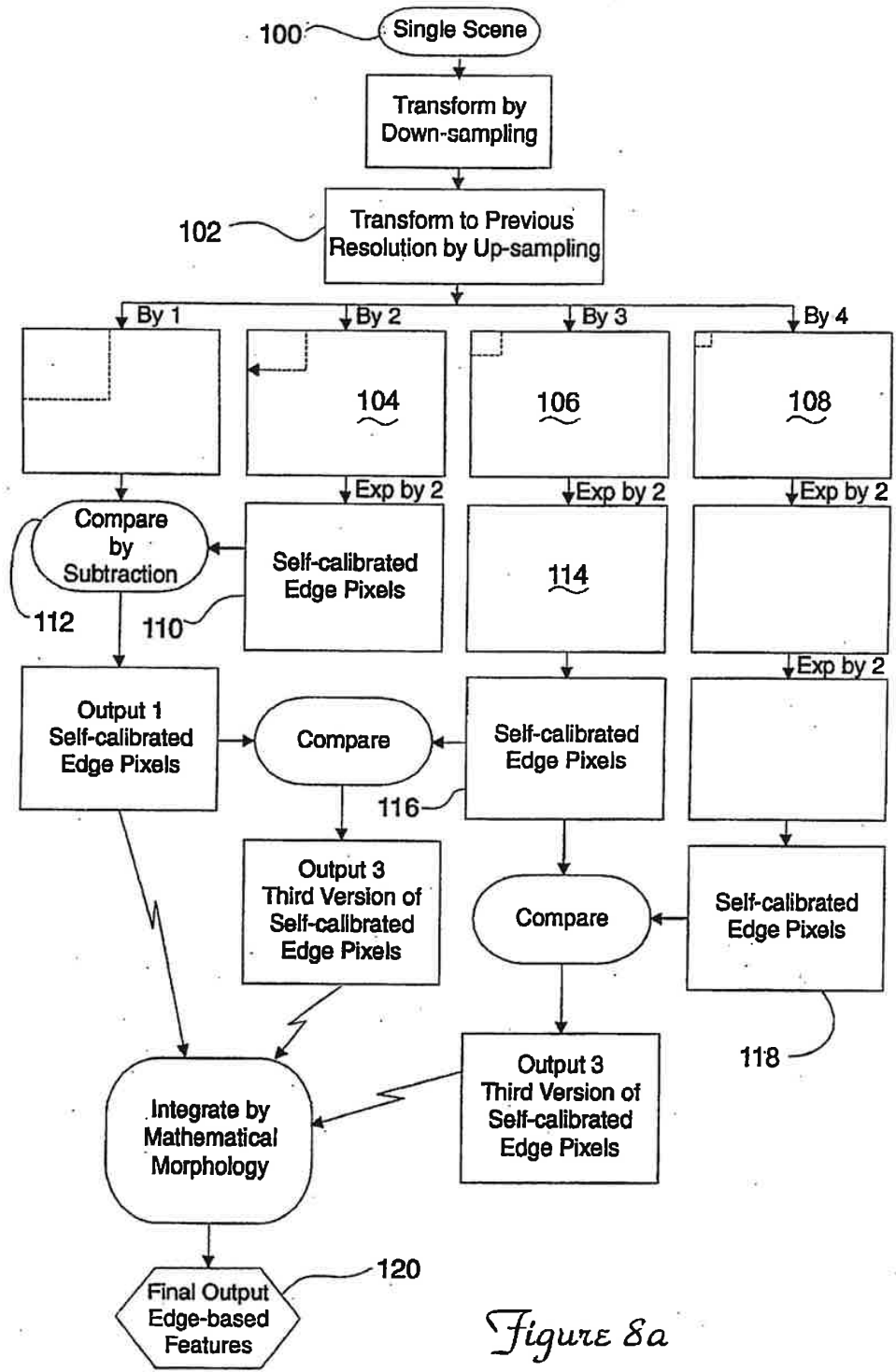


Figure 8a



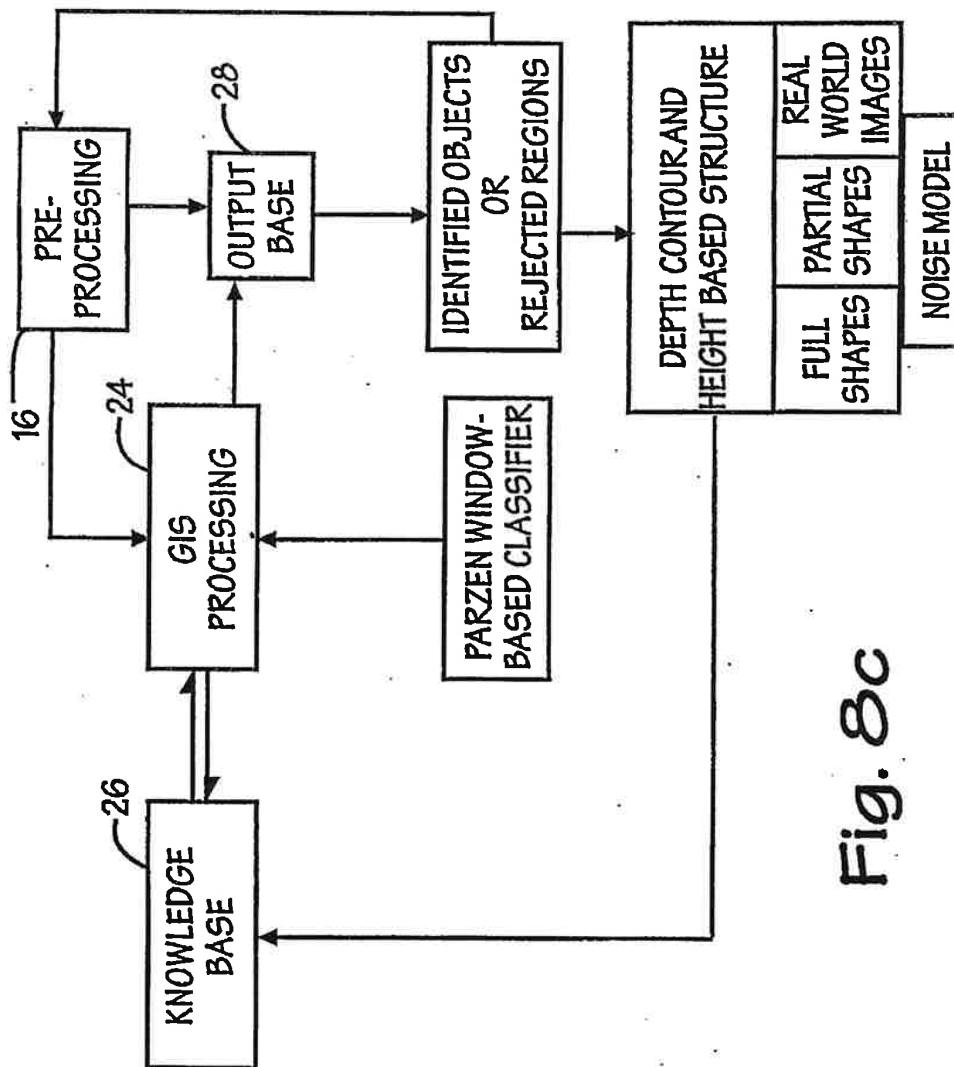
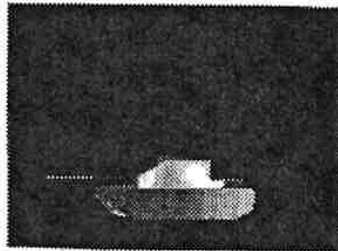


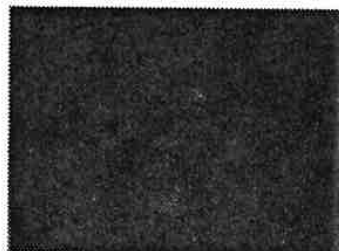
Fig. 8c



*Figure 8d*



*Figure 11a*



*Figure 11b*



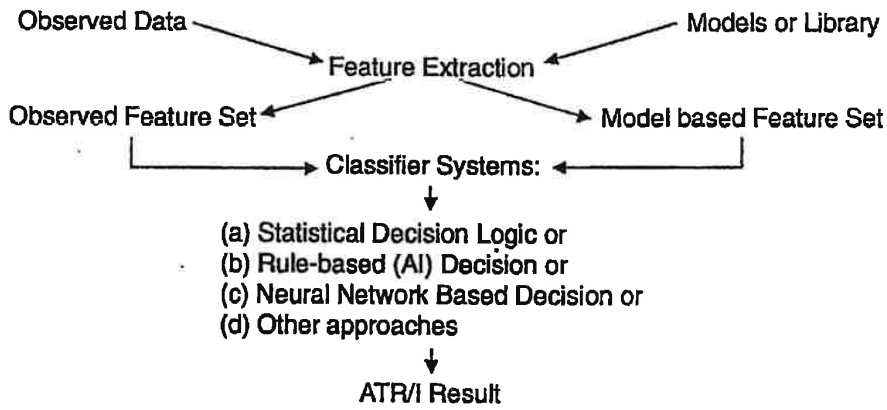


Figure 10a

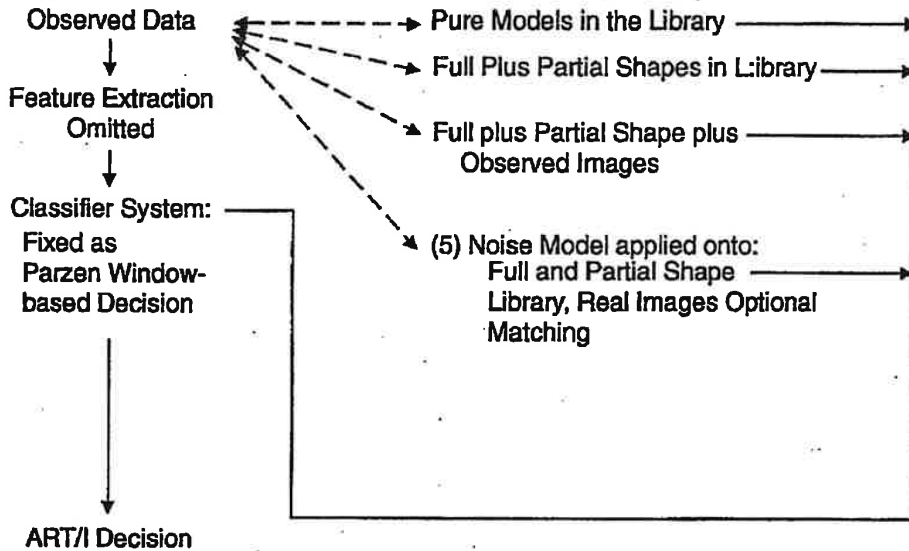


Figure 10b

## METHOD FOR IDENTIFYING OBJECTS AND FEATURES IN AN IMAGE

### FIELD OF INVENTION

The present invention pertains to object and feature identification in an image and, more particularly, to scene segmentation and object/feature extraction by generating uniform regions from a single band image or a plurality thereof using a self-determining, self-calibrating, improved stable structure, pseudo multispectral color, and multilevel resolution processing technique, and associated matching methods for object identification.

### BACKGROUND OF THE INVENTION

An image is basically a data matrix of  $m$  rows and  $n$  columns. An element of that image matrix is called a picture element, or a pixel. An image becomes meaningful when a user is able to partition the image into a number of recognizable regions that correspond to known natural features, such as rivers and forests, or to man-made objects. Once this higher-level of image generalization is completed, each distinct feature/object, being a uniform field, can be identified. The process by which such a uniform field is generated is generally referred to as segmentation. The process by which a segmented region is matched with a rule set or a model is referred to as identification.

Dozens of techniques have been used by researchers to perform image segmentation. They can be grouped into three major categories: (1) class-interval based segmentors, (2) edge-based segmentors, and (3) region-based segmentors.

A given image has 0 (zero) as the minimum pixel value and 255 as the maximum pixel value. By mapping all pixels whose intensity values are, say, between 0 and 20 into one category, a simple thresholding method can be used to perform image segmentation.

An edge is generally defined as the difference between adjacent pixels. Edge-based image segmentation is performed by generating an edge map and linking the edge pixels to form a closed contour. A review of this class of segmentors can be obtained from Farag. (*Remote Sensing Reviews*, Vol. 6, No. 1-4, 1992, pp. 95-121.)

Region-based segmentation reverses the process of edge-based segmentation, because it starts with the interior of a potential uniform field rather than with its outer boundary. The process generally begins with two adjacent pixels and one or more rules used to decide whether merging of these two candidates should occur. One of the examples of this class of segmentors can be found in Tenorio using a Markov random field approach. (*Remote Sensing Reviews*, Vol. 6, No. 1-4, 1992, pp. 141-153.)

All conventional segmentors share the following fundamental features:

- 1) the segmentation process is generally performed on a single band image;
- 2) the segmentation process follows well-defined mathematical decision rules;
- 3) except for simple thresholding, all segmentors are computationally expensive and/or intensive; and
- 4) none of the conventional techniques is self-determining or self-calibrating.

If segmentation is defined as the process of generating distinct uniform fields from a scene, a human visual system that is based on color perception should also be considered a segmenter. In contrast to mathematics-based segmentation

schemes, color-based segmentation relies on the use of three spectrally-derived images. These multiple images are, in most cases, generated from a physical device called a multispectral sensor. The advantage of this method over mathematical segmentors is its ability to perform scene segmentation with minimal or no mathematical computation.

For purposes of clarity throughout this discussion, it should be understood that the concept of three spectrally-derived (color) images, while representing the preferred embodiment, is merely a subset of a more general concept: any composite having component ranges which may be transformed into two or more respective component parts and then projected into a common space.

Color-based segmentors require input of three spectrally distinct bands or colors. A true color picture can be generated from a scene taken by three registered bands in the spectral regions of blue, green and red, respectively. Then, they are combined into a composite image using three color filters: red, green and blue. The resultant color scene is indeed a segmented scene because each color can represent a uniform field.

The above discussion is related to region-based segmentation. In edge-based segmentation, all of the conventional techniques use well-defined mathematical formulae to define an edge. After edges are extracted, another set of mathematical rules is used to join edges and/or eliminate edges in order to generate a closed contour to define a uniform region. In other words, none of the conventional techniques uses the scene itself to define an edge even though, in a more global point of view, an edge is, in fact, defined by the scene itself.

If a region or an edge can be generated from the content of the scene itself, it should be possible to integrate both region-based and edge-based segmentation methods into a single, integrated process rather than using two opposing philosophies.

Object identification is a subsequent action after segmentation to label an object using commonly-accepted object names, such as a river, a forest or an M-60 tank. While object recognition can be achieved from a variety of approaches (such as statistical document functions and rule-based and model-based matching), all of these conventional methods require extracting representative features as an intermediate step toward the final object identification. The extracted features can be spectral reflectance-based, texture-based and shape-based. Statistical pattern recognition is a subset of standard multivariable statistical methods and thus does not require further discussion. A rule-based recognition scheme is a subset of conventional, artificial intelligence (AI) methods that enjoyed popularity during the late 1980s. Shape analysis is a subset of model-based approaches that requires extraction of object features from the boundary contour or a set of depth contours. Sophisticated features include Fourier descriptors and moments. The effectiveness of depth information was compared to boundary-only based information, Wang, Gorman and Kuhl (*Remote Sensing Reviews*, Vol. 6, No. 1-4, pp. 129+). In addition, the classifier performance between range moments and Fourier descriptors was contrasted.

An object is identified when a match is found between an observed object and a calibration sample. A set of calibration samples constitutes a (calibration) library. A conventional object library has two distinct characteristics: 1) it is feature based and 2) it is full-shape based. The present invention reflects a drastically different approach to object identification because it does not require feature extraction as an

intermediate step toward recognition and it can handle partially-occluded objects.

Feature extraction uses fewer but effective (representative) attributes to characterize an object. While it has the advantage of economics in computing, it runs the risk of selecting wrong features and using incomplete information sets in the recognition process. A full-shape model assumes that the object is not contaminated by noise and/or obscured by ground clutter. This assumption, unfortunately, rarely corresponds to real-world sensing conditions.

Depth contours are used for matching three-dimensional (3-D) objects generated from a laser radar with 3-D models generated from wireframe models. In real-world conditions, any image is a 3-D image because the intensity values of the image constitute the third dimension of a generalized image. The difference between a laser radar based image and a general spectral-based image is that the former has a well-defined third dimension and the latter does not.

It has been proven that the majority of objective discrimination comes from the boundary contour, not the depth contour (Wang, Gorman and Kuhl, *Remote Sensing Review*, Vol. 6, Nos. 1-4, pp. 129-?, 1992(?)). Therefore, the present invention uses a generalized 3-D representation scheme to accommodate the general image. This is accomplished by using the height above the ground (called height library) as an additional library to the existing depth library. The resultant library is called a dual depth and height library.

It would be advantageous to provide a much simpler, more effective and more efficient process for image segmentation, one that achieves an integration between region-based and edge-based segmentation methodologies which, heretofore, have been treated as mutually exclusive processes.

It would also be advantageous to generate uniform regions of an image so that objects and features could be extracted therefrom.

It would also be advantageous to provide a method for segmenting an image with minimal mathematical computation and without requiring two or more spectrally-derived images.

It would also be advantageous to provide a flexible and arbitrary scheme to generate colors.

It would also be advantageous to use the human phenomenon of color perception to perform scene segmentation on only one spectral band.

It would be advantageous to provide an object identification scheme that does not rely on a predetermined number of features and fixed characteristics of features.

It would also be advantageous to provide an object identification scheme to facilitate object matching either in a full-shape or partial-shape condition.

It would also be advantageous to provide an object identification system that is both featureless and full and partial shape based.

It would also be advantageous to provide a mathematical model that can handle both featureless and full/partial shape cases.

It would also be advantageous to provide a library construction scheme that is adaptable to both featureless and full/partial shape based object recognition scenarios.

It would also be advantageous to provide a dual library (depth and height) to perform general 3-D object recognition using any type of image.

It would also be advantageous to provide a full object identification system that is capable of integrating the previously described novel segmentation and novel object recognition subsystems.

#### SUMMARY OF THE INVENTION

In accordance with the present invention, there is provided a Geographical Information System (GIS) processor to perform scene segmentation and object/feature extraction. GIS has been called a collection of computer hardware, software, geographical data and personnel designed to efficiently manipulate, analyze, and display all forms of geographically referenced information. The invention features the use of the fundamental concept of color perception and multi-level resolution in self-determining and self-calibration modes. The technique uses only a single image, instead of multiple images as the input to generate segmented images. Moreover, a flexible and arbitrary scheme is incorporated, rather than a fixed scheme of segmentation analysis. The process allows users to perform digital analysis using any appropriate means for object extraction after an image is segmented. First, an image is retrieved. The image is then transformed into at least two distinct bands. Each transformed image is then projected into a color domain or a multi-level resolution setting. A segmented image is then created from all of the transformed images. The segmented image is analyzed to identify objects. Object identification is achieved by matching a segmented region against an image library. A featureless library contains full shape, partial shape and real-world images in a dual library system. The depth contours and height-above-ground structural components constitute a dual library. Also provided is a mathematical model called a Parzen window-based statistical/neural network classifier, which forms an integral part of this featureless dual library object identification system. All images are considered three-dimensional. Laser radar based 3-D images represent a special case.

Analogous to transforming a single image into multiple bands for segmentation would be to generate multiple resolutions from one image and then to combine such resolutions together to achieve the extraction of uniform regions. Object extraction is achieved by comparing the original image and a reconstructed image based on the reduced-resolution image. The reconstruction is achieved by doubling the pixel element in both x and y directions. Edge extraction is accomplished by performing a simple comparison between the original image and the reconstructed image. This segmentation scheme becomes more complex when two or more sets of pair-wise comparisons are made and combined together to derive the final segmentation map. This integration scheme is based on mathematical morphology in the context of conditional probability.

To accommodate featureless and full/partial shape based object identification, the present invention proposes the use of a mixture of full-shape and partial-shape models plus real-world images as a calibration library for matching against the segmented real-world images. Moreover, in accordance with the invention, the library is constructed in the image domain so that features need not be extracted and real-world images can be added freely to the library. The invention further provides a mathematical model for the classifier using the Parzen window approach.

#### BRIEF DESCRIPTION OF THE DRAWINGS

The file of this patent contains at least one drawing executed in color. Copies of this patent with color drawings will be provided by the patent and Trademark Office upon request and payment of the necessary fee.

A complete understanding of the present invention may be obtained by reference to the accompanying drawings, when considered in conjunction with the subsequent detailed description, in which:

5

- FIG. 1 is a monochrome original image;
- FIG. 2 is a bit-reduced and reconstituted image (of FIG. 1) without compression, based on three identical bands, in accordance with the present invention;
- FIG. 3 is a multispectral original image;
- FIG. 4 is a compressed-transformed version of FIG. 3 with compression in accordance with the present invention;
- FIG. 5 is a compressed-transformed, three-band, monochrome, reconstituted version of FIG. 3;
- FIG. 6 is a block diagram depicting the multi-level preprocessing functions incorporated in the invention;
- FIG. 7 is a set of updated images in three dimensions (partials);
- FIGS. 8a and 8b, taken together, are a flow chart of self-determining, self-calibrating, edge-pixel generation and object extraction operations, in accordance with the present invention;
- FIG. 8c is a block diagram of the GIS processing system of the present invention showing system architecture and library details;
- FIG. 8d is an image generated by the process depicted in the processing loop of FIG. 6;
- FIG. 9 is a typical scene characteristics (SC) curve;
- FIGS. 10a and 10b are flow charts of a prior art approach and the approach of the present invention, respectively, to image and library matching techniques in accordance with the system depicted in FIG. 8c; and
- FIGS. 11a and 11b depict range and height libraries, respectively.

#### DESCRIPTION OF THE PREFERRED EMBODIMENT

In conventional multispectral images, an object emits or radiates electromagnetic radiation when its temperature is above 0° K. The radiation can be divided into numerous subsets according to any specified wavelength intervals. A conventional color photograph is a composite of three broad wavelength intervals: red from 0.6 to 0.7 micron; green from 0.5 to 0.6 micron; and blue from 0.4 to 0.5 micron. Using any three wavelength regions other than the aforementioned red/green/blue combination yields a set of colors that differs significantly from the set produced by blue, green and red spectral regions. All such deviations are called false colors. It follows that any three bands can generate false colors. The human-perceived true color set is a special case.

Reducing the interval between adjacent wavelength regions results in two images being very similar. Since any wavelength regions can be used selectively to generate multispectral images, generation of false-color images can be a random process. The present invention reflects the discovery that false color images can be generated from a single band image.

Referring now to the FIGS., and specifically to FIG. 6, there is shown a functional block diagram of the preprocessing technique that is the subject of copending patent application Ser. No. 08/066,691, filed May 21, 1993.

The first component of the system is means for accepting various information sources as input to a second-generation GIS system, shown at reference numeral 12. The system accepts multiple data sources 12 for one common geographical area. The sources can be existing maps, geo-coded, socio-economic data such as census tracts, and various images such as LANDSAT and SPOT satellite imagery. The most common information sources are images and maps.

6

This component 12 allows all data to conform to a common format: a layer of information is equivalent to a data matrix.

The input data 12 is applied to a data storage device, such as a system memory, a hard disk, or an optical disk, and/or to one or more conventional display devices, both storage and display devices being represented by reference numeral 14. Disks and memory 14 are used for efficient storage and retrieval.

All of the appropriate image processing and remote sensing analysis techniques can be used as preprocessors, shown generally as reference numeral 16, but consisting of a first, second and third preprocessor 18, 20 and 22 to the main GIS system processor 24, which performs the above-discussed GIS-based image analysis. Preprocessing transforms the incoming observed data into a format in which objects are readily extractable. If images are properly aligned, however, preprocessing levels 1 and 2 need not be performed at all.

If the images are "raw", of course, preprocessing is required. The level 1 preprocessor 18 is used to convert vector data to image (raster) data, to correct geometric and spectral errors, to perform resolution matching, to zoom, rotate and scale (so as to align the separate images with one another), and to filter and transform images, if necessary.

The level 2 preprocessor 20 is used for edge detection, special purpose feature separation, linear combination and multi-resolution functions. Image data must be preprocessed to the point that objects are readily extractable by the main processor. While the majority of level 2 preprocessing is to be performed using the segmenter of the main system, external system processors, not shown, can be used to perform similar functions.

A multi-level resolution analysis method is used to define edges and then extract edge-based objects. The level 2 preprocessor 20 provides the main processor 24 with a binary image. Background of zero intensity value is used to represent non-edge based object, and 255 to represent objects of strong edgeness.

The third level preprocessor 22 can be conceived as a "clearing house" for all incoming data. Regions are processed in such a way as to generate a scene structure. Once all of the data sets are processed, each individual region in any layer can communicate with any other region in any layer. While many methods are available to provide this function, the inventive system uses an object-based segmentation scheme to generate regions for each individual layer. Each region is given a set of feature attributes which includes spectral intensity, size/shape/texture information of the region, and locational information of the centroid and the individual pixels in the region.

The object extraction system discussed above is a seamless integration between raster image processing and vector GIS processing, heretofore not achieved by other researchers.

One or more rules are used to interrogate the attribute database. The main system 110 accepts any appropriate means for object identification, as long as it uses the regions generated by level 3 preprocessor 108 as the basis of information analysis.

The invention provides an environment for parallel processing in the level 3 preprocessor 22 and the main processor 24. In the preferred embodiment, the program is written for a parallel processor manufactured by the Transputer Company, but it should be understood that any appropriate parallel hardware/software system can be used. Parallel processing is not a required feature of a GIS system, but it

has been found to be a desirable feature for any information extraction system.

Connected to GIS processor 24 are a set of self-determining and self-calibrating segmentation schemes and an image library comprising a mixture of model-based images and real-world images, known as a knowledge base or library 26 and an output base 28, both described in greater detail hereinbelow.

In operation, a single, monochrome, original image (FIG. 1) is input to the GIS processor 24 (FIG. 6) as data 12 or as a level 2 preprocessed image 20, and then transformed into two or more pseudo bands using various compression methods. Such compression methods include, but are not limited to, taking a square root or applying compression factors from a general logarithm transformation. A false color image, therefore, can be generated, for example, from the following three transforms of a single image: (1) square root; (2) log; and (3) double log.

The user is therefore able to select a simple number to generate a new image. Accordingly, generation of a false color image based on three bands becomes a simple procedure based on selection of three numbers (e.g., 25, 50 and 100).

Table I contains an example of compression factors. These factors were generated by a generalized log scale of 75. Each pixel value between 0 and 255 is mapped and transformed by its appropriate compression factor (value), so that an original image having up to 256 pixel values can be represented as a transformed image of no more than 18 pixel values.

TABLE I

Single Image Transformations  
Transformation (increases left to right, then down)

0	0	0	0	4	4	36	37	37	41	41	41	41	41	41	73
73	73	77	77	77	77	77	77	77	109	109	110	110	110	110	110
114	114	114	114	114	114	114	114	114	114	114	146	146	146	146	146
146	146	146	146	146	146	146	146	150	150	150	150	150	150	150	150
150	150	150	150	150	182	182	182	182	182	182	182	182	182	182	182
182	182	182	182	182	183	183	183	183	183	183	183	183	183	183	187
187	187	187	187	187	187	187	187	187	187	187	187	187	187	187	219
219	219	219	219	219	219	219	219	219	219	219	219	219	219	219	219
219	219	219	219	219	219	219	219	219	219	219	219	219	219	219	219
219	219	219	219	219	219	219	219	219	219	219	219	219	219	219	219
223	223	223	223	223	223	223	223	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255

A false color composite image can be generated using a conventional bit mapping scheme (FIGS. 8a and 8b), such as the one shown below.

1. Code any number of bits from a square root transformed band 50 (FIG. 8b) as one of the primary colors (red, green or blue). Example: color three bits in the red band.
2. Code any number of bits from a log-transformed band 50 as one of the two remaining primary colors. Example: color three bits in the green band.
3. Code any number of bits from a double-log transformed band as the last remaining primary color. Example: color two bits in the blue band.
4. Combine the above three bands 52 into a composite image 54 and display it 56 on a color monitor, not shown. (This step is identical to the conventional color image generation procedure.)
5. Digitally capture the displayed color composite image 58.
6. Store the digital image 58 in memory or a data storage device (e.g., a hard disk), for subsequent segmentation and object/feature extraction analysis.

The resultant image is shown in FIG. 2.

A distinct feature of the present invention is a flexible color generation scheme. The above-described transformation scheme is essentially a data compression procedure. False color generation is an arbitrary method. Thus, a generalized data compression scheme can be devised in accordance with the concept of the logarithm. For instance, a compression factor of 10 can be selected as the equivalent of taking a square-root transform. Similarly, a compression factor of 100 can be selected as the equivalent of taking a single log transform.

The color image generation procedure differs significantly from conventional procedures because the latter relies on the availability of three images, each of which corresponds precisely to a given wavelength region. In other words, conventional color image generation techniques must have multiple, original images, whereas the present invention requires only one image.

Another distinct feature of the present invention is real time segmentation from a single band source. As previously noted, conventional segmentation procedures are extremely time consuming because sophisticated segmentors require intensive computations in finding edges and/or merging regions. As a significant departure, however, the present invention requires no computations beyond compression to perform image segmentation. The inventive method uses a color to represent a region. That color can be generated from the merger of three bands in the color domain using a lookup table (Table I) procedure, rather than numerical computations or logic statement decisions.

Segmentation is a method for partitioning an image into a number of uniform regions. It follows that a color representation of the original image is a segmented image. Two simple and appropriate segmentation schemes can be used to perform segmentation based on the present invention.

The first method is simple thresholding. Zero can be specified as a class by itself, or an interval, say, from one to five can represent another class. This simple and yet extremely effective segmentation scheme is made possible because the image is already segmented in the color domain. Each color is represented by a digital number or by a class interval.

Two initial segmentation algorithms, LOCAL and GLOBAL, having much in common, are discussed herein simultaneously.

Segmentation starts at the upper left pixel (which defines the first region) and proceeds from left to right across each row. When a row is complete, the next row down is processed.

When the region affiliation of a pixel is to be determined, the pixel above it and to the left are considered. One of four possibilities will result:

- 1) The pixel will define a new region.
- 2) It will merge into the region of the pixel above.
- 3) It will merge into the region of the pixel to its left.
- 4) The region of the pixel above and the region to its left will be merged into one grand region, with this pixel being included.

Diagonal tone values are never considered and diagonal merging is not a possibility.

The following algorithm description refers to pixels above and to the left of the current pixel. Obviously, the top row will have none above and the left column will have none to the left. The algorithm interprets such cases as "exceeding the threshold." In other words, pixels outside the frame are assumed to have infinitely different tone, with the result that a boundary is always generated around the image. In the case of GLOBAL segmentation, the following description refers to "changing the tone" of pixels. It should be understood that this change is temporary and in effect only for the duration of initial segmentation. All pixels revert to their original, true values after initial segmentation is complete. Initial segmentation is performed as follows:

- 1) Initialize the first region to be the upper left pixel. Go on to the next pixel to the right, which will be called the "current pixel".
- 2) Examine the tone difference between the current pixel and the (possibly revised) tones of the pixel above it and the pixel to the left of it. Compare these differences to a predetermined threshold.
- 3) If both tone differences exceed the threshold, the current pixel generates a new region.
- 4) If exactly one tone difference is less than or equal to the threshold, absorb the current pixel into the region to which that neighbor belongs. If GLOBAL segmentation is being performed, change the tone of the current pixel to the tone of that region.
- 5) If both tone differences are less than or equal to the threshold, absorb both the current pixel and whichever of the two regions was most recently created into the older region. If GLOBAL segmentation is being performed, change the tone of the current pixel and of all pixels in the most recent region to the tone of the older region.

In step no. 5 (above) of the initial segmentation, merging the region of the pixel above with that of the pixel to the left is not a trivial undertaking. Passing through the entire frame, reclassifying perhaps thousands of pixels, would be terribly time consuming. Therefore, a group map vector is maintained. Each member of this array is the "true" region membership of the region corresponding to that position in the array. Thus, suppose that the fifth element of this vector is equal to three. That means that pixels formerly considered to belong to region five are now considered to belong to region three. In this way, updating regions to reflect merger involves updating only this relatively small group map vector, rather than the entire image as segmented so far.

The initial segmentation steps are next illustrated with code fragments extracted directly from the source module SEG\_GLOB.C, the GLOBAL segmenter. The LOCAL seg-

menter is essentially identical except that tones are never changed to group leaders.

The image is stored as a vector. The first row is first, followed by subsequent rows. Therefore, the pixel to the left of the current pixel has a subscript one less than that of the current pixel. The pixel above has a subscript ncols less. For example, the tone of the current pixel is compared to those of the pixel to the left and the pixel above, as follows:

```

10
if (abs ((int) *pix_ptr - (int) *(pix_ptr - 1)) >      /*Left*/
cut)
if (abs ((int) *pix_ptr -
(int) *(pix_ptr - ncols)) > cut)                    /*Above*/

```

If both of the differences exceed the threshold, a new group (region) is processed. First, increment the region counter and verify that the memory allocated for storing region information has not been exceeded. Second, set the group pointer for this pixel equal to the number of the new group (region). Finally, set the group mapping pointer to be this group. In other words, it has its own identity and has not yet been absorbed into another region.

```

25
if (++ngps > maxgrps)
goto EXIT;
*grp_ptr++ = ngps;          /*Group ID for this pixel*/
grpmap[ngps] = ngps;      /*Group map vector*/
30

```

If the tone difference with the pixel above does not exceed the threshold, but the tone difference to the left does, this pixel is merged into the region above. This is done by changing the group ID of this pixel to that of the pixel above. Furthermore, if this is GLOBAL merging, the tone of this pixel must be changed to that of the pixel above.

```

40 *grp_ptr = *(grp_ptr - ncols); /*No top edge so merge up*/
++grp_ptr; /*Point to next pixel*/
*pix_ptr = *(pix_ptr - ncols); /*Propagate leader tone*/

```

If the tone difference with the pixel in the left does not exceed the threshold, but the tone difference above does exceed the threshold, merge this pixel into the region to the left. The steps are similar to the previous case.

```

50 *grp_ptr = *(grp_ptr - 1); /*No left edge so merge left*/
++grp_ptr; /*Point to next pixel*/
*pix_ptr = *(pix_ptr - 1); /*Propagate leader tone*/

```

The last case is the most difficult. When both tone differences do not exceed the threshold, the region above and the region to the left merge with each other, due to the current pixel. The most recently created region is absorbed into the older region, regardless of which is above and which to the left. This is an arbitrary decision, as either 0 choice (or even a random choice) could be possible. However, absorbing the younger into the older speeds the remapping process, described below, by limiting the number of regions that must be checked. The first step is therefore to find the older region (the smaller region number) and the younger. In GLOBAL segmentation, the new region's tone is that of the absorbing group.



```

small = grpmap[(grp_ptr-ncols)]; /*Most likely order*/
big = grpmap[(grp_ptr + 1)];
if (big < SMALL) | /*but not guaranteed*/
    temp = big;
    big = small;
    small = temp;
    leader = *(pix_ptr-1); /*Propagate leader tone*/
    |
else
    leader = *(pix_ptr-ncols);

```

This pixel is classified as belonging to the "small" region. For GLOBAL segmentation, change its tone to that of the new region. The pixels above and to the left should both be in the same region already, so nothing else need be done.

```

*grp_ptr++ = small; /*This pixel's region number*/
*pix_ptr = leader; /*Propagate leader tone*/
If (big == small) /*If above and left groups same, done*/
    continue;

```

If this is GLOBAL segmentation, the tone of the "big" group's pixels must be changed to the tone of the new region. There is no need to process all of the pixels in the image processed so far. Only the pixels in the row above, to the right of the current pixel, and those in the row to the left of the current pixel can affect future decisions.

```

c = ncols - col + 1; /*This row, to left of current pixel*/
while (++c)
    if (grpmap[(grp_ptr-c-1)] == big)
        *(pix_ptr-c) = leader;
c = col; /*And along row above which remains*/
while (--c)
    if (grpmap[(grp_ptr-ncols-1+c)] == big)
        *(pix_ptr-ncols+c) = leader;

```

The final step in dealing with the merging of two established regions is to update the group mapping vector to reflect the fact of the absorption. This loop requires is the most time in this algorithm. As each new region is defined, `grpmap[ng]=ng` is initialized, and the updating never increases the value of any element of `grpmap`. Thus it is guaranteed that `grpmap[k]<=k`.

```

for (temp=big; temp<=ngps; temp++)
    if (grpmap[temp] == big)
        grpmap[temp] = small;

```

After processing every pixel in the image as described above, the resulting group map vector is filled with holes. Absorption is not reversible. Every region number that was absorbed is an unused number, so the empty spaces must be compressed.

The algorithm for compressing the empty spaces is straightforward. Pass through the entire group map vector. When an element is found whose value is not equal to its subscript, it is a compressed-out region. That is, the region of the subscript was absorbed into the region of the value. To complete compression, increment a counter of zapped regions, place the counter in a vector called `omit_grp` in such a way that `omit_grp[ig]` is the number of regions up to that point that have been absorbed and reassign region numbers in `grpmap` by subtracting the corresponding element of `omit_grp`. The code for doing this is as follows:

```

count = 0;
for (ig=1; ig<=ngps; ig++) |
    if (grpmap[ig] != ig) /*if this group has been absorbed*/
        ++count; /*then count it*/
    omit_grp[ig] = count;
    grpmap[ig] = omit_grp[grpmap[ig]] /*Compress*/
    |

```

The final step is to remap all of the pixels in the image according to the final group map. This is easily accomplished as follows:

```

temp = nrows * ncols;
while (temp--)
    idptr[temp] = grpmap[idptr[temp]];

```

The core algorithm is extremely similar to the LOCAL initial segmentation. The image is passed through the same way, examining pixels above and to the left of the current pixels. The only difference is that tones of pixels are not compared as in the initial segmentation. Rather, region IDs are examined and the merging criterion checked. If both of a pair of pixels belong to the same region, there is obviously no border between them. If they are in different regions the criterion must be evaluated. If there is a tone difference, subtract their mean tones and compare this to the cutoff. If the criterion is size-based, determine whether their sizes fit the criterion, and so forth. Otherwise, the core algorithm of region growing is identical to LOCAL, initial segmentation. Mean tones and sizes are not updated as merging takes place during the top-to-bottom operation. All merging decisions are based on the regions as they exist at the beginning.

A slight complication occurs when four regions are lined up and their mean tones are 50, 60, 48, 58. Letting the cutoff be 10, the first two will be merged as will the last two. When the top-to-bottom pass is completed, one new region will have a mean tone of 55 and the other will have a mean of 53. These two new regions certainly meet the merging criterion. Thus, another top-to bottom pass, cleaning up the newly-created regions that meet the current criterion, is performed and repeated until no more mergers take place. This iteration is important to ensuring stability across similar images.

The second method is a simple region-growing method. Neighboring pixels can be merged together if their absolute difference is zero or within a specified number. This one-pass region-growing segmentation yields a segmentation map that corresponds to a visual segmentation of the color map.

The overall scene characteristics (SC) profile is a sloping L-shaped curve with three distinct segments: 1) a steep slope indicating the fast merging of interior pixels; 2) a gentle slope indicating a slower rate of merging among objects and features; and 3) an "inflection" point between these two segments indicating the emergence of the object sets in segmentation analyses. Since real-world objects and features now exist as distinct subsets, they cannot be easily merged by linearly increasing cutoff values. The scene structure is, therefore, very stable, and thus called a stable structure, or optimal segmentation of the scene.

From the foregoing, it is clear that if a set of algorithms is available to perform such multistage segmentations with the cutoff value increasing linearly from one iteration to the next, a scene characteristics profile can be generated as described above. The stable structure of the scene can be analytically determined by identifying and analyzing the inflection point within the curve. (In this context the term

inflection point does not have the rigid meaning of changing slope characteristics from a convex to a concave structure, but signifies a marked change in the magnitude of the slope from one section of curve to the next.) The effort is based precisely on the fact that this task-generation of stable structures of the scene can be accomplished with a set of artificial intelligence (AI)-based algorithms.

It should be understood that the scene characteristics curve may itself be transformed (e.g., log, square root) in any combination or derivative form without departing from the scope of the present invention. Similarly, the stopping points for processing can be obtained by a simple criterion (e.g., slope change > 1.0) or by combining suitable criteria of multiple curves.

#### Object Extraction Using Segmented Images

In the conventional approach, once the color image is digitally segmented, each region can be described by a set of feature attributes using size, shape and locational descriptors. An object is extracted if the given attributes match a model specified in an expert system.

An innovative approach of the present invention is to use a single-color (also known as "single band") or a single-feature-based image as the basis for generating additional objects. Once an image is represented by a set of colors, a series of images can be generated, each of which images is represented by a given color. For instance, if a color or a number corresponds to certain material that is used to build a road, then all of the road pixels (or similar pixels) can be extracted and implanted onto another image whose background is filled with zero values. When this road or similar-class map is isolated, region growing or other mathematical operations can be performed such as buffering or connecting broken components to generate new images.

An analogy to segmentation using multiple bands derived from a single band in a color domain is segmentation using multiple resolution from a single image.

Multi-level resolutions can be generated from a single image by a down sampling technique (such as generating a new, reduced-size image by mapping every other point into a new matrix). Using the newly created image as the basis for generating another down-sampled image, another lower resolution image of the original scene is obtained.

Self-determining and self-calibrating edges are a result of expanding the reduced-resolution image to a full-size image by doubling a pixel in both x and y directions and then subtracting the original matrix from the newly expanded matrix.

By generating three additional levels of resolutions from one scene, three sets of edge-based images can be obtained, each of which images is generated from two adjacent varying-resolution image pairs.

Once it is determined that edge-based pixels can be generated from an image by applying the above-discussed down-sampling expansion and matrix subtraction process, additional (third-generation) edge-pixel based images can be generated from the second-generation (edge-pixel based) images.

All of the edge-based images can be expanded to a full-resolution image by multiplying a factor of 2, 4, 16, . . . , respectively, in both x and y directions.

An edge-pixel based image is a general image composed of pixels of varying intensity values; therefore, a simple thresholding operation can be performed either to retain or to eliminate certain pixels. In general, only strong edges are

retained in order to generate objects that have strong contrasts against the background.

An object is generally represented by a uniform region. While the above-discussed methodology begins with edge extraction, the end result can be expressed in terms of a region. This is particularly obvious when an object is a point feature or a linear feature, rather than a large-size area feature. In order to make the edge-based pixels correspond to the area base of a real-world object, certain spatial, morphological operations can be applied to the edge-based images. The following are examples of object generation by creating regions from the edge-based pixels using spatial morphological processors:

- a) Binary Image Generation: Generating a binary image from a greytone image can be achieved by performing a simple thresholding operation. For instance, after weak edge points are eliminated, the values of all pixels whose intensity values are greater than zero can be changed into a value of 255 (black).
- b) Connected Components Identification: On the binary image, an operation can be performed to merge all contiguous pixels to become one uniform region; then the resultant discrete regions can be labeled using a region identification code.
- c) Feature Attributes Generation for Each Region: After a region is generated, a set of feature attributes can be generated to describe the size, shape and location of the region.
- d) Connecting Individual Regions: In certain cases, two or more individual regions are separated by a short distance but must be connected to form one uniform region. For this, a mathematical morphological operation is performed using a rectangle centered on each pixel to extend its spatial base to connect spatially separated regions.
- e) Chopping a Region into a Number of Disconnected Regions: A region can be chopped into a number of spatially separated regions. In general, separation is made to occur at the location where the width is small.
- f) Edge Cleaning and Filling: Once a region is determined, a smoothing operation can be performed on the boundary contour while simultaneously filling the missing pixels.
- g) Buffering: A buffering operation creates an outer boundary paralleling the existing boundary contour.

Another innovative approach of the present invention is using newly-generated images, described above, to extract additional objects from the original image. The newly-created image can be used as an input to the original segmentation analysis, creating an additional information layer to perform object extraction. For instance, if a river has been labeled from the single feature image and a buffer around the river is generated around the river boundary contour, the buffer can be used to infer that a given object is located within a predetermined distance from the river bank. As shown in FIG. 8d, a river (black) has a thin-outline buffer zone in cyan.

The present invention is a process by which first a single-based image is segmented in the color domain that requires an input of two or more related bands; second, a color composite, generated from the multiple bands, is used as the basis for real-time segmentation; and third, intelligent object extraction is performed by using an image that is generated from a single-color represented feature.

As stated hereinabove, for purposes of clarity throughout this discussion, it should be understood that the concept of three spectrally-derived (color) images, while representing the preferred embodiment, is merely a subset of a more general concept: any composite having component ranges



which may be transformed into two or more respective component parts and then projected into a common space.

The process can best be understood with reference to the FIGS. The input components accept one single image (FIG. 1) or multiple images that can be combined to generate a single image (FIG. 3), as the input. The input image(s) is stored in the memory or a physical device such as a hard disk or an optical disk.

Any composite having component ranges (e.g., color) break (transform) into three (3) parts, then project into common space.

The image display device is a means for visualizing the input image(s) and the subsequently processed images. A conventional color monitor in conjunction with a graphic adapter is sufficient to perform this function.

pseudo band generation using a generalized data compression scheme generates a data compression lookup table once a compression factor is given by the user. These data compression tables can also be generated offline for immediate use.

In the preferred embodiment, a given image follows an eight-bit structure; the user selects only three bits for a given band. A transformation scheme must be provided to perform this fewer-bit mapping to generate a new image. In general, this is an extremely simple procedure.

Once two or more derived bands are generated and fewer-bit transformations are performed, two or three transformed bands are selected for mapping into a color domain using red, green or blue primary colors as optical filters. Thus, two or three images are combined into a color composite. Once this color composite is displayed on a color monitor, it is generally a true or false color image.

The digital image capture component provides a means by which a color image (FIG. 3) is digitally captured. This captured image is equivalent to an additional band over and beyond the original set of images.

The segmenter, depicted in FIG. 6 as either the level 3 processor 22 or the GIS processor 24, provides a means by which a color or a group of colors is isolated as one uniform region. Any conventional segmenter is appropriate for this operation.

The object feature descriptor generator provides a means by which a region is described by a set of feature attributes such as size, shape and location descriptors.

The knowledge base or preset models provide a means by which a physical object is described in terms of a set of rules or set of models. The models can be three-dimensional, full image, and/or partial image or a mixture of models and real world images or a mixture of partial and full images.

It is possible to generate images and extract features based on three-dimensional images. Such techniques, in accordance with the present invention, use a dual library to form part or all of the knowledge base 26, which is not based on features. Therefore, it is called a featureless recognition system. For example, the two libraries used in the preferred embodiment relate to range and height, FIGS. 11a and 11b, respectively.

Object recognition is generally achieved by matching an observed (real world) image against a set of preset models (a library). A library can be generated from a set of physical models or a set of wireframe models. A wireframe model of an object can be built from a set of points, lines and surfaces. The result of an orthographic projection of a wireframe model is a boundary contour. A differential orthographic projection, according to a set of maximum range limits, yields a set of depth contours. The difference between two depth contours is called a depth or class interval. In addition

to depth contours, a 3-D object can also be described by Fourier descriptors and moments, as discussed by Wang, Gorman and Kuhl (*Remote Sensing Reviews*, Vol. 6, pp. 229-250, 1992.). For object recognition, it is important to know to what extent the classifier relies on the information from the boundary contour alone, versus from the entire set of the depth contours.

The library 26 can also include rules for use in an expert system. The rules available can go far beyond simple specification of properties such as size, tone, shape and texture. They can include spatial relationships between objects. (AN ENGINE IS INSIDE A TANK BODY AND IS ABOVE A TREAD). Special classes which are interdependent collections of two or three other classes may also be defined. (A TANK CONSISTS OF A GUN AND A BODY AND A TREAD.) A wide variety of interactive interrogation is also available. (WHICH TANKS ARE NOT TOUCHING A TREE?)

Referring now also to FIG. 7, in the preferred embodiment, the library 26 is three-dimensional. Not only are objects represented in three dimensions in this library 26, but so too are partial images representative of portions of the objects. Thus, along with the full, three-dimensional image of each object, can be stored a partial image that is cut or cropped up to 30% from the bottom of the image. Likewise, a second partial image represents the full image less up to 30% of the leftmost portion thereof. Finally, a third partial image represents the full image less up to 30% of the rightmost portion thereof. Thus, the full library representation of each object actually consists of four components: the full image representation and three partial image representations.

The library or knowledge base 26 is updatable by the GIS processor 24. The updating procedure occurs when a fraction of real world images (full images or partials) is incorporated into the library 26. The resultant library is a mixture of model-based and real-world images. The matching operation is accomplished by using a modified k-nearest neighbor classifier, as described in Meisel (*Computer Oriented Approaches to Pattern Recognition*, Academic Press, 1972). Thus, the library can be updated as it is used, and therefore contains a more precise reflection of actual images, the more it is used.

The matcher, which is functionally located between the segmenter (the GIS processor 24) and the knowledge base or library 26, and which can be located physically in either of those components, provides a means by which an observed set of feature attributes or a segmented but featureless image is matched with a given preset model for object recognition.

The present invention has been reduced to practice to result in an ATR/I system capable of performing target identification using LADAR data beyond one kilometer, up to at least 2.2 kilometers. The system superiority comes from a drastically different approach to ATR/I processes from conventional methods.

For the past 20 years, the vast majority of ATR/I efforts have been spent on issues relating to feature extraction, observed feature set, model-based feature set, and classifier systems, shown in FIG. 10a. The underlying assumption has been that only an insignificant difference exists between the model library and observed data. The results have not been encouraging.

In the past 15 years, however, each of the ATR/I components has been examined in depth, from both theoretical and empirical considerations. As a result, a working ATR/I system is achieved by confronting the complex issue on the relationship between the observed data (FIG. 10b) and the

pure models in the library, and by treating the rest of the system components as constants. In other words, the underlying assumption for the present invention is that a significant difference exists between the model library and the observed data.

To recognize or identify the observed target M, the conditional probability P(Ti | M) must be determined for each target or significant clutter i. The posterior probability of Ti given M is defined by the Bayes rule:

$$P(Ti | M) = \frac{P(M | Ti) P(Ti) P(M)}{\sum_j P(M | Tj) P(Tj)}$$

where P(Ti) is the prior probability of the target or significant clutter, Ti.

To estimate P(M | Ti), a Parzen window density estimation scheme is used. Each target Ti is represented by a set of significant geometric distortions of Ti, TID = {Ti<sub>d</sub> | d ∈ D}, where D is a set of all significant geometric distortions caused by viewing angle Θ and object occlusion o. Each instance of the geometrically distorted target Ti<sub>d</sub> is represented by its expected measurement Mi<sub>d</sub>, where

$$M_{i_d} = T_{i_d} + N'$$

and N' is the estimated noise component for the distorted target Ti<sub>d</sub>.

The conditional probability P(M | Ti) can be estimated by

$$\sum_{j \in D} P(M | M_{i_j}) / |D|$$

where |D| is the cardinality of set D, i.e., the total number of the component in set D. P(M | M<sub>i<sub>d</sub></sub>) is normally represented by a Gaussian distribution having mean M<sub>i<sub>d</sub></sub> and a proper variance defined based on the number of training sample Mi<sub>d</sub>. The exact form of Parzen window method has been given by Meisel (1972), to which a minor modification has been made in the area of estimating the sigma parameter.

A partial shape library is employed to represent f(T, Θ, o). For example, the shape library contains a subset of the full shape constructed from varying azimuth and depression angles. For the ground-to-ground case, the azimuth angle is 40°; the depression angle is fixed at 5°. In addition, the shape library contains three partial shape components: 1) each full shape is removed 30% from the bottom; 2) each full shape is removed 30% from the lefthand side; and 3) each full shape is cropped 30% from the righthand side. These partial shape components are added to the full shape library by means of software control rather than by using a manual method.

Through the generation of the distorted target Tid using the wireframe models, the nonlinear distortion function f, and the interaction between the measurements and the geometric distortion parameters Θ and o can be straightforwardly handled offline during the training phase of algorithm development. In the offline training process, a database of the expected target models are developed for the online target recognition and identification analysis (conditional probability estimation).

The remaining issue is to estimate the measurement result for a distorted target Ti<sub>d</sub>:

$$M_{i_d} = T_{i_d} + N'$$

To determine M<sub>i<sub>d</sub></sub>, the estimated noise component N' must be determined. Many approaches have been proposed for the estimation of the measurement noise. Codrington and Tenorio (1992) used a Markov Random Field model after

Geman and Geman (1984). In this model, noise is a function of the spatial structure of the image. Geman and Geman pointed out the duality between Markov Random Field and Gibbs distributions and employed a MAP estimation paradigm to perform image restoration. The estimation, together with some assumptions about the noise model, resulted in the energy function format. In Codrington and Tenorio, the estimation is conducted by the optimization of an objective energy function consisting of a likelihood term determined by a random noise term and a smoothness term determined by the underlying image structure, which makes a smoother image structure more likely.

The energy function approach is extended to determine the measurement results. The energy function approach has the ability to incorporate many different objective functions into a single cost function to be minimized, as described in Kass, Witkin, and Terzopoulou (1988), Friedland and Rosenfield (1992) and Leclerc (1989).

Creating an energy function framework includes the definition of energy components to perform the appropriate subtasks. A combination of two components is used: the first component attempts to minimize the cost related to boundary contour shape variation; and the second component minimizes the internal image structure of the target. Thus,

$$E(x) = w_1 * E_b(x) + w_2 * E_s(x)$$

where W<sub>1</sub> and W<sub>2</sub> are weights and E<sub>b</sub>(x) and E<sub>s</sub>(x) are the boundary and structure energy components, respectively; and where x is the image value of the distorted target Ti<sub>d</sub>. The image value of the best estimated measurement result M<sub>i<sub>d</sub></sub> is the value y, which minimizes E(x).

The functional form and the parameters of the energy functions E(x) are determined during the training phase by acquiring a set of measurements of known targets under known geometric distortions. The best function is the function which yields the minimum mean squared error between the estimated values and the true measurements.

Once w<sub>1</sub> and w<sub>2</sub> are estimated, the Markov Random Field model can be applied to the existing partial shape library to create a more realistic model based shape library (images) to match against the observed LADAR images for ATR/I analysis. A simulation has been conducted using a noise model-based shape library to identify observed LADAR images. This simulation is done simply by adding a sample of observed LADAR images to the wireframe models-based partial shape library. A significant (15%–30%) improvement in correct target identification rate is obtained by replacing the original pure model library with this mixed (observed plus models) library.

The output 28 provides a means by which the result of matching is an output element which can be displayed, stored or input into the segmenter 22, 24 or back to the updated library 26.

The feedback loop 28, 22, 24, 26 provides a means by which the output 28 becomes an input into the segmenter 22, 24 for extracting additional objects.

Referring now to FIG. 8a, there is depicted a flow chart for the self-determining/self-calibrating uniform region generator and object extractor.

A single scene is provided, step 100. As described in greater detail in the aforementioned copending patent application, Ser. No. 08/066,691, various information sources can be used as input to a second-generation GIS system. In fact, combined image and map data can be used to represent the image entered into the system.

The system accepts multiple data sources for one common geographical area. The sources can be existing maps, geo-

coded, socio-economic data such as census tracts, and various images such as LANDSAT and SPOT satellite imagery. The most common information sources are images and maps.

The single scene generally has well-defined edges and is in the red spectrum. It should be understood, however, that multiple bands for the scene can also be provided. Once the scene has been provided to the system, step 100, down-sampling occurs, step 102. For the first down-sample, step 104, alternate pixels are input to the system. In this way resolution is lowered. Similarly, the image is again down-sampled, step 106, so that every second image pixel from step 104 is input to the system. Finally, the image is again down-sampled, step 108, so that every second image pixel from step 106 is input to the system.

At this point, the image formed by step 104 is expanded back to the original resolution, step 110, by doubling each of the pixels so that the number of pixels in the transformed image is equal to the number of pixels in the original image.

Edge-mapping occurs by subtracting the original image from the image of step 110, which is shown as step 112. Similarly, the image of step 106 is expanded, step 114, and the result is subtracted from the image of step 104, resulting in the image of step 116. Similarly, for the image of step 108, the number of pixels is expanded, compared and subtracted from the image of step 106 to create the image of step 118.

Thus, the image of step 116 can be expanded to the original size merely by successively doubling the number of pixels two times. Similarly, the image of step 118 can be expanded to the original image by successively doubling the number of pixels three times. The expanded images can then be integrated, step 120, to create or define the common edge or edges in the original image. Such integration is accomplished by the use of mathematical morphology procedures, as are well known in the art.

At this point, the scene provided in step 100 can be provided to another band (e.g., near infrared such as 2.0 microns, such as provided by Landsat band 7 of Thematic Mapper). The image processed by such a second band is generated as previously mentioned to provide an integrated edge map, step 120. The two resulting images from step 120 are then merged together in accordance with standard union/intersection principles in a so-called conditional probability technique.

If the scene provided in step 100 is then provided to any pseudo band or any real band, 122, 124, 126, as the new original image 100, then multi-level resolution features are extracted 128, 130, 132, respectively and steps 102-120 are repeated for each band.

All bands are integrated 136 to extract multi-band based, self-generating, self-calibrated, edge-based, feature-less regions, which are applied (step 139) to the GIS processor 24 (FIG. 6), to extract objects 140.

As mentioned above, it is also possible to generate images and extract features based on three-dimensional images, using a dual library which is not based on features.

The majority of image processing techniques assume that input data has 8-bit information (i.e., the intensity range of a pixel is 0 to 255). Beyond this range (e.g., a 12-bit image), specialized processors must be designed to analyze the data, and specialized hardware must be used to display the image. The present invention allows 16-bit data to be processed into a set of 8-bit images. In addition, using LADAR depth data, this 8-bit mode approach provides a reliable environment for segmentation and object identification.

In real world conditions, an object is 3-dimensional. From a given angle, the object is actually only a 2½-dimensional

object, because the other half of an object cannot be perceived until it is rotated. Given that an object has 2½-dimensional information, a set of depth contours can be used to present the object; each contour-based plane is perpendicular to the optical axis. The last contour (away from the sensor location) is called the boundary contour. In a 2-D image domain such as obtained from FLIR (forward looking infrared) imagery, the boundary contour is generally referred to as a silhouette.

The principal use of boundary contour is to perform object recognition using shape information. The variables used to describe a given contour are called shape descriptors. Conventionally, researchers use Fourier descriptors and moments as shape descriptors. Another approach is to use a neural network to perform classification analysis by using binary silhouette-based images as input without having to extract feature attributes. While this boundary contour-based method is extremely effective in recognizing airplane types, it is not effective for ground vehicle recognition. For this reason, researchers use depth information to perform object recognition. Laser radar is an appropriate sensor for generating depth information; therefore, the image is called a range image, as oppose to an intensity-based image.

Wang, Gorman and Kuhl (ibid) conducted an experiment using wireframe models of ground vehicles. The experimental results show that using the silhouette information alone, the classifier can achieve a correct recognition rate of ranging from 72 to 78 percent. By adding depth information to the classifier system, an increase of an average rate of three percentage points can be expected if depth moments are used.

In a classification analysis using real world LADAR images against a set of wireframe-based ground vehicles, similar results have been obtained regarding the contribution of boundary contour: approximately 75 percent correct classification rate. While a much higher correct identification rate has been achieved using LADAR range images—approximately 95 percent in a 6-vehicle scenario—the most important information source is still the boundary contour.

A 16-bit pixel can be reformatted into a set of 8-bit pixels by manipulating an 8-bit number at a time using a shift-right method. For instance, from a 16-bit binary coded data, a number can be generated from the first 8 bits; next, by shifting to the right one bit and using the same 8-bit range, another number can be generated. The final 8-bit number represents the highest 8 bits of the original 16-bit number. This 8-bit shift-right technique is illustrated as follows:

TABLE II

ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
B0	0	-----														I
B1		1	-----													I
B2			2	-----												I
B3				3	-----											I
B4					4	-----										I
B5						5	-----									I
B6							6	-----								I
B7								7	-----							I
B8									8	-----						I

In terms of an image, taking 8 bits at a time is similar, in a very general sense, to the down-sampling multi-resolution

technique described above. The difference is that using this method, the dimensions of the image remain the same, whereas using a down-sampling approach, the size of the image (i.e., number of pixels) is gradually reduced by a factor of four.

A segment can be used to determine whether an object of certain size and shape is contained in a candidate interval. Then the automatic searching process can be terminated.

From the aforementioned data transformation model, each original 16-bit data item is decomposed into nine reduced resolution numbers: B0 through B8. (In LADAR data structure, a B0-based image is equivalent to the 8-bit LADAR AM channel data, and B8 is close to its FM channel data.) However, empirical data show that the B8-based images have less noise as compared to the corresponding FM image.

The self-determining segmentation method of the present invention assumes that an object has been detected by a FLIR sensor; therefore, the approximate centroid of the object is known. Under this assumption, the intensity value of the object centroid can be obtained. This value now represents the center of the 2½-D object.

The next step is to use a depth interval of "centroid value ±2" to create a binary image using the forming rule set:

any (depth) value within the designated depth interval becomes 1;

any (depth) value outside of the designated depth interval is set to 0. (1)

The final segmentation map is generated by multiplying the B0 image by the B8-derived binary image, or

$$\text{Segmented Image } B0\_Seg = B0 \times B0\_Binary. \quad (2)$$

The foregoing ranging method can be set in an automated mode. The process begins with an arbitrary number and a preset class interval. Then the next search is performed on the next interval, which is created by adding a number to the original arbitrary number. For example, the first search interval is set as 11-15 (or  $13 \pm 2$ ). Then the second search interval is 12-16.

Equation (2), above, is equivalent to using a mask created by the B8 to perform segmentation using the B0 data set. Thus, segmentation of B0 is entirely determined by B8. The generation of the binary is based on a simple thresholding principle, shown as Equation (1), above. Human intervention is minimal.

In general, the image B0<sub>15</sub> Seg has clutter attached at the bottom and both sides. Additional processing is thus required to create a clutter-free boundary contour. For this, an intelligent segmenter is used which is capable of merging neighboring pixels and subsequently performing region absorption based on size, shape and other criteria.

As a rule, in LADAR images, the edge value between object pixels is much smaller than its counterpart outside of the object. The area outside of the object is generally called the background. Therefore, using a thresholding value of approximately 10 would merge all of the object pixels to form a relatively large uniform region. At this stage, however, some background pixels are still likely to be attached to the object.

Accordingly, the next step is to use an extreme size difference penalty function to merge a relatively small-sized region into a relatively large-sized background. It should be noted that this merge is based on size criterion, not intensity or depth criterion. Since the clutter attached to the object may exhibit itself as multiple layers, this extreme size difference penalty function may have to be performed a number of times.

In certain cases, a small-sized penalty function may have to be performed to merge all small-sized clusters into one region. Since in LADAR images the clutter usually comes from the bottom of the object, this newly-formed region is usually extremely elongated. This is in opposition to the object that is either a square or a rectangle with minimal elongation. Then, the next step is to merge the elongated region with the general background using a shape criterion.

The last step is to merge relatively large-sized background objects with the general background using the same extreme size difference penalty function.

For this post-B0\_Seg segmentation analysis to be successful, the approximate distance between the object and the sensor location should be known. This information is needed to set the size parameter for the target, and a range value depending on the orientation of the object. The following program is a sample general penalty function-based segmentation method that is used to generate high-quality boundary contour from LADAR Images.

As noted earlier, an object is recognized only when a match occurs between an observed object (sensor dependent) and an element of a library. In addition, a library element can be represented by a set of descriptors (Fourier or moments, or any variables) or simply a model image without going through an intermediate step of feature extraction.

A means by which a match or no match is determined is generally referred to as a classifier or as classification logic. A simple form of a classifier can be a minimum distance criterion using a simple distance computation formula. However, this minimum distance classifier can be adapted for any feature-based classification system. For example, a Fourier descriptor-based classifier can be executed in a general framework of a minimum distance classifier because the Fourier coefficients can be used to compute a distance from an observed object to an element of a shape library.

A minimum distance classifier is also known as a nearest neighbor classifier because a minimum distance defines the nearest neighbor. If two or more library elements are used to compute the distance from an observed object to a given element in the library, this classifier is generally known as a K-nearest neighbor classifier.

Conventional pattern recognition techniques to perform a matching analysis use a set of features to describe an object. This set of features is then compared to another set of features that describe an element of a library or training set. An alternative to this approach is to match a raw image (instead of the extracted features) to raw models (also without feature extraction) in a library to determine the best match element. This is an innovative approach because heretofore it has not been successful in pattern recognition.

To be successful in using this feature-free method, the models must be prepared in such a way that they are realistic representations of real world objects. In addition, this realistic representation is judged from a visual and graphic point of view, rather than from measurable features. This is a significant departure from conventional approaches to object recognition.

This invention proposes two visual and graphic based representations of an object to perform object recognition. The first model uses depth contours to present a 3-D object. The second model uses structural components arranged by height above the ground to present a general object. The combination of these two models leads to a generalized 3-D representation of an object. For example, in a depth contour-based representation, the Z-axis coincides with the optical axis; whereas in a structural component representation, the Z-axis is approximately perpendicular to the optical axis.

A 3-D object can be constructed from three data files: a point file, a line file and a surface file. A line is constructed from a set of points and a surface is constructed from a set of lines. Furthermore, a surface so constructed is a planar surface.

A depth contour-based model can be constructed by an orthographic projection of the 3-D wireframe model that intersects a set of perpendicular planes. The distance between a pair of planes is called a depth interval. The result of this 3-D representation is analogous to using a set of contour lines to represent a terrain feature, such as a conic hill being represented by a set of concentric circles. Unlike topographic contour lines, pixels having varying greytone values are used in the preferred embodiment to present a 3-D object.

A tank can be conceptualized as an object that has these structural elements: a gun, a turret, a body, a track component and a set of wheels in the track. Using the same wireframe model data files, each structural component can be labeled using a set of surfaces. In addition, the centroid of the structural component can be used to represent the height of the object above the datum, which can be specified by the user.

Following the same projection method, and using the centroid information of each component, a model can be generated of a given object in terms of the spatial arrangement of its structure components. A model can be projected in terms of a set of viewpoints, each of which is obtained from a combination of a specific azimuth angle and a specific depression angle.

The above discussion requires both a range/depth library and a height/structural library to present a 3-D object for matching an observed object that possesses depth information, such as a laser radar image. If the observed object does not have depth information (e.g., a FLIR image), only the height component of the dual library is applicable to a matching analysis. However, this dual library concept is still valid if only the boundary contour is used to represent the depth library component of the system.

To match against elements in a depth contour library, the boundary contour of an observed object in a range image must be extracted first. After the boundary contour is extracted, the original depth-coded pixels are applied to the space enclosed by the boundary contour. The last step is to perform a filtering (e.g., a 3 by 3 median filter) to remove the noise in the range image.

For matching against the height library, the above processed range image must be segmented using an appropriate segmenter to bring out the structure component of the observed object. The segmented range image is rescaled and greytone-coded according to the values of the y-axis of the centroids of all segmented regions. A set of matching examples with this dual library system is shown as FIG. 11a, depicting portions of an M60 tank (broadside view) from the height library and as FIG. 11b, depicting the same image from the contour library.

Using feature descriptors, researchers generally select a minimum distance classifier to perform the matching analysis. Since this invention proposes the use of a graphically-represented global image as an input and the library component as well, to perform a matching analysis without measurable features, two dissimilar classifiers are proposed. The first is a modified K-nearest neighbor classifier following the work of Meisel (Computer Oriented Approaches to Pattern Recognition, Academic Press, 1972). The second is a standard back propagation neural network. Meisel's work belongs to a general Parzen window methodology for which

a major reference is Parzen's work entitled "An Estimation of a Probability Density Function and Mode", *Annals of Mathematical Statistica*, 33, pp. 1056-1076.

While this invention is not a new classifier, per se, two most appropriate classifiers are identified herein that should be an integral part of an object recognition system. The classifiers are unique in two aspects:

- 1) The use of visual, graphic representation of objects in both the observed component and the library component to perform matching analysis. This innovation eliminates the traditional process of feature extraction, a drastic and significant departure from the traditional pattern recognition paradigm.
- 2) The use of a dual library, range library and height library, to present a 3-D object.

Conventional pattern recognition systems have the following components:

- (1) Input data,
- (2) Feature extractor,
- (3) Classifier,
- (4) Training set or library, and
- (5) Output.

The present invention replaces the above set of components with:

- (1) Input data,
- (2) Self-determining and self-calibration segmenter,
- (3) Intelligent segmenter using general penalty functions,
- (4) Visual and graphic representation of the observed objects and the library models,
- (5) Designated classifier (not any classifier),
- (6) Dual 3-D library,
- (7) Output, and
- (8) Feedback loop linking the output to the segmenter component.

The present invention can integrate the two aforementioned object generation approaches into one system. Using the system components of the object extractor, multiple bands are generated from a single band image. Inputting these multiple bands into the self-determining/self-calibrating edge-pixel and uniform-region extractor yields multiple sets of regions from the original single scene. These multiple sets of regions provide the basis for extracting objects in the vector processing domain using rule sets and/or image libraries.

With multiple sets of input, a new object (spatial union based) can be defined in terms of the union of multiple individual regions. Using the same principle, a new object (spatial intersection based) can be defined in terms of the intersection portion of multiple individual regions.

Spatial normalization is performed by taking the ratio of pairs between the mesotexture, as described in U.S. Pat. No. 5,274,715, granted to the present inventor, of two adjacent cells in two separate directions as shown in the following example:

Original MTX Matrix			
mt(1,1)	mt(1,2)	mt(1,3)	mt(1,4)
mt(2,1)	mt(2,2)	mt(2,3)	mt(2,4)
mt(3,1)	mt(3,2)	mt(3,3)	mt(3,4)
mt(4,1)	mt(4,2)	mt(4,3)	mt(4,4)
Row-wider Normalization			
mt(1,2)/mt(1,1)	mt(1,3)/mt(1,2)	mt(1,4)/mt(1,3)	
mt(2,2)/mt(2,1)	mt(2,3)/mt(2,2)	mt(2,4)/mt(2,3)	
mt(3,2)/mt(3,1)	mt(3,3)/mt(3,2)	mt(3,4)/mt(3,3)	



-continued

mt(4,2)/mt(4,1)	mt(4,3)/mt(4,2)	mt(4,4)/mt(4,3)	
Column-wide Normalization			
mt(2,1)/mt(1,1)	mt(2,2)/mt(1,2)	mt(2,3)/mt(1,3)	mt(2,4)/mt(1,4)
mt(3,1)/mt(2,1)	mt(3,2)/mt(2,2)	mt(3,3)/mt(2,3)	mt(3,4)/mt(2,4)
mt(4,1)/mt(3,1)	mt(4,2)/mt(3,2)	mt(4,3)/mt(3,3)	mt(4,4)/mt(3,4)

Each transformed matrix is called a simple structure because the majority of the background elements are represented by a value close to 1. In addition, cells having values significantly different from 1 most likely contain a "foreign object" or belong to an interface zone.

For ground-to-ground FLIR images, the column-wide simple structure reflects a scene structure composed of the Sky Zone, the Sky/Tree Interface, the Tree Zone, the Tree/Ground Interface, the Ground Zone, and the Ground/Ground Interface. If the airplane rolls while acquiring images, however, the Sky/Tree Interface and the Tree/Ground Interface lines will not be pure east-west straight lines. As a result, the original ground-to-ground simple structure may not match a general air-to-ground scene structure. Therefore, it will not be a good model for predicting target locations using structure as an inference engine.

The proposed isotropic transformation combines both row and column directions into one edge-based index to detect "foreign objects" existing in all directions: horizontal, vertical and diagonal. The specific normalization algorithm is as follows:

$$\gamma(i) = [\sum_{k \in N(i)} \text{meso}(i) \text{meso}(j) / (\sum_{k \in N(i)} f_j)]$$

where N(i) is the neighboring cell set for the cell i and meso(k) is the mesotexture index for the cell k.

The image understanding (IU) Model #1 has the following specifications:

- 1) Three Neighboring Interface Rows:
  - First Row: Sky/Tree Interface
  - Second Row: Tree/Tree Interface
  - Third Row: Tree/Tree Interface or Tree/Ground Interface
- 2) Target Row Detection Principles:
  - a) Target row(s) are located at the last Interface Row or on one or two rows below the last of the Triple Interface Rows. Effective Rows are the last two Triple Interface plus two rows below.
  - b) If there is no Triple Interface, possible Effective Rows are the Sky/Tree Interface plus four rows.
  - c) The column-wide SOS (sum of squares) Ratio Difference can be used to find the Target Row: Usually, the maximum or last of the three very large SSRD (sum of squares of ratio differences) values is/are linked to the Tree/Ground Interface.
- 3) Target Column Detection Principles
  - a) Row-wide SSRD global maximum or local maximum is the target column.
  - b) Local cluster of maxima with a 2 by 2 cell is a potential target location. Search is limited to Effective Rows.
- 4) Miscellaneous Universal Rules, applicable to all models:
  - a) Mesotexture maxima Effective Rows are restricted global maxima.
  - b) Mesotexture maxima row/column coincidence are target locations.

c) Mesotexture first Rank row/column coincidence are target locations.

d) Row and Column-wide Minimum pairwise ranked correlation are target column or rows.

5 Accordingly, once the two simple structures are replaced by an isotropic normalization derived interface values, the above-described scene structure and its associated target detection rules are invalid.

Beyond simple operations of union and/or intersection, objects can be generated having a spatial base between the region of union and the region of intersection. Once edge-based and region-based objects are generated, information integration can be performed at a higher level (an object level) as opposed to a pixel level. The vehicle used for performing this process is the IMAGE system described in copending application, Ser. No. 08/066,691.

After the detection of the scene structure and content, geometric evidence can be extracted from the detected scene structure and regions. The geometric evidence for each cell can be defined as it is "inside tree region", "outside tree region", "inside sky region", "outside sky region", "inside road region", "outside road region", "inside ground region", "outside ground region", "nearby tree-ground region", "nearby sky-ground region", etc. The evidence can be defined as fuzzy membership functions. Fuzzy memberships are defined for: "inside tree region and outside tree region." To determine the fuzzy membership, the shortest distance between the cell of interest and the tree boundary is measured. When the distance is small, the "inside tree region" evidence has a higher value, and the "outside tree region" has a lower value. As the distance increases, the confidence value of the "inside tree region" is progressively decreased and the confidence value of the "outside tree region" is progressively increased.

6 A geometric reasoning process is to be applied to each detected potential target. The reasoning process integrates the potential target probability with the geometric evidence derived from the scene structure and content. A probability reasoning approach based on the Bayes Belief Network paradigm is used to perform this multi-source information integration. The input information is integrated and updated through Bayes rules rather than the ad hoc method. For example, each node has a vector of input conditions and a vector of output conditions. The mapping between the input conditions and the output conditions is through a conditional probability (CP) matrix. The CP matrix encodes the relationship between the input and the output conditions. Prior probabilities can be assigned to the output conditions of each node. The belief of a node is the probability of the output conditions in the node. The belief is updated incremental as new evidence is gathered.

As an example, a Bayes network configured as a tree structure can be used for target detection application. A root node name "detection node" is used to make the final target vs. clutter decision. The probability of the potential target derived from the ratio image map is used as a prior probability of the root node. The root node is supported by five evidence nodes: a sky node, a tree node, a ground node, a road node, and a tree-ground node. Each of these nodes provides one aspect of the geometric evidence. The input conditions of the sky node are "inside sky region" and "outside sky region" which can be determined from the fuzzy membership functions described above if sky regions exist. The output conditions of the sky node are "target confirm" and "target reject" which provides contextual information to support the target detection decision. The relationship between the "inside sky region", "outside sky

region" conditions and the "target confirm" and "target reject" conditions are encoded in a 2 by 2 CP matrix with the following elements:

I	{	P inside sky region I	}	P	{	inside sky region I	}	I
		target confirm				target reject		
I	{	P outside sky region I	}	P	{	outside sky region I	}	I
		target confirm				target reject		

The conditional probability elements can be determined by human expert or by an off-line training process. For example, it is obvious that if the ground target is confirmed it is unlikely to be inside sky region. Therefore, we will assign a very small value to P(inside sky region I target confirm). The principles of the operations for the other evidence are similar to the sky node.

The relationship between the "target confirm" and "target region" conditions and the "target" or "clutter" conditions are converted to evidence of target vs. clutter decision. This evidence along the prior target probability will be used to generate the output target vs. clutter decision.

The output of the network is a probability vector of P(target) and P(clutter)=1-P(target). A detection threshold can be defined in such a way that we call an object "target" if P(target)> detection threshold and call an object "clutter" otherwise. The detection threshold can be set for the detection system based on the criteria defined earlier to achieve the best compromise between the detection sensitivity and the false alarm rate.

Since other modifications and changes varied to fit particular operating requirements and environments will be apparent to those skilled in the art, the invention is not considered limited to the example chosen for purposes of disclosure, and covers all changes and modifications which do not constitute departures from the true spirit and scope of this invention.

Having thus described the invention, what is desired to be protected by Letters Patent is presented in the subsequently appended claims.

What is claimed is:

1. A method of generalizing objects or features in an image, the steps comprising:
  - a) retrieving an original image, each pixel of which has a value represented by a predetermined number of bits, n;
  - b) transforming said original image into at least two distinct bands, each of said pixels in each of the band-images comprising less than n bits;
  - c) transforming said original image into at least two distinct resolutions by a series of down sampling schemes;
  - d) projecting each of said transformed, band-images, into a composite domain;
  - e) expanding each down-sampled image back to previous resolution by doubling, tripling, or quadrupling the pixels in both the x and y direction;
  - f) creating a composite image from all of said transformed, band-images; and
  - g) creating edge-based images by a series of comparisons and integrating among all resultant edge-based images.
2. The method of generalizing objects or features in an image in accordance with claim 1, the steps further comprising:
  - h) digitizing said composite image, if necessary; and p1 i) analyzing regions in said segmented images to identify objects.

3. The method of generalizing objects or features in an image in accordance with claim 1, the steps further comprising:

- g) coordinating said composite image with independently-generated information to identify features and objects.
4. The method of generalizing objects or features in an image in accordance with claim 1, wherein said transforming step (b) comprises compressing said original image, each of said bands using a different compression factor or technique, and said transforming step (c) comprises extracting said original image, each of said bands using a different down-sampling factor or technique.
5. The method of generalizing objects or features in an image in accordance with claim 4, wherein said compressing of said original image is accomplished by extracting the square root of the pixel values thereof.
6. The method of generalizing objects or features in an image in accordance with claim 4, wherein said compressing of said original image is accomplished by extracting the log of the pixel values thereof.
7. The method of generalizing objects or features in an image in accordance with claim 4, wherein said compressing of said original image is accomplished by extracting the double log of the pixel values thereof.
8. The method of generalizing objects or features in an image in accordance with claim 4, wherein said compressing of said original image is accomplished by filtering said pixel values thereof.
9. The method of generalizing objects or features in an image in accordance with claim 1, wherein said transforming step (b) comprises bit reduction and remapping without compression.
10. A self-calibrating, self-determining method of generalizing objects or features in an image, the steps comprising:
  - a) retrieving an original image in pixel form;
  - b) transforming said original image into at least one stable structure band;
  - c) executing a predetermined algorithm with said transformed image to perform iterative region-growing or region-merging by interrogating said image with a set of linearly-increasing color values;
  - d) generating groups having a set of values indicating a number of regions in each segmented image;
  - e) monitoring a slope and slope change of a scene characteristic (SC) curve;
  - f) establishing at least one stopping point;
  - g) projecting each of said transformed, band-images, into a composite domain; and
  - h) creating a composite image from all of said transformed, band-images.
11. The self-calibrating, self-determining method of generalizing objects or features in an image in accordance with claim 10, wherein said stopping point occurs when the slope or slope change of said SC curve is greater than zero.
12. The self-calibrating, self-determining method of generalizing objects or features in an image in accordance with claim 10, the steps further comprising:

29

- i) digitizing said composite image, if necessary; and
- j) analyzing regions in said segmented images to identify objects.

13. The self-calibrating, self-determining method of generalizing objects or features in an image in accordance with claim 12, the steps further comprising: p1 k) coordinating said composite image with independently-generated information to identify features and objects.

14. The self-calibrating, self-determining method of generalizing objects or features in an image in accordance with claim 10, wherein said transforming step (b) comprises

30

compressing said original image, each of said bands using a different compression factor or technique.

15. The method of generalizing objects or features in an image in accordance with claim 10, wherein said library comprises data representative of full images or a combination of portions thereof.

16. The method of generalizing objects or features in an image in accordance with claim 15, wherein said library further comprises a noise model for facilitating matching of an actual image with stored library elements.

\* \* \* \* \*





US005781665A

# United States Patent [19]

Cullen et al.

[11] Patent Number: 5,781,665

[45] Date of Patent: Jul. 14, 1998

[54] APPARATUS AND METHOD FOR CROPPING AN IMAGE

[75] Inventors: Mark F. Cullen, Bethany, Conn.; Mayur N. Patel, Los Angeles, Calif.

[73] Assignee: Pitney Bowes Inc., Stamford, Conn.

[21] Appl. No.: 519,903

[22] Filed: Aug. 28, 1995

[51] Int. Cl.<sup>6</sup> G06K 9/00; G06K 9/34; G06K 9/38; G06K 9/40

[52] U.S. Cl. 382/254; 348/268; 358/445; 380/23; 382/115; 382/118; 382/171; 382/173; 382/174; 382/266; 382/271; 382/282; 382/286; 382/291; 395/761

[58] Field of Search 382/171, 173, 382/254, 266, 271, 115, 118, 174, 282, 286, 291; 348/268; 358/445; 380/23; 395/761

### [56] References Cited

#### U.S. PATENT DOCUMENTS

5,159,667	10/1992	Borrey et al.	395/761
5,315,393	5/1994	Mican	348/268
5,343,283	8/1994	Van Dorsselaer et al.	358/445
5,420,924	5/1995	Berson et al.	380/23

5,481,622 1/1996 Gerhardt et al. 382/171

Primary Examiner—Leo H. Boudreau  
Assistant Examiner—Daniel G. Mariam  
Attorney, Agent, or Firm—Robert H. Whisker; Melvin J. Scolnick; Robert Meyer

### [57] ABSTRACT

A method and apparatus for cropping an image in digital form. The image to be cropped is represented as a first digital array which is operated on by an edge enhancement transformation to generate a second, binary digital array wherein edges of the image are emphasized. The second digital array is then partitioned into predetermined segments which are typically rows and columns of the array and the pixel values of each row and column are summed to generate brightness sums. The second digital array is then partitioned into a first, brighter, central group of rows and a second, less bright group consisting of upper and lower borders of rows; and a third, brighter, central group of column and left and right borders of columns in accordance with predetermined criteria relating to the brightness sums. The boundaries between the borders and the central groups are then applied to the first digital array and only those pixel values corresponding to pixel values common to the first and third groups are output to generate a cropped image.

16 Claims, 6 Drawing Sheets

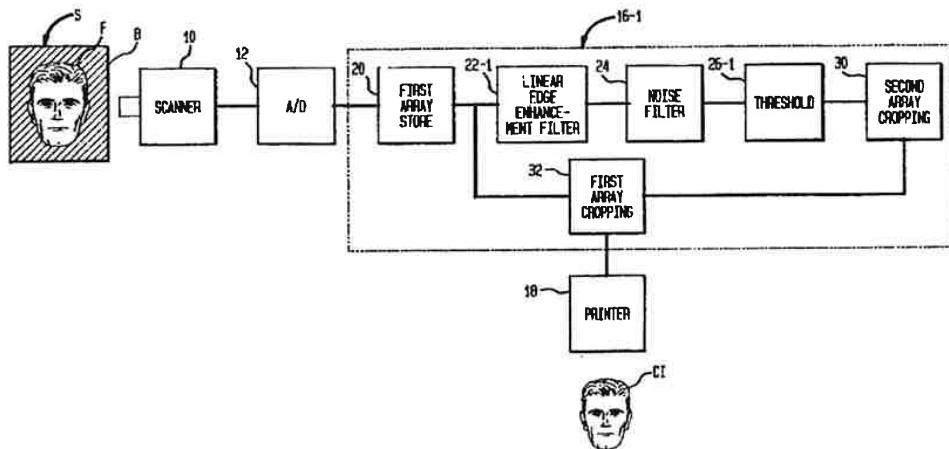


FIG. 1

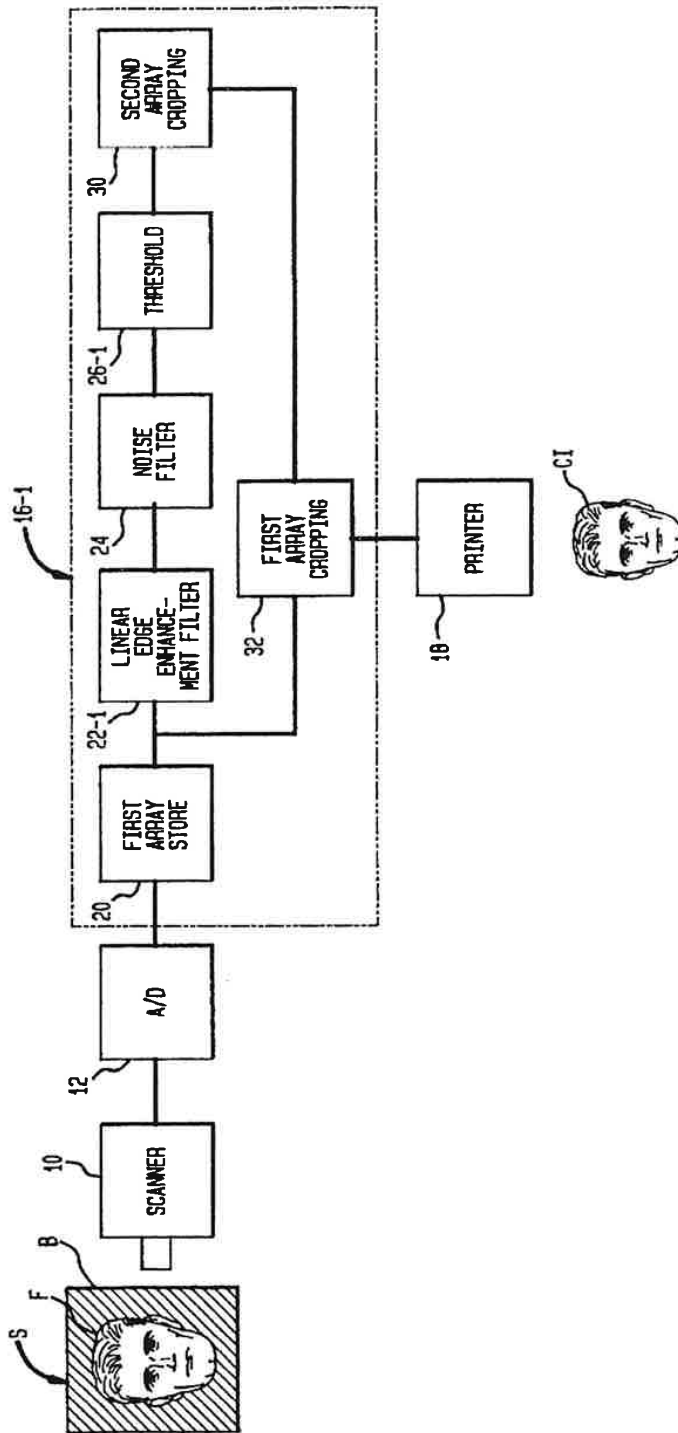


FIG. 2

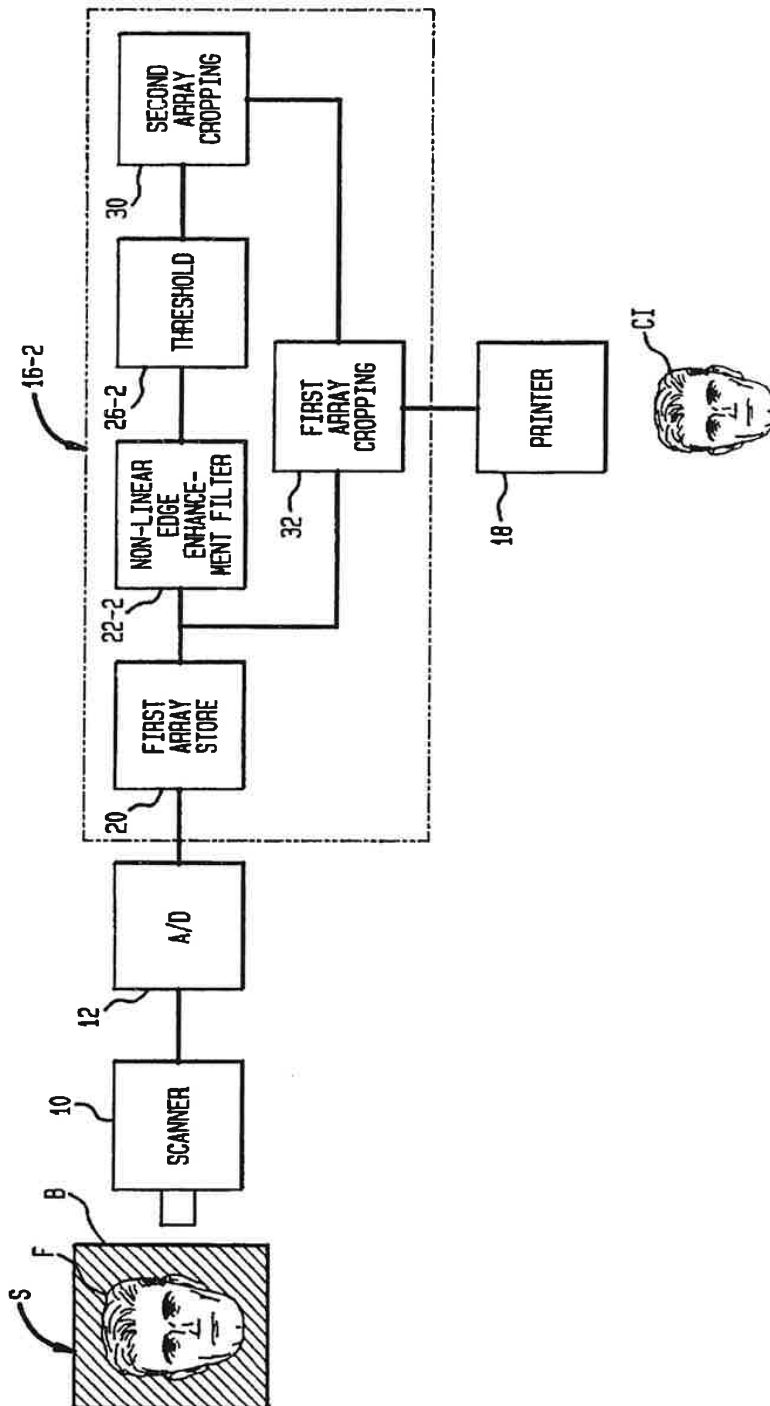


FIG. 3

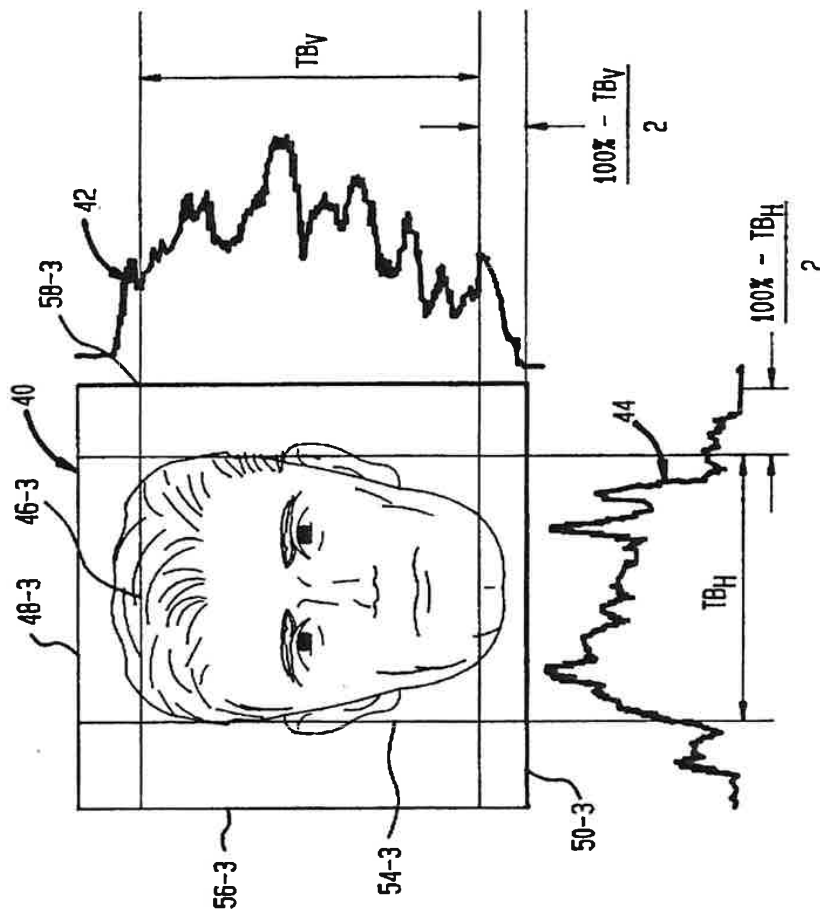


FIG. 4

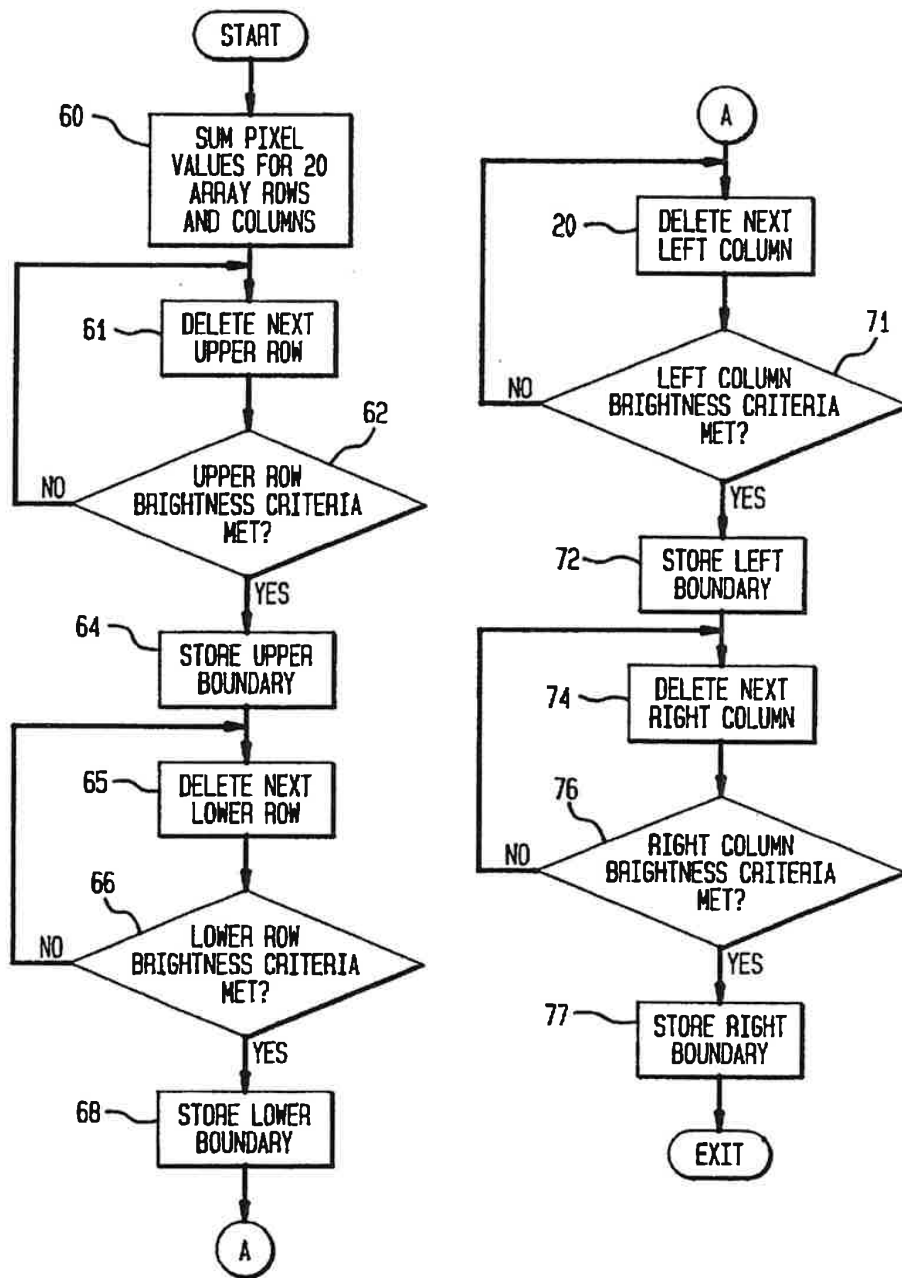


FIG. 5

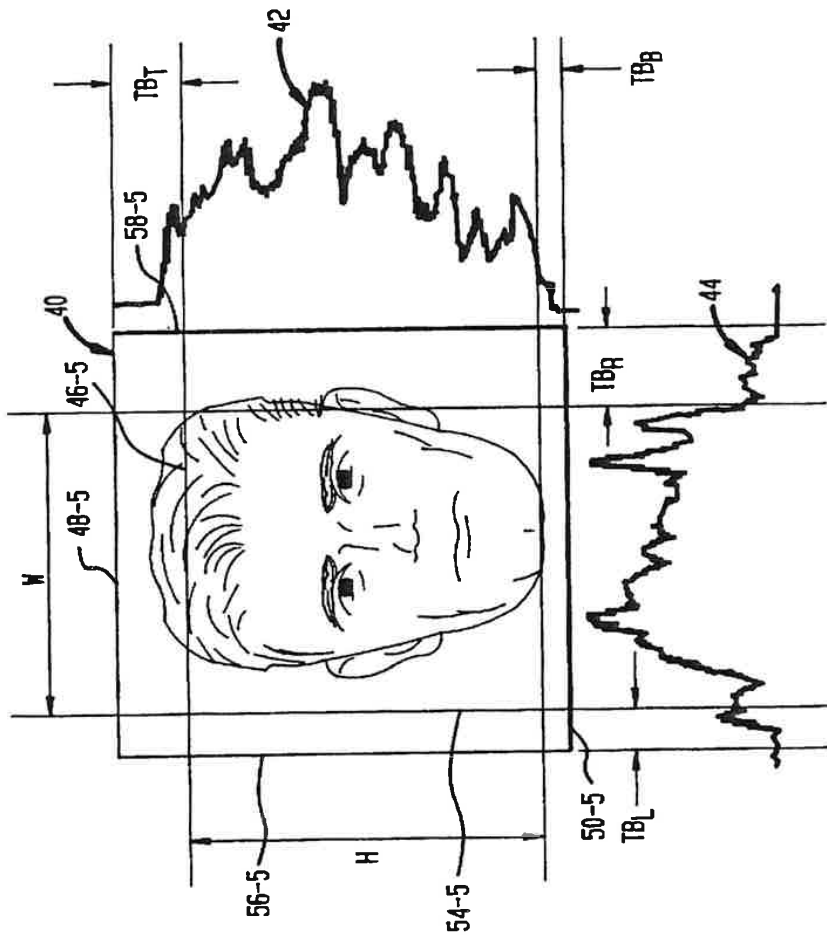
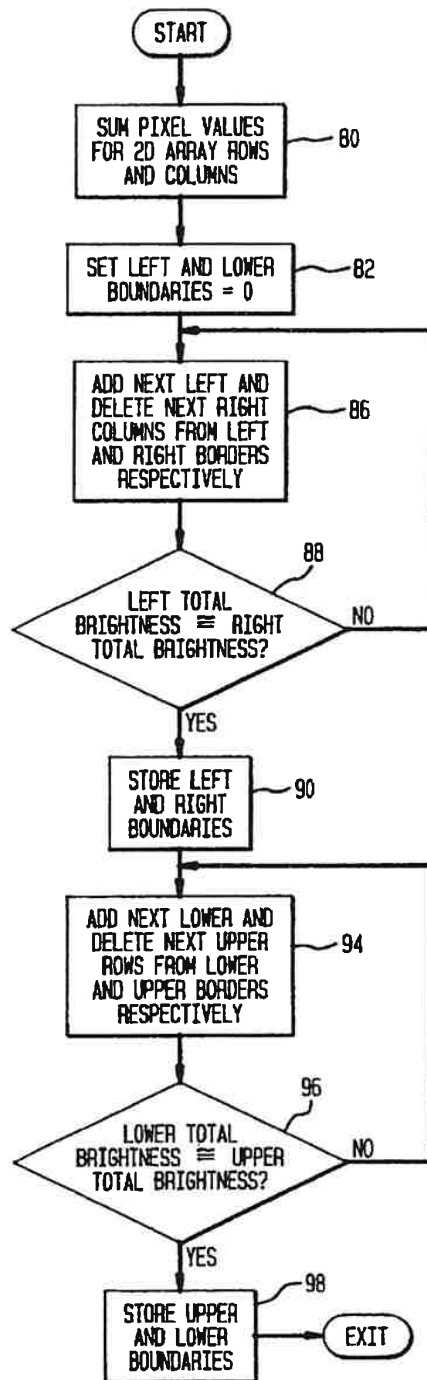


FIG. 6



## APPARATUS AND METHOD FOR CROPPING AN IMAGE

### BACKGROUND OF THE INVENTION

The subject invention relates to an apparatus and method for cropping an image to remove portions of the image which contain relatively little detail (i.e. have a low information content). More particularly it relates to a method and apparatus for cropping an image of a person's face.

U.S. Pat. No. 5,420,924; to: Berson et. al; issued: May 30, 1995 discloses an identification card which includes an image of a person to be identified together with an encrypted digital representation of that image. Such a card can be verified by decrypting the digital representation and displaying it for comparison with the image on the card. Preferably the digital representation is stored on the card in the form of a two dimensional barcode. In order to reduce the amount of area on the card consumed by the barcode needed, it would be very desirable to crop the image to eliminate as much of the image background as possible.

Other applications where it would be desirable to crop an image will be readily apparent to those skilled in the art.

Thus, it is an object of the subject invention to provide a method for automatically cropping an image.

One known method of producing such cropped images is to have a skilled operator manually crop an image by physically cutting away portions of a photographic image, or by electronically manipulate a digital array representing the image through a computer system. Another method would be to have a skilled technician initially create a closely focused image which contains minimal amounts of background. These approaches, however, require high degrees of judgment and care which could prove to be unduly expensive for applications where large numbers of identification cards must be produced, as where the identification card also serves as a drivers license.

### BRIEF SUMMARY OF THE INVENTION

The above object is achieved and the disadvantages of the prior art are overcome in accordance with the subject invention by means of an apparatus and method for cropping an image which is represented as a digital array of pixel values. The digital array is first processed to produce a second digital array which corresponds to a transformation of the image to enhance edges in the image. (i.e. The boundaries between areas of uniform or gradually changing intensity are emphasized while variations within such areas are de-emphasized.) The second digital array is then partitioned into predetermined segments and the pixel values for each segment are summed to obtain a brightness sum for each of the segments. The segments are then divided into a first, higher brightness group and a second, lower brightness group in accordance with predetermined criteria relating to the brightness sums. A group of the first digital array which corresponds to the first group of the second digital array is then identified and at least part of the group of the first digital array is then output to generate a cropped image.

In accordance with one aspect of the subject invention a threshold is applied to each value output by the edge enhancement transformation so that the second digital array is an array of binary pixel values.

In accordance with another aspect of the subject invention the edge enhancement transformation includes applying a non-linear edge enhancement filter to the first digital array

and the applied threshold is selected as a function of the background portion of the image.

In accordance with still another aspect of the subject invention the edge enhancement transformation includes applying a linear edge enhancement filter to the first digital array and then applying a noise filter to output of the linear edge enhancement filter.

In accordance with still another aspect of the subject invention the segments are horizontal rows of the second digital array and the first group of segments is a continuous group of rows which contain a predetermined fraction of the total brightness of the second digital array.

In accordance with still yet another aspect of the subject invention the segments are horizontal rows (or vertical columns) of the second digital array and the second group of segments is a predetermined number of the rows (or columns) divided into two contiguous outboard subgroups of the rows (or columns), the subgroups having equal brightness.

Since the density of detail (i.e. information content) of an image closely correlates to the density of edges in that image, those skilled in the art will readily recognize that the above summarized invention clearly achieves the above object and overcomes the disadvantages of the prior art. Other objects and advantages of the subject invention will be apparent to those skilled in the art from consideration of the attached drawings and the detailed description set forth below.

### BRIEF DESCRIPTION OF THE DRAWINGS

Various preferred embodiments of the subject invention are shown in the following figures wherein substantially identical elements shown in various figures are numbered the same.

FIG. 1 shows a schematic block diagram of a preferred embodiment of the subject invention.

FIG. 2 shows a schematic block diagram of a second preferred embodiment of the subject invention.

FIG. 3 is an illustration of one method of cropping an image in accordance with the subject invention.

FIG. 4 is a flow diagram of the method of FIG. 3.

FIG. 5 is an illustration of another method of cropping an image in accordance with the subject invention.

FIG. 6 is a flow diagram of the method of FIG. 5.

### DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS OF THE SUBJECT INVENTION

Turning to FIGS. 1 and 2, two preferred embodiments of the subject invention are shown. In each of these embodiments a subject S is scanned by conventional scanner 10 whose output is converted to digital values by conventional A/D converter 12. Preferably subject S consist of face F of a person to be identified by an identification card and background B, which is preferably a uniform, substantially featureless screen or the like.

The output of A/D converter 12 is stored in first array store 20 which is comprised in both cropping apparatus 16-1 and 16-2. Store 20 stores a first array of pixel values, which are preferably greyscale values. Apparatus 16-1 and 16-2 process the first array to provide an output to printer 18 (or other suitable output device) to provide cropped image CI, as will be described further below.

Turning to FIG. 1 the output of store 20 is transformed by linear edge enhancement filter 22-1, noise filter 24 and



threshold 26-1 to generate a second digital array of binary pixel values corresponding to an image of subject S having its edges enhanced.

The second digital array is then cropped by second array cropping element 30, which sums the pixel values for each of a plurality of predetermined segments into which the second digital array have been partitioned to obtain a brightness sum for each of the segments. These segments are divided into at least a first group having a relatively higher total brightness and a second group having a relatively lower total brightness. The boundaries between the first and second groups of the second digital array are then applied to the first digital array by first array cropping apparatus 32 to identify a group of the first digital array which corresponds to the first group of the second digital array. The identified group of the first array is then output to printer 18 to produce cropped image CI.

In a preferred embodiment element 30 may further crop the second digital array by partitioning the second digital array into a set of segments which are then divided into a third relatively bright and forth relatively less bright groups and element 32 outputs only those pixel values of the first digital array which correspond to values of the second digital array common to the first and third groups, as will be described further below.

Linear edge enhancement filter 22-1 begins the transformation of the first digital array by successively convolving the first digital array with each of four 3x3 masks shown in Table I below. The results of these four convolution operations are then summed to provide an output.

(The functioning of such edge enhancement filters, and of the nonlinear filter which will be described with respect to FIG. 2 are well known in the art and need not be discussed further here for an understanding of the subject invention.)

TABLE I

mask1	mask2	mask3	mask4
$\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$	$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$

The output of filter 22-1 is then applied to noise filter 24 which is preferably a conventional "blurring" filter to remove artifacts which might be interpreted as false edges. The output of noise filter 24 is then applied to threshold 26-1 to produce a binary second digital array. Threshold values of approximately 200 have provided satisfactory results where the pixel values of first digital array represented a 256-level greyscale.

Turning to FIG. 2 a first digital array is produced, and a second digital array is cropped and applied to the first digital array to produce cropped image CI in a manner substantially identical to that described with respect to FIG. 1 and apparatus 16-2 differs from apparatus 16-1 only in the manner in which the second digital array is generated. In apparatus 16-2 the first digital array is applied to non-linear edge enhancement filter 22-2 which sequentially convolves two 3x3 masks, shown in Table II below, with the first digital array. The absolute values of these convolution operations are then summed to provide the output of filter 22-2. In another embodiment of the subject invention the RMS value of the convolution operations may be taken as the output. Threshold 26-2 is then applied to the output of filter 22-2 to generate the second digital array. By using a non-linear filter

and appropriately selecting threshold 26-2 apparatus 16-2 eliminates the need for a noise filter.

TABLE II

mask1	mask2
$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$

Determination of the proper threshold value to use in apparatus 16-2 is a function of background B. For a given background color, lighting conditions, and camera position and parameters, threshold 26-2 may be calibrated by repeatedly generating a second digital image of background B only; i.e. without a foreground subject, and adjusting threshold 26-2 to minimize the number of asserted noise pixels. This calibration process can readily be automated by a person skilled in the art. Where background B is known threshold 26-2 may be preset; or, where background B may vary threshold 26-2 can be calibrated in the field.

Preferably apparatus 16-1 and 16-2 are implemented by programming a general purpose digital computer to carry out the various functions illustrated. Programming of such a computer to implement appropriate sub-routines to carry out the illustrated functions would be a routine matter for a person of ordinary skill in the art and need not be discussed further here for an understanding of the subject invention.

FIG. 3 is an illustration of one manner in which second array cropping element 30 can operate. Image 40 is a line drawing representation corresponding to the second digital array and showing an edged enhanced image of subject S. By emphasizing edges and de-emphasizing variations within areas of constant or slowly varying intensity image 40

concentrates brightness (i.e. asserted pixels) in areas of high detail, thus emphasizing face F and particularly high detail areas such as the eyes of face F. Pixel values are summed for the rows and columns of the second digital array to obtain brightness sums for image 40. Histogram 42 represents a plot of horizontal row brightness sums as a function of vertical position and histogram 44 represents a plot of vertical column brightness sums as a function of horizontal position. (Histograms shown in FIGS. 3 and 5 are intended as illustrative only and are not actually derived from the line drawing representations shown.)

In accordance with the embodiment of the subject invention illustrated in FIG. 3 the second digital array is first partitioned into two groups of horizontal rows of pixel values; a first, central, brighter group 46-3 and a second, less bright group consisting of upper border 48-3 and lower border 50-3 in accordance with criteria which require that first group 46-3 contain a predetermined fraction TB<sub>1</sub> of the total brightness of image 40 and that the remaining fraction of the total brightness be evenly divided between upper border 48-3 and lower border 50-3.

The columns of the second digital array are then divided into a third, central brighter group 54-3 and a forth, less

bright group consisting of left border 56-3 and right border 58-3. The criteria for dividing the columns into groups are similar to the criteria applied to the rows with group 54-3 containing a predetermined fraction  $TB_n$  of the total brightness of image 40 and borders 56-3 and 58-3 having the remaining total brightness evenly divided between them.

Once these groups are identified the borders between groups are applied to the first digital array by cropping element 32; which outputs only those pixel values of the first digital array which correspond to pixel values common to both central common brighter groups 46-3 and 54-3 to printer 18 to generate cropped image CI.

Values of 80% for  $TB_n$  and  $TB_n$  have been found to provide substantial reduction in the number of pixels required to represent cropped image CI while still retaining sufficient detail so that cropped image CI is easily recognizable.

FIG. 4 shows a flow diagram of the operation of cropping element 30 in implementing the embodiment described above with respect to FIG. 3. At 60 element 30 sums pixel values for the second array rows and columns to generate row and column brightness sums. Then at 61 the next (i.e. outer most remaining) upper row is deleted and at 62 element 30 tests to determine if the upper row brightness criteria have been met; that is, for the preferred embodiment described above, has approximately 10% of the total brightness been deleted. If the criteria has not been met element 30 returns to 61 to delete the next upper row, and, if the criteria has been met, at 64 stores the upper border between group 46-3 and upper border 48-3. Then at step 65, 66 and 68 the lower boundary between border 50-3 and central group 46-3 is determined and stored in the same manner. Then at step 70, 71 and 72; and at steps 74, 76, and 77 the columns of the second digital array are divided into central group 54-3 and borders 56-3 in the same manner. Then, as described above identified boundaries are applied to the first digital array to generate cropped image CI.

(Those skilled in the art will recognize that, since only whole rows or columns can be deleted the above described brightness criteria (and those described below with respect to FIGS. 5 and 6) will, in general, only be met approximately.)

In other embodiments of the subject invention values for fractions  $TB_n$  and  $TB_n$  can be unequal and the total brightness in boundaries 48-3 and 50-3, and 56-3 and 58-3 need

not be equal. In embodiments where face F is symmetrically positioned central groups 46-3 and 54-3 may simply be positioned symmetrically about the horizontal and vertical axes of image 40 by deleting the outermost pairs of rows or columns until the predetermined fraction of the total image brightness is left.

FIG. 5 shows an illustration of an other embodiment of the invention wherein cropping element 30 operates on the second digital array to divide the rows into a first, central, brighter group 46-5 having a predetermined height H and a second, less bright group consisting of upper border 48-5 and lower border 50-5; and to divide the columns into a third, central, brighter group 54-5 having a predetermined width W; and a fourth less bright group consisting of left border 56-6 and right border 58-5.

FIG. 6 shows a flow diagram of the operation of cropping element 30 on the second digital array in the embodiment described with respect to FIG. 5. At 80 element 30 sums the pixel values for the second digital array rows and columns. Then at 82 the left end lower borders are set equal to zero. That is group 46-5 is initially assumed to begin at the left edge of image 40 and group 54-5 is initially assumed to begin at the lower edge of image 40. Then at 86 the next (i.e. outermost) column is added to the left border and the next (i.e. innermost) column is deleted from the right border; and at 88 element 30 tests to determine if the left border total brightness equals the right border total brightness as closely as possible. If not element 30 returns to 86 to delete and add the next pair of columns; and, if the total brightness of the left and right borders are equal, stores the left and right boundaries between groups 54-3 and borders 56-3 and 58-3 at step 90. Then at steps 94, 96, and 98 element 30 operates on the rows of the second digital array to divide the pixel values into groups corresponding to groups 46-3 and border 48-3 and 50-3 in the same manner.

In other embodiments of the subject invention the boundaries between the central and border groups of the rows and columns may be taken at the outermost peaks of histograms 42 and 44 respectively and still other embodiments of the subject invention the second digital array may be partitioned into segments other than rows and columns. For example, the segments may be taken as concentric, and annular rings of approximately equal area and the image may be cropped radially.

#### EXAMPLE

TABLE III

	Pixel Area Reduction: (Area measured in pixels <sup>2</sup> )								
	Manual Crop 12 pixels/side			Auto-Cropped (90%)		Auto-Cropped (85%)		Auto-Cropped (80%)	
	Original Area	Area	% of Original	Area	% of Original	Area	% of Original	Area	% of Original
Albert	32279	24215	75.02%	16256	50.36%	13221	40.96%	10816	33.51%
Eric	32279	24215	75.02%	18445	57.14%	15038	46.59%	13066	40.48%
GeorgeH	32279	24215	75.02%	23760	73.61%	20808	64.46%	15960	49.44%
James	32279	24215	75.02%	19398	60.09%	16872	52.27%	15080	46.72%
Lady	22879	16159	70.63%	15729	68.75%	13700	59.88%	11500	50.26%
Mayur	32279	24215	75.02%	19602	60.73%	16912	52.39%	14214	44.03%
Steve	32279	24215	75.02%	18207	56.41%	15194	47.07%	12544	38.86%
Theresa	32279	24215	75.02%	18048	49.72%	12669	39.25%	9890	30.64%
Averages:					59.60%		50.36%		41.74%

TABLE IV

Compressed File Reduction: (Tested using JPEG with Q factor = 60)									
	Manual Crop 12 pixels/side		Auto-Cropped (90%)		Auto-Cropped (85%)		Auto-Cropped (80%)		
	Original bytes	% of Original	bytes	% of Original	bytes	% of Original	bytes	% of Original	
Albert	1262	1004	79.56%	833	66.01%	753	59.67%	691	54.75%
Eric	992	742	74.80%	635	64.01%	573	57.76%	440	44.35%
GeorgeH	1130	893	79.03%	889	78.67%	799	70.71%	599	53.01%
James	1233	831	67.40%	723	58.64%	641	51.99%	642	52.07%
Lady	767	539	70.27%	609	79.40%	542	70.66%	423	55.15%
Mayur	1107	843	76.15%	794	71.73%	639	57.72%	550	49.68%
Steve	1196	896	74.92%	779	65.13%	631	52.76%	477	39.88%
Teresa	937	721	76.95%	590	62.79%	493	52.61%	433	46.21%
Averages:			74.88%		68.32%		59.24%		49.39%

Table III shows examples of the subject invention where subject's faces were scanned to generate a first, 169x191, 256 greyscale level array of pixel values. The first array was operated on by a four mask linear filter and a conventional noise filter as described above, and a threshold of 200 applied to generate a second, binary array. The second array was then cropped to central groups of rows and columns having the various percentages of total brightness shown. For each percentage of total brightness the percentage of pixels in the cropped image (i.e. pixels common to the two central groups) is given. A predetermined fixed cropping of 12 pixels/side is also shown for purposes of comparison.

Table IV shows the same percentages where the images are also compressed using the well known JPEG compression algorithm; demonstrating substantial benefits even with compression of the images.

The embodiments of the subject invention described above have been given by way of illustration only, and those skilled in the art will recognize numerous other embodiments of the subject invention from the detailed descriptions set forth above and the attached drawings. Accordingly, limitations on the subject invention are found only in the claims set forth below.

What is claimed:

1. A method of cropping an image, said image being represented as a first digital array of pixel values, said method comprising the steps of:
  - a) processing said first digital array to produce a second digital array of pixel values, said processing including applying an edge enhancement transformation to said first digital array;
  - b) partitioning said second digital array into predetermined segments;
  - c) summing pixel values for each of said segments to obtain a brightness sum for each of said segments;
  - d) dividing said segments into first and second groups in accordance with predetermined criteria relating to said brightness sums;
  - e) identifying a group of said first digital array corresponding to said first group of said second digital array;
  - f) outputting at least a part of said group of said first digital array to generate a cropped image.
2. A method as described in claim 1 wherein said processing further includes applying a threshold to each value output by said edge enhancement transformation whereby said second digital array is an array of binary pixel values.

3. A method as described in claim 2 wherein said edge enhancement transformation comprises applying a non-linear edge enhancement filter to said first digital array.

4. A method as described in claim 3 wherein said image includes a substantially featureless background and said threshold is selected as a function of said background.

5. A method as described in claim 2 wherein said edge enhancement transformation comprises applying a linear edge enhancement filter to said first digital array and then applying a noise filter to output of said linear edge enhancement filter.

6. A method as described in claim 1 wherein said segments are horizontal rows or vertical columns of said second digital array.

7. A method as described in claim 6 wherein said first group of segments is a contiguous group of said rows or of said columns containing a predetermined fraction of the total brightness of said second digital array.

8. A method as described in claim 7 wherein said predetermined fraction is approximately equal to 80 percent.

9. A method as described in claim 6 wherein said second group of segments consists of two contiguous borders outboard of said rows or columns, said borders having equal brightness.

10. A method of cropping an image, said image being represented as a first digital array of pixel values, said method comprising the steps of:

- a) processing said first digital array to produce a second digital array of pixel values, said processing including applying an edge enhancement transformation to said first digital array;
- b) partitioning said second digital array into predetermined horizontal rows of pixel values;
- c) summing pixel values for each of said horizontal rows to obtain brightness sums for each of said rows;
- d) dividing said rows into first and second groups in accordance with predetermined criteria relating to said horizontal row brightness sums;
- e) partitioning said second digital array into predetermined vertical columns;
- f) summing pixel values for each of said vertical columns to obtain brightness sums for each of said columns;
- g) dividing said columns into third and fourth groups in accordance with predetermined criteria relating to said column brightness sums;
- h) identifying a part of said first digital array corresponding to pixels common to said first and third groups of said second digital array;

9

i) outputting said part of said first digital array to generate a cropped image.

11. A method as described in claim 10 wherein said processing further includes applying a threshold to each value output by said edge enhancement transformation whereby said second digital array is an array of binary pixel values.

12. A method as described in claim 11 wherein said edge enhancement transformation comprises applying a non-linear edge enhancement filter to said first digital array.

13. A method as described in claim 12 wherein said image includes a substantially featureless background and said threshold is selected as a function of said background.

14. A method as described in claim 11 wherein said edge enhancement transformation comprises applying a linear edge enhancement filter to said first digital array and then applying a noise filter to output of said linear edge enhancement filter.

15. A method as described in claim 1 wherein said image is an image of a human face.

10

16. An apparatus for cropping an image, said image being represented as a first digital array of pixel values, said apparatus comprising:

- a) means for processing said first digital array to produce a second digital array of pixel values, said processing including applying an edge enhancement transformation to said first digital array;
- b) means for cropping said second digital array to form first and second groups of predetermined segments of said second digital array in accordance with predetermined criteria relating to brightness of said segments;
- c) means for identifying a group of said first digital array corresponding to said first group of said second digital array;
- d) means for outputting at least a part of said group of said first digital array to generate a cropped image.

\* \* \* \* \*



US005880858A

**United States Patent** [19]  
**Jin**

[11] **Patent Number:** **5,880,858**  
[45] **Date of Patent:** **Mar. 9, 1999**

[54] **METHOD OF AUTO-CROPPING IMAGES FOR SCANNERS**

[75] **Inventor:** Yuan-Chang Jin, Hsinchu, Taiwan

[73] **Assignee:** Mustek Systems Inc., Hsin-Chu, Taiwan

[21] **Appl. No.:** 1,979

[22] **Filed:** Dec. 31, 1997

[51] **Int. Cl.<sup>6</sup>** ..... H04N 1/04

[52] **U.S. Cl.** ..... 358/487; 358/453; 358/465

[58] **Field of Search** ..... 358/465, 466, 358/487, 474, 453, 462, 467; 348/64; 382/169, 172, 175, 176

[56] **References Cited**

**U.S. PATENT DOCUMENTS**

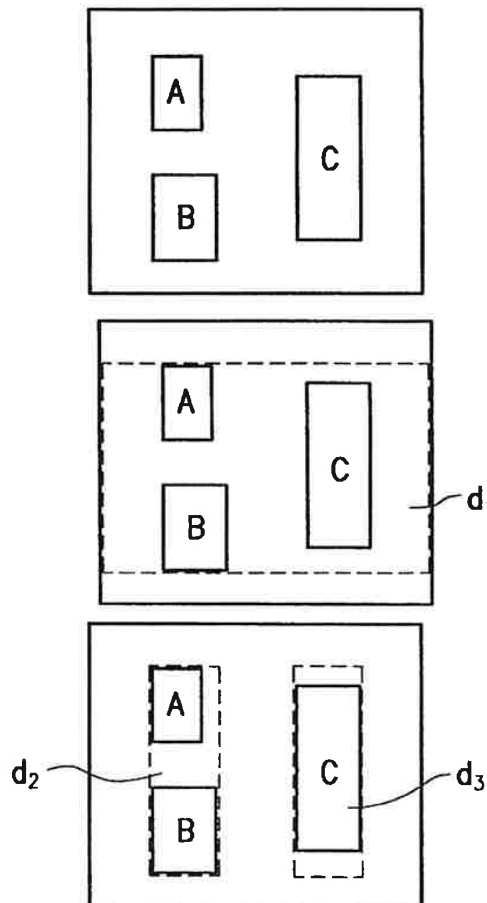
5,270,688 12/1993 Dawson et al. .... 345/150

*Primary Examiner*—Scott Rogers  
*Assistant Examiner*—Jerome Grant, II  
*Attorney, Agent, or Firm*—Ladas & Parry

[57] **ABSTRACT**

A method of auto-cropping images appropriate for scanners is disclosed, that features use of the image-division method for carrying out auto-cropping. According to the method, the images of the objects after pre-scanning can be auto-cropped, and therefore do not require further manual operation. The interference due to the background color of the cover and other redundant images can also be reduced. Furthermore, the present invention is also appropriate for scanning positive and negative films, and films are disposed on frames can be scanned properly.

**9 Claims, 7 Drawing Sheets**



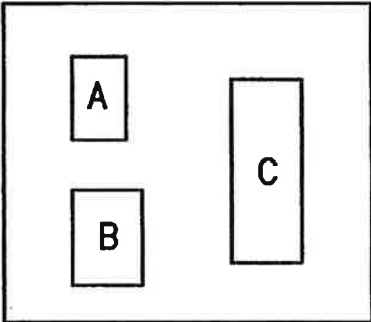


FIG. 1A

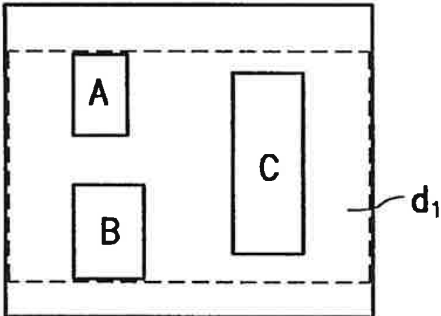


FIG. 1B

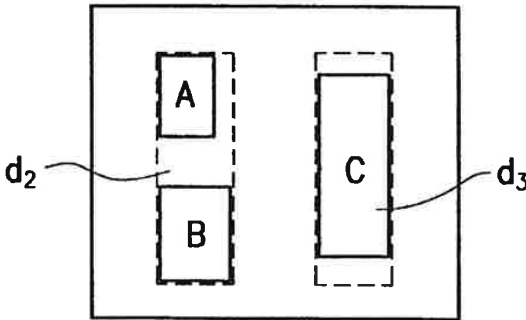


FIG. 1C

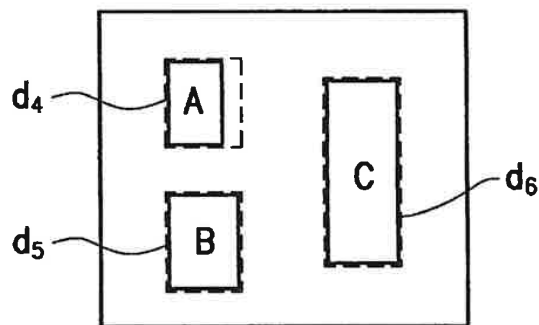


FIG. 1D

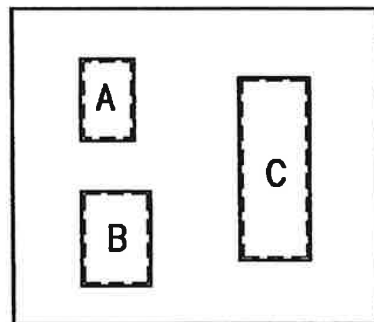


FIG. 1E

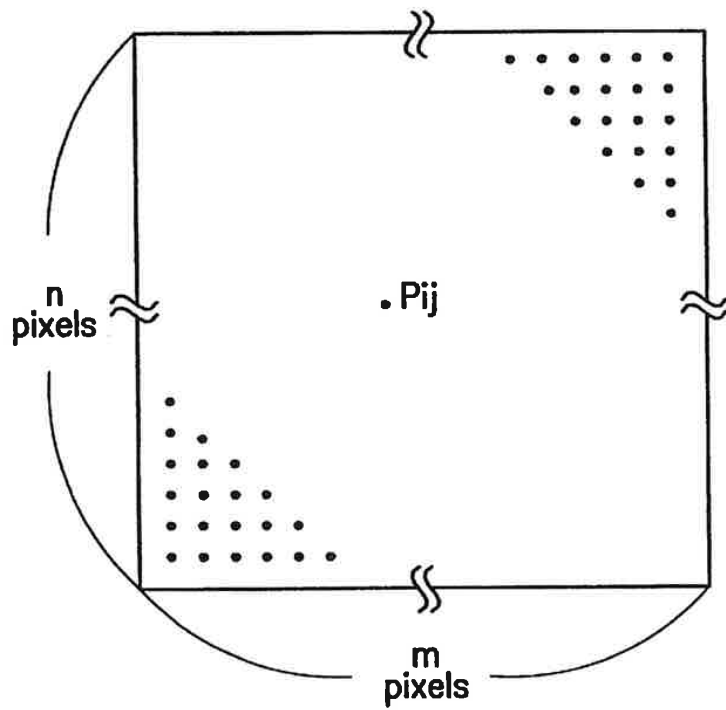


FIG. 2



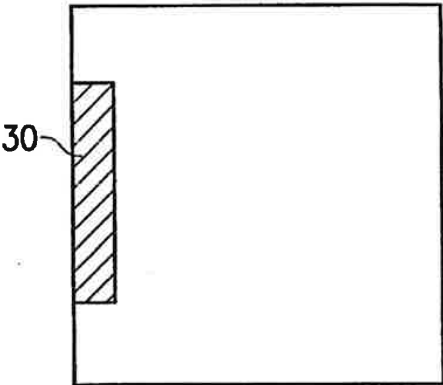


FIG. 3A

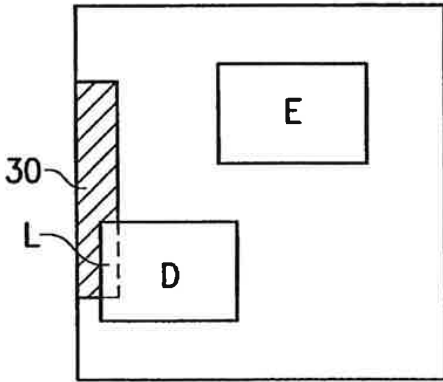


FIG. 3B

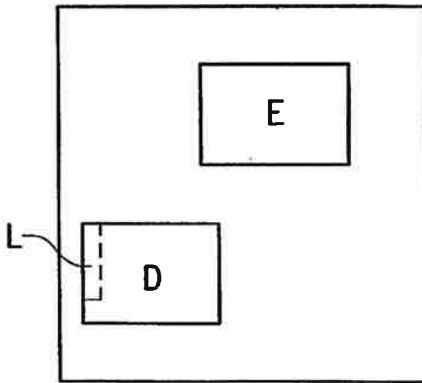


FIG. 3C

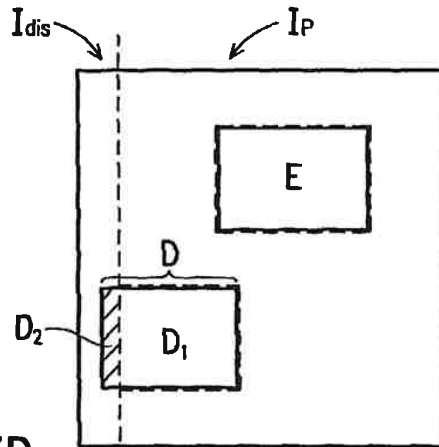


FIG. 3D

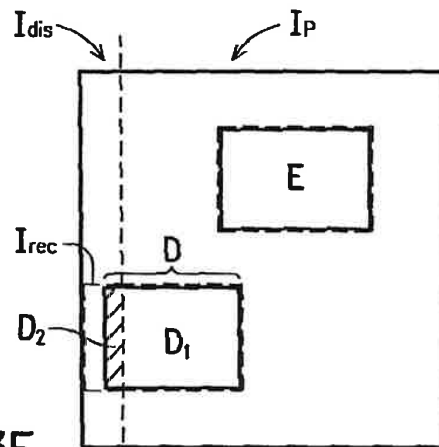


FIG. 3E

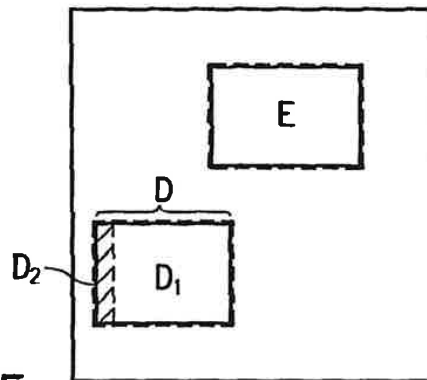


FIG. 3F

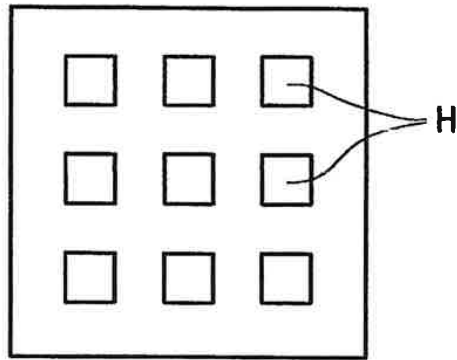


FIG. 4A

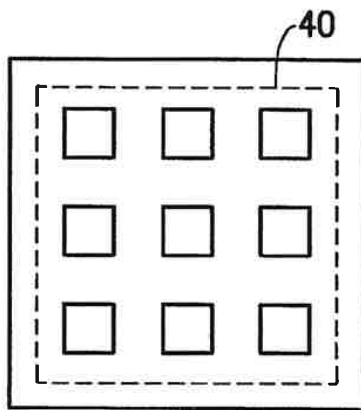


FIG. 4B

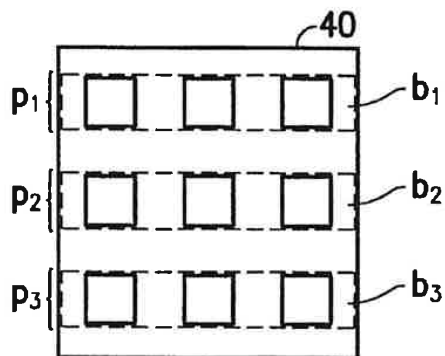


FIG. 4C

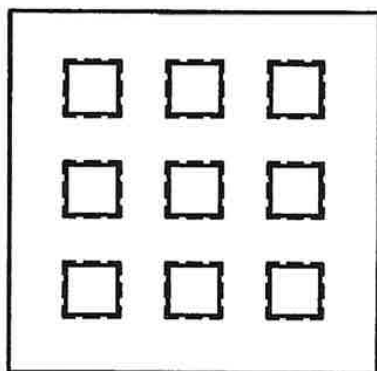


FIG. 4D

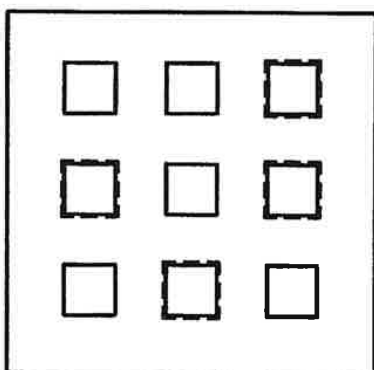


FIG. 4E

## METHOD OF AUTO-CROPPING IMAGES FOR SCANNERS

### BACKGROUND OF THE INVENTION

#### 1. Field of the Invention

The present invention generally relates to an image-processing method. More particular, the present invention relates to a method of auto-cropping images for scanners, accordingly the images prescanned by a scanner can be located and cropped automatically without further manual operation by the user.

#### 2. Description of the Related Art

Generally speaking, after putting objects in a scanner, for example documents and drawings, the scanner prescans the objects and displays the images of the objects in a preview window. For conventional scanners, users must crop the prescanned images manually by using pointers such as a mouse to select the required images. Cropping the images manually is very inconvenient. Especially, in the case of scanners with batch-scan functions, users waste a great deal of time cropping the images, while batch-scanning the objects. Without an auto-cropping function, users are subject to inconvenience and work efficiency is degraded.

### SUMMARY OF THE INVENTION

Accordingly, the object of the present invention is to provide a method of auto-cropping images for scanners, such that the prescanned images displayed in a preview window can be auto-cropped. Therefore, users don't have to crop the images, thus increasing convenience and efficiency.

The disclosed method can be applied to scanners with or without covers. In addition, interference due to the background color of the cover can also be solved.

Furthermore, the present invention is appropriate for scanners that scan positive films and negative films, and films with or without frames can also be determined and the images selected properly.

In order to achieve the above objects, one method of auto-cropping images for scanners according to the present invention is proposed, wherein the steps are as follows:

- (a) provide a prescanned image, wherein the pixels of the image are processed by the image division method to obtain at least a low threshold and a high threshold;
- (b) compare the pixels in every horizontal row with the low threshold, wherein the number of any pixel that is larger than the low threshold is recorded respectively to obtain a dot-intension number for every row, and the dot-intension number of every row is compared with a limit, whereby the rows with dot-intension numbers that exceed the limit are cropped, and the prescanned image is divided into several image regions;
- (c) compare the pixels of every vertical column in every divided image region with the low threshold, wherein the number of the pixels that exceed the low threshold is recorded respectively to obtain a dot-intension number for every column, and the dot-intension number of every column is compared with the limit, whereby the columns with dot-intension numbers that exceed the limit are cropped, and every one of the divided image region is further divided into several cropped regions; and
- (d) iterate steps (b) and (c) to further divide the cropped regions horizontally and vertically, wherein the iterating process stops when every horizontal and vertical

division of the cropped regions can not form any new divided region.

In order to achieve the above objects, another method of auto-cropping images for scanners wherein that must crop films with or without frames is also proposed, wherein the steps are as follows:

- (I) the frame detection step comprises the following sub-steps: (1a) processing all pixels in a preview window to obtain a low threshold and a high threshold; (1b) comparing the pixels in every horizontal row with the low threshold, and recording the number of pixels that exceed the low threshold respectively to obtain a dot-intension number with respect to every row, then comparing the dot-intension number of every row with a limit, and cropping the rows with dot-intension numbers that exceed the limit, and recording the row number of every cropped regions; (1c) transforming the size of the regions for disposing films into pixel dots; and (1d) comparing every one of the horizontal row numbers recorded with the pixel dot-number of the region for disposing a film, and if at least two of the horizontal row numbers and the dot-number have a difference that falls within a specific range, the frames are then detected.

After confirming that the films are disposed in frames, the auto-cropping step is carried out, comprising the following sub-steps: (2a) processing the pixels of the prescanned image by the image division method to obtain at least a low threshold and a high threshold; (2b) comparing the pixels in every horizontal row with the low threshold, and recording the number of pixels that exceeds the low threshold respectively to obtain a dot-intension numbers with respect to every row, then comparing the dot-intension number of every row with a limit, and cropping the rows with dot-intension numbers that exceed the limit, and dividing the prescanned image into several divided image regions; (2c) comparing the pixels in every vertical column of every the divided image region with the low threshold, and recording the number of pixels that exceed the low threshold respectively to obtain a dot-intension number with respect to every column, and comparing the dot-intension number of every column with the limit, thereby the columns with dot-intension numbers exceeding the limit are cropped, and further dividing every divided image region into several cropped regions; (2d) iterating steps 2b and 2c to further divide the cropped regions horizontally and vertically, wherein the iterating process stops when every horizontal and vertical division doing to the cropped regions can not form any new divided region; and (2e) comparing the pixels in every cropped region with the high threshold, and recording the number of the pixels that exceed the high threshold to serve as a sum number, and if the sum number of every cropped region exceeds the total pixel dots in a cropped region for a certain proportion, then no film is disposed at the cropped position, whereby every cropped region can be checked sequentially to locate the correct position of the films; wherein if in step I the frames are not detected, then the steps described steps 2a-2d in process II are carried out to crop the film regions properly.

### BRIEF DESCRIPTION OF THE DRAWINGS

Other objects, features and advantages of the present invention will become apparent by way of the following detailed description of the preferred but non-limiting embodiment. The following description is made with reference to the accompanying drawings.

FIG. 1A to FIG. 1E illustrate the process of the first embodiment according to the present invention.

FIG. 2 illustrates the pixel distribution in a preview window of a scanner.

FIG. 3A to FIG. 3F illustrate the process of the second embodiment according to the present invention.

FIG. 4A to FIG. 4E illustrate the process of the third embodiment according to the present invention.

### DETAILED DESCRIPTIONS OF THE INVENTION

#### First Embodiment

Referring to FIG. 1A, after a scanner prescans some objects A, B, and C, the images of A, B, and C are displayed in a preview window. For conventional scanners, users must crop the images of objects A, B, and C with additional steps.

FIG. 2 illustrates the pixel distribution in a preview window of a scanner. In the first embodiment, the horizontal resolution of the preview window has  $m$  pixels, and the vertical resolution of the preview window has  $n$  pixels. Any pixel in the preview window is indicated as  $P_{ij}$ , wherein  $i, j$  indicates the coordinates and  $1 \leq i \leq m, 1 \leq j \leq n$ . The images in the preview window consist of the pixels.

In the first embodiment, after prescanning the objects A, B, and C, the images A, B, and C are displayed in a preview window, as shown in FIG. 1.

After prescanning, the steps for completing the auto-cropping method are as follows.

Step (a)

An image-division method is carried out to process the pixels to obtain at least a low threshold and a high threshold. The image-division method will be described in another section.

Step (b)

In every horizontal row, the pixels are compared with the low threshold, and the number of pixels that exceed the low threshold is recorded respectively. Therefore, each row has a corresponding dot-intension number. The dot-intension number in every row is compared with a limit, thereby selecting the rows with dot-intension numbers that exceed the limit.

In the first embodiment, a cropped region  $d_1$  is obtained, as shown in FIG. 1B.

Step (c)

Next, the cropped region  $d_1$  serves as a region for processing. In every vertical column of the region  $d_1$ , the pixels are compared with the low threshold, and the number of pixels that exceed the low threshold is recorded respectively. Therefore, each column has a corresponding dot-intension number. Then, the dot-intension number in every column is compared with the limit, thereby selecting the columns with dot-intension numbers that exceed the limit. Consequently, the region  $d_1$  is further divided into regions  $d_2$  and  $d_3$  as shown in FIG. 1C.

Step (d)

Then, the regions  $d_2$  and  $d_3$  are used as the processing regions, and the above step (a) is carried out again. In every horizontal row of the regions  $d_2$  and  $d_3$  the pixels are compared with the low threshold, and the number of pixels that exceed the low threshold is recorded respectively. Therefore, each row has a corresponding dot-intension number. The dot-intension number in every row is compared with a limit, thereby selecting the rows with dot-intension numbers that exceed the limit. Consequently, the regions  $d_2$  and  $d_3$  are divided into three regions  $d_4, d_5$ , and  $d_6$ , as shown in FIG. 1D. It is obvious that the images of objects B and C are well cropped as the regions  $d_5$  and  $d_6$ .

Next, the regions  $d_4, d_5$ , and  $d_6$  are used as the regions for processing, and the above step (b) is carried out again. In every vertical column of the regions  $d_4, d_5$ , and  $d_6$ , the pixels are compared with the low threshold, and the number of pixels that exceed the low threshold is recorded respectively. Therefore, each column has a corresponding dot-intension number. Then, the dot-intension number in every column is compared with the limit, thereby selecting the columns with dot-intension numbers that exceed the limit. Finally, the images of objects A, B, and C are cropped properly. For the final cropped regions corresponding to the images of A, B, and C, further division in these regions can not be formed, even when the steps (a) and (b) are repeated. Consequently, auto-cropping processing is completed, as shown in FIG. 1E.

The image division method mentioned above is described hereinafter.

In general, a fixed threshold is adopted in the conventional image-division method. The image data are compared with the fixed threshold to divide the image data into two groups.

The image-division method applied in the present invention is different from the conventional one. Its processing steps will be described as follows.

The image-division method comprises the following steps:

- (a) First, a plurality of initial values  $a_1 \sim a_n$  are selected corresponding to a plurality of sets  $S_1 \sim S_n$ , serving as central characteristic values of the above sets, where  $a_1 < \dots < a_n$ .
- (b) After calculating the color values of every one of the pixels  $P_{ij}$  in the preview window (FIG. 2), for example taking the average of the color values, the pixel characteristic values corresponding to all pixels are obtained respectively.
- (c) The differences between every pixel characteristic value and every central characteristic value are calculated and then absolute values of the differences are taken. The central characteristic value most close to the pixel characteristic value is selected, and then the pixel characteristic value is assigned to the set corresponding to the central characteristic value. Therefore, all the pixel characteristic values are assigned to their corresponding sets.
- (d) The pixel characteristic values in every set ( $S_1 \sim S_n$ ) are averaged to obtain corresponding mean characteristic values  $T_1 \sim T_n$  respectively.
- (e) The mean characteristic values  $T_i$  are compared with the central characteristic values  $a_i$  respectively (where,  $i=1 \sim n$ ). If every absolute value  $|T_i - a_i|$  is within a specific value, for example 1, then the minimal mean characteristic value  $T_1$  will replace the low threshold, and the maximal mean characteristic value  $T_n$  will replace the high threshold, such that the object for image-division is achieved. If the absolute value  $|T_i - a_i|$  is greater than the specific value, then the central characteristic values  $a_1 \sim a_n$  are replaced by the mean characteristic values  $T_1 \sim T_n$ , and the steps (c), (d), and (e) are carried out again.

In general, the image 2-division method can be carried out by providing two initial values, for example, let  $a_1$  and  $a_2$  equal 3 and 250. If the image intension requires being divided more finely, more initial values can be added between the initial values  $a_1$  and  $a_2$ . Therefore, the image-division method according to the present invention can provide suitable low threshold and high threshold according to the characteristic of the images.

## Second Embodiment

The background color of the cover of a scanner usually is not black, such that the prescanned images will be subject to interference by the background color of the cover. In addition, the extra or redundant images such as the connection lines will be scanned and appear in the preview window.

The above-mentioned problems can be solved according to the method described hereinafter.

First, a background image is obtained by scanning the closed cover without any objects in the scanner. The background image is shown in FIG. 3A, wherein the redundant images such as connection lines are indicated as numeral 30.

Next, objects D and E are scanned by the scanner, and the prescanned image is shown in FIG. 3B. The image in FIG. 3B and the background image in FIG. 3A are processed by subtracting to cancel out the interference due to redundant image and background color of the cover. The result image, which serves as a prescanned image, is shown in FIG. 3C.

There is an overlapping portion L of the object D and the redundant image (for example, connection lines).

In order to further reduce the disturbance due to the overlapping portion L, a portion of the prescanned image  $I_{dis}$  is masked out with the width of some pixel dots. Thus, the object D is divided into two parts  $D_1$  and  $D_2$ . The width of the masked image  $I_{dis}$  should be larger than that of connection lines. For high resolution, the width of the masked image can be 18 pixel dots. For low resolution, the width of the mask image can be 9 pixel dots. Then, the method described in the first embodiment is carried out to process the unmasked image  $I_{ps}$ , such that the image of objects  $D_1$  and E are cropped automatically, as shown in FIG. 3D.

Finally, the masked image  $I_{dis}$  is processed to crop the object  $D_2$ . The cropped region of the object  $D_1$  is expanded toward the masked region  $I_{dis}$  horizontally or vertically, such that the object  $D_2$  at the masked image region  $I_{dis}$  is cropped as a recovery region  $I_{rec}$ . In the second embodiment, the cropped region of the object  $D_1$  is expanded toward the masked region  $I_{dis}$  horizontally, as shown in FIG. 3E. Then, the recovery region  $I_{rec}$  is processed according to the method described in the first embodiment. Next, the object  $D_2$  can be cropped properly, thus completing the cropping of the objects D and E, as shown in FIG. 3F.

## Third Embodiment

When scanners scan films, the film usually disposed in a frame for convenience. FIG. 4A illustrates a frame for loading films, the portions H is where the film is disposed.

While scanning positive films without using a frame, the background color of prescanned image is white, and therefore a reversal (complement) operation is carried out to the pixels in the preview window. Then, the prescanned image after carrying out a reversal operation is processed according to the auto-cropping method described in first embodiment.

While scanning negative films without using a frame, the background color of the prescanned image is black, and thus it is processed according to the auto-cropping method described in first embodiment, without carrying out a reversal operation.

In a scanner for scanning films, the tube (light source) is disposed in the cover, and while scanning positive (negative) films, the positions of the frames in the preview images are black (white). Therefore, the general method described above can not be applied for scanning positive and negative films. Before scanning films, films with or without frames must be distinguished.

## Frame Detection Method

The method for detecting frames will be described in detail, with reference to FIG. 4A to FIG. 4E. The image of films disposed in a frame after scanning is shown in FIG. 4A, wherein the frame portion is black or white depending on the type of film (positive or negative).

In general, the frame can not cover the scanning window of a scanner completely, therefore, the light source will be transparent in the part of the region uncovered by the frame. This light leakage will interfere with the detection of the frame. To reduce the interference, the margin of the frame is masked out, and the remaining image 40 enclosed by a dashed line is used for frame detection, as shown in FIG. 4B. If the films are negative, the image 40 has to be processed by carrying out a reversal operation first, and the frame detection then follows. If the films are positive, a reversal operation is not required for processing the image 40, and the frame detection is carried out directly.

The steps (a) and (b) described in first embodiment are used to process the image 40 to select the rows with dot-intension numbers that exceed the limit. For the third embodiment, three regions b1, b2, and b3 are cropped, as shown in FIG. 4C. (The number of the cropped regions depends on the type of frame.)

In a frame, the portions H for disposing films have specific sizes. In the third embodiment, the size is about 3.6 cm, for example. Therefore, the pixel dot-number after mapping to the preview window can be expressed as  $pc = (3.6/2.54) \times \text{prescan\_dpi}$ , wherein  $\text{prescan\_dpi}$  is the prescanned resolution (dots per inch) of the scanner.

In FIG. 4C, the cropped regions b1, b2, and b3 have corresponding pixel dots p1, p2, and p3 along their vertical columns respectively. If at least two pixel dots conform to the following condition  $(pc - sn) \leq pi \leq (pc + sn)$ , then the frame is detected, wherein  $i=1-3$ , and sn is a specific number, here sn is 3 for example.

Auto-cropping method while a frame is detected

After detecting a frame, if the films are positive, then the prescanned image is processed by the auto-cropping method. If the films are negative, then the reversal of prescanned image must be carried out first, and then the prescanned image after reversal is processed by auto-cropping method.

The auto-cropping method applied to positive films is similar to that applied to negative films, therefore, the one applied to positive films is described hereinafter.

First, the prescanned image is processed by the auto-cropping method described in the first embodiment, and all the positions where films can be disposed are cropped, as shown in FIG. 4D.

Because not all the positions are disposed with films, an examination process is required to get rid of the cropped positions without films disposed. The process is described as follows:

The image pixels of every cropped position are compared with a high threshold, the numbers of the pixels that exceed the high threshold are recorded as a sum number. If the sum number of a cropped position is larger than the total pixel dots in a cropped position for a certain proportion, then no film is disposed at the cropped position. By this way, every cropped position can be checked sequentially, and therefore the correct position of the film can be cropped properly, as shown in FIG. 4E.

While the invention has been described in terms of what is presently considered to be three most practical and preferred embodiments, it is to be understood that the invention need not be limited to the disclosed embodiments.

On the contrary, it is intended to cover various modifications and similar arrangements included within the spirit and scope of the appended claims, the scope of which should be accorded the broadest interpretation so as to encompass all such modifications and similar structures.

What is claimed is:

1. A method of auto-cropping images for scanners comprising the steps of:

- a. providing a prescanned image wherein the pixels of said image are processed by an image-division method to obtain at least a low threshold and a high threshold;
- b. comparing the pixels in every horizontal row with said low threshold, wherein the number of the pixels that exceed said low threshold is recorded respectively to obtain a dot-intension number for every row, and the dot-intension number of every row is compared with a limit, whereby cropping the rows with a dot-intension number that exceeds said limit, and dividing the prescanned image into several divided image regions;
- c. comparing the pixels in every vertical column of every said divided image region with said low threshold, and the number of the pixels that exceed said low threshold is recorded respectively to obtain a dot-intension number for every column, then the dot-intension number of every column is compared with said limit, and the columns with dot-intension numbers that exceed said limit are cropped, and every said divided image region is further divided into several cropped regions; and
- d. iterating steps b and c to further divide the cropped regions horizontally and vertically, and stopping the iterating process when every horizontal and vertical division in the cropped regions can not form any new divided region.

2. The method as claimed in claim 1, wherein said image-division method comprises the steps of:

- a. selecting a plurality of different initial values with respect to a plurality of sets one to one, serving as the central characteristic values of said sets;
- b. calculating the color values of every pixel to obtain the pixel characteristic values corresponding to every pixel;
- c. calculating every said pixel characteristic value with said central characteristic values by subtraction and continuously taking the absolute values thereof, whereby selecting the central characteristic value closest to said pixel characteristic value, and by this way, every pixel characteristic value is assigned to the corresponding set of central characteristic values, which is closest to the pixel characteristic value;
- d. averaging the pixel characteristic values in every set respectively to obtain corresponding mean characteristic values; and
- e. comparing every said mean characteristic value in every set with the corresponding central characteristic value, wherein if every difference is within a specific range, then the minimal mean characteristic value serves as said low threshold, and the maximal mean characteristic value serves as said high threshold, and if at least one of the differences exceeds said specific range, then every central characteristic value of said sets is replaced by said mean characteristic values of said sets and the steps c, d, and e are iterated.

3. The method as claimed in claim 1, when the scanners have covers, said method further comprises the steps of:

- a. prescanning a background image and storing it, before putting any objects in a scanner;

b. detecting whether said cover is closed or not, wherein if not closed, then the steps a-d in claim 1 are carried out to auto-crop the images, and if closed, then the process comprises the following steps:

- i. calculating the background image and a prescanned image by subtraction; and
- ii. carrying out the steps a-d in claim 1 to the image after subtraction to complete the auto-cropping process.

4. The method as claimed in claim 3, wherein, after carrying out step ii, the margin of said prescanned image is masked out first and then the step ii is carried out, and after completing the cropping of the unmasked region, the cropping regions are expanded toward the masked regions horizontally or vertically to crop the images in the masked regions serving as recovery regions, and the recovery regions are processed by step ii to crop the masked image, thereby completing the cropping of all images.

5. A method of auto-cropping images for scanners appropriate for scanning films with or without frames comprising the steps of:

(I). confirming the type of the films first, and detecting frames, and if frames are detected, then step (II) is carried out;

(II). preprocessing a prescanned image according to the film type, and carrying out the following steps:

- 2a. processing the pixels of said prescanned image by the image-division method to obtain at least a low threshold and a high threshold;
- 2b. comparing the pixels in every horizontal row with said low threshold, and recording respectively the number of the pixels that exceed said low threshold to obtain a dot-intension number for every row, then comparing the dot-intension number of every row with a limit, thereby cropping the rows with dot-intension numbers that exceed said limit, and dividing the prescanned image into several divided image regions;
- 2c. comparing the pixels in every vertical column of every said divided image region with said low threshold, then respectively recording the number of the pixels that exceed said low threshold to obtain a dot-intension number for every column, then comparing the dot-intension number of every column with said limit, thereby cropping the columns with dot-intension numbers that exceed said limit, and further dividing every said divided image region into several cropped regions;
- 2d. iterating steps 2b and 2c to further divide the cropped regions horizontally and vertically, and stopping the iterating process when every horizontal and vertical division of the cropped regions can not form any new divided region; and
- 2e. comparing the pixels in every cropped region with said high threshold, and recording the number of the pixels that exceed the high threshold to serve as a sum number, and if said sum number of every cropped region is greater than the total of pixel dots in a cropped region for a certain proportion, then no film is disposed at the cropped position, whereby every cropped region can be checked sequentially to locate the correct position of said film;

wherein if in step I the frames are not detected, then the steps described steps 2a-2d in process II are carried out to crop the films regions properly.

6. The method as claimed in claim 5, wherein the frame detection step I comprises the following sub-steps:



9

- 1a. processing all pixels in a preview window to obtain a low threshold and a high threshold;
  - 1b. comparing the pixels in every horizontal row with said low threshold, and respectively recording the number of the pixels that exceed said low threshold to obtain a dot-intension number with respect to every row, then comparing the dot-intension number of every row with a limit, thereby cropping the rows with dot-intension numbers that exceed said limit, and recording the row number of every cropped regions;
  - 1c. transforming the size of the regions for disposing films into pixel dots; and
  - 1d. comparing every one of said horizontal row number recorded with said pixel dot-number of the region for disposing a film, and if the difference between at least two of said horizontal row numbers and said dot-number is within a specific range then frames are detected.
7. The method as claimed in claim 5, wherein the image-division method comprises the following steps:
- a. selecting a plurality of initial values with respect to a plurality of sets one to one to serve as the central characteristic values of said sets;
  - b. calculating the color values of every pixel to obtain the pixel characteristic values corresponding to every pixel;
  - c. calculating every said pixel characteristic value with said central characteristic values by subtraction and continuously taking the absolute values, thereby selecting the central characteristic value closest to said pixel characteristic value, whereby every pixel characteristic

10

- value is put into the corresponding set of the central characteristic value, that is closest to the pixel characteristic value;
  - d. averaging the pixel characteristic values in every set respectively to obtain corresponding mean characteristic values; and
  - e. comparing every said mean characteristic value in every set with the corresponding central characteristic value, if every difference is within a specific range, then the minimal mean characteristic value serves as said low threshold, and the maximal mean characteristic value serves as said high threshold, and if at least one of the differences exceeds said specific range, then every central characteristic value of said sets is replaced by said mean characteristic values of said sets, then the steps c, d, and e are iterated.
8. The method as claimed in claim 5, wherein, after confirming the type of films in step 1, if said films are negative, then said prescanned image is subjected to a reversal operation first, and the frame detection process is carried out successively; while if said films are positive, then the frame detection is carried out successively without a reversal operation.
9. The method as claimed in claim 5, wherein if the films are negative, then said prescanned image is subjected to a reversal operation, and the cropping process are carried out successively; while if the films are positive, then the cropping process are carried out to said prescanned image without a reversal operation.

\* \* \* \* \*



US006335985B1

(12) **United States Patent**  
**Sambonsugi et al.**

(10) **Patent No.: US 6,335,985 B1**  
(45) **Date of Patent: Jan. 1, 2002**

(54) **OBJECT EXTRACTION APPARATUS**

- (75) **Inventors:** Yoko Sambonsugi, Yamato; Toshiaki Watanabe, Yokohama; Takashi Ida, Kawasaki, all of (JP)
- (73) **Assignee:** Kabushiki Kaisha Toshiba, Kawasaki (JP)
- (\*) **Notice:** Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) **Appl. No.: 09/222,876**

(22) **Filed: Dec. 30, 1998**

(30) **Foreign Application Priority Data**

Jan. 7, 1998 (JP) ..... 10-001847  
 Jul. 7, 1998 (JP) ..... 10-192061

(51) **Int. Cl.<sup>7</sup> ..... G06K 9/46**

(52) **U.S. Cl. .... 382/190; 386/109**

(58) **Field of Search .... 382/190, 191, 382/192, 189, 193, 194, 291, 220, 258; 386/109, 111, 112**

(56) **References Cited**

**U.S. PATENT DOCUMENTS**

- 4,783,833 A \* 11/1988 Kawabata et al.
- 5,274,466 A 12/1993 Ida et al.
- 5,317,397 A \* 5/1994 Odaka et al. .... 348/416
- 5,331,436 A 7/1994 Ida et al.
- 5,650,829 A \* 7/1997 Sugimoto et al. .... 348/699
- 5,953,488 A \* 9/1999 Seto ..... 386/109
- 5,960,081 A \* 9/1999 Vynne et al. .... 380/10

**FOREIGN PATENT DOCUMENTS**

JP 8-241414 9/1996

**OTHER PUBLICATIONS**

- Roland Mech, et al. "A Noise Robust Method for Segmentation of Moving Objects in Video Sequences", International Conference on Acoustics, Speech and Signal Processing (ICASSP97), vol. 4, Apr. 1997, pp. 2657-2660.
- Naohiro Amamoto, et al. "Detecting Obstructions and Tracking Moving Objects by Image Processing Technique", IEICE Trans. On Fundamentals of Elec., Comm., and Computer Sciences (A), vol. J81-A No. 4, Apr. 1998, pp. 527-535.
- T. Echigo et al. "Region Segmentation of the Spatio-Temporal Image Sequence for the Video Mosaicing", IEICE Conference, D-12-81, Sep. 1997, p. 273.
- Takashi Ida, et al. "Self-Affine Mapping System for Object Contour Extraction", Research and Development Center, Toshiba Corporation, pp. 1-3.

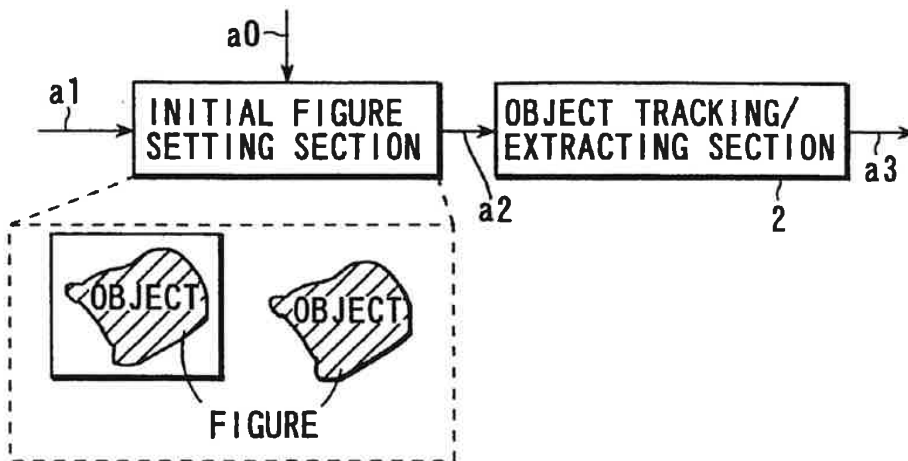
\* cited by examiner

*Primary Examiner*—Andrew W. Johns  
*Assistant Examiner*—Seyed Azarian  
 (74) *Attorney, Agent, or Firm*—Oblon, Spivak, McClelland, Maier & Neustadt, P.C.

(57) **ABSTRACT**

Rectangles  $R(i-1)$ ,  $R(i)$ , and  $R(i+1)$  are set to surround three temporally continuous frames  $f(i-1)$ ,  $f(i)$ , and  $f(i+1)$ . Difference images  $fd(i-1, i)$  and  $fd(i, i+1)$  are obtained on the basis of the inter-frame differences between the current frame  $f(i)$  and the first reference frame  $f(i-1)$  and between the current frame  $f(i)$  and the second reference frame  $f(i+1)$ . Background regions are respectively determined for polygons  $Rd(i-1, i)=R(i-1)$  or  $R(i)$  and  $Rd(i, i+1)=R(i)$  or  $R(i+1)$ , and the remaining regions are selected as object region candidates. By obtaining the intersection between these object region candidates, an object region  $O(i)$  on the current frame  $f(i)$  can be extracted.

36 Claims, 20 Drawing Sheets



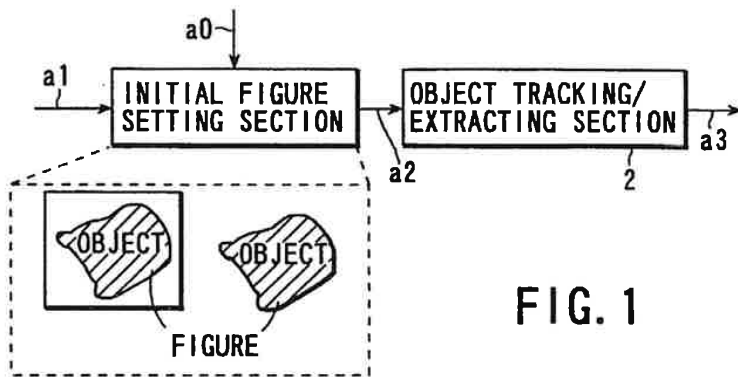


FIG. 1

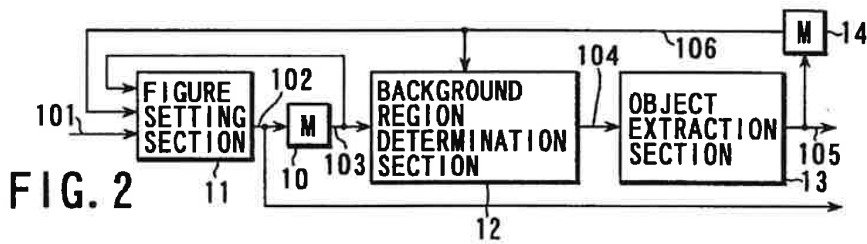


FIG. 2

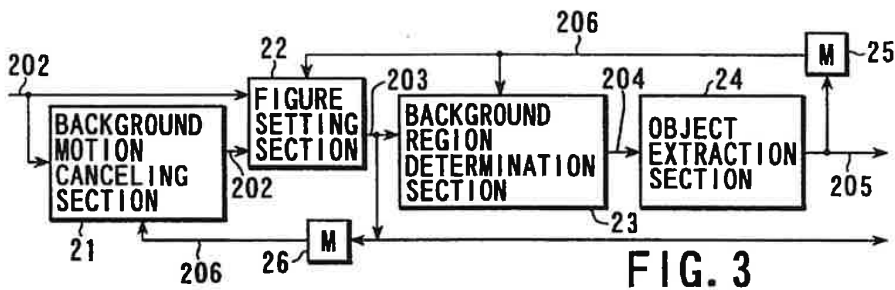


FIG. 3

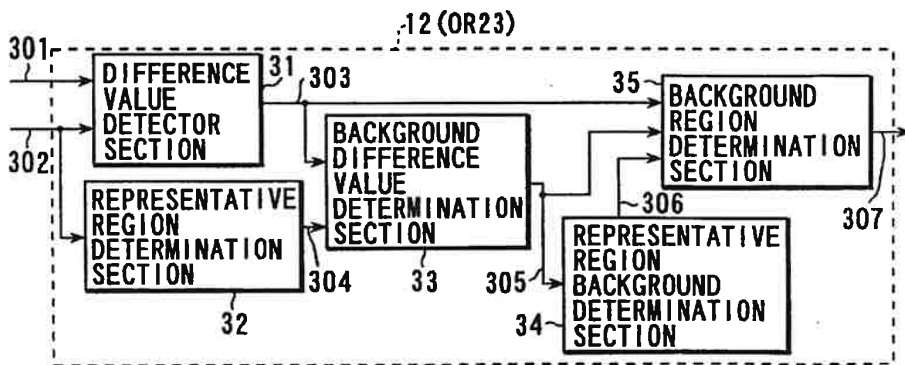


FIG. 4A

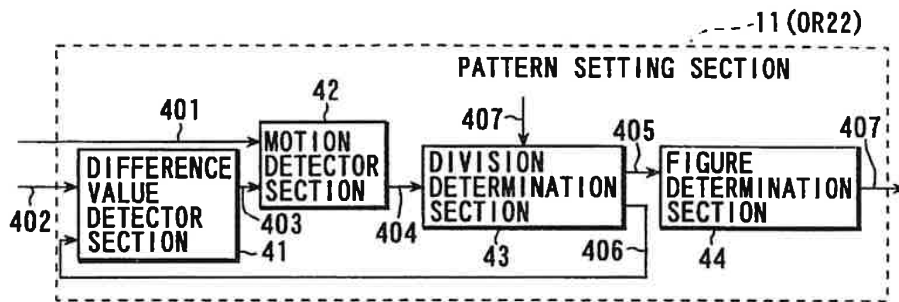
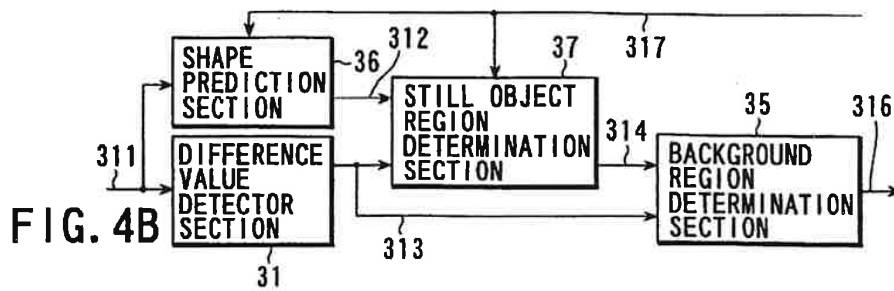


FIG. 5

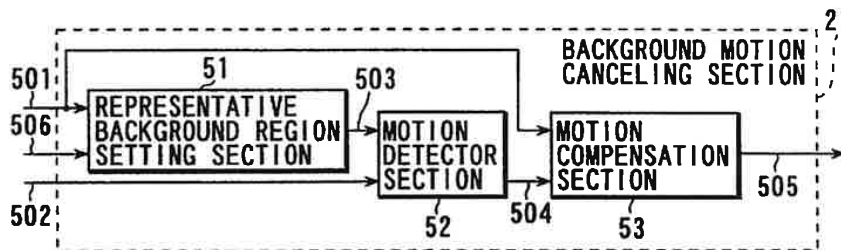


FIG. 6

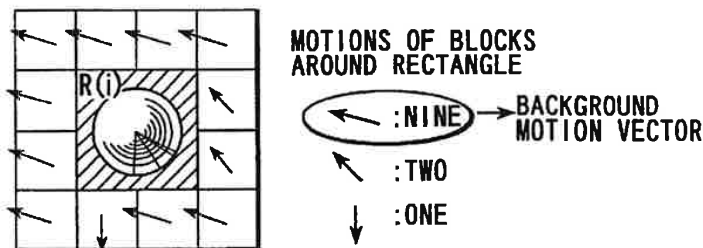


FIG. 7

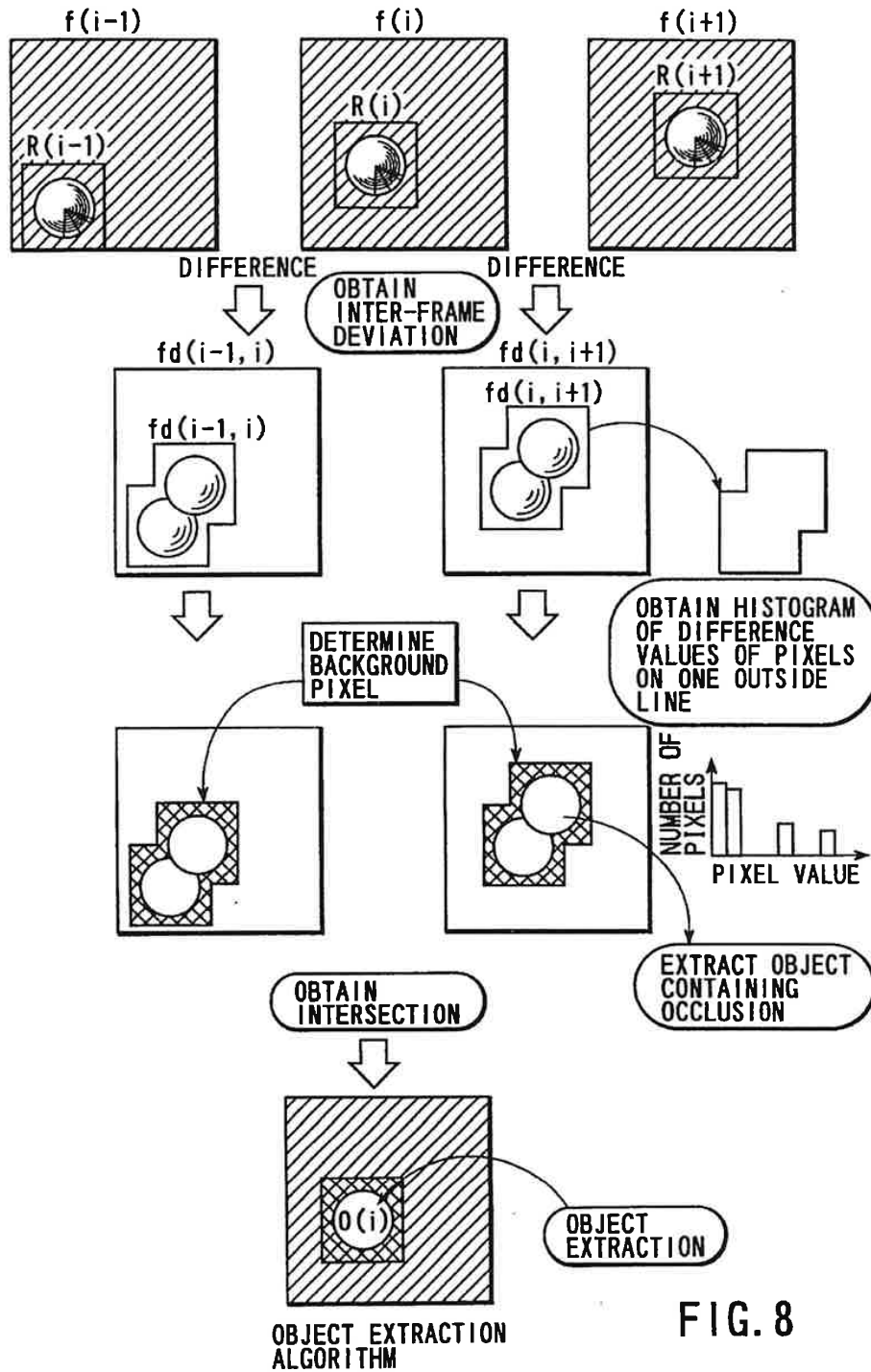


FIG. 8

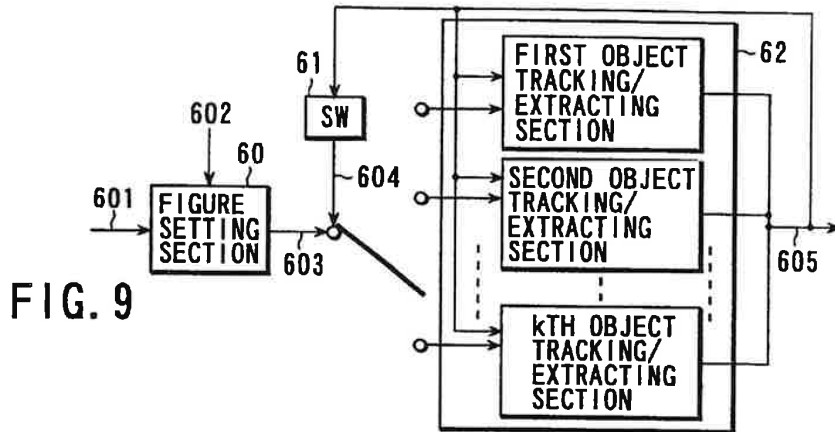


FIG. 9

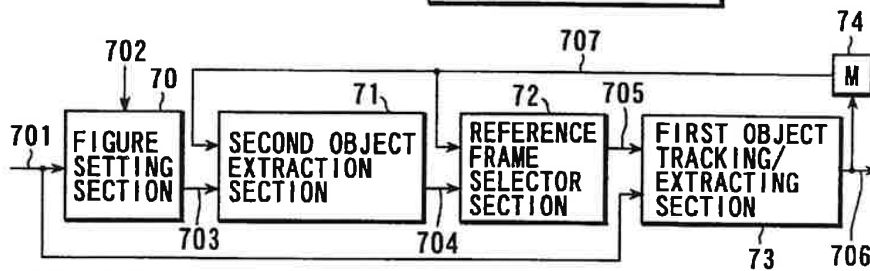


FIG. 10

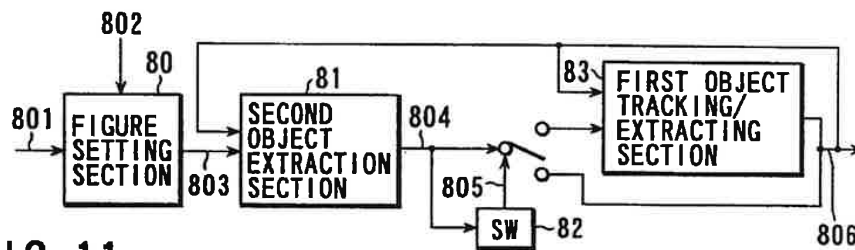


FIG. 11

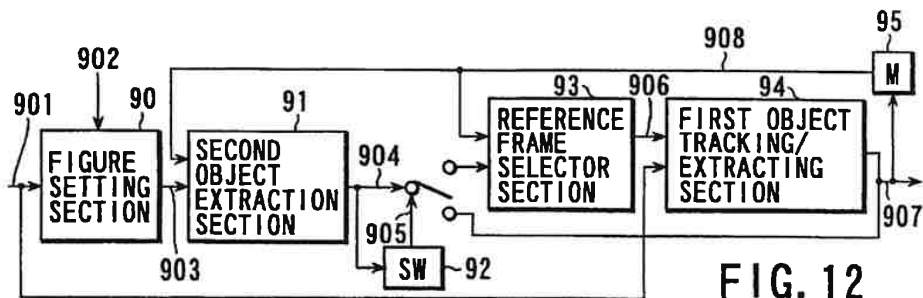


FIG. 12

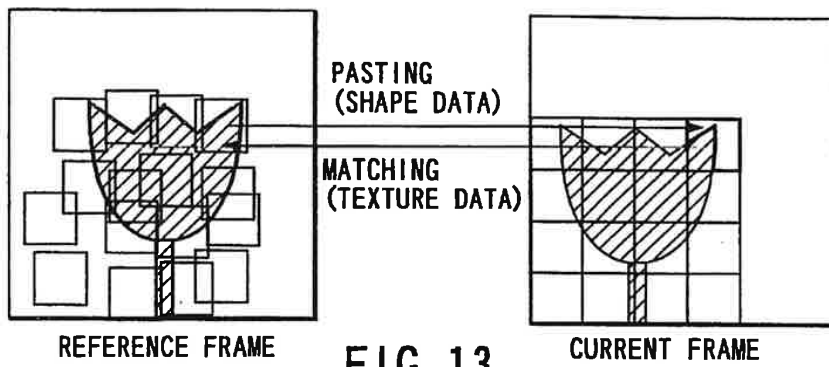


FIG. 13

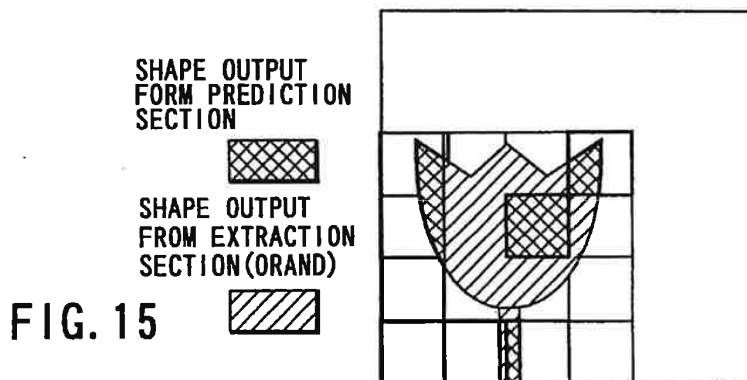
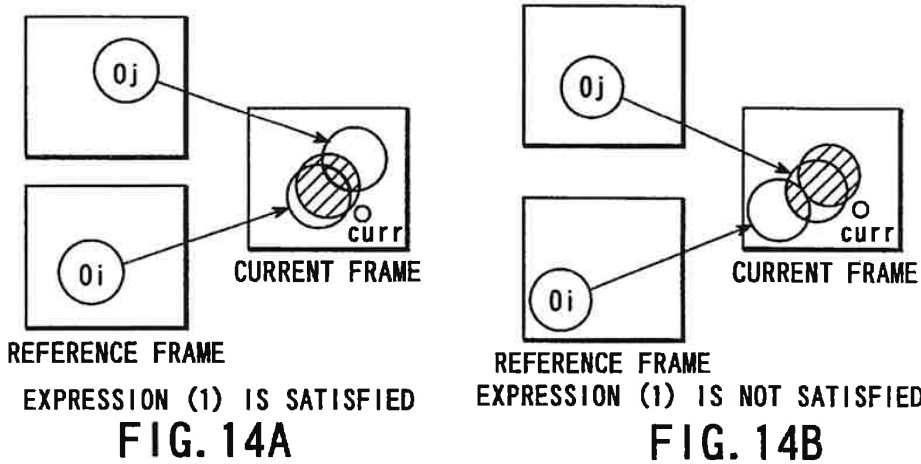


FIG. 15

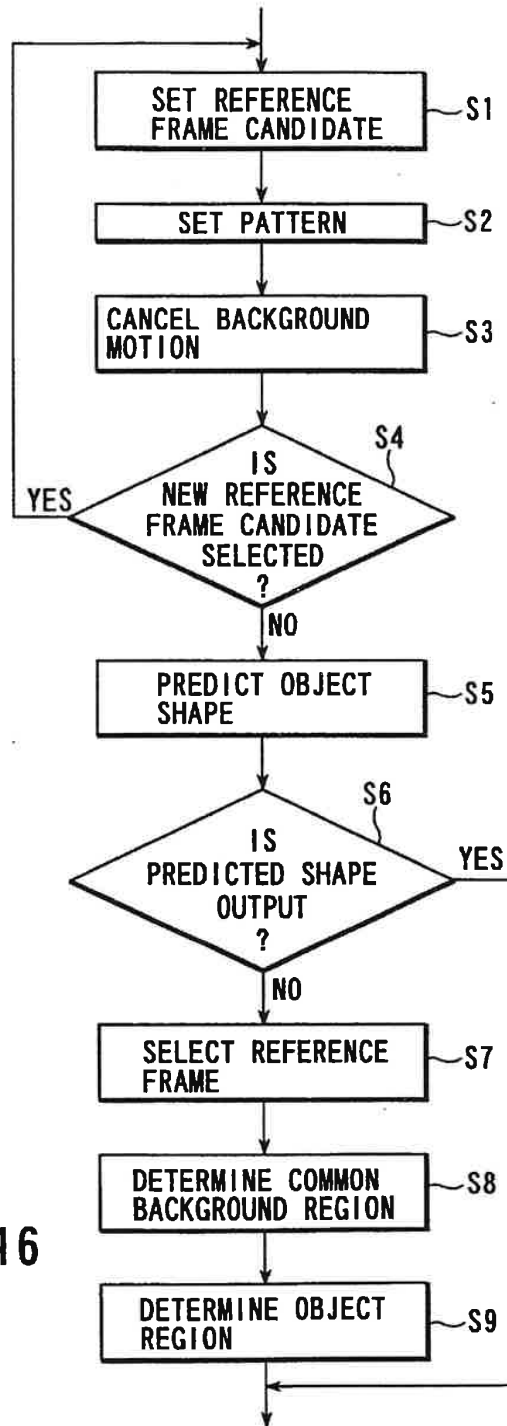


FIG. 16



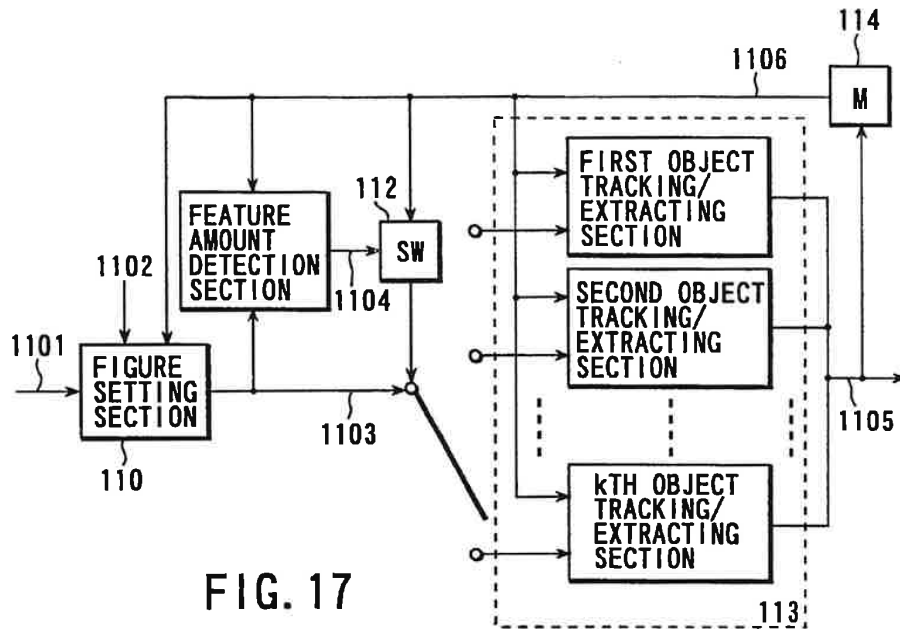


FIG. 17

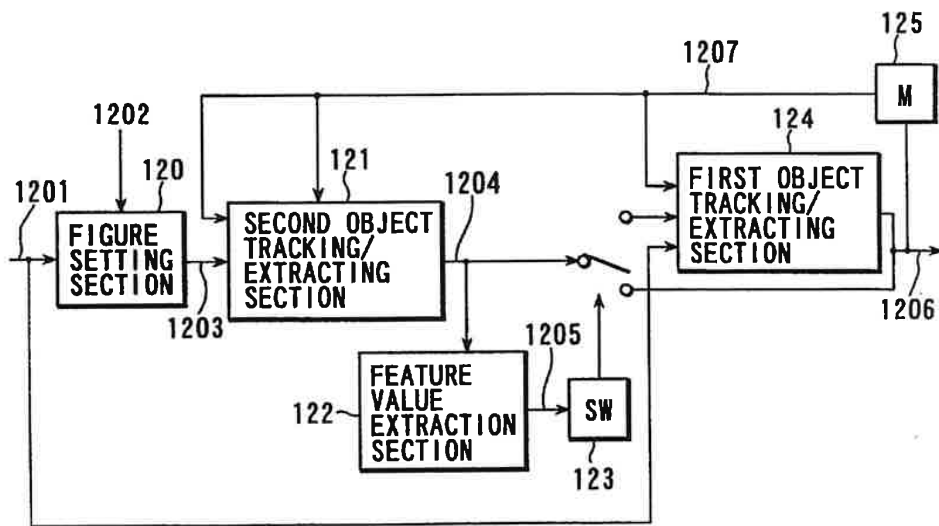


FIG. 18

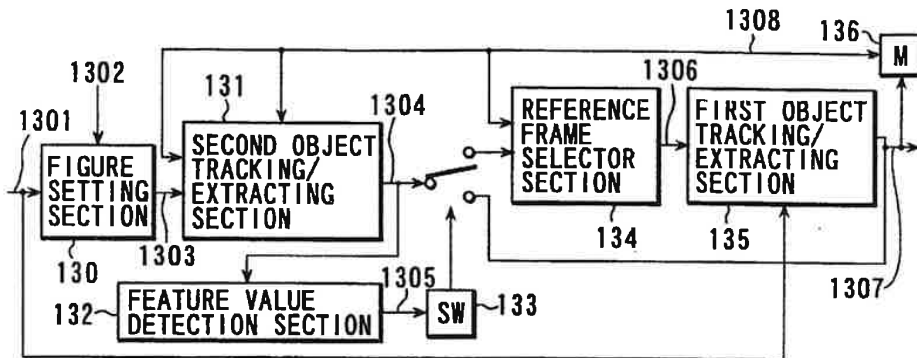


FIG. 19

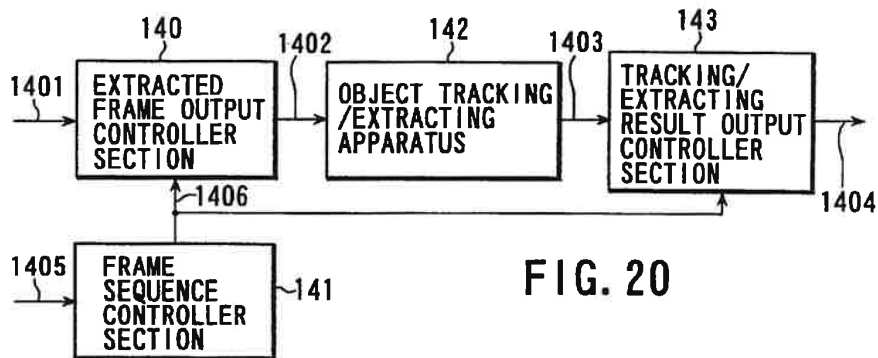


FIG. 20

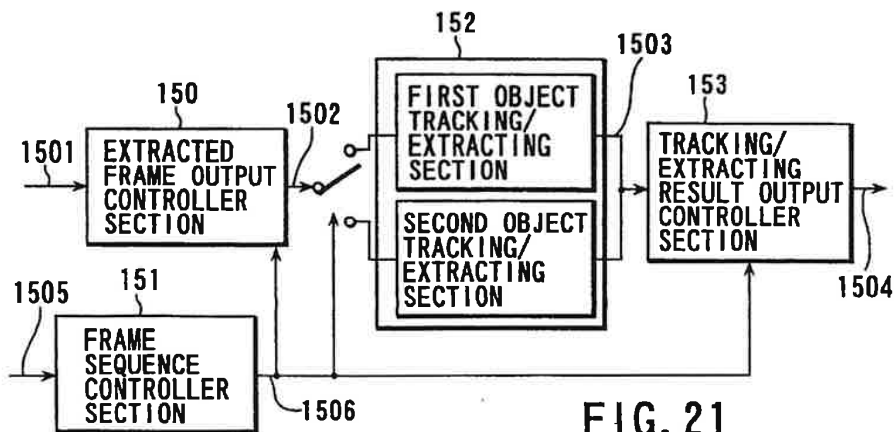


FIG. 21

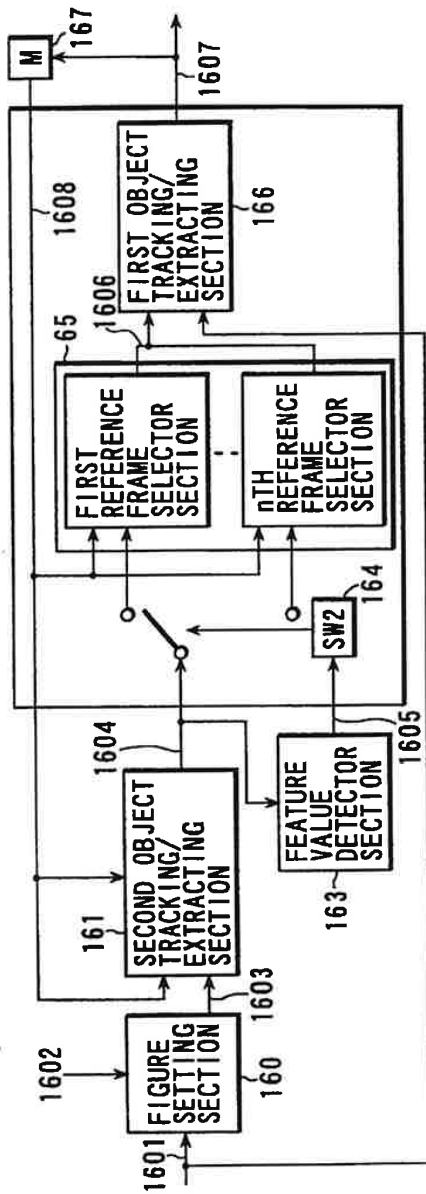


FIG. 22

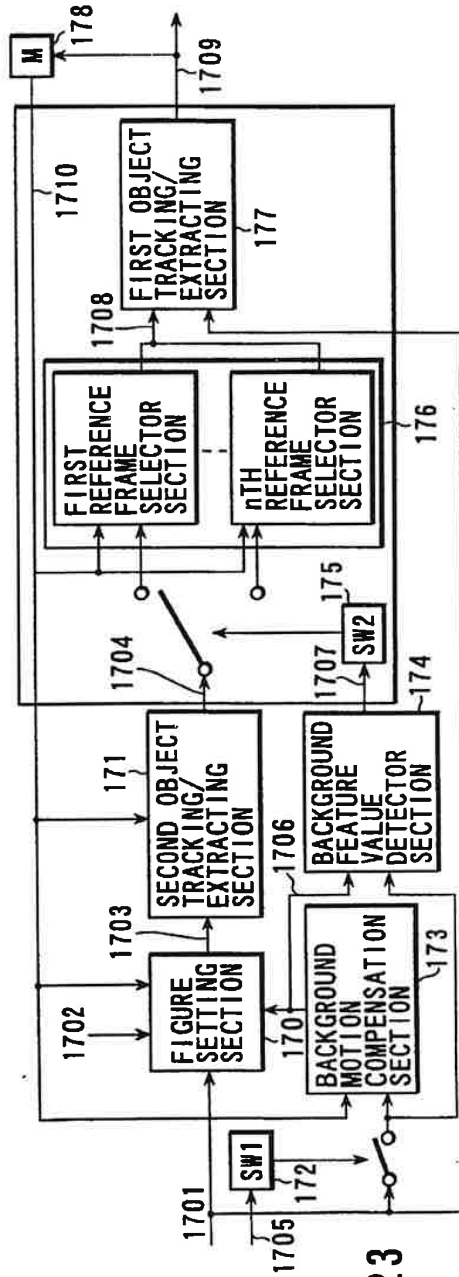


FIG. 23

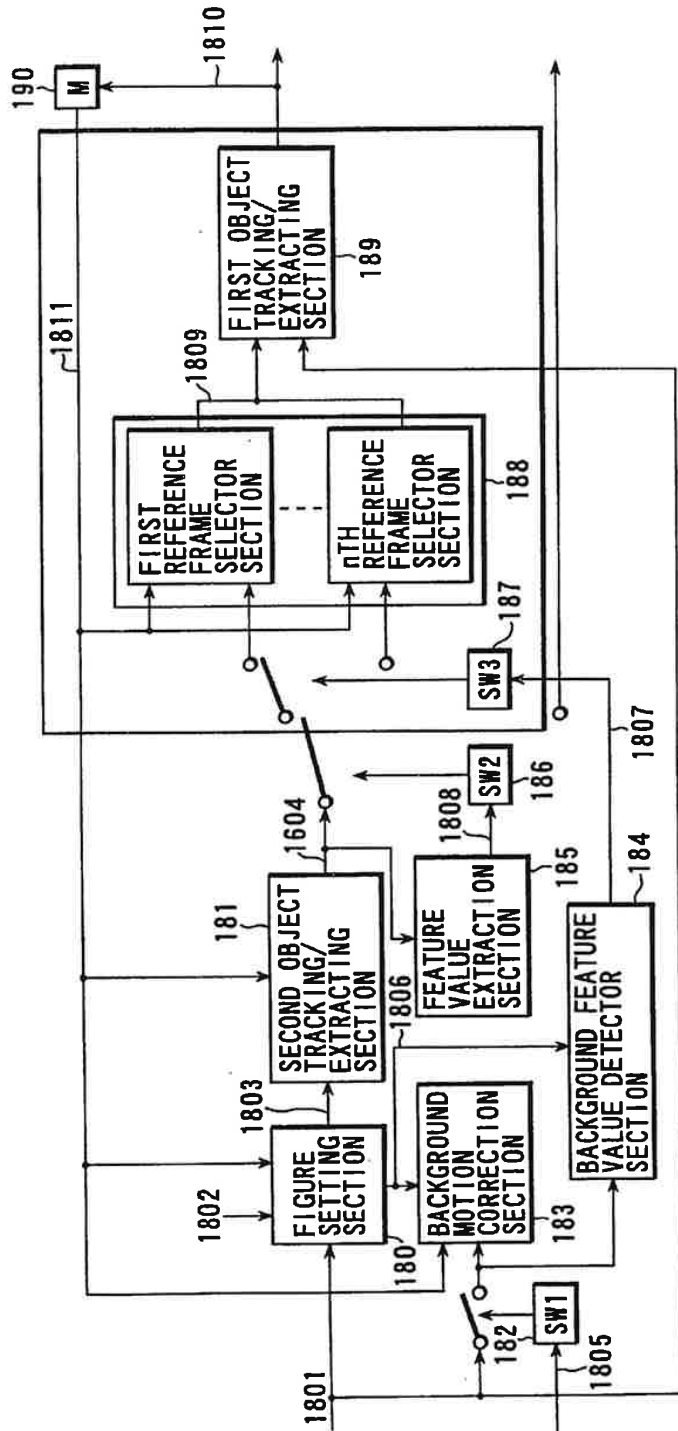
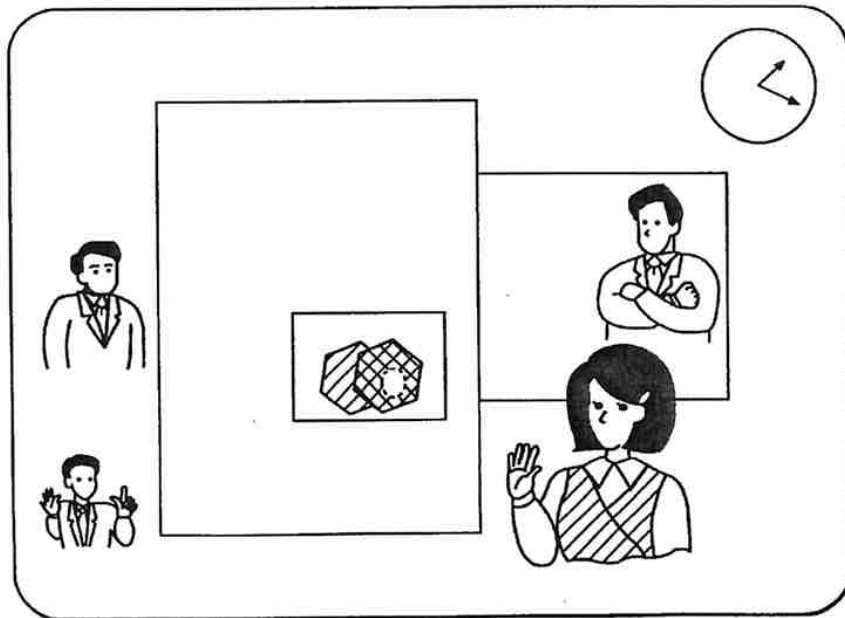
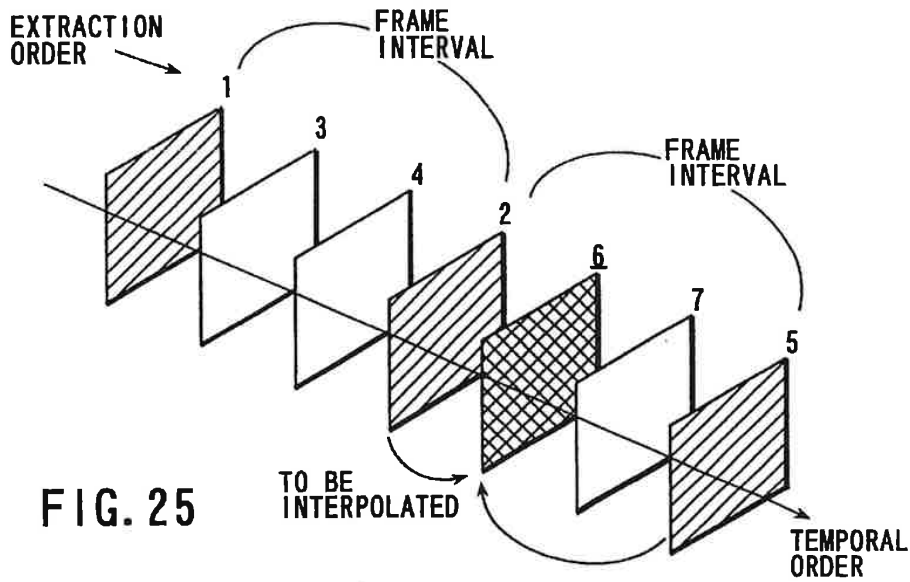


FIG. 24



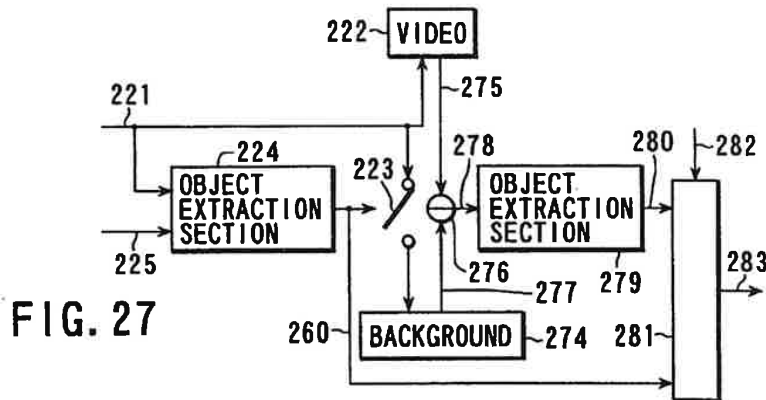


FIG. 27

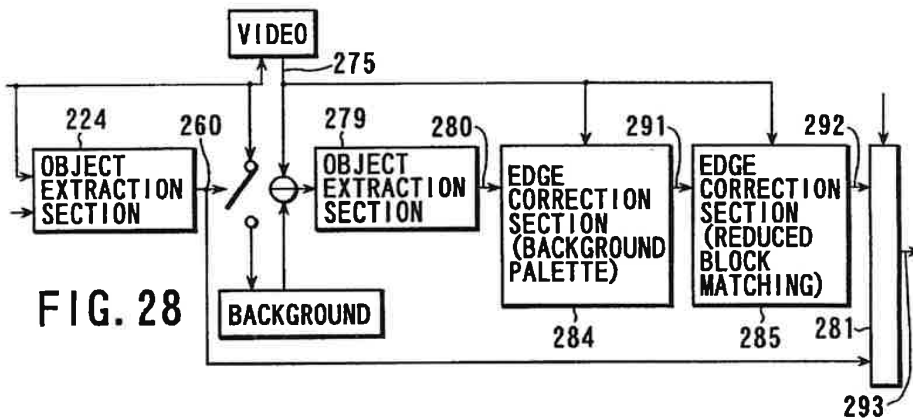


FIG. 28

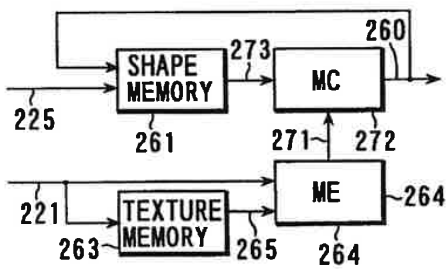


FIG. 29

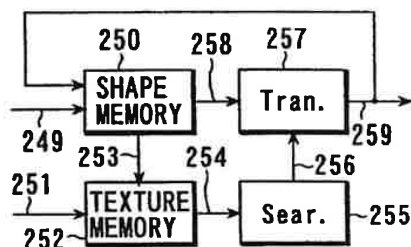


FIG. 30

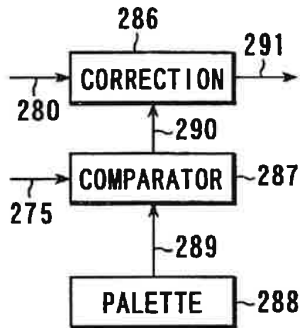


FIG. 31

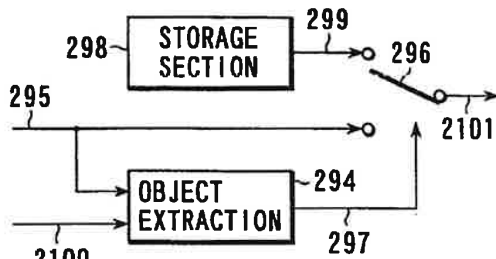


FIG. 32

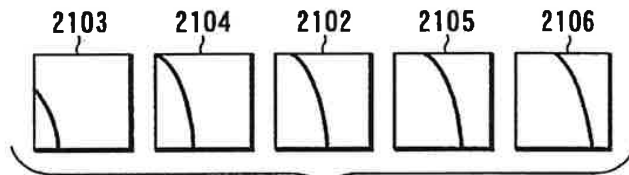


FIG. 33

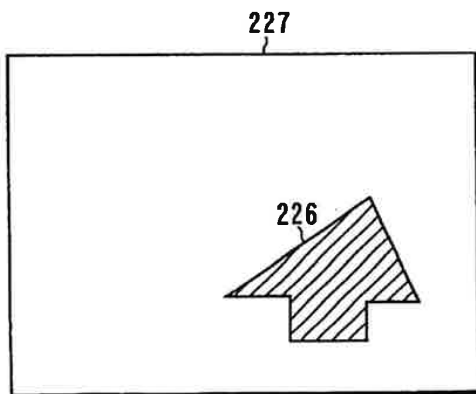


FIG. 34

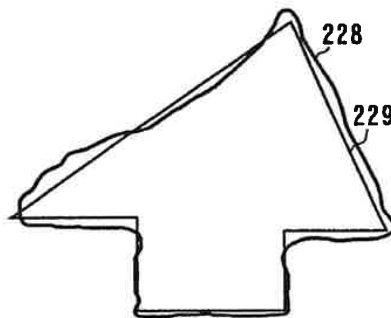


FIG. 35

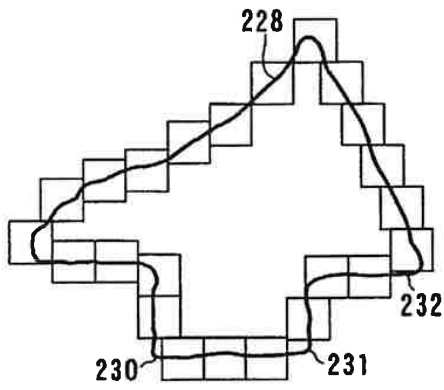


FIG. 36

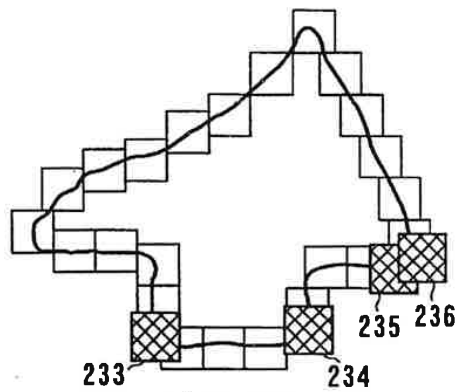


FIG. 37

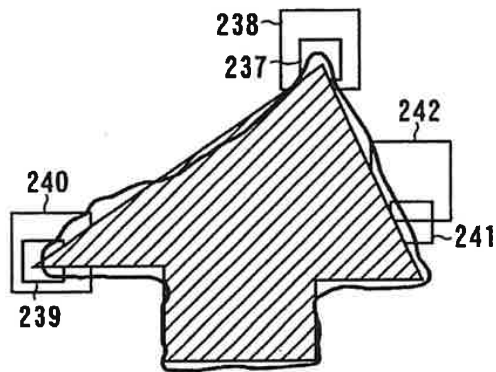


FIG. 38

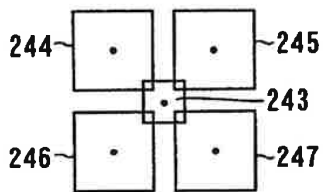


FIG. 39

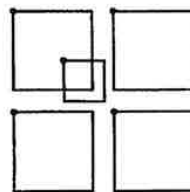


FIG. 40



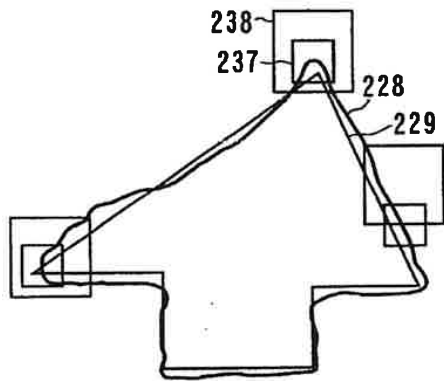


FIG. 41

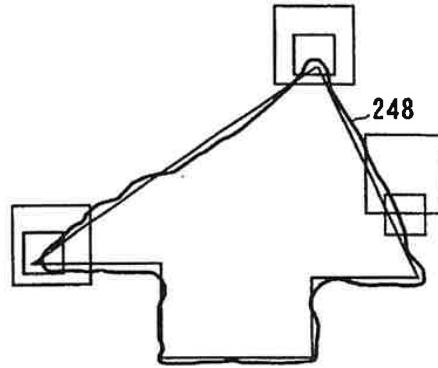


FIG. 42

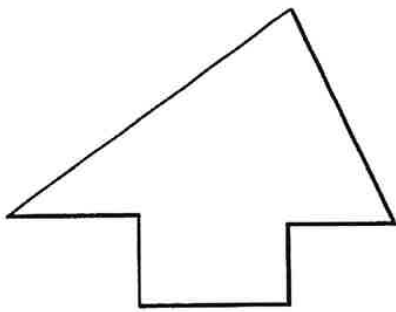


FIG. 43

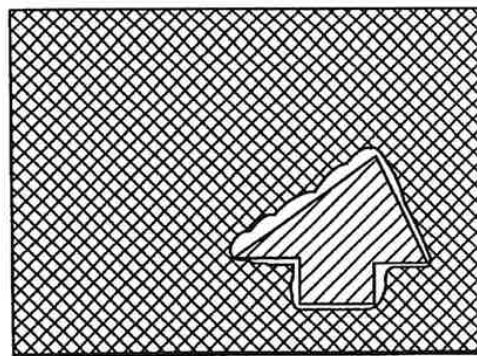


FIG. 44

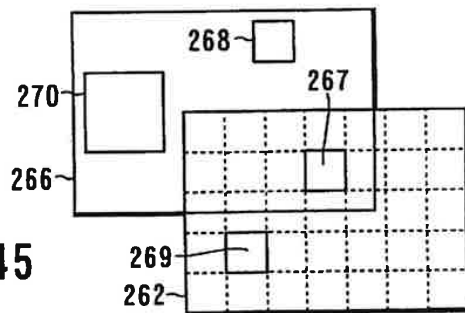


FIG. 45

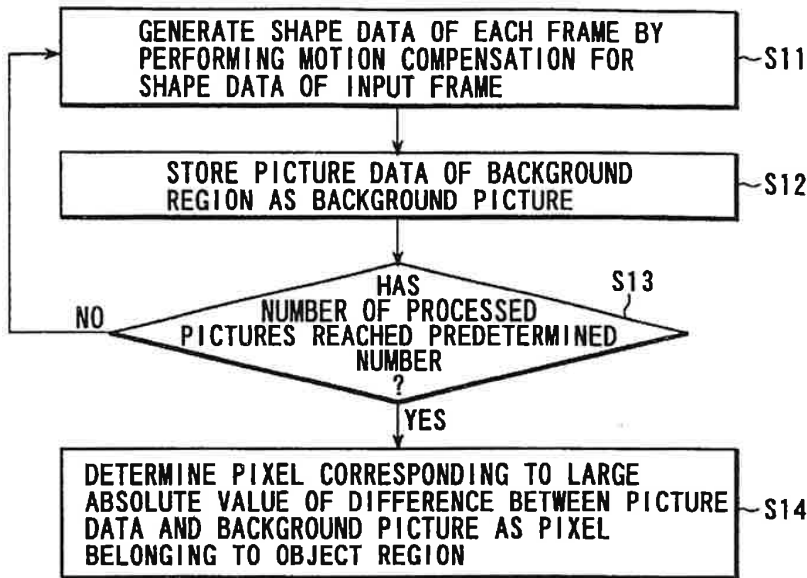


FIG. 46

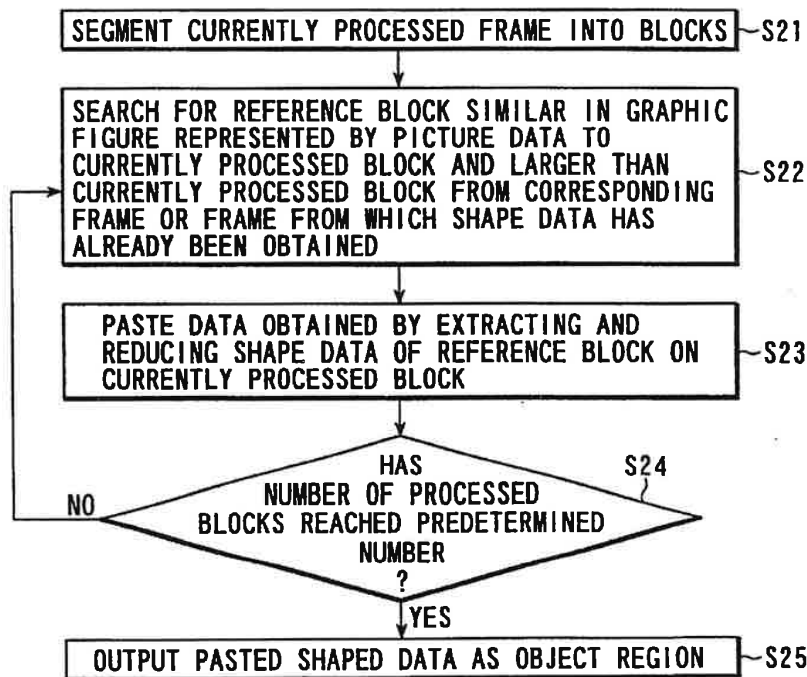


FIG. 47

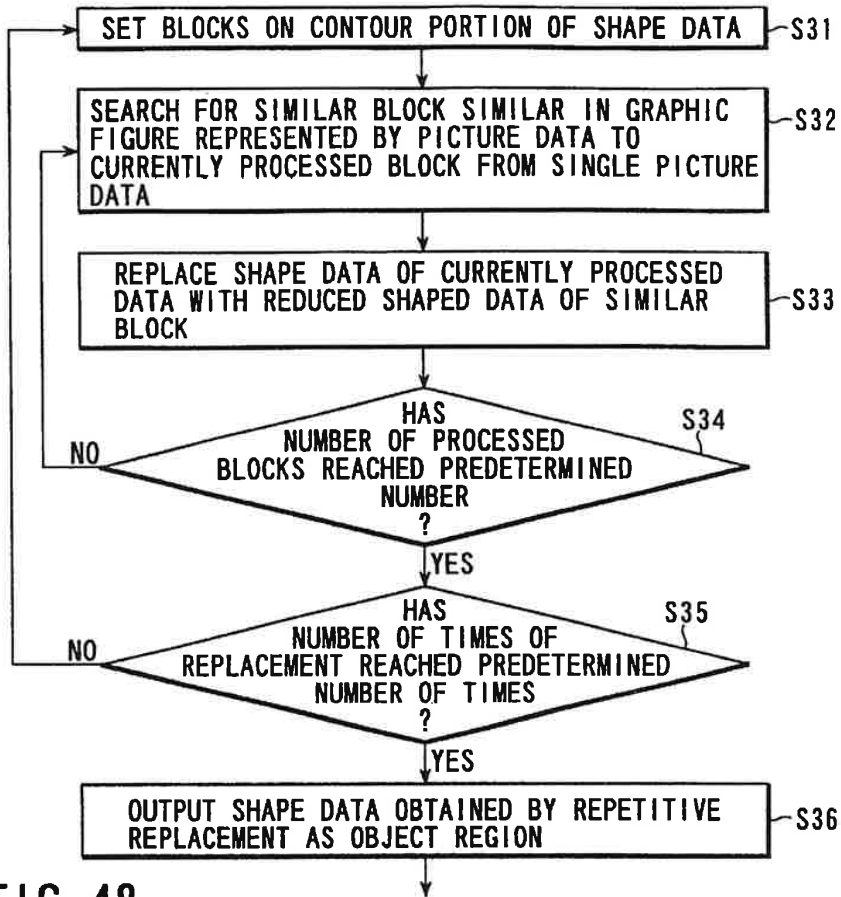


FIG. 48

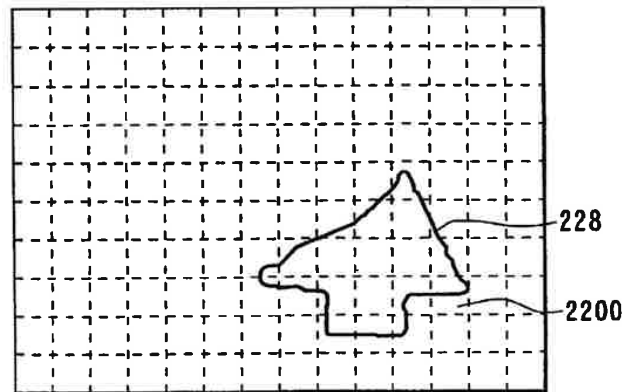


FIG. 49

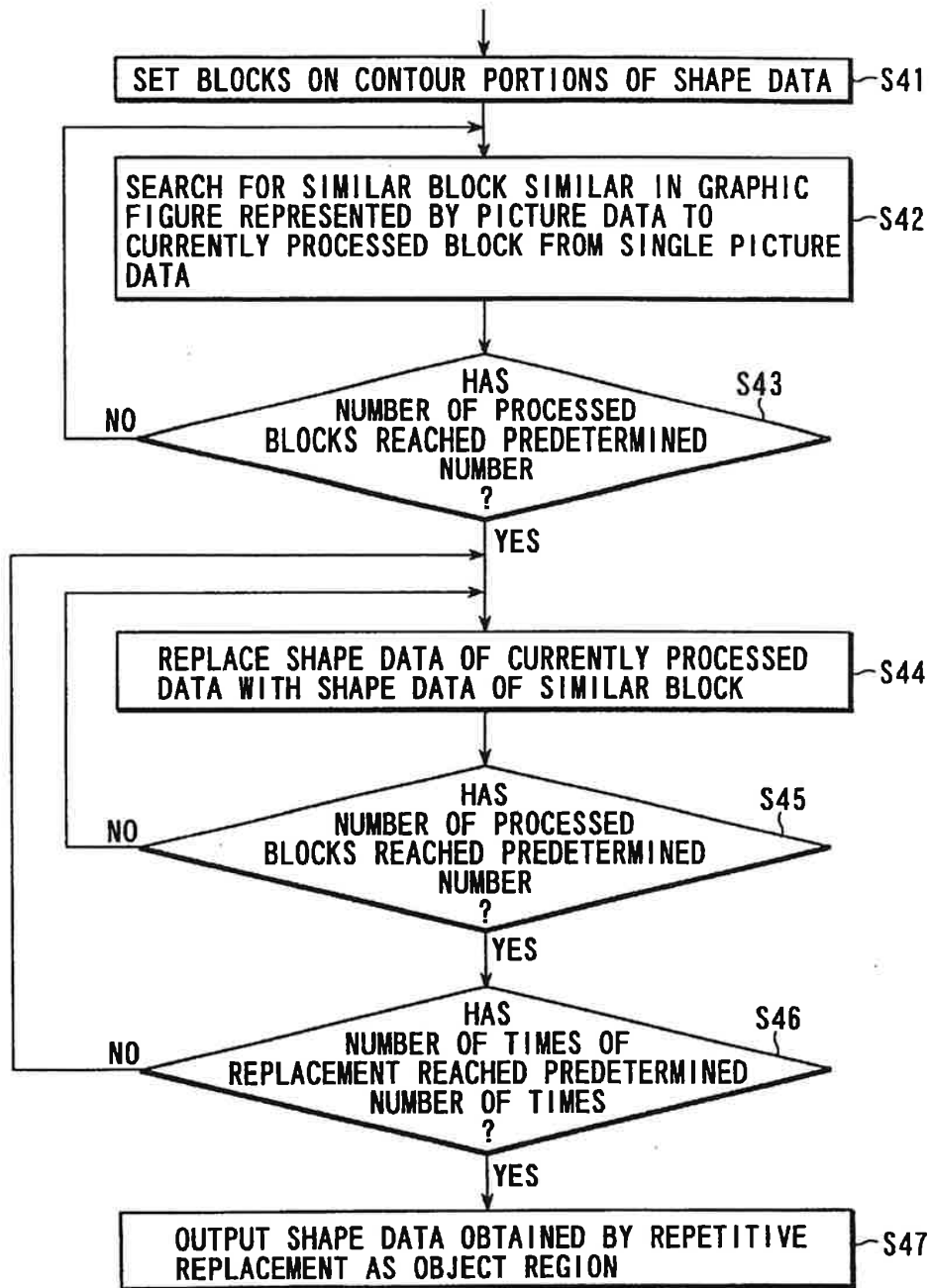
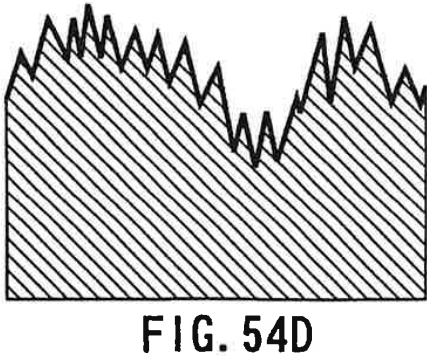
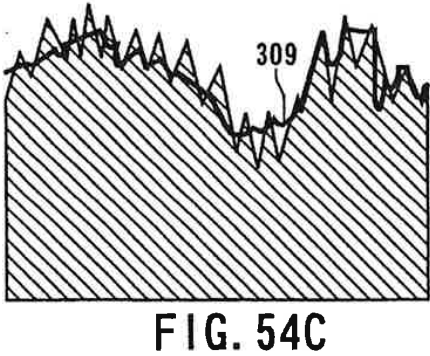
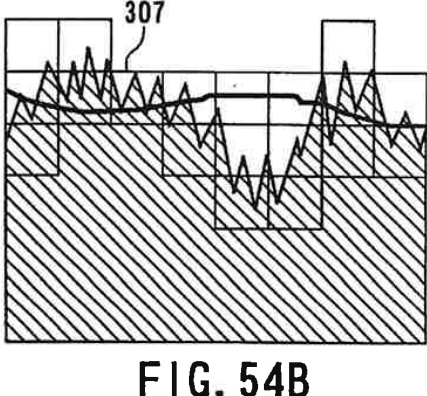
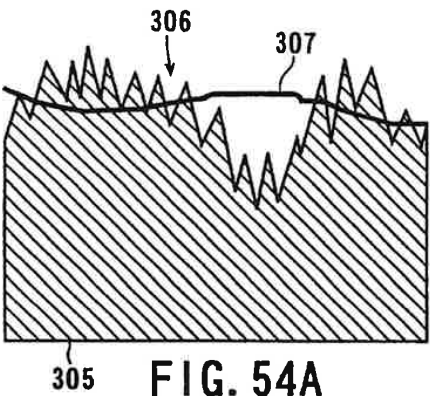
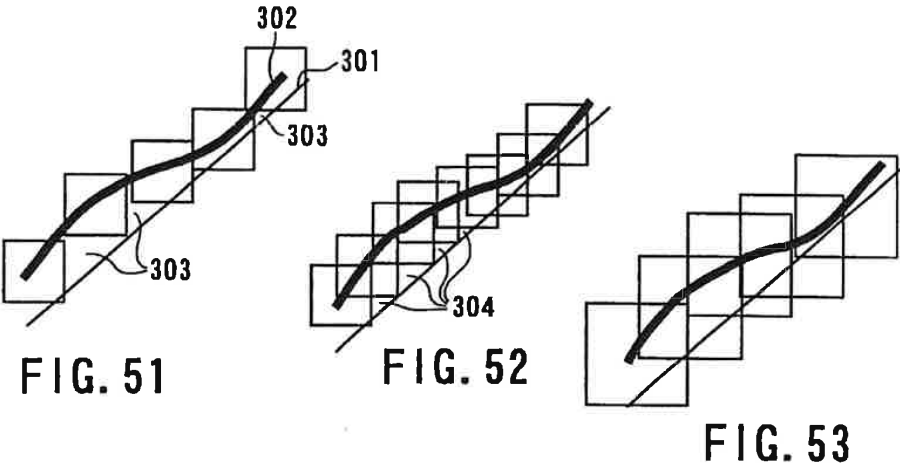


FIG. 50



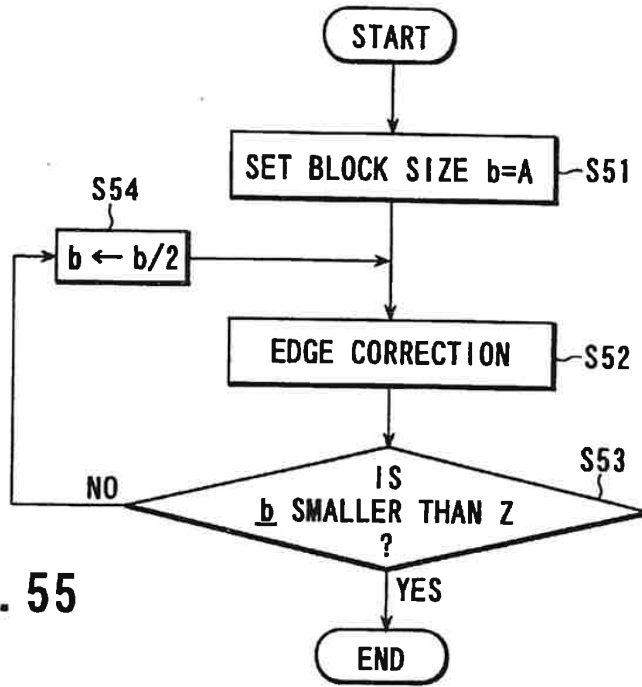


FIG. 55

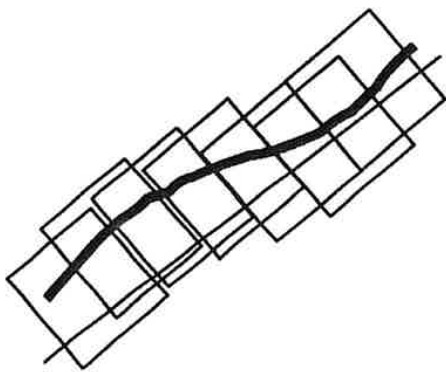


FIG. 56

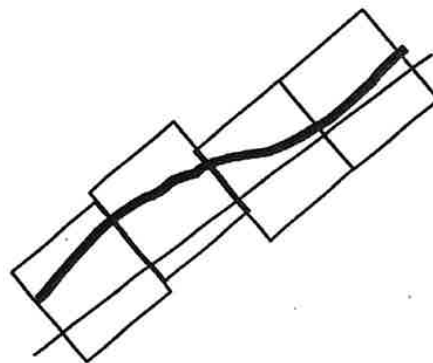


FIG. 57

## OBJECT EXTRACTION APPARATUS

## BACKGROUND OF THE INVENTION

The present invention relates to an object extraction apparatus and, more particularly, to an object extraction apparatus for detecting the position of a target object from input moving picture and tracking/extracting a moving object.

An algorithm for tracking/extracting an object in moving picture has conventionally been proposed. This is a technique of extracting only a given object from a picture including various objects and a background. This technique is useful for a process and editing of moving picture. For example, a person extracted from moving picture can be synthesized with another background.

As a method used for object extraction, the region dividing technique using region segmentation of the spatio-temporal image sequence (Echigo and Hansaku, "region segmentation of the spatio-temporal image sequence for video mosaic", THE 1997 IEICE SYSTEM SOCIETY CONFERENCE, D-12-81, p. 273, September, 1997) is known.

In this region dividing method using region segmentation of the spatio-temporal image sequence, moving picture is divided into small regions according to the color texture in one frame of the moving picture, and the regions are integrated in accordance with the relationship between the frames. When a picture in a frame is to be divided, initial division must be performed. This greatly influences the division result. In this region dividing method using region segmentation of the spatio-temporal image sequence, initial division is changed by using this phenomenon in accordance with another frame. As a result, different division results are obtained, and the contradictory divisions are integrated in accordance with the motion between frames.

If, however, this technique is applied to tracking/extracting of an object in moving picture without any change, a motion vector is influenced by an unnecessary motion other than the motion of the moving object as a target. In many cases, therefore, the reliability is not satisfactorily high, and erroneous integration occurs.

A moving object detecting/tracking apparatus using a plurality of moving object detectors is disclosed in Jpn. Pat. Appln. KOKAI Publication No. 8-241414. For example, this conventional moving object detecting/tracking apparatus is used for a monitoring system using a monitor camera. This apparatus detects a moving object from an input moving picture and tracks it. In this moving object detecting/tracking apparatus, the input moving picture is input to a picture segmenting section, an inter-frame difference type moving object detector section, a background difference type moving object detector section, and a moving object tracking section. The picture segmenting section segments the input moving picture into blocks each having a predetermined size. The division result is sent to the inter-frame difference type moving object detector section and the background difference type moving object detector section. The inter-frame difference type moving object detector section detects the moving object in the input picture by using the inter-frame difference in units of difference results. In this case, to detect the moving object without being influenced by the moving speed of the moving object, the frame intervals at which inter-frame differences are obtained are set on the basis of the detection result obtained by the background difference type moving object detector section. The background difference type moving object detector section detects

the moving object by obtaining the difference between the moving object and the background picture created by using the moving picture input so far in units of division results. An integration processor section integrates the detection results obtained by the inter-frame difference type moving object detector section and the background difference type moving object detector section to extract the motion information about the moving object. After the object is detected from each frame, the moving object tracking section makes the correction moving objects on the respective frames correspond to each other.

In this arrangement, since a moving object is detected by using not only an inter-frame difference but also a background difference, the detection precision is higher than that in a case wherein only the inter-frame difference is used. However, owing to the mechanism of detecting an object in motion from overall input moving picture by using an inter-frame difference and background difference, the detection result of the inter-frame difference and background difference are influenced by unnecessary motions other than the motion of the target moving object. For this reason, a target moving object cannot be properly extracted/tracked from a picture with a complicated background motion.

Another object extraction technique is also known, in which a background picture is created by using a plurality of frames, and a region where the difference between the pixel values of the background picture and input picture is large is extracted as an object.

An existing technique of extracting an object by using this background picture is disclosed in "MOVING OBJECT DETECTION APPARATUS, BACKGROUND EXTRACTION APPARATUS, AND UNCONTROLLED OBJECT DETECTION APPARATUS", Jpn. Pat. Appln. KOKAI Publication No. 8-55222.

According to this technique, the moving picture signal of the currently processed frame is input to a frame memory for storing one-frame picture data, a first motion detection section, a second motion detection section, and a switch. A video signal one frame ahead of the current frame is read out from the frame memory and input to the first motion detection section. The background video signals generated up to this time are read out from the frame memory prepared to hold background pictures and is input to the second motion detection section and the switch. Each of the first and second motion detection section extracts an object region by using, for example, the difference value between the two input video signals. Each extraction result is sent to a logical operation circuit. The logical operation circuit calculates the AND of the two input video data, and outputs it as a final object region. The object region is also sent to the switch. The switch selects signals depending on an object region as follows. For a pixel belonging to the object region, the switch selects a background pixel signal. In contrast to this, for a pixel that does not belong to the object region, the switch selects the video signal on the currently processed frame, and the signal is sent as an overwrite signal to the frame memory. As a result, the corresponding pixel value in the frame memory is overwritten.

According to this technique, as disclosed in Jpn. Pat. Appln. KOKAI Publication No. 8-55222, as the processing proceeds, more accurate background pictures can be obtained. At the end, the object is properly extracted. However, since the background picture is mixed in the object in the initial part of the moving picture sequence, the object extraction precision is low. In addition, if the motion of the object is small, the object picture permanently remains in the background picture, and the extraction precision remains low.

3

As described above, in the conventional object extraction/tracking method, owing to the mechanism of detecting an object in motion from the overall input moving picture, the detection result of the inter-frame difference and background difference are influenced by unnecessary motions other than the motion of the target moving object. For this reason, a target moving object cannot be properly extracted/tracked.

In the object extraction method using background pictures, the extraction precision is poor in the initial part of a moving picture sequence. In addition, if the motion of the object is small, since a background picture remains incomplete, the extraction precision remains low.

#### BRIEF SUMMARY OF THE INVENTION

It is an object of the present invention to provide an object extraction apparatus for moving picture which can accurately extract/track a target object without being influenced by unnecessary motions around the object.

It is another object to provide an object extraction apparatus which can accurately determine a background picture and obtain a high extraction precision not only in the late period of a moving picture sequence but also in the early period of the moving picture sequence regardless of the magnitude of the motion of an object.

According to the present invention, there is provided an object extraction apparatus comprising a background region determination section for determining a first background region common to a current frame as an object extraction target and a first reference frame that temporally differs from the current frame on the basis of a difference between the current frame and the first reference frame, and determining a second background region common to the current frame and a second reference frame that temporally differs from the current frame on the basis of a difference between the current frame and the second reference frame, an extraction section for extracting a region, in a picture on the current frame, which belongs to neither the first background region nor the second background region as an object region, and a still object detection section for detecting a still object region.

In this object extraction apparatus, two reference frames are prepared for each current frame as an object extraction target, and the first common background region commonly used for the current frame and the first reference frame is determined on the basis of the first difference image between the current frame and the first reference frame. The second common background region commonly used for the current frame and the second reference frame is determined on the basis of the second difference image between the current frame and the second reference frame. Since the object region on the current frame is commonly included in both the first and second difference images, the object region on the current frame can be extracted by detecting a region, of the regions that belong to neither the first common background region nor the second common background region, which is included in the image inside figure of the current frame. If this object region corresponds to a still object, a still object region is detected when there is no difference between the preceding object region and the current object region.

In this manner, a region that does not belong to any of the plurality of common background regions determined on the basis of the temporally different reference frames is determined as an extraction target object to track the object. This allows accurate extraction/tracking of the target object without any influences of unnecessary motions around the target object.

4

It is preferable that this apparatus further comprise a background correction section for correcting motion of a background on the reference frame or the current frame such that the motion of the background between each of the first and second reference frames and the current frame becomes relatively zero. With this background correction section set on the input stage of the figure setting section or background region determination section, even if background moving picture gradually changes between continuous frames as in a case wherein, for example, a camera is panned, the pseudo background moving picture can be made constant between these frames. Therefore, when the difference between the current frame and the first or second reference frame is obtained, the backgrounds of these frames can be canceled out. This allows common background region detection processing and object region extraction processing without any influences of changes in background. The background correction section can be realized by motion compensation processing.

In addition, the background region determination section preferably comprises a detector section for detecting difference values between the respective pixels, in a difference image between the current frame and the first or second reference frame, which are located near a contour of a region belonging to the image inside figure on the current frame or the image inside figure on the first or second reference frame, and a determination section for determining a difference value for determination of the common background region by using the difference values between the respective pixels near the contour, and determines the common background region from the difference image by using the determined difference value as a threshold value for background/object region determination. By paying attention to the difference values between the respective pixels near the contour in this manner, a threshold value can be easily determined without checking the entire difference image.

The figure setting section preferably comprises a segment section for segmenting the image inside figure of the reference frame into a plurality of blocks, a search section for searching for a region on the input frame in which an error between each of the plurality of blocks and the input frame becomes a minimum, and a setting section for setting figures surrounding a plurality of regions searched out on the input frame. With this arrangement, an optimal new figure for an input frame as a target can be set regardless of the shape or size of the initially set figure.

The present invention further comprises a prediction section for predicting a position or shape of the object on the current frame from a frame from which an object region has already been extracted, and a selector section for selecting the first and second reference frames to be used by the background region determination section on the basis of the position or shape of the object on the current frame which is predicted by the prediction section.

By selecting proper frames as reference frames to be used in this manner, a good extraction result can always be obtained.

Letting  $O_i$ ,  $O_j$ , and  $O_{curr}$  be objects on reference frames  $f_i$  and  $f_j$  and a current frame  $f_{curr}$  as an extraction target, optimal reference frames  $f_i$  and  $f_j$  for the proper extraction of the shape of the object are frames that satisfy

$$(O_i \cap O_j) \subseteq O_{curr}$$

That is, frames  $f_i$  and  $f_j$  whose objects  $O_i$  and  $O_j$  have an intersection belonging to the object  $O_{curr}$ .

In addition, the present invention is characterized in that a plurality of object extraction sections for performing



object extraction by different methods are prepared, and object extraction is performed while these object extraction sections are selectively switched. This apparatus preferably uses a combination of first object extraction sections for performing object extraction by using the deviations between the current frame and at least two reference frames that temporally differ from the current frame and second object extraction sections for performing object extraction by predicting an object region on the current frame from a frame having undergone object extraction using inter-frame prediction. With this arrangement, even if the object is partially still, and no difference between the current frame and each reference frame can be detected, compensation for this situation can be made by the object extraction section using inter-frame prediction.

When a plurality of object extraction sections are prepared, it is preferable that this apparatus further comprise an extraction section for extracting a feature value of a picture in at least a partial region of the current frame as the object extraction target from the current frame, and switch the plurality of object extraction sections on the basis of the extracted feature value.

If, for example, it is known in advance whether a background moves or not, the corresponding property is preferably used. If there is a background motion, background motion compensation is performed. However, perfect compensation is not always ensured. Almost no compensation may be given for a frame exhibiting a complicated motion. Such a frame can be detected in advance in accordance with the compensation error amount in background motion compensation, and hence can be excluded from reference frame candidates. If, however, there is no background motion, this processing is not required. This is because if another object moves, wrong background motion compensation may be performed, or even an optimal frame for reference frame selection conditions may be excluded from reference frame candidates, resulting in a decrease in extraction precision. In addition, one picture may include various properties. The object motions and textures partly differ. For these reasons, the object may not be properly extracted by using the same tracking/extracting method and apparatus and the same parameter. It is therefore preferable that the user designate a portion of a picture which has a special property, or a difference in a picture be automatically detected as a feature value, and tracking/extracting methods be partly switched in units of, e.g., blocks in each frame to perform object extraction or the parameter be changed on the basis of the feature value.

If a plurality of object extraction sections are switched on the basis of the feature value of a picture in this manner, the shapes of objects in various pictures can be accurately extracted.

Assume that the first object extraction section using the deviations between the current frame and at least two reference frames that temporally differ from the current frame and the second object extraction section using inter-frame prediction are used in combination. In this case, the first and second object extraction sections are selectively switched and used on the basis of the prediction error amount in units of blocks in each frame as follows. When the prediction error caused by the second object extraction section falls within a predetermined range, the extraction result obtained by the second object extraction section is used as an object region. When the prediction error exceeds the predetermined range, the extraction result obtained by the first object extraction section is used as an object region.

The second object extraction section is characterized by performing inter-frame prediction in a sequence different

from an input frame sequence such that a frame interval between a reference frame and the current frame as the object extraction target is set to a predetermined number of frames or more. With this operation, since the motion amount between frames increases as compared with a case wherein inter-frame prediction is sequentially performed in the input frame sequence, the prediction precision can be increased, resulting in an increase in extraction precision.

In some cases, an object motion is too small or complicated to be coped with by the shape prediction technique using inter-frame prediction depending on the frame intervals. If, for example, a shape prediction error exceeds a threshold value, the prediction precision can be increased by increasing the interval between a target frame and the extracted frame used for prediction. This leads to an increase in extraction precision. In addition, if there is a background motion, reference frame candidates are used to obtain the background motion relative to the extracted frame to perform motion compensation. However, the background motion may be excessively small or complicated depending on the frame intervals, and hence background motion compensation may not be performed with high precision. In this case as well, the motion compensation precision can be increased by increasing the frame intervals. If the sequence of extracted frames is adaptively controlled in this manner, the shape of an object can be extracted more reliably.

In addition, according to the present invention, there is provided an object extraction apparatus for receiving moving picture data and shape data representing an object region on a predetermined frame of a plurality of frames constituting the moving picture data, comprising a readout section for reading out moving picture data from a storage unit in which the moving picture data is stored, and performing motion compensation for the shape data, thereby generating shape data in units of frames constituting the readout moving picture data, a generator section for generating a background picture of the moving picture data by sequentially overwriting picture data in a background region of each frame, determined by the generated shape data, on a background memory, and a readout section for reading out the moving picture data again from the storage unit on which the moving picture data is recorded, obtaining a difference between each pixel of each frame constituting the readout moving picture data and a corresponding pixel of the background picture stored in the background memory, and determining a pixel exhibiting a difference whose absolute value is larger than a predetermined threshold value as a pixel belonging to the object region.

In this object extraction apparatus, in the first scanning processing of reading out the moving picture data from the storage unit, a background picture is generated in the background memory. The second scanning processing is then performed to extract an object region by using the background picture completed by the first scanning. Since the moving picture data is stored in the storage unit, an object region can be extracted with a sufficiently high precision from the start of the moving picture sequence by scanning the moving picture data twice.

The present invention further comprises an output section for selectively outputting one of an object region determined by shape data of each of the frames and an object region determined on the basis of an absolute value of a difference from the background picture as an object extraction result. Depending on the picture, the object region determined by the shape data obtained by the first scanning is higher in extraction precision than the object region obtained by the second scanning using the difference from the background

picture. The extraction precision can therefore be further increased by selectively outputting the object region obtained by the first scanning and the object region obtained by the second scanning.

Furthermore, according to the present invention, there is provided an object extraction apparatus for receiving moving picture data and shape data representing an object region on a predetermined frame of a plurality of frames constituting the moving picture data, and sequentially obtaining shape data of the respective frames by using frames for which shape data have already been provided or from which shape data have already been obtained as reference frames, comprising a division section for segmenting a currently processed frame into a plurality of blocks, a search section for searching for a similar block, for each of the blocks, which is similar in figure represented by picture data to the currently processed block and is larger in area than the currently processed block, from the reference frame, a paste section for pasting shape data obtained by extracting and reducing shape data of each similar block from the reference frame on each block of the currently processed frame, and an output section for outputting the pasted shaped data as shape data of the currently processed frame.

This object extraction apparatus performs search processing in units of blocks in the current frame as an object extraction target to search for a similar block that is similar in graphic figure represented by picture data (texture) to the currently processed block and larger in area than the currently processed block. The apparatus also pastes the data obtained by extracting and reducing the shape data of each similar block searched out on the corresponding block of the currently processed frame. Even if the contour of an object region, given by shape data, deviates, the position of the contour can be corrected by reducing and pasting the shape data of each similar block larger than the currently processed block in this manner. If, therefore, the data obtained when the user approximately traces the contour of an object region on the first frame with a mouse or the like is input as shape data, object regions can be accurately extracted from all the subsequent input frames.

Moreover, according to the present invention, there is provided an object extraction apparatus for receiving picture data and shape data representing an object region on the picture, and extracting the object region from the picture data by using the shape data, comprising a setting section for setting blocks on a contour portion of the shape data, and searching for a similar block, for each of the blocks, which is similar in graphic figure represented by the picture data to each block and is larger than the block, from the same picture, a replace section for replacing the shape data of each of the blocks with shape data obtained by reducing the shape data of each of the similar blocks, a repeat section for repeating the replacement by a predetermined number of times, and an output section for outputting shape data obtained by repeating the replacement as corrected shape data.

The position of the contour provided by shape data can be corrected by performing replacement processing using similar blocks based on block matching within a frame. In addition, since the block matching is performed within a frame, a search for similar blocks and replacement can be repeatedly performed for the same blocks. This can further increase the correction precision.

Additional objects and advantages of the invention will be set forth in the description which follows, and in part will be obvious from the description, or may be learned by practice of the invention. The objects and advantages of the invention

may be realized and obtained by means of the instrumentalities and combinations particularly pointed out hereinafter.

#### BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWING

The accompanying drawings, which are incorporated in and constitute a part of the specification, illustrate presently preferred embodiments of the invention, and together with the general description given above and the detailed description of the preferred embodiments given below, serve to explain the principles of the invention.

FIG. 1 is a block diagram showing the basic arrangement of an object tracking/extracting apparatus for moving picture according to the first embodiment of the present invention;

FIG. 2 is a block diagram showing the first example of the arrangement of the object tracking/extracting apparatus according to the first embodiment;

FIG. 3 is a block diagram showing the second example of the arrangement of the object tracking/extracting apparatus according to the first embodiment;

FIGS. 4A and 4B are block diagrams each showing an example of the detailed arrangement of a background region determination section incorporated in the object tracking/extracting apparatus according to the first embodiment;

FIG. 5 is a block diagram showing an example of the detailed arrangement of a figure setting section incorporated in the object tracking/extracting apparatus according to the first embodiment;

FIG. 6 is a block diagram showing an example of the detailed arrangement of a background motion canceling section incorporated in the object tracking/extracting apparatus according to the first embodiment;

FIG. 7 is a view showing an example of a representative background region used by the background motion canceling section incorporated in the object tracking/extracting apparatus according to the first embodiment;

FIG. 8 is a view for explaining the operation of the object tracking/extracting apparatus according to the first embodiment;

FIG. 9 is a block diagram showing the first object tracking/extracting apparatus for moving picture according to the second embodiment of the present invention;

FIG. 10 is a block diagram showing the second object tracking/extracting apparatus for moving picture according to the second embodiment;

FIG. 11 is a block diagram showing the third object tracking/extracting apparatus for moving picture according to the second embodiment;

FIG. 12 is a block diagram showing the fourth object tracking/extracting apparatus for moving picture according to the second embodiment;

FIG. 13 is a view for explaining an object prediction method used by the object tracking/extracting apparatus according to the second embodiment;

FIGS. 14A and 14B are views for explaining a reference frame selection method used by the object tracking/extracting apparatus according to the second embodiment;

FIG. 15 is a view showing an example of the object extraction result obtained by switching the first and second object extraction sections in the object tracking/extracting apparatus according to the second embodiment;

FIG. 16 is a flow chart for explaining the flow of object tracking/extracting processing for moving picture using the object tracking/extracting apparatus according to the second embodiment;

FIG. 17 is a block diagram showing the first object tracking/extracting apparatus for moving picture according to the third embodiment of the present invention;

FIG. 18 is a block diagram showing the second object tracking/extracting apparatus for moving picture according to the third embodiment;

FIG. 19 is a block diagram showing the third object tracking/extracting apparatus for moving picture according to the third embodiment;

FIG. 20 is a block diagram showing the fifth object tracking/extracting apparatus for moving picture according to the third embodiment;

FIG. 21 is a block diagram showing the sixth object tracking/extracting apparatus for moving picture according to the third embodiment;

FIG. 22 is a block diagram showing the fourth object tracking/extracting apparatus for moving picture according to the third embodiment;

FIG. 23 is a block diagram showing still another example of the arrangement of the object tracking/extracting apparatus for moving picture according to the third embodiment;

FIG. 24 is a block diagram showing still another example of the arrangement of the object tracking/extracting apparatus for moving picture according to the third embodiment;

FIG. 25 is a view for explaining an example of an extracted frame sequence based on frame sequence control applied to the object tracking/extracting apparatus for moving picture according to the third embodiment;

FIG. 26 is a view showing an application of the object tracking/extracting apparatus for moving picture according to the third embodiment;

FIG. 27 is a block diagram showing an object extraction apparatus according to the fourth embodiment of the present invention;

FIG. 28 is a block diagram showing an example of the arrangement of the object extraction apparatus according to the fourth embodiment to which edge correction processing is applied;

FIG. 29 is a block diagram showing an example of the arrangement of a motion compensation section applied to the object extraction apparatus according to the fourth embodiment;

FIG. 30 is a block diagram showing an example of the arrangement of an object extraction section based on reduced block matching which is applied to the object extraction apparatus according to the fourth embodiment;

FIG. 31 is a block diagram showing an edge correction circuit using a background palette and used in the object extraction apparatus according to the fourth embodiment;

FIG. 32 is a block diagram showing an image synthesizing apparatus applied to the object extraction apparatus according to the fourth embodiment;

FIG. 33 is a view for explaining the principle of edge correction using separation degrees and used in the object extraction apparatus according to the fourth embodiment;

FIG. 34 is a view showing the overall processing image to be processed by the object extraction apparatus according to the fourth embodiment;

FIG. 35 is a view showing the contour drawn by an operator and used in the fourth embodiment;

FIG. 36 is a view showing the state of block setting (first scanning) used in the fourth embodiment;

FIG. 37 is a view showing the state of block setting (second scanning) used in the fourth embodiment;

FIG. 38 is a view for explaining similar blocks used in the fourth embodiment;

FIG. 39 is a view for explaining a search range of similar blocks used in the fourth embodiment;

FIG. 40 is a view for explaining another search range of similar blocks used in the fourth embodiment;

FIG. 41 is a view showing the state of a shape picture before replacement/conversion, which is used in the fourth embodiment;

FIG. 42 is a view showing the state of shape picture after replacement/conversion, which is used in the fourth embodiment;

FIG. 43 is a view showing an extracted contour in the fourth embodiment;

FIG. 44 is a view showing a portion of an extracted background color in the fourth embodiment;

FIG. 45 is a view for explaining motion compensation used in the fourth embodiment;

FIG. 46 is a flow chart showing an object extraction method using a background picture and used in the fourth embodiment;

FIG. 47 is a flow chart showing an object extraction method based on motion compensation and used in the fourth embodiment;

FIG. 48 is a flow chart showing an object extraction method using reduced block matching within frames in the fourth embodiment;

FIG. 49 is a view showing another example of block setting used in the fourth embodiment;

FIG. 50 is a flow chart for explaining edge correction;

FIG. 51 is a view showing an example of block setting;

FIG. 52 is a view showing another example of block setting;

FIG. 53 is a view showing still another example of block setting;

FIGS. 54A to 54D are views showing the process of searching for the contour of an object region;

FIG. 55 is a flow chart for explaining a method of gradually reducing a block size;

FIG. 56 is a view showing still another example of block setting; and

FIG. 57 is a view showing still another example of block setting.

#### DETAILED DESCRIPTION OF THE INVENTION

FIG. 1 shows the overall arrangement of an object tracking/extracting apparatus for moving picture according to the first embodiment of the present invention. This object tracking/extracting apparatus is designed to track the motion of a target object from an input video signal, and comprises an initial figure setting section 1 and an object tracking/extracting section 2. The initial figure setting section 1 is used to initially set a figure that surrounds a target object to be tracked/extracted with respect to an input video signal a1 on the basis of an externally input initial figure setting indication signal a0. A figure having an arbitrary shape such as a rectangular, circular, or elliptic shape is set on the initial frame of the input video signal a1 so as to surround the target object on the basis of the initial figure setting indication signal a0. As a method of inputting the initial figure setting indication signal a0, the following method can be used: a method of allowing the user to directly write, with a pointing

11

device such as a pen or mouse, a figure on the screen on which the input video signal **a1** is displayed; or a method of designating the position and size of an input figure by using such a pointing device. With this operation, an object to be tracked/extracted can be easily designated from outside on the initial frame picture on which the target object appears.

Initial figure setting can also be realized by detecting, for example, the contours of the face or body of a person or animal by general frame picture analysis processing and automatically setting a figure to surround the object, instead of figure input operation performed by the user.

The object tracking/extracting section **2** tracks/extracts the object with reference to the image inside figure set by the initial figure setting section **1**. In this case, in moving object tracking/extracting processing, attention is focused on the object designated by the figure, and the motion of the object is tracked. The target moving object can therefore be extracted/tracked without any influences of the unnecessary motions of neighboring objects other than the target moving object.

FIG. **2** shows a preferable arrangement of the object tracking/extracting section **2**.

As shown in FIG. **2**, this object tracking/extracting section comprises memories (**M**) **10** and **14**, a figure setting section **11**, a background region determination section **12**, and an object extraction section **13**.

The figure setting section **11** is used to sequentially set figures for input frames by using arbitrary frames input and subjected to figure setting in the past as reference frames. The figure setting section **11** receives a current frame picture **101**, an image inside figure of a reference frame, its position **103**, and an object extraction result **106** of the current frame, and outputs image data **102** inside an arbitrary figure of the current frame. More specifically, in the figure setting processing performed by the figure setting section **11**, a region on the current frame picture which exhibits the minimum error with respect to the image **103** inside figure of the reference frame is searched out on the basis of the correlation between the image **103** inside figure of the reference frame and the current frame picture **101**, and a figure that surrounds the region is set for the current frame picture. The figure to be set may be any one of the following shapes: a rectangle, a circle, an ellipse, a region surrounded by an edge, and the like. For the sake of simplicity, a rectangle is taken as an example in the following case. The detailed arrangement of the figure setting section **11** will be described with reference to FIG. **5**. Note that if any figure that surrounds an object is not to be used, the entire image is an image inside figure, and any position need not be input and output.

The memory **10** saves at least three frames that have been already input and undergone already figure setting. The saved information includes the pictures of the figure-set frames, the positions and shapes of the set figures, images inside figures, and the like. The memory **10** may save only the intra-frame pictures instead of the overall pictures of the input frames.

The background region determination section **12** uses at least two arbitrary frames of the frames that temporally differ from a current frame as reference frames for each current frame as an object extraction target, and obtains the difference between each reference frame and the current frame, thereby determining a background region common to each reference frame and the current frame. The background region determination section **12** receives an image inside arbitrary figure of the current frame, its position **102**, images

12

inside arbitrary figures of at least two frames, and their positions **103**, which are saved in the memory **10**, together with the object extraction result **106** obtained from at least two frames, and outputs background regions **104** common to the images inside figures of the current frames and at least two frames. More specifically, when first and second frames are to be used as reference frames, a first background region commonly used as a background region in both the current frame and the first reference frame is determined from the first difference image obtained by calculating the inter-frame difference between the current frame and the first reference frame. In addition, a second background region commonly used as a background region in both the current frame and the second reference frame is determined from the second difference image obtained by calculating the inter-frame difference between the current frame and the second reference frame.

The detailed arrangement of the background region determination section **12** will be described later with reference to FIG. **4**. A method of obtaining a common background by using a background memory is also available.

Note that if any figure that surrounds an object is not to be used, the entire image is an image inside figure, and any position need not be input and output.

The object extraction section **13** is used to extract only an object region from the image inside figure of the current frame by using the common background region determined by the background region determination section **12**. The object extraction section **13** receives the background regions **104** common to the current frame and at least two frames, and outputs the object extraction result **106** associated with the current frame. Since the object region on the current frame is commonly included in both the first and second difference images, the object region on the current frame can be extracted by detecting a region, of the regions that do not belong to the first and second common background regions, which is included in the image inside figure of the current frame. This operation is based on the fact that regions other than common background regions become object region candidates. More specifically, a region other than the first common background region on the first difference image becomes an object region candidate, and a region other than the second common background region on the second difference image becomes an object region candidate. Therefore, a region where the two object region candidates overlap can be determined as the object region of the current frame. As the object extraction result **106**, information indicating the position and shape of the object region can be used. In addition, the picture in the object region may actually be extracted from the current frame by using the information.

The memory **14** saves at least two object extraction results, and is used to feed back the already extracted results so as to increase the extraction precision.

An object extraction/tracking processing method used in this embodiment will be described below with reference to FIG. **8**.

Assume that three temporally continuous frames  $f(i-1)$ ,  $f(i)$ , and  $f(i+1)$  are used to extract an object from the current frame  $f(i)$ .

First of all, figure setting processing is performed by the figure setting section **11**. Figure setting processing is performed by respectively using arbitrary reference frames for the three frames  $f(i-1)$ ,  $f(i)$ , and  $f(i+1)$  to set rectangles  $R(i-1)$ ,  $R(i)$ , and  $R(i+1)$  so as to surround the objects on the respective frames. Note that the rectangular figures  $R(i-1)$ ,

$R(i)$ , and  $R(i+1)$  are pieces of information about positions and shapes, but are not present as images.

A common background region is then determined by the background region determination section 12.

In this case, first of all, the inter-frame difference between the current frame  $f(i)$  and the first reference frame  $f(i-1)$  is calculated to obtain a first difference image  $fd(i-1, i)$ . Likewise, the inter-frame difference between the current frame  $f(i)$  and the second reference frame  $f(i+1)$  is calculated to obtain a second difference image  $fd(i, i+1)$ .

When the first difference image  $fd(i-1, i)$  is obtained, since the pixel values of portions of the current frame  $f(i)$  and first reference frame  $f(i-1)$  which are common in pixel value are canceled out, the difference value between the pixels becomes zero. If, therefore, the frames  $f(i-1)$  and  $f(i)$  have substantially the same background, an image corresponding to the OR of the image inside figure of the rectangle  $R(i-1)$  and the image inside figure of the rectangle  $R(i)$  basically remains in the first difference image  $fd(i-1, i)$ . As shown in FIG. 8, the figure surrounding this remaining image is a polygon  $Rd(i-1, i)=R(i-1)$  or  $R(i)$ . The background region common to the current frame  $f(i)$  and the first reference frame  $f(i-1)$  is the entire region other than the actual object region (the region in the form of the number 8 obtained by overlapping two circles in this case) in the polygon  $Rd(i-1, i)$ .

In the second difference image  $fd(i, i+1)$  as well, an image corresponding to the OR of the image inside figure of the rectangle  $R(i)$  and the image inside figure of the rectangle  $R(i+1)$  remains. The figure surrounding this remaining image becomes a polygon  $Rd(i, i+1)=R(i)$  or  $R(i+1)$ . The background region common to the current frame  $f(i)$  and the second reference frame  $f(i+1)$  is the entire region other than the actual object region (the region in the form of the number 8 obtained by overlapping two circles in this case) in the polygon  $Rd(i, i+1)$ .

Subsequently, the background region common to the current frame  $f(i)$  and the first reference frame  $f(i-1)$  is determined from the first difference image  $fd(i-1, i)$ .

There is required a difference value as a threshold value to be used for determining a common background region/object region. This value may be input by the user or may be automatically set by detecting picture noise and properties. In this case, one threshold value need not be determined for one frame but may be determined partially in accordance with the properties of a portion of a picture. The properties of a picture include edge intensity, difference pixel dispersion, and the like. In addition, a threshold value may be obtained by using a figure for tracking an object.

In this case, a difference value serving as a threshold value for determining a common background region/object region is obtained, and the region of a pixel having a difference value equal to or smaller than the threshold value is determined as a common background region. This threshold value can be determined by using the histogram of the difference values of the respective pixels along one outer line of the polygon  $Rd(i-1, i)$  of the first difference image  $fd(i-1, i)$ , i.e., the contour of the polygon  $Rd(i-1, i)$ . The abscissa of the histogram represents the pixel values (difference values); and the ordinate, the numbers of pixels having the respective pixel values. For example, a difference value corresponding to the half of the total number of pixels on the contour of the polygon  $Rd(i-1, i)$  is determined as the above threshold value. In this manner, a threshold value can be easily determined without checking the distribution of pixel values throughout the first difference image  $fd(i-1, i)$ .

By using this threshold value, the common background region in the polygon  $Rd(i-1, i)$  of the first difference image  $fd(i-1, i)$  is determined. A region other than the common background region is an object region including an occlusion. With this operation, the region in the polygon  $Rd(i-1, i)$  is divided into the background region and the object region. The pixel values of the background and object regions are respectively converted into binary images of "0" and "1".

Similar processing is performed for the second difference image  $fd(i, i+1)$ . The background region common to the current frame  $f(i)$  and the second reference frame  $f(i+1)$  is determined, and the region in the polygon  $Rd(i, i+1)$  is converted into a background region having a pixel value of "0" and an object region having a pixel value of "1".

After this processing, object extraction is performed by the object extraction section 13.

In this case, AND processing for the binary image in the polygon  $Rd(i-1, i)$  and the binary image in the polygon  $Rd(i, i+1)$  is performed in units of pixels. With this processing, the intersection of the objects including the occlusions is obtained, thereby extracting an object  $O(i)$  on the current frame  $f(i)$ .

In this case, all the regions other than the object regions in the frame difference images are obtained as common background regions. However, only the image inside figure may be extracted from each frame, and the difference between the respective images inside figures may be calculated in consideration of the positions of the images inside figures on the frames. In this case, only the common background regions in the polygon  $Rd(i-1, i)$  and the polygon  $Rd(i, i+1)$  are determined.

As described above, in this embodiment, object extraction is performed by the ORAND method in consideration of images inside figures as follows:

- 1) obtaining the difference images between the current frame and at least two reference frames, i.e., the first and second reference frames, which temporally differ from the current frame, thereby obtaining the OR of the images inside figures of the current and first reference frames and the OR of the images inside figures of the current and second reference frames, and
- 2) extracting the target object region from the image inside figure of the current frame by AND processing for the difference images obtained by OR processing for these images inside figures.

In addition, the temporal relationship between the current frame and the two reference frames is not limited to that described above. For example, two frames  $f(i-m)$  and  $f(i-n)$  temporally preceding the current frame  $f(i)$  may be used as reference frames, or two frames  $f(i+m)$  and  $f(i+n)$  temporally following the current frame  $f(i)$  may be used as reference frames.

Referring to FIG. 8, assume that the frames  $f(i-1)$  and  $f(i)$  are used as reference frames, and the same processing as that described above is performed for the difference images between the reference frames and the frame  $f(i+1)$ . In this case, the object can be extracted from the frame  $f(i+1)$ .

FIG. 3 shows the second example of the arrangement of the object tracking/extracting section 2.

The main difference from the arrangement shown in FIG. 2 is to additionally arrange a background motion canceling section 21. The background motion canceling section 21 serves to correct the motions of the backgrounds of each reference frame and the current frame so as to cancel out their motions.

The apparatus shown in FIG. 3 will be described in detail below.

The background motion canceling section 21 receives images inside arbitrary figures of at least two frames that temporally differ from a current frame 201, together with positions 206 of the images inside figures, and outputs pictures 202 obtained by canceling the motions of the backgrounds of these two frames. The detail arrangement of the background motion canceling section 21 will be described later with reference to FIG. 6.

A figure setting section 22 corresponds to the figure setting section 11 in FIG. 2. The figure setting section 22 receives the current frame 201, at least the two pictures 202 obtained by canceling the motions of the backgrounds, and object extraction results 206 based on the pictures 202, and outputs images 203 representing the inside of the regions of the current frame and at least the two pictures 202 which are surrounded by arbitrary figures.

A memory 26 holds the images inside arbitrary figures and their positions.

The background region determination section 23 corresponds to the background region determination section 12 in FIG. 2. The background region determination section 23 receives the images inside arbitrary figures and their positions 203, and the object extraction results 206 based on the pictures 202, and outputs background regions 204 common to the current frame and at least the two pictures 202. An object extraction section 24 corresponds to the object extraction section 13 in FIG. 2. The object extraction section 24 receives the background regions 204 common to the current frame and at least the two pictures, and outputs an object extraction result 205 based on the current frame. A memory 25 saves at least two object extraction results. The memory 25 corresponds to the memory 14 in FIG. 2.

With this background motion canceling section 21, even if background moving picture gradually changes between continuous frames as in the case wherein a camera is panned, the pseudo background moving picture can be made constant between the frames. Therefore, when the difference between the current frame and a reference frame is obtained, the backgrounds of these frames can be canceled out. This allows common background region detection processing and object region extraction processing without any influences of changes in background.

Note that the background motion canceling section 21 may be connected to the input stage of a background region determination section 23 to eliminate the motion of the background of each reference frame in accordance with the current frame.

FIG. 4A shows an example of the detailed arrangement of the background region determination section 12 (or 23).

A difference value detector section 31 is used to obtain the difference between the current frame and the first and second reference frames described above. The difference value detector section 31 receives images inside arbitrary figures of frames that temporally differ from the current frame and their positions 302, and object extraction results 301 based on the frames that temporally differ from the current frame, and outputs a difference value 303 between the images inside arbitrary figures of the frames that temporally differ from the current frame. As this difference value, for example, the luminance difference between the frames, color variation, optical flow, or the like can be used. By using the object extraction results based on the frames that temporally differ from the current frame, an object can be extracted even if the object does not change between the frames. Assume that an inter-frame difference is used as a difference value.

In this case, a portion belonging to the object and exhibiting zero inter-frame difference indicates that the object is standing still. Therefore, the same results as the object extraction results based on the frames that temporally differ from the current frame can be obtained.

A representative region determination section 32 receives an image inside arbitrary figure of the current frame and its position 302, and outputs the background of the image inside arbitrary figure as a representative region 304. As this representative region, a region that is expected to contain the most background in the image inside arbitrary figure is selected. For example, a belt-like region on the outermost portion of the image inside figure, like the contour of the figure on the difference image described with reference to FIG. 8, is set. Since the figure is set to surround the object, the possibility that the figure is a background is high.

A background difference value determination section 33 receives the representative region 304 and the difference value 303, and outputs a difference value for determining a background. A background difference value is determined as follows. As described with reference to FIG. 8, the histogram of the difference values of the difference values in the representative region is formed. Then, for example, a region having a difference value, i.e., a difference value, corresponding to the number of pixels equal to or more than the half (majority) of the total number of pixels is determined as a background region.

A representative region background determination section 34 receives the background difference value 305, determines a representative region background 306, and outputs it. The background region of the representative region is determined depending on whether the region corresponds to the background difference value determined in advance. A background region determination section 35 receives the difference value 303, the background determination threshold value 305, and the representative region background 306, and outputs a background 307 of a region other than the representative region. The background region other than the representative region is determined by a growth method based on the representative region. If, for example, an undetermined pixel adjacent to a determined pixel in the inward direction of the figure coincides with the background difference value, the undetermined pixel is determined as a background pixel. Pixels that are not adjacent to the background and pixels that do not coincide with the background difference value are determined as pixels other than the background. Alternatively, a pixel may be simply determined depending on whether it corresponds to the background difference value determined in advance. By performing determination inwardly from the contour of a figure on a difference image, the extent to which the background region inwardly extends in the image inside figure can be determined.

In contrast to this, an object region protruding outwardly from the contour of the figure is detected. If, for example, an undetermined pixel adjacent to a pixel determined as a pixel other than a background in the direction of the outside of the figure does not coincide with the background difference value, this pixel is determined as a pixel other than the background. A pixel which is not adjacent to a pixel other than the background or coincides with the background difference value is determined as a background pixel. By performing pixel determination outwardly from the contour of the figure on the difference image in this manner, the extent to which an image outside the figure extends as a region outside the background can be determined. In this case, a difference value must also be obtained outside the

figure. For this reason, an arbitrary figure may be increased in size by several pixels to set a new figure that can reliably surround the object, and a difference value may be obtained only within the figure, or a difference value may be simply obtained in an overall frame. Alternatively, a difference value may be obtained in advance only within the figure, and the above processing may be performed while a difference value is sequentially obtained in performing determination outside the figure. Obviously, when the object does not protrude from the figure, for example, when no pixel is present except for background pixels on the contour, processing outside the figure need not be performed.

If an object or a part thereof is standing still between the current frame and a reference frame, the difference between the current frame and the reference frame cannot be detected so that the shape of the object may not be properly extracted. A method of detecting an object on the current frame by using already extracted reference frames will therefore be described with reference to FIG. 4B.

FIG. 4B shows a background region determination section 12 (or 23) having a still object region detector section 37. According to this section, a difference value detector section 31 receives an image inside figure and its position on the current frame and images inside figures and their positions 311 on at least two temporally different frames, and detects difference values 313 between the images inside figures of the current and reference frames.

A shape predicting section 36 receives the image inside figure and its position on the current frame, the images inside figures and their positions 311 on at least the two temporally different frames, and an image and its position 317 on an already extracted frame, predicts an object shape 312 on a frame, of the frames temporally different from the current frame, from no object has been extracted yet, and outputs the predicted shape.

A still object region detector section 37 receives the predicted object shape 312, the difference values 313 between the reference frames and the current frame, and the object shape 317 of the already extracted frame, and determines an object region 314 that is still with respect to the current frame from at least the two frames.

A background region determination section 35 receives the object region 314 with respect to the current frame, which is associated with at least the two frames and the difference values 313 between the reference frames and the current frame, determines a background region 316 common to at least the two frames and the current frame, and outputs it.

Assume that an object has been extracted from a reference frame. Consider a region of the current frame in which the inter-frame difference with respect to the reference frame zero. If the same position on the reference frame is part of the object, the corresponding region on the current frame can be extracted as part of the still object. In contrast to this, if this region on the reference frame is part of a background, the corresponding region on the current frame is a background.

If, however, no object has been extracted from the reference frame, a still object or part of an object cannot be extracted by the above method. In this case, an object shape on the reference frame from which no object has been extracted can be predicted by using another frame from which an object has already been extracted, and it can be determined that the corresponding portion is part of the object. As a prediction method, for example, the block matching method or affine transform method which is often used to code a picture.

For example, the block matching method shown in FIG. 13 can be conceived. By predicting the shape of the object in this manner, a region where no inter-frame difference is detected can be determined as part of a still object or background.

If any figure that surrounds an object is not to be used, the entire image is an image inside figure, and any position need not be input and output. This shape prediction can be performed by using the same method as that used to select a reference frame. In addition, in an embodiment in which a given object extraction method is switched to another object extraction method, the object shape obtained by another object extraction method can be used.

FIG. 5 shows an example of the detailed arrangement of the figure setting section 11 (or 22).

A division section 41 receives an image inside arbitrary figure of a frame that temporally differs from the current frame and its position 402, and outputs segmented pictures 403. The image inside arbitrary figure may be segmented into two or four equal parts. Alternatively, edges may be detected to segment the image. Assume that the image is divided into two equal figures, the divided figures will be referred to as blocks. A motion detector section 42 receives the segmented image inside arbitrary figure and its position 403, and the image inside arbitrary figure of the current frame and its position 401, and outputs the motion of the segmented image and an error 404. In this case, the position of each block which corresponds to the current frame is searched out to minimize the error, thereby obtaining the motion and the error. A division determination section 43 receives the motion, the error 404, and an object extraction result 407 based on the frame that temporally differs from the current frame, and outputs a determination result 406 indicating whether to segment the image inside arbitrary figure of the frame that temporally differs from the current frame. If it is determined that the image is not segmented, the division determination section 43 outputs a motion 405. In this case, if the object extraction result based on the frame that temporally differs from the current frame is not contained in a given segmented block, the block is eliminated from the figure. In another case, if the obtained error is equal to or larger than a threshold value, the block is further segmented to obtain the motion again. Otherwise, the motion of the block is determined. A figure determination section 44 receives the motion 405, and outputs the image inside figure of the current frame and its position 407. In this case, the figure determination section 44 obtains the positional correspondence between each block and the current frame, and determines a new figure to contain all the blocks at the corresponding positions. The new figure may be a rectangle or circle that is effective for the unity of all the blocks and contains all of them.

In this manner, the image inside figure of each reference frame is segmented into a plurality of blocks, a region where the error between each block and the current frame is minimized is searched out, and a figure surrounding a plurality of regions that are searched out is set for the current frame. This allows a new figure having an optimal shape to be set for the input frame subjected to figure setting regardless of initially set figure shapes and sizes.

It suffices if a reference frame to be used for figure setting is a frame for which a figure has already been set and which temporally differs from the current frame. A frame temporally following the current frame may be used as a reference frame for figures setting as in the case wherein forward prediction and backward prediction are used in general coding techniques.



FIG. 6 shows an example of the detailed arrangement of the background motion canceling section 21.

A representative background region setting section 51 receives a temporally different image inside arbitrary figure and its position 501, and outputs a representative background region 503. The representative background region is a region representing the global motion in an arbitrary figure, i.e., representatively represents the motion of the background in the figure. If, for example, an arbitrary figure is a rectangle, a belt-like frame region having a width corresponding to several pixels is set to surround a rectangle, as shown in FIG. 7. Alternatively, several pixels outside the figure may be used. A motion detector section 52 receives a current frame 502 and the representative background region 503, and outputs a motion 504. In the above case, the motion of the belt-like frame region around the rectangle with respect to the current frame is detected. The frame region may be detected as one region. Alternatively, as shown in FIG. 7, the frame region may be divided into a plurality of blocks, and the averaged motion of the respective blocks may be output, or a motion representing the majority may be output.

A motion compensation section 53 receives the temporally different frame 501 and the motion 504, and outputs a motion compensated picture 505. The motion of the temporally different frame is eliminated by using the motion obtained in advance in accordance with the current frame. Motion compensation may be block matching motion compensation or motion compensation using affine transform.

As described above, in this embodiment, a target object can be accurately extracted/tracked by relatively simple processing without any influences of unnecessary motions other than the motion of the target object as follows: (1) tracking the object by using a figure approximately surrounding the object instead of a contour of the object, (2) setting an arbitrary figure for the current frame, determining background regions common to the images inside figures of the current frame and at least the two frames, and extracting the object of the current frame, (3) canceling the motions of the backgrounds of at least the two temporally different frames, (4) detecting the difference value of the images inside arbitrary figures, determining a representative region, determining a difference value corresponding to the images inside figures of the current frame and at least the two frames and the backgrounds at their positions, and determining a background on the basis of the relationship between the difference value and the representative region, (5) segmenting each image inside figure, detecting the motion of the image inside arbitrary figure or part of the segmented image inside figure, determining whether to segment the image inside arbitrary figure or part of the segmented image inside figure, and determining the image inside arbitrary figure and its position of the current frame, and (6) setting a region representing the background, detecting the motion of the background, and forming a picture by canceling the motion of the background of each of the temporally different frames.

In addition, a procedure for object extracting/tracking processing of this embodiment can be implemented by software control. In this case as well, the basic procedure is the same as that described above. After initial figure setting is performed, figure setting processing may be sequentially performed for each input frame. Concurrently with or after this figure setting processing, background region determination processing and object extraction processing may be performed.

The second embodiment of the present invention will be described next.

The first embodiment includes only one object extraction section based on the ORAND method. In some case, however, satisfactory extraction performance may not be obtained by using this section alone depending on input pictures. According to the ORAND method in the first embodiment, a common background is set on the basis of the difference between the current frame subjected to object extraction and the first reference frame that temporally differs from the current frame. In addition, a common background is set on the basis of another current frame and the second reference frame that temporally differs from another current frame. A method of selecting these first and second reference frames is not specifically limited. Depending on the selected first and second reference frames, the object extraction results greatly vary, and any satisfactory result may not be obtained.

The second embodiment is obtained by improving the first embodiment to extract an object with high precision regardless of input pictures.

The first example of the arrangement of an object tracking/extracting apparatus according to the second embodiment will be described first with reference to the block diagram of FIG. 9.

Only the arrangement corresponding to the object tracking/extracting section 2 of the first embodiment will be described below.

A figure setting section 60 is identical to the figure setting section 11 in the first embodiment described with reference to FIG. 2. The figure setting section 60 receives a frame picture 601 and a FIG. 602 set for an initial frame or another input frame, sets a figure for the frame picture 601, and outputs it. A switching section 61 receives a result 605 of object extraction that has already been performed, and outputs a signal 604 for switching to the object extraction section to be used on the basis of the result.

An object tracking/extracting section 62 is made up of first to *K*th object tracking/extracting sections, as shown in FIG. 9. These object tracking/extracting sections perform object extraction by different methods. The object tracking/extracting sections 62 include at least a section using the ORAND method described in the first embodiment. As object tracking/extracting sections using other methods, a section using a shape predictive method based on block matching, a section using an object shape predictive method based on affine transform, and the like can be used. In these shape predictive methods, the position or shape of an object region on the current frame is predicted by inter-frame prediction between a frame having undergone object extraction and the current frame, and the object region is extracted from an image inside FIG. 603 of the current frame on the basis of the prediction result.

FIG. 13 shows an example of how shape prediction is performed by block matching. The image inside figure of the current frame is segmented into blocks having the same size. Each block that is most similar in texture to a corresponding block of the current frame is searched out from a reference frame from which the shape and position of an object have already been extracted. Shape data representing an object region on this reference frame has already been created. The shape data is obtained by expressing the pixel value of each pixel belonging to the object region as "255"; and the pixel value of each of the remaining pixels as "0". Shape data corresponding to the searched out block is pasted to the corresponding position on the current frame. Such a texture search and shape data pasting processing are performed for all the blocks constituting the image inside figure of the current frame to fill the image inside figure of the current



frame with shape data for discriminating the object region from the background region. By using this shape data, therefore, a picture (texture) corresponding to the object region can be extracted.

Assume that when operation similar to that of the first object tracking/extracting section is performed, the extraction precision is high. In this case, the switching section 61 operates to select the first object tracking/extracting section. Otherwise, the switching section 61 operates to select another object tracking/extracting section. If, for example, the first object tracking/extracting section is an object shape predicting section based on block matching, switching of the object tracking/extracting sections may be controlled in accordance with the magnitude of a matching error. If this section is an object shape predicting section based on affine transform, the object tracking/extracting sections can be switched in accordance with the magnitude of the estimation error of an affine transform coefficient. The switching operation of the switching section 61 is not performed in units of frames but is performed in units of small regions in each frame, e.g., blocks or regions segmented on the basis of luminances or colors. With this operation, the object extraction methods to be used can be selected more finely, and hence the extraction precision can be increased.

FIG. 10 shows the second example of the moving object tracking/extracting apparatus according to the second embodiment.

A figure setting section 70 is identical to the figure setting section 11 in the first embodiment described with reference to FIG. 2. The figure setting section 70 receives a picture 701 and a FIG. 702 set for an initial frame or another input frame, sets a figure for the frame picture 701, and outputs it.

A second object extraction section 71 is used to extract an object region by shape prediction using the block matching method or affine transform. The second object extraction section 71 receives an image inside FIG. 703 of the current frame which is input from the figure setting section 70 and the shape and position 707 of an object on another reference frame having undergone extraction processing, and predicts the shape and position of the object from the image inside FIG. 703 of the current frame.

A reference frame selector section 72 receives the predicted shape and position 704 of the object on the current frame which are predicted by the second object extraction section 71 and the shape and position 707 of the object that have already been extracted, and selects at least two reference frames. A method of selecting reference frames will be described below.

Reference symbols  $O_i$ ,  $O_j$ , and  $O_{curr}$  denote objects on frames  $i$ ,  $j$ , and a currently extracted frame  $curr$ , respectively. Deviations  $d_i$  and  $d_j$  between two temporally different reference frames  $f_i$  and  $f_j$  are calculated, and these deviations are ANDed to extract an object from a current frame  $f_{curr}$ . As a result, the overlap between the objects  $O_i$  and  $O_j$  is extracted by AND processing for the temporally different frames, in addition to the desired object  $O_{curr}$ . Obviously, if  $O_i \cap O_j = \phi$ , i.e., if there is no overlap between the objects  $O_i$  and  $O_j$ , and the overlap between the objects  $O_i$  and  $O_j$  becomes an empty set, no problem arises.

If, however, there is an overlap between the objects  $O_i$  and  $O_j$  and the overlap is located outside the object to be extracted,  $O_{curr}$  and  $O_i \cap O_j$  remain as extraction results.

In this case, as shown in FIG. 14A, no problem is posed when there is no region common all to the background region ( $O_{curr} \cap (O_i \cap O_j) = \phi$ ) of the object  $O_{curr}$  and the objects  $O_i$  and  $O_j$ . If, however, as shown in FIG. 14B, there is a region common all to the background region

( $O_{curr} \cap (O_i \cap O_j) \neq \phi$ ) of the object  $O_{curr}$  and the objects  $O_i$  and  $O_j$ , the object  $O_{curr}$  is extracted in a wrong shape, as indicated by the hatching.

The optimal reference frames  $f_i$  and  $f_j$  for extraction of an object in a correct shape are frames that satisfy

$$(O_i \cap O_j) \cap O_{curr} \quad (1)$$

That is, they are the frames  $f_i$  and  $f_j$  that make the overlap between the objects  $O_i$  and  $O_j$  belong to the object  $O_{curr}$  (FIG. 14A).

In addition, when two or more reference frames are to be selected,

$$(O_i \cap O_j \cap \dots \cap O_k) \cap O_{curr} \quad (2)$$

The shape of an object can therefore be reliably extracted by selecting reference frames that satisfy expression (1) or (2) on the basis of the prediction result on the position or shape of the object on the current frame subjected to object extraction.

A first object tracking/extracting section 73 receives at least two reference frames 705 selected by the reference frame selector section 72 and the current picture 701, extracts an object by the ORAND method, and outputs its shape and position 706.

A memory 74 holds the shape and position 706 of the extracted object.

FIG. 11 shows the third example of the arrangement of the object tracking/extracting apparatus according to the second embodiment.

As shown in FIG. 11, this object tracking/extracting apparatus comprises a figure setting section 80, a second object extraction section 81, a switching section 82, and a first object extraction section 83. The figure setting section 80, the second object extraction section 81, and the first object extraction section 83 respectively correspond to the figure setting section 70, the second object extraction section 71, and the first object tracking/extracting section 73 in FIG. 10. In this case, with the switching section 82, the extraction results obtained by the second object extraction section 81 and the first object extraction section 83 are selectively used.

More specifically, the figure setting section 80 receives a picture 801 and the shape and position 802 of an initial figure, and outputs the shape and position 803 of the figure. The second object extraction section 81 receives the shape and position 803 of the figure and the shape and position 806 of an already extracted object, predicts the predicted shape and position 804 of an object that has not been extracted, and outputs them. The switching section 82 receives the shape and position 804 of the object which are predicted by the second object extraction section 81, and outputs a signal 805 for switching or not switching to the first object extraction section 83. The first object extraction section 83 receives the shape and position 806 of the already extracted object and the predicted shape and position 804 of the object that has not been extracted, determines the shape and position 805 of the object, and outputs them.

The switching operation of the switching section 82 may be performed in units of blocks as in the above case, or may be performed in units of regions segmented on the basis of luminances or colors. For example, switching may be determined on the basis of the predictive error in object prediction. More specifically, if the predictive error in the second object extraction section 81 that performs object extraction by using inter-frame prediction is equal to or smaller than a predetermined threshold value, the switching section 82 operates to use the predicted shape obtained by the second

object extraction section 81 as an extraction result. If the predictive error in the second object extraction section 81 exceeds the predetermined threshold value, the switching section 82 operates to make the first object extraction section 83 perform object extraction by the ORAND method. The extraction result is then output to an external unit.

FIG. 15 shows examples of the extraction results obtained when the extraction sections to be used are switched for each block as a unit of prediction on the basis of a matching error.

In this case, each crosshatched portion indicates the object shape predicted by the second object extraction section 81, and the hatched portion indicates the object shape obtained by the first object extraction section 83.

FIG. 12 shows the fourth example of the arrangement of the moving object tracking/extracting apparatus according to the second embodiment.

This object tracking/extracting apparatus has the reference frame selecting section shown in FIG. 10 in addition to the arrangement in FIG. 11.

A figure setting section 90 receives a picture 901 and the shape and position 902 of an initial figure, and outputs the shape and position 903 of the figure. A second object extraction section 91 receives the shape and position 903 of the figure and the shape and position 908 of an already extracted object, and predicts the shape and position 904 of an object that has not been extracted. A switching section 92 receives the predicted shape and position 904 of the object, checks whether the precision of the predicted object is satisfactorily high, and outputs a switch signal 905 for switching the object extraction output obtained by the second object extraction section. A reference frame selector section 93 receives the predicted shape and position 904 of the object that has not been extracted, selects the shape and position 906 of an object based on at least two reference frames or those of a predicted object, and outputs them. An object tracking/extracting section 94 receives the current picture 901 and the shape and position 906 of the object based on at least the two reference frames or the predicted object, extract an object, and outputs the shape and position 907 of the object. A memory 95 holds the shape and position 907 of the extracted object or the shape and position 904 of the predicted object.

A procedure for an object tracking/extracting method in this case will be described below with reference to FIG. 16. (Step S1)

As reference frame candidates, frames that temporally differ from the current frame are set in advance. These candidates may be all the frames other than the current frame or may be several frames preceding/following the current frame. For example, reference frame candidates are limited to a total of five frames, i.e., the initial frame, the three frames preceding the current frame, and the one frame following the current frame. If, however, the number of previous frames is less than three, the number of future frames as candidates is increased accordingly. If there is no frame following the current frame, four frames preceding the current frames are set as candidates. (Step S2)

First of all, the user sets a figure, e.g., a rectangle, on the initial frame in which the object to be extracted is drawn. A figure is set on each subsequent frame by dividing the initially set figure into blocks, matching the blocks, and pasting each block to the corresponding position. The object is tracked by setting a new rectangle to surround all the pasted blocks. Figures for object tracking are set on all the reference frame candidates. If an object tracking figure for each future frame is obtained by using the object every time

the object is extracted, an extraction error can be prevented more effectively. In addition, the user inputs the shape of the object on the initial frame.

Assume that the frame from which the object is to be extracted is the current frame, and the object has already been extracted from each previous frame, but no object has been extracted from the future frame.

(Step S3)

A proper region is set around the figure on each reference frame candidate. The motion of the background with respect to the current frame is detected to eliminate the background in the figure on the reference frame. The motion of the background is detected by the following method. A region having a width corresponding to several pixels is set around the figure. This region is matched with the current frame. A motion vector exhibiting the minimum matching error is detected as the motion of the background.

(Step S4)

An extraction error caused when the background motion is not properly eliminated can be prevented by removing any reference frame that exhibits a large motion vector detection error in canceling the background motion from the candidates. In addition, if the number of reference frame candidates decreases, new reference frame candidates may be selected again. If figure setting and background motion elimination have not been performed for a new reference frame candidate, figure setting and background motion elimination must be performed.

(Step S5)

The shape of the object on the current frame from which the object has not been extracted and the shape of the object on each reference frame candidate preceding the current frame are predicted. The rectangle set on the current frame or the preceding reference frame candidate is segmented into, e.g., blocks, and block matching is performed with a frame (previous frame) from which the object has already been extracted, and the corresponding object shape is pasted, thereby predicting the object shape. An extraction error can be prevented more effectively by predicting the object on each future frame by using the object every time it is extracted.

(Step S6)

At this time, any block exhibiting a small prediction error outputs the predicted shape as an extraction result without any change. If an object shape is predicted in units of blocks, block distortion may occur owing to matching errors. In order to prevent this, the video signal may be filtered to smooth the overall object shape.

Rectangle segmentation in object tracking and object shape prediction may be performed in a fixed block size, or may be performed by hierarchical block matching with a matching threshold value.

The following processing is performed for each block exhibiting a large prediction error.

(Step S7)

Temporary reference frames are set from the reference frame candidates, and each set of reference frames that satisfy expression (1) or (2) is selected. If any set of all the reference frame candidates does not satisfy either expression (1) or (2), a set having the minimum number of pixels in  $O_i \cap O_j$  may be selected. Reference frame candidates are preferably combined to select frames that minimize motion vector detection errors in canceling the motion of the background. More specifically, if there are reference frame sets that satisfy expression (1) or (2), for example, a set that exhibits a smaller motion vector detection error in canceling the motion of the background may be selected. Assume that two frames are selected as reference frames in the following description.

(Step S8)

When reference frames are selected, the inter-frame difference between each reference frame and the current frame is obtained, and attention is given to the inter-frame difference in the set figure. The histogram of the absolute values of the deviations of one-line pixels outside the set figure is obtained, and the absolute value of a majority difference is set as the difference value of the background region, thereby determining the background pixels of the one-line pixels outside the set figure. A search is performed inwardly from the background pixels of the one-line pixels outside the set figure to determine any pixel having the same difference value as that of the adjacent background region as a background pixel. This search is sequentially performed until no pixel is determined as a background pixel. This background pixel is a background region common to the current frame and one reference frame. At this time, since the boundary between the background region and the remaining portions may become unnatural, the video signal may be filtered to smooth the boundary or eliminate excess or noise regions. (Step S9)

When background regions common to the respective reference frames are obtained, a region that is not contained in the two common background regions is detected and extracted as an object region. This result is output for a portion that does not use the object shape predicted in advance to output the overall object shape.

If there is no matching between the portion using the shape obtained from the common backgrounds and the portion using the predicted object shape, filtering can make the output result look nice.

As described above, according to the second embodiment, an object can be accurately extracted regardless of input pictures, or reference frames suitable for object extraction can be selected.

The third embodiment of the present invention will be described next.

The first example of an object tracking/extracting apparatus according to the third embodiment will be described first with reference to the block diagram of FIG. 17.

In this arrangement, the feature value of a picture in at least a partial region is extracted from the current frame subjected to object extraction, and a plurality of object extraction sections are switched on the basis of the feature value.

As shown in FIG. 17, this object tracking/extracting apparatus comprises a figure setting section 110, a feature value extraction section 111, a switching section 112, a plurality of object tracking/extracting sections 113, and a memory 114. The figure setting section 110, the switching section 112, and the plurality of object tracking/extracting sections 113 respectively correspond to the figure setting section 60, the switching section 61, and the plurality of object tracking/extracting sections 62 in the second embodiment in FIG. 9. This apparatus differs from that of the second embodiment in that the object tracking/extracting sections to be used are switched on the basis of the feature value of the picture of the current frame which is extracted by the feature value extraction section 111.

The figure setting section 110 receives an extracted frame 1101, an initial FIG. 1102 set by the user, and an extraction result 1106 based on the already extracted frame, sets a figure for the extracted frame, and outputs the figure. The figure may be a geometrical figure such as a rectangle, circle, or ellipse, or the user may input an object shape to the figure setting section 110. In this case, the figure may not have a precise shape but may have an approximate shape. The

feature value extraction section 111 receives an extracted frame 1103 in which a figure is set and the extraction result 1106 based on the already extracted frame, and outputs a feature value 1104. The switching section 112 receives the feature value 1104 and the extraction result 1106 based on the already extracted frame, and controls inputting the extraction result 1106 based on the already extracted frame to the object tracking/extracting section.

Upon reception of the feature value of the overall picture, the switching section 112 detects the properties of the picture, and can use them for control on inputting of the picture to a proper object tracking/extracting section. The portion inside a figure is segmented into portions each having a proper size, and the feature value may be applied in units of segmented figure portions. The feature value includes a dispersion, luminance gradient, edge intensity, and the like. In this case, these values can be automatically calculated. Alternatively, the user may visually perceive the properties of the object and input them to the switching section 112. If, for example, a target object is a person, his/her hair exhibiting unclear edges may be designated to specially select a parameter for extraction, and extraction may be performed after edge correction is performed as pre-processing.

The feature value may be associated with a portion (background portion) outside the set figure as well as portions (object and its surrounding) inside figure.

Each of the plurality of (first to kth) object tracking/extracting sections 113 receives the extracted frame 1103 in which the figure is set and the extraction result 1106 based on the already extracted frame, and outputs a result 1105 obtained by tracking/extracting the object.

The plurality of object tracking/extracting sections 113 include a section for extracting an object by using the ORAND method, a section for extracting an object by using chromakeys, a section for extracting an object by block matching or affine transform, and the like.

In the first embodiment, a background pixel is determined by using the histogram of the inter-frame differences of the pixel values around the set figure. However, a pixel corresponding to an inter-frame difference equal to or smaller than a threshold value may be simply determined as a background pixel. In addition, in the first embodiment, background pixels (corresponding to difference values equal to or smaller than the predetermined value) are sequentially determined inwardly from the set figure. However, object pixels (corresponding to difference values equal to or larger than the predetermined value) may be sequentially determined outwardly from the figure, or an arbitrary operation sequence may be employed.

The memory 114 receives the result 1105 obtained by tracking/extracting the object, and saves it.

The reason why a better extraction result can be obtained by switching the tracking/extracting methods in accordance with the feature value indicating the properties of a picture will be described below.

If, for example, it is known in advance whether a background moves or not, the corresponding property is preferably used. When the background moves, the motion of the background is compensated, but perfect compensation may not be attained. Almost no motion compensation can be performed for a frame exhibiting a complicated motion. Such a frame can be known in advance from a background motion compensation error, and hence can be excluded from reference frame candidates. If, however, there is no background motion, this processing is not required. If another object is moving, erroneous background motion compensa-

tion may be performed. Alternatively, the corresponding frame may be excluded from reference frame candidates. Even if, therefore, this frame is optimal for reference frame selection conditions, the frame is not selected, resulting in a decrease in extraction precision.

In addition, one picture includes various properties. The motion and texture of an object may partly vary, and hence the object may not be properly extracted by the same tracking/extracting method and apparatus and the same parameters. For this reason, the user preferably designates a portion of a picture which has a special property. Alternatively, differences in a picture may be automatically detected as feature values to extract an object by partly switching tracking/extracting methods, or the parameters may be changed.

When the plurality of object tracking/extracting sections are switched in this manner, the shapes of objects in various pictures can be accurately extracted.

The second example of the moving object tracking/extracting apparatus according to the third embodiment will be described next with reference to the block diagram of FIG. 18.

A figure setting section 120 receives an extracted frame 1201, an initial FIG. 1202 set by the user, and an extraction result 1207 based on the already extracted frame, sets a figure for the extracted frame, and outputs the figure. A second object tracking/extracting section 121 is used to extract an object region by shape prediction such as the block matching method or affine transform. The second object tracking/extracting section 121 receives an extracted frame 1203 in which a figure is set and the extraction result 1207 based on the already extracted frame, and outputs an object tracking/extracting result 1204.

A feature value extraction section 122 receives the object tracking/extracting result 1204, and outputs a feature value 1205 of the object to a switching section 123. The switching section 123 receives the feature value 1205 of the object, and controls inputting the object tracking/extracting result 1204 to the first object tracking/extracting section. Assume that the second object tracking/extracting section 121 tracks/extracts an object shape by the block matching method. In this case, a feature value is regarded as a matching error, and the second object tracking/extracting section 121 outputs a portion exhibiting a small matching error as a predicted shape extraction result. Other feature values include parameters (fractal dimension and the like) representing the luminance gradient or dispersion of each block and texture complicity. When luminance gradient is to be used, input control is performed on the first object tracking/extracting section to use the result obtained by a first object tracking/extracting section 124 using the ORAND method with respect to a block having almost no luminance gradient. In addition, when an edge is detected to use information indicating the presence/absence or intensity of the edge as a feature value, input control is performed on the first object tracking/extracting section so as to use the result obtained by the first object tracking/extracting section 124 with respect to a portion having no edge or having a weak edge. In this manner, switching control can be changed in units of blocks or regions as portions of a picture. Adaptive control can be realized by increasing/decreasing the threshold value for switching.

The first object tracking/extracting section 124 receives the extracted frame 1201, the object tracking/extracting result 1204, and the extraction result 1207 based on the already extracted frame, and outputs a tracking/extracting result 1206 based on the already extracted frame to a

memory 125. The memory 125 receives the tracking/extracting result 1206 based on the extracted frame and saves it.

The third example of the arrangement of the object tracking/extracting apparatus according to the third embodiment will be described next with reference to the block diagram of FIG. 19.

This object tracking/extracting apparatus includes the reference frame selecting section described in the second embodiment in addition to the arrangement shown in FIG. 18. As shown in FIG. 19, the object tracking/extracting apparatus comprises a figure setting section 130, a second object tracking/extracting section 131, a feature value extraction section 132, a switching section 133, a reference frame selector section 134, a first object tracking/extracting section 135, and a memory 136.

The figure setting section 130 receives an extracted frame 1301, an initial FIG. 1302 set by the user, and an extraction result 1308 based on the already extracted frame, sets a figure for the extracted frame, and outputs the figure. The second object tracking/extracting section 131 is used to extract an object region by shape prediction such as the block matching method or affine transform. The second object tracking/extracting section 131 receives an extracted frame 1303 in which a figure is set and the extraction result 1308 based on the already extracted frame, and outputs an object tracking/extracting result 1304.

The feature value extraction section 132 receives the object tracking/extracting result 1304, and outputs a feature value 1305 of the object. The switching section 133 receives the feature value 1305 of the object, and controls inputting the object tracking/extracting result 1304 to the first object tracking/extracting section 135.

The reference frame selector section 134 receives the object tracking/extracting result 1304 to be sent to the first object tracking/extracting section 135 and the extraction result 1308 based on the already extracted frame, and outputs a reference frame 1306.

An example of the features of an object is motion complexity. When the object is to be tracked/extracted by the second object tracking/extracting section 131 using the block matching method, the first object extraction result is output with respect to a portion exhibiting a large matching error. If a portion of the object exhibits a complicated motion, the matching error corresponding to the portion increases. As a result, the portion is extracted by the first object tracking/extracting section 135. Therefore, the reference frame selecting methods to be used by the first object tracking/extracting section 135 are switched in accordance with this matching error as a feature value. More specifically, a reference frame selecting method is selected for only the portion to be extracted by the first object tracking/extracting section 135 instead of the overall object shape so as to satisfy expression (1) or (2) as a selection condition described in the second embodiment.

An example of the feature value of a background includes, for example, information indicating 1) a picture with a still background, 2) zooming operation, and 3) panning operation. The user may input this feature value, or the parameter obtained from the camera may be input as a feature value. The feature value of a background includes a background motion vector, the precision of a picture having undergone background motion compensation, the luminance distribution of the background, texture, edge, and the like. For example, reference frame selecting methods can be controlled in accordance with the precision of a picture having undergone background motion compensation which is

obtained as a feature value from the averaged difference between the picture having undergone background motion compensation and the picture before correction. For example, control is performed such that when the averaged difference is large, the corresponding frame may be excluded from reference frame candidates or a lower priority is assigned to the frame in frame selection. If the background is still or background motion compensation is perfectly performed for all the frames, the difference becomes zero. The same reference frame selecting method as that in the second embodiment can be used.

The first object tracking/extracting section 135 receives the extracted frame 1301, the reference frame 1306, and the extraction result 1308 based on the already extracted frame, and outputs a tracking/extracting result 1307 obtained from the extracted frame by the ORAND method to the first object tracking/extracting section 135. The memory 136 receives the tracking/extracting result 1307 based on the extracted frame, and holds it.

Of the above examples, the arrangement in which a plurality of reference frame selecting sections are switched in accordance with the feature value obtained from the output from the second object tracking/extracting section will be described as the fourth example of the arrangement of this apparatus with reference to FIG. 22.

A figure setting section 160 receives an extracted frame 1601, an initial FIG. 1602 set by the user, and a frame 1608 from which an object has already been extracted, and outputs a set FIG. 1603. A second object tracking/extracting section 161 is used to extract an object region by shape prediction such as the block matching method or affine transform. The second object tracking/extracting section 161 receives the frame 1608 from which the object has already been extracted, and outputs an object tracking/extracting result 1604. A feature value detector section 163 receives the object tracking/extracting result 1604, and outputs a feature value 1605 to a switching section 164. The switching section 164 receives the feature value 1605, and controls inputting the object tracking/extracting result 1604 to the reference frame selecting section.

Each of a plurality of reference frame selector sections 165 receives the object tracking/extracting result 1604 and the frame 1608 from which the object has already been extracted, and outputs at least two reference frames 1606.

A first object tracking/extracting section 166 is used to extract an object by the ORAND method. The first object tracking/extracting section 166 receives the reference frames 1606 and the extracted frame 1601, and outputs an object tracking/extracting result 1607 to a memory 167. The memory 167 receives the object tracking/extracting result 1607 and holds it.

Of the above cases, the case in which background information is obtained, and input control is performed on a plurality of reference frame selecting sections in accordance with the background motion compensation error will be described next.

A figure setting section 170 receives an extracted frame 1701, an initial FIG. 1702 set by the user, and a frame 1710 from which an object has already been extracted, and outputs a set FIG. 1703. A second object tracking/extracting section 171 receives the set FIG. 1703 and the frame 1710 from which the object has already been extracted, and outputs an object tracking/extracting result 1704. A switching section 172 receives background information 1705 designated by the user, and controls inputting the extracted frame 1701 to a background motion compensation section 173.

The background motion compensation section 173 receives the extracted frame 1701 and the frame 1710 from which the object has already been extracted, and outputs a frame 1706 having undergone background motion compensation.

A background feature value detector section 174 receives the extracted frame 1701 and the frame 1706 having undergone background motion compensation, and outputs a background feature value 1707 to a switching section 175. The switching section 175 receives the background feature value 1707, and controls inputting the object tracking/extracting result 1704 to a reference frame selector section 176. The reference frame selector section 176 receives the object tracking/extracting result 1704 and the frame 1710 from which the object has already been extracted, and outputs at least two reference frames 1708.

A first object tracking/extracting section 177 receives at least the two reference frames 1708 and the extracted frame 1701, and outputs an object tracking/extracting result 1709 to a memory 178. The memory 178 receives and holds the object tracking/extracting result 1709.

The fifth example of the arrangement of the object tracking/extracting apparatus according to the third embodiment will be described next with reference to the block diagram of FIG. 20.

An extracted frame output controller section 140 receives a picture 1401 and a sequence 1405 of frames to be extracted, and outputs an extracted frame 1402. A frame sequence controller section 141 receives the information 1405 about the frame sequence input by the user, and outputs a frame sequence 1406. An object tracking/extracting apparatus 142 is an object tracking/extracting method and apparatus for extracting/tracking a target object from a moving picture signal. The object tracking/extracting apparatus 142 receives the extracted frame 1402 and outputs a tracking/extracting result 1403 to a tracking/extracting result output controller section 143. The tracking/extracting result output controller section 143 receives the tracking/extracting result 1403 and the frame sequence 1406, rearranges the frame sequence to match with the picture 1401, and outputs the result.

A frame sequence may be input by the user or may be adaptively determined in accordance with the motion of the object. A frame interval at which the motion of the object can be easily detected is determined to extract the object. More specifically, the frame sequence is controlled to perform object extraction processing in a sequence different from the input frame sequence in such a manner that the frame interval between each reference frame and the current frame subjected to object extraction becomes two or more frames. With this operation, the prediction precision can be increased as compared with the case wherein shape prediction based on inter-frame prediction or ORAND computation is performed in the input frame sequence. In the case of the ORAND method, the extraction precision can be increased by selecting proper reference frames. Therefore, this method is especially effective for a shape prediction method based on inter-frame prediction using block matching or the like.

Depending on the frame interval, the motion becomes too small or complicated to be properly coped with by using the shape prediction method based on inter-frame prediction. If, therefore, a shape prediction error is not equal to or smaller than a threshold value, the prediction precision can be increased by increasing the interval between the current frame and the extracted frame used for prediction. As a result, the extraction precision can also be increased. If there

is a background motion, the background motion between each reference frame and the extracted frame is obtained and compensated. Depending on the frame interval, however, the background motion becomes too small or complicated to accurately perform background motion compensation. In this case as well, the motion compensation precision can be increased by increasing the frame interval. An object shape can be extracted more reliably by adaptively controlling the frame extraction sequence in this manner.

The sixth example of the arrangement of the object tracking/extracting apparatus according to the third embodiment will be described next with reference to the block diagram of FIG. 21.

An extracted frame output controller section 150 receives a picture 1501 and a frame extraction sequence 1505, and outputs an extracted frame 1502. A frame sequence controller section 151 receives information 1505 about the frame sequence input by the user, and outputs a frame sequence 1506. That is, the frame sequence controller section 151 receives the frame interval and determines a frame extraction sequence. Each of a plurality of object tracking/extracting apparatuses 152 is an object tracking/extracting method and apparatus for extracting/tracking a target object from a moving picture signal. Inputting of the extracted frame 1502 to each object tracking/extracting apparatus 152 is controlled in accordance with the frame sequence 1506, and the apparatus outputs a tracking/extracting result 1503. A tracking/extracting result output controller section 153 receives the tracking/extracting result 1503 and the frame sequence 1506, rearranges the frame sequence to match with the picture 1501, and outputs the result.

Skipped frames may be interpolated from already extracted frames, or may be extracted by the same algorithm upon changing the method of selecting reference frame candidates.

An example of the processing performed by the object tracking/extracting apparatus in FIG. 21 will be described below with reference to FIG. 25.

Referring to FIG. 25, the frames indicated by the hatching are future frames to be extracted at two-frame intervals. Skipped frames are extracted by the second object tracking/extracting apparatus. As shown in FIG. 25, after two frames on the two sides of a skipped frame are extracted, the skipped frame may be interpolated on the basis of the extraction result on the two frames, thereby obtaining an object shape. In addition, a parameter such as a threshold value may be changed, or these frames on the two sides of the skipped frame may be added to reference frame candidates to extract the skipped frame by the same method as that used for the frames on the two sides.

Another arrangement of the object tracking/extracting apparatus will be described next with reference to the block diagram of FIG. 24.

A switching section 182 receives background information 1805 designated by the user, and controls inputting an extracted frame 1801 to a background motion correction section 183. The background motion correction section 183 receives the extracted frame 1801 and a frame 1811 from which an object has already been extracted, and outputs a frame 1806 having undergone background motion compensation to a frame 1806. A background feature value detector section 184 receives the extracted frame 1801 and the frame 1806 having undergone background motion compensation, and outputs a feature value 1807. A switching section 187 receives the background feature value 1807, and controls inputting a tracking/extracting result 1804 to a reference frame selector section 188. A figure setting section 180

receives the extracted frame 1801, the frame 1811 from which the object has already been extracted, and an initial FIG. 1802 set by the user, and outputs an extracted frame 1803 on which a figure is set. A second object tracking/extracting section 181 receives the extracted frame 1803 on which the figure is set and the frame 1811 from which the object has already been extracted, and outputs the tracking/extracting result 1804. A feature value detector section 184 receives the tracking/extracting result 1804, and outputs a feature value 1808. A switching section 186 receives the feature value 1808, and controls inputting the tracking/extracting result 1804 to the reference frame selector section 188. The reference frame selector section 188 receives the object tracking/extracting result 1804 and the frame 1811 from which the object has already been extracted, and outputs at least two reference frames 1809.

A first object tracking/extracting section 189 receives at least the two reference frames 1809 and the extracted frame 1801, and outputs an object tracking/extracting result 1810 to a memory 190. The memory 190 holds the object tracking/extracting result 1810.

The following is the flow of processing.

The user roughly surrounds an object to be extracted on an initial frame. A rectangle on a subsequent frame is set by expanding the rectangle surrounding the already extracted object by several pixels in all directions. This rectangle is segmented into a blocks, and each block is matched with a corresponding block of the already extracted block. Then, the shape of the already extracted object is pasted at the corresponding position. The object shape (predicted object shape) obtained by this processing represents an approximate object. If the prediction precision is not equal to or smaller than the threshold value, the prediction precision may be increased by performing prediction again by using another frame.

If the prediction precision is high, all or part of the predicted shape is output as an extraction result without any change. This method can allow both tracking and extraction of the object.

In forming blocks in object tracking and object shape prediction, a rectangle may be segmented in a fixed block size, or hierarchical block matching based on a matching threshold value may be performed. Alternatively, a frame may be segmented in a fixed size, and only the blocks including the object may be used.

In consideration of a case wherein the prediction precision is low, the predicted object shape is expanded by several pixels to correct irregular portions and holes due to prediction errors are corrected. Predicted object shapes are set on all the reference frame candidates by this method. Every time an object is extracted, an object tracking figure for a future frame is newly obtained by using the extracted object, thereby preventing any extraction error. Note that this tracking figure is set to surround the object.

Assume that an object has already been extracted from a frame preceding each extracted frame, and no object has been extracted from the future frame.

Assume that reference frame candidates are five frames that temporally differ, at predetermined intervals, from and precede/follow each frame to be extracted at predetermined intervals. More specifically, reference frame candidates are limited to a total of five frames, e.g., the initial frame, the three frames preceding the current frame, and one frame following the current frame. If, however, the number of previous frames is less than three, the number of future frames is increased accordingly. If there is no future frame, four previous frames are set as candidates.

A proper region is set around an object on each reference frame candidate. The motion of a background between this region and the current frame is detected to eliminate the background in the figure on the reference frame. The background motion is detected by the following method. Matching is performed between the entire region excluding the object and the current frame. A motion vector exhibiting the minimum matching error is determined as the background motion.

Any reference frame exhibiting a large motion vector detection error in canceling a background motion is excluded from candidates to prevent any extraction error that is caused when elimination of a background motion is not proper. In addition, if the number of reference frame candidates decreases, new reference frame candidates may be selected again. If figure setting and background motion elimination have not been performed for a new reference frame candidate, figure setting and background motion elimination must be performed.

If it is known in advance that there is no background motion, this processing is not performed.

Temporary reference frames are set from the reference frame candidates, and each set of reference frames that satisfy expression (1) or (2) in the second embodiment is selected. If any set of all the reference frame candidates does not satisfy either expression (1) or (2), a set having the minimum number of pixels in  $O_f \cap O_t$  may be selected.

Reference frame candidates are preferably combined to select frames that minimize motion vector detection errors in canceling the motion of the background. More specifically, if there are reference frame sets that satisfy expression (1) or (2), for example, a set that exhibits a smaller motion vector detection error in canceling the motion of the background may be selected. If there is no multi-electron beam exposure motion, a frame on which an inter-frame difference can be satisfactorily detected is preferentially selected.

Assume that the object prediction precision is high, and part of the object is output without any change. In this case, a frame that satisfies the condition given by expression (1) or (2) is selected with respect to only a region where an object prediction result is not used as an extraction result.

The processing to be performed when two reference frames are selected will be described below.

When a reference frame is selected, the inter-frame difference between an extracted frame and the reference frame is obtained, and attention is paid to the inter-frame difference in the set figure.

The inter-frame difference is binarized with a set threshold value. The threshold value used for binarization may be constant with respect to a picture, may be changed in units of frames in accordance with the precision of background motion compensation. For example, if the precision of background motion compensation is low, since many unnecessary deviations are produced in the background, the threshold value for binarization is increased. Alternatively, this threshold value may be changed in accordance with the partial luminance gradient or texture of an object or edge intensity. For example, the threshold value for binarization is decreased for a relatively flat region, e.g., a region where the luminance gradient is small or a region wherein the edge intensity is low. In addition, the user may set a threshold value in consideration of the properties of an object.

Any pixel that is located outside the object tracking figure and has a difference value corresponding to an adjacent background region is determined as a background pixel. At the same time, any pixel that is located inside the object tracking figure and has not a difference value corresponding

to an adjacent background region is determined as a pixel other than a background pixel.

No inter-frame difference can be detected in a still region of an object. If, therefore, the inter-frame difference with respect to a frame used for prediction is zero, and the pixel of interest is located inside the object on the frame used for prediction, the pixel is determined as a still region pixel but is not added as a background pixel.

This background pixel corresponds to a background region common to the current frame and one reference frame. At this time, since the boundary between the background region and the remaining portions may become unnatural, the video signal may be filtered to smooth the boundary or eliminate unnecessary noise regions.

When background regions common to the respective reference frames are obtained, a region that is not contained in the two common background regions is detected and extracted as an object region. This result is output for a portion that does not use the object shape predicted in advance to extract the overall object shape. If there is no matching between the portion using the shape obtained from the common backgrounds and the portion using the predicted object shape, filtering can make the output result look nice.

Finally, the extraction sequence is rearranged into the input frame sequence, and the extraction object region is output.

The object shape extraction method and apparatus of the present invention can be used as an input means for object coding in MPEG-4 that has almost been standardized. For example, this MPEG-4 and object extraction technique are applied to a display system for displaying an object shape in the form of a window. Such a display system can be effectively applied to a multipoint conference system. Space savings can be achieved by displaying each person in the form of a person as shown in FIG. 26 rather than by displaying a text material and the person who is taking part in the conference at each point on a display with a limited size using rectangular windows. With the function of MPEG-4, only the person who is speaking can be enlarged and displayed, or the persons who are not speaking can be made translucent, thus making the user feel nice in using the system.

According to the third embodiment of the present invention, unnecessary processing can be omitted and stable extraction precision can be obtained by selecting an object using a method and apparatus in accordance with the properties of a picture. In addition, by removing the limitation associated with a temporal sequence, sufficient extraction precision can be obtained regardless of the motion of an object.

The third embodiment is designed to improve the performance of the first and second embodiments, and each of the arrangements of the first and second embodiments can be properly combined with the arrangement of the third embodiment.

FIG. 27 shows the first example of the arrangement of an object extraction apparatus according to the fourth embodiment of the present invention.

A texture picture 221 sensed by an external camera or read out from a storage medium such as a video disk and input to this object extraction apparatus is input to a recorder unit 222, a switching section 223, and an object extraction circuit 224 using motion compensation. The recorder unit 222 holds the input texture picture 221. For example, the recorder unit 222 is a hard disk or photomagnetic disk used for a personal computer. The recorder unit 222 is required to use the texture



picture 221 again afterward. If the texture picture 221 is recorded on an external storage medium, the recorder unit 222 need not be prepared, and the storage medium is used as the recorder unit 222. In this case, the texture picture 221 need not be input again to the recorder unit 222. A texture picture is generally called a video signal, which is formed by arranging pixels having luminances (Y) expressed as the values "0" to "255" in the raster order (from the upper left pixel of the picture to the right, and from the uppermost line to the lowermost line). This picture is called a texture picture to be discriminated from a shape picture (to be described later). For a texture picture, color differences (U, V, and the like) or colors (R, G, B, and the like) may be used instead of luminances.

On the first frame, a shape picture 225 on which a desired object to be extracted has been independently extracted by the user is input to the object extraction circuit 224 based on motion compensation. The shape picture is generated by arranging pixels in the raster order as in the case of a texture picture, with the pixel value of each pixel belonging to the object being expressed as "255" and the picture value of each of the remaining pixels being expressed as "0".

An embodiment in which a shape picture 25 on the first frame is generated will be described in detail below with reference to FIG. 34.

Assume that there are graphic figures in the background and foreground, and the operator wants to extract an object 226 in the form of a house. The operator traces a contour of the object 226, with a mouse or pen, on a picture 227 displayed on a monitor. A shape picture is obtained by substituting "255" for each pixel inside the contour and "0" for each pixel outside the contour. If the operator draws this contour with great care, the precision of this shape picture becomes high. Even if this precision becomes low to some degree, the precision can be increased by applying a method described in Takashi Ida and Yoko Sambonsugi, "SELF-AFFINE MAPPING SYSTEM FOR OBJECT CONTOUR EXTRACTION (SUMMARY)", Research and Development Center, Toshiba corporation.

FIG. 35 shows a line 228 drawn by the operator and a contour 229 of the object 226. Obviously, in this stage, the correct position of the contour 229 has not been extracted yet, but the contour 229 is shown to indicate the positional relationship with the line 228.

First of all, a block is allocated to contain the line 228. More specifically, when the frame is scanned in the raster order, and the line 228 is detected, i.e., the difference between a pixel value in the shape picture defined by the line 228 and an adjacent pixel value is detected, a block having a predetermined size is set around the corresponding pixel. In this case, if the current block overlaps an already set block, scanning is continued without setting the current block. As a result, blocks can be set such that the respective blocks touch each other without overlapping, as shown in FIG. 36. With this operation alone, portions 230, 231, and 232 are not contained in blocks. For this reason, scanning is performed again to detect contour portions that are not contained in blocks. If such a portion is detected, a block is set around the corresponding pixel. In the second scanning operation, however, even if the current block overlaps an already set block, the current block is set as long as the pixel serving as the center is not contained in the already set block. Referring to FIG. 37, blocks 233, 234, 235, and 236 indicated by the crosshatching are the blocks set by the second scanning operation. The block size may be fixed. However, if the number of pixels surrounded by the line 228 is large, a large block size may be set, and vice versa. In addition, if

the line 228 has few irregular portions, a large block size may be set, and vice versa. Alternatively, a large block size may be set for a picture having a flat graphic figure, and a small block size may be set for a picture having fine graphic figure.

When a block is set at an end of a screen, the block may protrude from the screen. In this case, an end of only this block is cut to form a rectangular block to prevent it from protruding the screen. In this case, a similar block is also set in the form of a rectangle.

The above method is a method of setting blocks on a shape picture.

Subsequently, similar blocks are searched out in units of blocks by using the texture picture. In this case, it is defined that given blocks having different block sizes are similar when one of the blocks is enlarged or reduced to have the same block size as that of the other block, the number of pixels of one block becomes almost equal to that of the corresponding pixels of the other block. For example, a block 238 has a texture picture similar in shape to that of a block 237 in FIG. 38. Likewise, a block 240 is similar to a block 239, and a block 242 is similar to a block 241. In this embodiment, a similar block is set to be larger than a block set on the contour. In searching for similar blocks, it suffices if a search is performed within a given range having four corners defined by blocks 244, 245, 246, and 247 near a block 243, as shown in FIG. 39, instead of the entire screen. FIG. 39 shows a case wherein the centers of the respective blocks are set as start points, and the start points of the blocks 244, 245, 246, and 247 are moved by a predetermined pixel width in all directions with respect to the start point of the block 243. FIG. 40 shows a case wherein a start point is set on the upper left corner of each block.

Any similar block that partly protrudes from a screen is excluded from search targets even if it is located in a search range. If a block is located at an end of a screen, all the similar blocks in a search range may be excluded from search targets. In this case, the search range is shifted to the inside of the screen for the block on the end of the screen.

Similar blocks can be searched out by a multi-step-search with a small computation amount. In this multi-step-search method, a search is performed to check errors first at discrete start points instead of searching the entire search range while shifting the start point in unit of pixels of half pixels. Then, start points only around a start point exhibiting a small error are shifted relatively finely to check errors. This operation is repeated to approach the position of the similar block.

In a search for a similar block, if the similar block is reduced every time, a long processing time is required. If, therefore, the entire picture is reduced in advance, and the resultant data is held in another memory, the above operation can be done by only reading out the data of a portion corresponding to the similar block from the memory.

FIG. 38 shows only the similar blocks for only the three blocks 237, 239, and 241. In practice, however, similar blocks are obtained for all the blocks shown in FIG. 37. The above description is about the method of searching for similar blocks. It should be noted that a search for similar blocks is performed by using a texture picture instead of a shape picture. Considering primary conversion of transferring a similar block to a block within a frame, the contour of the texture picture remains unchanged in this primary conversion.

A method of performing correction to match the contour of a shape picture with that of a texture picture by using the positional relationship between each block and a corresponding similar block will be described next.



Referring to FIG. 41, a contour 228 is the line drawn by the user. It suffices if this line is approximated to a correct contour 229. For this purpose, a portion of the shape picture which corresponds to a similar block 238 is read out, the portion is reduced to the same size as that of a block 237, thereby replacing the corresponding portion of the shape picture which corresponds to the block 237. Since this operation makes the contour approach an invariant set including the fixed point of primary conversion from the similar block to the block, the contour 228 approaches the contour 229. When one side of the similar block is twice as long as one side of the block, one replacing operation reduces the gap between the contour 228 and the correct contour 229 to almost  $\frac{1}{2}$ . FIG. 42 shows a contour 248 obtained by performing this replacing operation once for all the blocks. If this block replacement is repeated, the contour 248 further approaches the correct contour. Eventually, as shown in FIG. 43, the contour 248 coincides with the correct contour. In practice, since there is no need to reduce the gap between the two contours to a value smaller than the distance between pixels, replacing operation is terminated after replacement is performed a certain number of times. This technique is effective when the contour of a texture picture is contained in a  $(N \times N)$ -pixel block set on a shape picture. In this case, the maximum distance between the contour of the shape picture and that of the texture picture is about  $N/2$ . If the length of one side of a similar block is  $A$  times larger than that of one side of a corresponding block, the distance between the two contours is reduced to  $1/A$  per replacement. Letting  $x$  be the number of times replacement is performed, a state wherein the distance becomes smaller than one pixel can be expressed as follows:

$$(N/2) \times (1/A)^x < 1$$

where  $\wedge$  represents the power, i.e.,  $(1/A)$  is multiplied by  $x$  times. From the above inequality,

$$x \log (2/N) / \log (1/A)$$

If, for example,  $N=8$  and  $A=2$

$$x > 2$$

It therefore suffices if replacement is performed three times.

FIG. 30 is a block diagram showing this object extraction apparatus. First of all, a shape picture 249 input by the operator is recorded on a shape memory 250. In the shape memory 250, blocks are set in the manner described with reference to FIGS. 36 and 37. Meanwhile a texture picture 251 is recorded on a texture memory 252. The texture memory 252 sends a texture picture 254 of a block to a search circuit 255 upon referring to position information 253 of the block sent from the shape memory 250. At the same time, similar block candidates are also sent from the texture memory 252 to the search circuit 255, as described with reference to FIGS. 39 and 40. The search circuit 255 reduces each similar block candidate, calculates the error between each candidate and the corresponding block, and determines a candidate exhibiting the minimum error as a similar block. An example of this error is the absolute value sum of luminance value deviations or the value obtained by adding the absolute value sum of color difference deviations thereto. If color differences are also used, the precision can be increased as compared with a case wherein only luminances are used, even though the computation amount increases. This is because, even if the luminance difference is small at the contour of an object, a similar block can be properly determined when the color difference is large. Information

256 about the position of the similar block is sent to a reduction conversion circuit 257. A shape picture 258 on the similar block is also sent from the shape memory 250 to the reduction conversion circuit 257. The reduction conversion circuit 257 reduces the shape picture of the similar block. The reduced similar block is sent back to the shape memory 250 as a shape picture 259 whose contour has been corrected. The shape picture of the corresponding block is then overwritten. When this replacement in the shape memory 250 is performed a predetermined number of times, the corrected shape picture 259 is output to an external unit. The contents of the shape memory 250 may be overwritten in units of blocks. Alternatively, memories corresponding to two frames may be prepared. After the shape picture on the entire frame is copied from one memory to the other memory, the respective blocks on the contour portion may be replaced with the blocks obtained by reducing similar blocks.

This object extraction method will be described with reference to the flow chart of FIG. 48.

(Object Extraction Method Based On Matching of Reduced Blocks in Frames)

In step S31, blocks are set on the contour portion of shape data. In step S32, a similar block having picture data representing a graphic figure that is similar to that of the currently processed block is detected from the same picture data. In step S33, the shape data of the currently processed block is replaced with the data obtained by reducing the shape data of the similar block.

If it is determined in step S34 that the number of processed blocks reaches a predetermined number, the flow advances to step S35. Otherwise, the flow returns to step S32 upon setting the next block as a processing target.

If it is determined in step S35 that the number of times of replacement reaches a predetermined number of times, the flow advances to step S36. Otherwise, the flow returns to step S31 upon setting replaced shaped data as a processing target. In step S36, the shape data having undergone repetitive replacement is output as an object region.

This method is effective when an edge of a block matches with an edge of a similar block. If, therefore, a block has a plurality of edges, the edges do not properly match with each other in some case. Such a block is not replaced, and the input edges are held with any change. More specifically, the shape picture of each block is scanned horizontally and vertically in units of lines. Any block that has at least a predetermined number of lines each having two or more points at each of which a change from "0" to "255" or from "255" to "0" occurs is not replaced. In addition, even on the boundary between an object and a background, the luminance or the like may be uniform depending on the portion. In such a case as well, since no edge correction effect can be expected, any block in which the dispersion value of the texture picture is equal to or smaller than a predetermined value is not replaced, and the input edge is held without being changed.

If the error between a similar block and a corresponding block cannot be reduced to a predetermined value, an attempt to reduce the block may be abandoned, and the similar block may be obtained without any change in size. In this case, a similar block should be selected while the chance of overlapping of blocks is minimized. Although no edge correction effect can be expected from only blocks that are not reduced, when the edges of reduced blocks, whose edges have been corrected by reduction, are copied, the edges of even the blocks that have not been reduced can be indirectly corrected.

The flow chart of FIG. 48 shows the case wherein a shape picture is replaced immediately after a similar block is detected. A method of searching all blocks for similar blocks, and replacing a shape picture in all the blocks by holding the position information about the similar blocks of all the blocks will be described with reference to the flow chart of FIG. 50.

In this case, shape picture replacement can be repeated a plurality of number of times per search for similar blocks.

In step S41, blocks are set on the contour portion of shape data. In step S42, a similar block having picture data representing a graphic figure that is similar to that of the currently processed block is detected from the same picture data. If it is determined in step S43 that the similar block search processing is complete for all the blocks, i.e., the number of processed blocks reaches a predetermined number, the flow advances to step S44. Otherwise, the flow returns to step S42. In step S44, the shape data of the currently processed data is replaced with the data obtained by reducing the shape data of the similar block.

If it is determined in step S45 that replacement processing is complete for all the blocks, i.e., the number of processed blocks reaches a predetermined number, the flow advances to step S46. Otherwise, the flow returns to step S44. If it is determined in step S46 that the number of times all the blocks are replaced reaches a predetermined number of times, the flow advances to step S47. Otherwise, the flow returns to step S44. In step S47, the shape data obtained by repeating replacement/conversion is output as an object region.

A block setting method that can increase the edge correction precision will be described next.

As described above, in the method of setting blocks around the contour of a shape picture, a portion of a correct contour 301 may not be contained in any block, as shown in FIG. 51. In this case, a contour 302 of the shape picture is indicated by the thick line. Assume that an object is located on the lower right side of the contour, and a background is located on the upper left side of the contour. In this case, although a portion 303 that belongs to the background is erroneously set as an object portion, there is no possibility that the portion 303 be corrected, because it is not contained in any block. As described above, if there is a gap between a block and a correct contour, the corresponding portion cannot be properly corrected.

To reduce the gap between a block and a correct contour, a method of overlapping blocks to some extent may be used. In this method, since the number of blocks increases, a gap 304 decreases even though the computation amount increases. The extraction precision therefore increases. In this case, however, the gap is not completely eliminated.

The gap can also be reduced effectively by increasing the block size, as shown in FIG. 53. In this case, the above method of overlapping blocks is also used. In this case, the gap is completely eliminated by this method.

As described above, the contour correction range can be effectively increased by increasing the block size. If, however, the block size is excessively large, the shape of a contour contained in blocks is complicated, resulting in difficulty in detecting similar blocks. Such a case is shown in FIGS. 54A to 54D.

Referring to FIG. 54A, a hatched portion 305 represents an object region, and a white portion 306 represents a background region. A contour 307 of an input shape picture is indicated by the black line. As shown in FIG. 54A, the contour 307 of the shape picture is greatly away from the correct contour, and the correct contour has irregular por-

tions. In contrast to this, FIG. 54B shows the result obtained by arranging blocks by a method different from that described above. In this case, the picture is segmented into rectangular blocks such that the respective blocks do not overlap each other and produce no gap. The dispersion values in the texture picture are calculated in units of blocks. Any block that exhibits a dispersion value smaller than a predetermined value is canceled. In the case shown in FIG. 54B, therefore, only blocks exhibiting dispersion values larger than the predetermined value are left. A similar blocks is obtained for each of these blocks. For example, near a block 308, there is no graphic figure that is twice as large in the vertical and horizontal directions as the block 308. This applies to many other blocks. Even if, therefore, a portion exhibiting the minimum error is selected as a similar block, and the shape picture is repeatedly replaced/converted by using the positional relationship with the selected block, the resultant contour does not match with the correct contour, as shown in FIG. 54C. However, as compared with the contour 307 of the shape picture in FIG. 54A, the irregular portions of the contour of the texture picture is approximately reflected in a contour 309 of the shape picture in FIG. 54C after edge correction (to the extent that a valley is formed between left and right peaks). In this case, if the block size is decreased, even this approximate correction cannot be attained.

As described above, if a large block size is set to extend the correction range, the shape of a contour contained in blocks is complicated, resulting in difficult in detecting similar blocks. Consequently, only approximate edge correction can be performed. In such a case, edge correction is performed first with a large block size, and then edge correction performed upon decreasing the block size in accordance with the correction result. This operation can increase the correction precision. FIG. 54D shows the result obtained by performing correction upon reducing the block size to  $\frac{1}{2}$  that in FIG. 54C in the vertical and horizontal directions, and further performing correction upon reducing the block size to  $\frac{1}{4}$ . If correction is repeated while the block size is gradually decreased in this manner, the correction precision can be increased.

A method of gradually decreasing the block size will be described with reference to the flow chart of FIG. 55.

In step S51, block size  $b=A$  is set. In step S52, edge correction similar to the edge correction shown in FIG. 48 or 50 is performed. In step S53, the block size  $b$  is checked. If the block size  $b$  becomes smaller than  $Z$  ( $Z < A$ ), this processing is terminated. If the block size  $b$  is equal to or larger than  $Z$ , the flow advances to step S54. In step S54, the block size  $b$  is reduced to half, and the flow advances to step S52.

In the above case, a relatively large block size is set first, and correction is repeated while the block size is gradually decreased, thereby increasing the correction precision.

FIG. 56 shows a case wherein each block is tilted through  $45^\circ$  to hinder a gap from being formed between each block and a correct contour. As shown in FIG. 56, if the contour is inclined, the correct contour can be covered with the blocks by tilting the blocks without increasing the block size as much as in the case shown in FIG. 53. In this case, as shown in FIG. 56, the correct contour can be covered without any overlap between the blocks. By tilting sides of the blocks in the same direction as that of the contour of the shape picture in this manner, formation of gaps between the blocks and the correct contour can be suppressed. More specifically, when the inclination of the contour of an alpha picture is detected, and the contour is close to a horizontal

or vertical line, the blocks are directed as shown in FIG. 53. Otherwise, the blocks are tilted as shown in FIG. 56. Whether the contour is close to a horizontal or vertical line is determined by comparing the inclination with a threshold value.

The above description is about object extraction processing for the first frame. This technique is not limited to the first frame of moving picture and can be generally used for still picture. If block setting and a search for similar blocks are performed after each replacement such that when first replacement is performed for a shape picture, block setting and a search for similar blocks are performed again, and second replacement is performed, a better correction effect can be obtained although the computation amount increases.

Since it is preferable that the similar blocks are selected from the portion adjacent thereto, the range in which the similar blocks are searched for had better be changed in accordance with the block size. In other words, when the block size is large, the block searching range is widened. When the block size is small, the block searching range is narrowed.

In the present method, small holes or independent small regions are appeared in the shaping data as errors in the replacement processing for the shaping data. Thus, if the small holes or independent small regions are deleted from the shaping data before the steps S34, S35, S36, S45, S46, S47, S53, the correction accuracy is improved. A method of deleting the small holes or independent small regions can use a process for combining expansion and reducing or a decision-by-majority filter, which is described in Takagi and Shimoda, "Image Analysis Handbook" Tokyo University Press, January 1991, pp. 575-576 and pp. 677.

Alternatively, blocks may be set more easily, as shown in FIG. 49. That is, a frame is simply segmented into blocks, and a search for similar blocks and replacement processing are performed for only blocks containing a contour 228, e.g., a block 2200.

If an input texture picture has been compressed by fractal coding ("PICTURE REGION SEGMENTATION METHOD AND APPARATUS" in Jpn. Pat. Appln. KOKOKU Publication No. 08-329255), the compressed data contains information about similar blocks for the respective blocks. If, therefore, the compressed data is used for the similar blocks for the blocks containing the contour 228, there is no need to search for similar blocks.

The description of the object extraction apparatus for extracting an object from a picture will be continued by referring back to FIG. 27.

An object extraction circuit 242 based on motion compensation generates a shape picture 260 of each of the subsequent frames from the shape picture 25 of the first frame by using the motion vector detected from the texture picture 221.

FIG. 29 shows an example of the object extraction circuit 224 based on motion compensation. The shape picture 225 of the first frame is recorded on a shape memory 261. In the shape memory 261, blocks are set on the entire screen as in the case of a frame 262 in FIG. 45. The texture picture 221 is sent to a motion estimation circuit 264 and recorded on a texture memory 263. A texture picture 265 one frame ahead of the currently processed frame is sent to the motion estimation circuit 264. The motion estimation circuit 264 detects a reference block exhibiting the minimum error from a frame one frame ahead of the currently processed frame in units of the blocks of the currently processed frame. FIG. 45 shows an example of a block 267, and a reference block 268 selected from a frame 266 one frame ahead of the currently

processed frame. In this case, if the error is smaller than a predetermined threshold value, the reference block is set to be larger than the corresponding block. FIG. 45 also shows an example of a reference block 70 twice as large in the vertical and horizontal directions as a block 269.

Referring back to FIG. 29, information 271 about the position of the reference block is sent to a motion compensation circuit 272. A shape picture 273 of the reference block is also sent from the shape memory 261 to the motion compensation circuit 272. In the motion compensation circuit 272, if the reference block is equal in size to the corresponding block, the shape picture of the reference block is kept unchanged. If the reference block is larger in size than the corresponding block, the shape picture of the reference block is reduced and output as the shape picture 260 of the currently processed frame. In addition, for the next frame, the shape picture 260 of the currently processed frame is sent to the shape memory 261, and the shape picture on the entire frame is overwritten.

If each reference block is larger than the corresponding block, and a contour deviates from the correct position, correction can be effectively performed, as described with reference to FIGS. 41 and 42. Therefore, objects can be accurately extracted from all the frames of the moving picture sequence, which follows the shape picture of the first input frame. The present invention therefore eliminates the conventional inconvenience of lacking precision in early frames of a moving picture sequence and when the motion of an object is small.

Object extraction based on inter-frame motion compensation will be described with reference to FIG. 47.

In step S21, a currently processed frame is segmented into blocks. In step S22, a reference block that contains picture data representing a graphic figure similar to that of the currently processed block and has a size larger than that of the currently processed block is searched out from the respective frames or frames from which shape data have already been obtained. In step S23, the subblocks obtained by extracting shape data from the reference block and reducing the data is pasted on the currently processed block.

If it is determined in step S24 that the number of processed blocks reaches a predetermined number, the flow advances to step S25. Otherwise, the next block is set as a processing target, and the flow returns to step S22. In step S25, the pasted shape data is output as an object region.

In this embodiment, the respective frames are the first frames for which shape pictures are provided in advance. In addition, the reference block need not be a frame one frame ahead of the currently processed frame, and any frame from which a shape picture has already been obtained can be used, as described here.

The above description is about object extraction using motion compensation. The object extraction circuit 224 may use a method using inter-frame difference images as disclosed in "OBJECT TRACKING/EXTRACTING APPARATUS FOR MOVING PICTURE", Jpn. Pat. Appln. KOKAI Publication No. 10-001847 filed previously, as well as the method described above.

The description of the object extraction apparatus for extracting an object from moving picture according to this embodiment will be continued by referring back to FIG. 27.

The shape picture 260 is sent to a switching section 223 and a switching section 281. When the shape picture 260 is "0" (background), the switching section 223 sends the texture picture 221 to a background memory 274 to be recorded thereon. When the shape picture 260 is "255" (object), the texture picture 221 is not sent to the background

memory 274. When this processing is performed for several frames, and the shape picture 260 is accurate to some degree, a picture that contains no object but contains only a background portion is generated in the background memory 274.

A texture picture 275 is sequentially read out again from the recorder unit 222, starting from the first frame, or only frames from which the object designated by the operator is to be extracted are read out and input to a difference value 276. At the same time, a background picture 277 is read out from the background memory 274 and input to the difference value 276. The difference value 276 obtains a difference value 278 between pixels of the texture picture 275 and background picture 277 which are located at the same positions within frames. The difference value 278 is then input to an object extraction circuit 279 using a background picture. The object extraction circuit 279 generates a shape picture 280. This picture is generated by regarding each pixel larger than the threshold value predetermined by the absolute value of the difference value 278 as a pixel belonging to the object to allocate the pixel value "255" to it, and regarding other pixels as pixels belonging to the background to allocate the pixel value "0" to each of them. If color difference and color are to be used for the texture picture as well as luminance, the sum of the absolute values of the deviations between the respective signals is compared with a threshold value to determine whether each pixel is an object or background pixel. Alternatively, a threshold value is determined for each luminance or color difference. If the absolute value of the difference between luminance or color difference values is larger than the threshold value, the corresponding pixel is determined as an object pixel. Otherwise, the corresponding pixel is determined as a background pixel. The shape picture 280 generated in this manner is sent to a switching section 281. In addition, a selection signal 282 determined by the operator is externally input to the switching section 281. The switching section 281 selects either the shape picture 260 or the shape picture 280 in accordance with this selection signal 282. The selected picture is output as a shape picture 283 to an external unit. The operator displays each of the shape pictures 260 and 280 on a display or the like, and selects the more accurate one. Alternatively, the processing time can be saved as follows. The operator displays the shape picture 260 when it is generated. If this picture does not have a satisfactory precision, the shape picture 280 is generated. If the shape picture 260 has a satisfactory precision, the operator outputs the shape picture 260 as the shape picture 283 to the external unit without generating the shape picture 280. Selection may be performed in units of frames or moving picture sequences.

An object extraction method corresponding to the object extraction apparatus in FIG. 27 will be described with reference to the flow chart of FIG. 46.

(Object Extraction Method Using Background Picture)

In step S11, motion compensation is performed for the shape data on each input frame to generate shape data on each frame. In step S12, the picture data of the background region determined by the shaped data is stored as a background picture in the memory.

If it is determined in step S13 that the number of processed frames reaches a predetermined number, the flow advances to step S14. Otherwise, the next frame is set as a processing target, and the flow returns to step S11. In step S14, each pixel where the absolute value of the difference between the picture data and the background picture is large is determined as a pixel belonging to the object region, and other pixels are determined as pixels belonging to the background region.

In this embodiment, when, for example, the camera used for image sensing moves, the background moves. In this case, the motion of the overall background (global motion vector) is detected from a previous frame. In the first scanning, the background is shifted from the previous frame by the global motion vector and stored in the background memory. In the second scanning, the portion shifted from the previous frame by the global motion vector is read out from the background memory. If the global motion vector detected in the first scanning is recorded on the memory, and is read out in the second scanning, the time required to obtain the global motion vector can be saved. In addition, if, for example, the camera is fixed, and it is known in advance that the background is still, the operator operates the switch to inhibit the detection of a global motion vector so as to keep the global motion vector zero. This can further save the processing time. When a global motion vector is to be obtained with a precision of half pixel, the pixel density of a picture input to the background memory is doubled in the vertical and horizontal directions. That is, the pixel values of an input picture are alternately written in the background memory. If, for example, the background moves by 0.5 pixel in the horizontal direction on the next frame, the pixel values are alternately written between the previously written pixels. With this operation, at the end of the first scanning, some pixels may not be written even once in the background picture. In this case, the corresponding gaps are filled with pixels interpolated from neighboring pixels that have been written.

No pixel value is recorded on the background memory to substitute for a portion that is not written even once as a background region portion throughout the moving picture sequence even at the end of the first scanning regardless of whether a half-pixel motion vector is used or not. In the second scanning, such an undefined portion is always determined as an object portion. For this operation, the operator need not prepare a memory for storing an undefined portion and determine whether a given portion is undefined or not. Instead of this, the background memory may be initialized first with a pixel value  $(Y, U, V) = (0, 0, 0)$  that is expected to rarely appear in the background, and then the first scanning may be started. Since this initial pixel value is left in an undefined pixel, the pixel is automatically determined as an object pixel in the second scanning.

According to the above description, a background picture is to be generated in the background memory, even a pixel for which a background pixel value has already been substituted is overwritten with a pixel value as long as it belongs to the background region. In this case, the pixel values of the background in the late period of the moving picture sequence are recorded on the background memory in correspondence with every background portion regardless of whether it corresponds to the early or late period of the moving picture sequence. If the pixel values of the background in the early period of the moving picture sequence are completely the same as those in the late period, no problem arises. If, however, the camera moves very slowly or the brightness of the background gradually changes, and the pixel values slightly vary among frames, the pixel values of the background in the early period of the moving picture sequence greatly differ from those in the late period. If, therefore, this background memory is used, even a background portion is erroneously detected as an object portion in early frames of the moving picture sequence. For this reason, only the pixels that have not been defined even once as pixels belonging to the background region in the previous frames and are defined as pixels belonging to the back-

ground region for the first time in the currently processed frame are written in the background memory, and the pixels for which background pixel values have already been substituted are not overwritten. With this operation, since the pixel values of the background in the early period of the moving picture sequence are recorded on the background memory, an object can be properly extracted. When the background region of the currently processed frame is overwritten in the background memory in the second scanning in accordance with the object extraction result, the background of the currently processed process and the background of the frame immediately preceding the currently processed frame, which exhibit a high correlation, are compared with each other, thereby suppressing erroneous detection of the corresponding portion as an object portion. Overwriting in the second scanning is effective when the background slightly varies. If, therefore, the operator operates a switch to indicate that there is no background motion, overwriting is not performed. This switch may be commonly used as a switch for choosing between detecting a global motion vector or not detecting it.

Since the first scanning is performed to generate a background picture, all the frames need not necessarily be used. Even if skipping is performed every one or two frames, almost the same background picture can be obtained, and the processing time can be shortened.

If only the pixels, of the pixels belonging to the background region, which exhibit inter-frame differences equal to or smaller than a threshold value are recorded on the background memory, it prevents other objects entering the screen from being recorded on the background memory. If the object region detected is erroneously detected at a position closer to the object side than the actual position in the first scanning, the corresponding pixel values of the object are recorded on the background memory. For this reason, even pixels belonging to the background region are not input to the background memory if the pixels are located near the object region.

When only a background picture from which a person and the like belonging to a foreground are removed is required as in a picture photographed in a sightseeing area, the background picture recorded on the background memory is output to the external device.

The above description is about the first example of the arrangement of this embodiment. According to this example, a high extraction precision can be obtained not only in the late period of a moving picture sequence but also in the early period of the moving picture sequence. In addition, an object can be properly extracted even if the object moves little or does not move.

An example of how the generated shape picture 280 is corrected will be described next with reference to FIG. 28. Since this processing is the same as that described with reference to FIG. 27 up to the step of generating the shape picture 280, a description of the processing up to this step will be omitted.

The shape picture 280 is input to an edge correction circuit 284 using a background palette. In addition, the texture picture 275 is input to the edge correction circuit 284 using the background palette and an edge correction circuit 285 using reduced block matching. FIG. 31 is a block diagram showing the detailed arrangement of the edge correction circuit 284.

Referring to FIG. 31, the shape picture 280 is input to a correction circuit 286, and the texture picture 275 of the same frame is input to a comparator circuit 287. A background color 289 is read out from a memory 288 holding the

background palette and input to the comparator circuit 287. In this case, the background palette is a set of combinations of luminances (Y) and color differences (U, V) existing in the background portion, i.e., vectors:

(Y1, U1, V1)  
(Y2, U2, V2)  
(Y3, U3, V3)  
...

and is prepared in advance. More specifically, the background palette is a set of combinations of Y, U, and V of pixels belonging to the background region in the first frame. If, for example, Y, U, V each take 256 values, the number of combinations of these values becomes enormous, and the computation amount for the processing to be described later becomes large. For this reason, the values of Y, U, and V are quantized with a predetermined step size to limit the number of combinations. This is because some different vector values before quantization may become the same after quantization.

The comparator circuit 287 checks whether the vector obtained by quantizing Y, U, and V of each pixel of the texture picture 275 coincides with any one of the vectors sequentially sent from the memory 288 and registered in the background palette, i.e., any one of the background colors 289. A comparison result 290 obtained by checking whether the color of each pixel coincides with any of the background colors is sent from the comparator circuit 287 to the correction circuit 286. If the comparison result 290 indicates a background color, the correction circuit 286 replaces the pixel value of the pixel with "0" (background) and outputs it as a corrected shape picture 291 regardless of whether the pixel value of the corresponding pixel of the shape picture 280 is "255" (object). With this processing, when an object region protrudes into a background region in the shape picture 280 and is erroneously extracted, the background region can be properly separated. If, however, the background and the object have a common color, and this color of the object is also registered in the background palette, the portion corresponding of the object which corresponds to the registered color is also determined as a background portion. For this reason, the above palette is set as a temporary palette for the background in the first frame, and an object palette for the first frame is also generated by the same method as described above. Then, any color in the temporary palette for the background which is also included in the object palette is removed from the temporary palette for the background, and the resultant palette is used as a background palette. This can prevent any portion of the object from being determined as a background portion.

In consideration of a case wherein an error is included in a shape picture input for the first frame, pixels near the edge of the shape picture may not be used to generate a palette. In addition, the occurrence frequency of each vector may be counted, and any vector whose frequency is equal to or lower than a predetermined frequency may not be registered in the palette. If the quantization step size is excessively small, the processing time is prolonged or even a color very similar to a background color may not be determined as a background color because of the slight difference between the vector values. In contrast to this, if the quantization step size is excessively large, the number of vectors common to the background and the object increases too much. For this reason, several quantization step sizes are tried for the first frame, and a quantization step size that separates the background and object colors from each other as in the case of an input shape picture is selected.

In addition, since a new color may appear in the background or object in this process, the background palette may be updated in some frame.

Referring back to FIG. 28, the shape picture 291 is input to the edge correction circuit 285. Since the edge correction circuit 285 is identical to the circuit that receives the shape picture 249 equivalent to the shape picture 291 and the texture picture 251 equivalent to the texture picture 275 in the circuit shown in FIG. 30, a description thereof will be omitted. This circuit corrects a shape picture such that the edge of the shape picture coincides with the edge of the corresponding texture picture. A corrected shape picture 292 is sent to the switching section 281. A shape picture 293 selected from the shape pictures 292 and 260 is output from the switching section 281.

In this case, the edge correction circuits are arranged on the subsequent stage of the object extraction circuit 279. If these correction circuits are arranged on the subsequent stage of the object extraction circuit 224, the precision of the shape picture 260 can be increased.

In some rare cases, the extraction precision is decreased by edge correction. If the shape picture 280 and the shape picture 291 are also input to the switching section 281 in the circuit shown in FIG. 28 to prevent the degraded shape picture 292 from being output, the shape picture 280 for which no edge correction has been performed or the shape picture 291 for which only edge correction using a background palette has been performed can be selected.

FIG. 44 shows pixels corresponding to background colors registered in the background palette by the crosshatching. If the information in FIG. 44 is used in a search for similar blocks, which has been described with reference to FIGS. 30 and 29, the contour extraction precision can be further increased. When a graphic figure exists in a background, similar blocks may be selected along the edge of the graphic figure in the background instead of the edge between the object and the background. In such a case, in calculating the errors between the blocks and the blocks obtained by reducing the similar blocks, if both corresponding pixels have the same background color, the error between these pixels is not included in the calculation result. This prevents occurrence of an error even if the edge of the graphic figure in the background deviates. Therefore, similar blocks are properly selected such that the edge of the object matches with that of the background.

FIG. 32 shows an example of an image synthesizing apparatus incorporating an object extraction apparatus 294 of this embodiment. A texture picture 295 is input to a switching section 296 and the object extraction apparatus 294. A shape picture 2100 of the first frame is input to the object extraction apparatus 294. The object extraction apparatus 294 has the same arrangement as that shown in FIG. 27 or 28. The object extraction apparatus 294 generates a shape picture 297 of each frame and sends it to the switching section 296. A background picture 299 for synthesis is held in a recording circuit 298 in advance. The synthesis background picture 299 of the currently processed frame is read out from the recording circuit 298 and sent to the switching section 296. When a pixel of the shape picture has the pixel value "255" (object), the switching section 296 selects the texture picture 295 and outputs it as a synthetic picture 2101. When a pixel of the shape picture has the pixel value "0" (background), the switching section 296 selects the synthesis background picture 299 and outputs it as the synthetic picture 2101. With this operation, a picture is generated by synthesizing the object in the texture picture 295 with the foreground of the synthesis background picture 299.

FIG. 33 shows another example of edge correction. Assume that one of the blocks set as shown in FIG. 33 is a block 2102 in FIG. 33. In this case, blocks are separately set in the object region and the background region with a contour serving as a boundary. Blocks 2103, 2104, 2105, and 2106 are obtained by shifting this contour in the lateral direction. These blocks are shifted by different widths in different directions. The separation degree described on page 1408 in Fukui "Object Contour Extraction Based on Separation Degrees between Regions", THE TRANSACTIONS OF THE IEICE, D-II, Vol. J80-D-II, No. 6, pp. 1406-1414, June 1997 is obtained for each contour, and one of the contours corresponding to the blocks 2102 to 2106 which exhibits the maximum separation degree is used. With this operation, the contour of the shape picture matches with the edge of the texture picture.

As has been described above, according to the fourth embodiment, a high extraction precision can be obtained not only in the late period of a moving picture sequence but also in the early period. In addition, even if an object moves slightly or does not move, the object can be properly extracted. Furthermore, even if the contour of an object region input as shape data deviates, the position of the contour can be corrected by reducing the shape data of a similar block larger than the currently processed block and pasting the reduced data. With this operation, by only providing data obtained by approximately tracing the contour of the object region as shape data, object regions on all the subsequent input frames can be extracted with high precision.

Note that the first and fourth embodiments can be properly combined and used. In addition, all the procedures for the object extraction methods of the first to fourth embodiments can be implemented by software. In this case, the same effects as those of the first to fourth embodiments can be obtained by only installing computer programs for executing these procedures in a general computer through a recording medium.

As described above, according to the present invention, a target object can be accurately extracted/tracked without any influences of excess motions around the target object by tracking the object using a figure surrounding the object.

In addition, a high extraction precision can be obtained regardless of input pictures. Furthermore, a high extraction precision can be obtained not only in the late period of a moving picture sequence but also in the early period. Moreover, even if an object moves slightly or does not move, the object can be properly extracted.

Additional advantages and modifications will readily occur to those skilled in the art. Therefore, the invention in its broader aspects is not limited to the specific details and representative embodiments shown and described herein. Accordingly, various modifications may be made without departing from the spirit or scope of the general inventive concept as defined by the appended claims and their equivalents.

What is claimed is:

1. An object extraction apparatus for a moving picture, comprising:
  - a background region determination section which determines a first background region common to a current frame and a first reference frame, and a second background region common to the current frame and a second reference frame, the current frame containing a target object to be extracted from a moving picture signal, the first reference frame being temporally different from the current frame on the basis of a differ-

ence between the current frame and the first reference frame, the second reference frame being temporally different from the current frame on the basis of a difference between the current frame and the second reference frame, and the first background region and the second background region indicating a background in the moving picture; and

an extraction section which extracts a region, in a picture on the current frame, which belongs to neither the first background region nor the second background region as an object region.

2. The apparatus according to claim 1, which comprises a still object determination section which determines pixels of the current frame as the object region when pixels of one of the first and second reference frames belongs to the object region, and which determines the pixels of the current frame as the background region when the pixels of one of the first and second reference frames belongs to the background region, using a predetermined shape of the object of the one of the first and second reference frames in a case that the difference between the pixels of the current frame and the pixels of the one of the first and second reference frames is small.

3. The apparatus according to claim 2, wherein the still object determination section uses the predetermined shape of the object, when the shape of the object of the one of the first and second reference frames has already been extracted, and a shape of the object of one of the first and second reference frames which is created from the frame, from which the shape of the object has been extracted, by a block matching method, when the object region is not extracted.

4. The apparatus according to claim 1, further comprising a background correction section which corrects motion of a background on each of the first and second reference frames or the current frame such that the motion of the background between each of the first and second reference frames and the current frame becomes relatively zero.

5. The apparatus according to claim 1, wherein the background region determination section includes a determination section which determines the common background region using a predetermined threshold value.

6. The apparatus according to claim 5, wherein the background region determination section includes a setting on which sets the threshold value to a larger value than the predetermined threshold value when the difference of the current frame is larger than a predetermined value, and to a smaller value than it when the difference is smaller.

7. The apparatus according to claim 5, wherein the background region determination section includes a dividing section which divides the current frame into a plurality of regions, which measures a difference between each of the regions and each of corresponding regions of one of the first and second reference frames, and which sets the threshold value to a larger value than a predetermined value when the difference is larger than a predetermined value and to a smaller value when it is smaller.

8. The apparatus according to claim 1, further comprising a prediction section which predicts a position or shape of the object on the current frame from a frame from which the object region has already been extracted, and a selection section which selects the first and second reference frames to be used by said background region determination section on the basis of the position or shape of the object on the current frame which is predicted by said prediction section.

9. The apparatus according to claim 1, wherein said apparatus further comprises an initial figure setting section which sets a figure surrounding the target object on an initial

frame of the moving picture signal, and a figure setting section which sets on one of the first and second reference frames a figure surrounding a region on each input frame of the moving picture signal which corresponds to an image inside figure of one of the first and second reference frames that temporally differs from the input frame on the basis of a correlation between the input frame and the image inside figure, and said object region extraction section extracts a region, in the image inside figure, which belongs to neither the first background region nor the second background region as an object region.

10. The apparatus according to claim 8, wherein said initial figure setting section sets a figure surrounding the target object on the basis of an external input.

11. An object extraction apparatus for a moving picture comprising:

an initial figure setting section which sets a figure surrounding a target object on an input frame of a moving picture signal;

a figure setting section which sets, on the input frame, a figure surrounding a region on the input frame of the moving picture signal and corresponding to an image inside figure of a reference frame that temporally differs from the input frame for each input frame on the basis of a correlation between the input frame and the image inside figure;

a background region determination section which determines a first background region common to a current frame as an object extraction target and a first reference frame and a second background region common to the current frame and a second reference frame, the first reference frame being temporally different from the current frame on the basis of a difference between the current frame and the first reference frame, the second reference frame being temporally different from the current frame on the basis of a difference between the current frame and the second reference frame, and the first background region and the second background region indicating a background in the moving picture;

a first object extraction section which extracts a region, in the image inside figure of the current frame, which belongs to neither the first background region nor the second background region, as an object region;

a second object extraction section which extracts an object region from the image inside figure on the current frame as the object extraction target by using a method different from that used by said first object extraction section; and

a switching section which selectively switches the first and second object extraction sections.

12. The apparatus according to claim 11, which further comprises a feature extraction section which extracts a feature value of a picture in at least a partial region of the current frame as the object extraction target from the current frame, and wherein said switching section selectively switches said first and second object extraction sections on the basis of the extracted feature value.

13. The apparatus according to claim 11, wherein said second object extraction section includes a prediction section which uses a frame, from which the object region has already been extracted, as a reference frame, to predict a position or shape of the object on the current frame as the object extraction target from the reference frame.

14. The apparatus according to claim 13, wherein said first and second object extraction sections are selectively switched and used in units of blocks of each frame on the