BAR CODE LABEL

U.S. PATENT APPLICATION

SERIAL NUMBER	FILING DATE	CLASS	GROUP ART UNIT
08/411,369	03/27/95	382	2613

APPLICANT

GIORDANO BERETTA, PALO ALTO, CA; VASUDEV BHASKARAN, MOUNTAIN VIEW, CA; KONSTANTINOS KONSTANTINIDES, SAN JOSE, CA.

CONTINUING DATA*****************
VERIFIED

_____

FOREIGN/PCT APPLICATIONS*********
VERIFIED

_____

FOREIGN FILING LICENSE GRANTED 04/26/95

STATE OR COUNTRY	SHEETS DRAWING	TOTAL CLAIMS	INDEPENDENT CLAIMS	FILING FEE RECEIVED	ATTORNEY DOCKET NO.
CA	7	36	4	$1,158.00	10940893

ADDRESS

RECORDS MANAGER
LEGAL DEPARTMENT 20BO
HEWLETT PACKARD COMPANY
P O BOX 10301
PALO ALTO CA 94303-0890

TITLE

TEXT AND IMAGE SHARPENING OF JPEG COMPRESSED IMAGES IN THE FREQUENCY DOMAIN

This is to certify that annexed hereto is a true copy from the records of the United States Patent and Trademark Office of the application which is identified above.

By authority of the
COMMISSIONER OF PATENTS AND TRADEMARKS

Date   Certifying Officer

BAR CODE LABEL	U.S. PATENT APPLICATION		

SERIAL NUMBER	FILING DATE	CLASS	GROUP ART UNIT
08/411,369	03/27/95	382	2608

APPLICANT

GIORDANO BERETTA, PALO ALTO, CA; VASUDEV BHASKARAN, MOUNTAIN VIEW, CA; KONSTANTINOS KONSTANTINIDES, SAN JOSE, CA.

CONTINUING DATA******************
VERIFIED

FOREIGN/PCT APPLICATIONS**********
VERIFIED

FOREIGN FILING LICENSE GRANTED 04/26/95

STATE OR COUNTRY	SHEETS DRAWING	TOTAL CLAIMS	INDEPENDENT CLAIMS	FILING FEE RECEIVED	ATTORNEY DOCKET NO.
CA	7	36	4	$1,158.00	10940893

ADDRESS

MARGER JOHNSON MCCOLLOM STOLOWITZ
1030 SW MORRISON ST
PORTLAND OR 97205

TITLE

TEXT AND IMAGE SHARPENING OF JPEG COMPRESSED IMAGES IN THE FREQUENCY DOMAIN

This is to certify that annexed hereto is a true copy from the records of the United States Patent and Trademark Office of the application which is identified above.

By authority of the
COMMISSIONER OF PATENTS AND TRADEMARKS

Date _____ Certifying Officer _____

08/411369
PATENT APPLICATION

ATTORNEY DOCKET NO. __1094893-1__

IN THE U.S. PATENT AND TRADEMARK OFFICE
Patent Application Transmittal Letter

COMMISSIONER OF PATENTS AND TRADEMARKS
Washington, D.C. 20231

Sir:

Transmitted herewith for filing under 37 CFR 1.53(b) is a(n): (X) Utility   ( ) Design

                (X) original patent application,

                ( ) continuing application,

                    ( ) continuation-in-part

                    ( ) continuation or   ( ) divisional

              of S/N_____ filed _____.

INVENTOR(S):   Giordano Beretta et al.

TITLE: TEXT AND IMAGE SHARPENING OF JPEG COMPRESSED IMAGES IN THE FREQUENCY DOMAIN

Enclosed are:

(X) The Declaration and Power of Attorney.   (X) signed   ( ) unsigned or partially signed

(X) _7_ sheets of (X) formal drawings ( ) informal drawings   (one set)

( ) Information Disclosure Statement and Form PTO-1449   ( ) Associate Power of Attorney

( ) Priority document(s) ( ) _(Other)_____ (fee $_____)

CLAIMS AS FILED BY OTHER THAN A SMALL ENTITY				
(1) FOR	(2) NUMBER FILED	(3) NUMBER EXTRA	(4) RATE	(5) TOTALS
TOTAL CLAIMS	36 — 20	16	X $ 22	$ 352
INDEPENDENT CLAIMS	4 — 3	1	X $ 76	$ 76
ANY MULTIPLE DEPENDENT CLAIMS	0		$ 240	$ 0
BASIC FEE: Design ($300.00); Utility ($730.00)				$ 730
TOTAL FILING FEE				$ 1,158
OTHER FEES				$ 0
TOTAL CHARGES TO DEPOSIT ACCOUNT				$ 1,158

Charge $ __1,158__ to Deposit Account 08-2025. At any time during the pendency of this application, please charge any fees required or credit any over payment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.19, 1.20 and 1.21. A duplicate copy of this sheet is enclosed.

"Express Mail" label no. __TB396300465__

    Date of Deposit __March 27, 1995__

I hereby certify that this is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Commissioner of Patents and Trademarks, Washington, D.C. 20231.

By _____

Typed Name: Linda A. Iimura

Respectfully submitted,

Giordano Beretta et al.

By _____

C. Douglass Thomas

Attorney/Agent for Applicant(s)
Reg. No. 32,947

Date: March 27, 1995

Telephone No.: (415) 857-8129

TEXT AND IMAGE SHARPENING OF JPEG COMPRESSED IMAGES IN THE FREQUENCY DOMAIN

RELATED APPLICATION DATA

This application incorporates subject matter disclosed in commonly-assigned application entitled METHOD FOR SELECTING JPEG QUANTIZATION TABLES FOR LOW BANDWIDTH APPLICATIONS, Ser. No. 08/936517 filed on even date herewith.

BACKGROUND OF THE INVENTION

This invention relates to data compression using the JPEG compression standard for continuous - tone still images, both grayscale and color.

A committee known as "JPEG," which stands for "Joint Photographic Experts Group," has established a standard for compressing continuous-tone still images, both grayscale and color. This standard represents a compromise between reproducible image quality and compression rate. To achieve acceptable compression rates, which refers to the ratio of the uncompressed image to the compressed image, the JPEG standard adopted a lossy compression technique. The lossy compression technique was required given the inordinate amount of data needed to represent a color image, on the order of 10 megabytes for a 200 dots per inch (DPI) 8.5" × 11" image. By carefully implementing the JPEG standard, however, the loss in the image can be confined to imperceptible areas of the image, which produces a perceptually loss less uncompressed image. The achievable compression rates using this technique are in the range of 10:1 to 50:1.

Figure 1 shows a block diagram of a typical implementation of the

JPEG compression standard. The block diagram will be referred to as a compression engine. The compression engine 10 operates on source image data, which represents a source image in a given color space such as CIELAB. The source image data has a certain resolution, which is determined by how the image was captured. Each individual datum of the source image data represents an image pixel. The pixel further has a depth which is determined by the number of bits used to represent the image pixel.

The source image data is typically formatted as a raster stream of data. The compression technique, however, requires the data to be represented in blocks. These blocks represent a two-dimensional portion of the source image data. The JPEG standard uses 8x8 blocks of data. Therefore, a raster-to-block translation unit 12 translates the raster source image data into 8x8 blocks of source image data. The source image data is also shifted from unsigned integers to signed integers to put them into the proper format for the next stage in the compression process. These 8x8 blocks are then forwarded to a discrete cosine transformer 16 via bus 14.

The discrete cosine transformer 16 converts the source image data into transformed image data using the discrete cosine transform (DCT). The DCT, as is known in the art of image processing, decomposes the 8x8 block of source image data into 64 DCT elements or coefficients, each of which corresponds to a respective DCT basis vector. These basis vectors are unique 2-dimensional (2D) "spatial waveforms," which are the fundamental units in the DCT space. These basis vectors can be intuitively thought to represent unique images, wherein any source image can be decomposed into a weighted sum of these unique images. The discrete cosine transformer uses the forward discrete cosine (FDCT) function as shown below, hence the name.

$$Y[k,l] = \frac{1}{4}C(k) \bullet C(l)\left[\sum_{x=0}^{7}\sum_{y=0}^{7}S(x,y) \bullet \cos\frac{(2x+1)k\pi}{16}\cos\frac{(2y+1)l\pi}{16}\right]$$

where: $C(k), C(l) = 1/\sqrt{2}$ for $k,l = 0$; and

5           $C(k), C(l) = 1$ otherwise

The output of the transformer 16 is an 8x8 block of DCT elements or coefficients, corresponding to the DCT basis vectors. This block of transformed image data is then forwarded to a quantizer 20 over a bus 18. The quantizer 20 quantizes the 64 DCT elements using a 64-element quantization table 24, which must be specified as an input to the compression engine 10. Each element of the quantization table is an integer value from one to 255, which specifies the stepsize of the quantizer for the corresponding DCT coefficient. The purpose of quantization is to achieve the maximum amount of compression by representing DCT coefficients with no greater precision than is necessary to achieve the desired image quality. Quantization is a many-to-one mapping and, therefore, is fundamentally lossy. As mentioned above, quantization tables have been designed which limit the lossiness to imperceptible aspects of the image so that the reproduced image is not perceptually different from the source image.

The quantizer 20 performs a simple division operation between each DCT coefficient and the corresponding quantization table element. The lossiness occurs because the quantizer 20 disregards any fractional remainder. Thus, the quantization function can be represented as shown in Equation 2 below.

$$Y_Q[k,l] = \text{Integer Round}\left(\frac{Y[k,l]}{Q[k,l]}\right)$$

where Y(k,l) represents the (k,l)-th DCT element and Q(k,l) represents the corresponding quantization table element.

To reconstruct the source image, this step is reversed, with the quantization table element being multiplied by the corresponding quantized DCT coefficient. The inverse quantization step can be represented by the following expression:

$$Y'[k, l] = Y_Q[k, l] \, Q_E[k, l].$$

As should be apparent, the fractional part discarded during the quantization step is not restored. Thus, this information is lost forever. Because of the potential impact on the image quality of the quantization step, considerable effort has gone into designing the quantization tables. These efforts are described further below following a discussion of the final step in the JPEG compression technique.

The final step of the JPEG standard is an entropy encoding, which is performed by an entropy encoder 28. The entropy encoder 28 is coupled to the quantizer 20 via a bus 22 for receiving the quantized image data therefrom. The entropy encoder achieves additional lossless compression by encoding the quantized DCT coefficients more compactly based on their statistical characteristics. The JPEG standard specifies two entropy coding methods: Huffman coding and arithmetic coding. The compression engine of Fig. 1 assumes Huffman coding is used. Huffman encoding, as is known in the art, uses one or more sets of Huffman code tables 30. These tables may be predefined or computed specifically for a given image. Huffman encoding is a well known encoding technique that produces high levels of lossless compression. Accordingly, the operation of the entropy encoder 28 is not further described.

Referring now to Fig. 2, a typical JPEG compressed file is shown

generally at 34. The compressed file includes a JPEG header 36, the quantization (Q) tables 38 and the Huffman (H) tables 40 used in the compression process, and the compressed image data 42 itself. From this compressed file 34 a perceptually indistinguishable version of the original source image can be extracted when an appropriate Q-table is used. This extraction process is described below with reference to Fig. 3.

A JPEG decompression engine 43 is shown in Fig. 3. The decompression engine essentially operates in reverse of the compression engine 10. The decompression engine receives the compressed image data at a header extraction unit 44, which extracts the H tables, Q tables, and compressed image data according to the information contained in the header. The H tables are then stored in H tables 46 while the Q tables are stored in Q tables 48. The compressed image data is then sent to an entropy decoder 50 over a bus 52. The Entropy Decoder decodes the Huffman encoded compressed image data using the H tables 46. The output of the entropy decoder 50 are the quantized DCT elements.

The quantized DCT elements are then transmitted to an inverse quantizer 54 over a bus 56. The inverse quantizer 54 multiplies the quantized DCT elements by the corresponding quantization table elements found in Q tables 48. As described above, this inverse quantization step does not yield the original source image data because the quantization step truncated or discarded the fractional remainder before transmission of the compressed image data.

The inverse quantized DCT elements are then passed to an inverse discrete cosine transformer (IDCT) 57 via bus 59, which transforms the data back into the time domain using the inverse discrete cosine transform (IDCT). The inverse transformed data is then transferred to block-to-raster translator 58 over a bus 60 where the blocks of DCT elements are translated into a raster string of decompressed source image
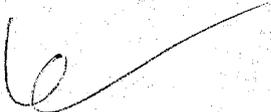
data. From the decompressed source image data, a facsimile of the original source image can be reconstructed The reconstructed source image, however, is not an exact replication of the original source image. As described above, the quantization step produces some lossiness in the

5   process of compressing the data. By carefully designing the quantization tables, however, the prior art methods have constrained the loss to visually imperceptible portions of the image. These methods, and their shortcomings, are described below.

The JPEG standard includes two examples of quantization tables,

10   one for luminance channels and one for chrominance channels. See International Organization for Standardization: "Information technology - digital compression encoding of continuous - tones still images - part 1: Requirements and Guidelines, " ISO/IEC IS10918-1, October 20, 1992. These tables are known as the K.1 and K.2 tables, respectively. These

15   tables have been designed based on the perceptually lossless compression of color images represented in the YUV color space.

These tables result in visually pleasing images, but yield a rather low compression ratio for certain applications. The compression ratio can be varied by setting a so-called Q-factor or scaling factor, which is

20   essentially a uniform multiplicative parameter that is applied to each of the elements in the quantization tables. The larger the Q-factor the larger the achievable compression rate. Even if the original tables are carefully designed to be perceptually lossless, however, a large Q-factor will introduce artifacts in the reconstructed image, such as blockiness in

25   areas of constant color or ringing in text-scale characters. Some of these artifacts can be effectively cancelled by post-processing of the reconstructed image by passing it through a tone reproduction curve correction stage, or by segmenting the image and processing the text separately. However, such methods easily introduce new artifacts.

Therefore, these methods are not ideal.

As a result of the inadequacy of the Q-factor approach, additional design methods for JPEG discrete quantization tables have been proposed. These methods can be categorized as either perceptual, which

5   means based on the human visual system (HVS) or based on information theory criteria. These methods are also designated as being based on the removal of subjective or statistical redundancy, respectively. These methods are discussed in copending application entitled "Method for Selecting JPEG Quantization Tables for Low Bandwidth Applications,"

10 commonly assigned to the present assignee, incorporated herein by reference.

Quantization is not the only cause of image degradation. The color source image data itself might be compromised. For scanned colored images, the visual quality of the image can be degraded because of the

15 inherent limitations of color scanners. These limitations are mainly of two kinds: limited modulation transfer function (MTF) and misregistration. The modulation transfer function refers to the mathematical representation or transfer function of the scanning process. There are inherent limitations in representing the scanning process by

20 the MTF and these limitations are the main cause of pixel aliasing, which produces fuzzy black text glyphs of grayish appearance. Misregistration, on the other hand, refers to the relative misalignment of the scanner sensors for the various frequency bands. For example, the Hewlett Packard Scan Jet IIc™ has a color misregistration tolerance of +/- 0.076

25 mm for red and blue with respect to green. This amount of misregistration is significant considering the size of an image pixel (e.g., 0.08 mm at 300 dots per inch (dpi)).

These limitations significantly degrade text in color images because sharp edges are very important for reading efficiency. The visual quality

of text can be improved, however, using prior art edge enhancement techniques. Edge enhancement can be performed in either the spatial or frequency domain. In the spatial domain (i.e., RGB), edge crispening can be performed by discrete convolution of the scanned image with an
5 edge enhancement kernel. This approach is equivalent to filtering the image with a high-pass filter. However, this technique is computationally intensive. An M x N convolution kernel, for example, requires MN multiplications and additions per pixel.

For edge sharpening in the frequency domain, the full image is first
10 transformed into the frequency domain using the Fast Fourier Transform (FFT) or the Discrete Fourier Transform (DFT), low frequency components are dropped, and then the image is transformed back into the time domain. This frequency domain method, as with the spatial domain method, is also computationally intensive. Moreover, it uses a
15 transformation different than that required by the JPEG standard.

Accordingly, the need remains for a computationally efficient method for improving the visual quality of images, and in particular text, in scanned images.

20 SUMMARY OF THE INVENTION

The invention is a method of compressing and decompressing images which comprises using one quantization table (Q_E) for compressing the image and a second quantization table (Q_D) for decompressing the image. In general, compression and decompression are performed in
25 conformance with the JPEG standard. The second quantization table Q_D is related to the first quantization table according to the following general expression:

$$Q_D = S \times Q_E + B,$$

where S is a scaling matrix having each element S[k,l] formed according to the following expression:

$$S[k,l]^2 = V^*[k,l]/V_Y[k,l]$$

where V^* is a variance matrix of a reference image and V_Y is a variance matrix of a scanned image; and where B is a brightness matrix, which can include zero or non-zero elements. By using the scaling matrix S, the high-frequency components of the DCT elements can be "enhanced" without any additional computational requirements. According to the invention, the quantization table Q_D is transmitted with the encoded quantized image data, and is used in decompression to recover the image.

The reference image is a preselected continuous-tone image, either grayscale or color depending on the images to be processed. The reference image is rendered into a target image file. The target image file is not generated by a scanner, so the data therein is not compromised by any of the inherent limitations of a color scanner. Thus, the variance of the target image data, which is a statistical representation of the energy or frequency content of the image, retains the high-frequency components. The reference image can be any continuous-tone image, but in the preferred embodiment the reference image includes text with a serif font because the serif font has good visual quality which the method preserves.

The scanned image, although it can be any image, in the preferred embodiment is a printed version of the reference image. Thus, the variance of the scanned image represents the energy or frequency composition of the reference image but which is compromised by the inherent limitations of the scanner. The scaling matrix, therefore, boosts the frequency components that are compromised by the scanning process.

A preferred embodiment of the invention is described herein in the context of a color facsimile (fax) machine. The color fax machine includes a scanner for rendering a color image into color source image data that represents the color image, a compression engine that compresses the color source image data to compressed image data, a means for encapsulating the compressed image data, and a means for transmitting the encapsulated data. The compression engine includes means for storing two quantization tables. The first quantization table is used to quantize the image data transformed using the discrete cosine transform (DCT). The second quantization table is encapsulated with the encoded quantized image data for use in decompressing the image. The second quantization table is related to the first quantization table in the manner described above. When used to transmit and receive color images between two locations, the machine transfers the images with higher quality than prior systems.

The second quantization table can be precomputed and stored in the compression engine, in which case there are no additional computational requirements for the compression engine to implement the image enhancing method of the invention. This capability results in a lower cost color facsimile product than is possible using the prior art image enhancement techniques.

The foregoing and other objects, features and advantages of the invention will become more readily apparent from the following detailed description of a preferred embodiment of the invention which proceeds with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram of a prior art JPEG compression engine.

Fig. 2 is a drawing of a typical format of a JPEG compressed file.

Fig. 3 is a block diagram of a prior art JPEG decompression engine.

Fig. 4 is a flow chart of a method of forming a scaled quantization table according to the invention.

Fig. 5 is a drawing of a JPEG compressed file including a quantization table scaled according to the invention.

Fig. 6 is a block diagram of a JPEG decompression engine according to the invention.

Fig 7 is a block diagram of a color fax machine including JPEG compression and decompression engines according to the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Overview of the Quantization Process

The text and image enhancing technique according to the invention is integrated into the decoding or inverse quantization step that is necessarily required by the JPEG standard. The invention integrates the two by using two different quantization tables: a first quantization table (Q_E) for use in quantizing the image data during the compression step and a second quantization table (Q_D) for use during the decode or inverse quantization during the decompression process. The difference between the two tables, in particular the ratio of the two tables, determines the amount of image enhancing that is done in the two steps. By integrating the image enhancing and inverse quantization steps, the method does not require any additional computations than already required for the compression and decompression processes.

In order to understand the operation of the invention, the following mathematical derivation is necessary. Let Q_D be the second quantization table used during the decoding or inverse quantization step. Then let Q_D be related to the first quantization table Q_E, used during the quantization step, by the following expression:
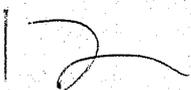
$$Q_D = (S \times Q_E) + B \qquad\qquad (1)$$

where S is a scaling matrix, which scales each element of the first quantization table Q_E to a corresponding element in the second quantization table Q_D. The scaling matrix S is not used in a true matrix multiplication; rather, the multiplication is an element-by-element multiplication. Each element in the first quantization table Q_E has a corresponding element in the scaling matrix S that when multiplied together produce the corresponding element in the second quantization table Q_D.

The matrix B is a so-called brightness matrix because it can affect the brightness of the image by changing the DC level of the DCT elements. The elements of the B matrix can include zero or non-zero values depending on the desired brightness. For purposes of the following discussion and derivation, however, it will be assumed that the B matrix contains zero elements only to simplify the derivation.

The text and image enhancing technique of the invention uses a variance matrix to represent the statistical properties of an image. The variance matrix is an M x M matrix, where each element in the variance matrix is equal to the variance of a corresponding DCT coefficient over the entire image. The variance is computed in the traditional manner, as is known in the art.

The edge enhancement technique in essence tries to match the variance matrix of a decompressed image ($V_Y[k,l]$) with a variance matrix of a reference image ($V^*[k,l]$). The technique tries to match the two by scaling the quantization table in the manner described above. In order to do this, the method takes advantage of the relationship between the uncompressed image and the compressed image. The following derivation

will make this relationship clear.

Let $V^*[k,l]$ denote the variance of the $[k,l]$ frequency component of a reference image. Ideally, this image contains those critical attributes that the technique seeks to preserve, for example, text. This variance matrix is of an ideal or reference image in that it is not rendered into color source image data by a scanner but, instead, is rendered into its ideal form by software, described further below. Thus, the color source image data of the reference image does not suffer from the image degradation due to the inherent limitations of the scanner. Therefore, the variance of the reference image retains the high-frequency characteristics of the original reference image.

The method produces a resulting decompressed image that has approximately the same variance as the variance of the reference by modifying the quantization table. Thus, the method produces the following relationship:

$$V_Y[k, l] = V^*[k, l] \tag{2}$$

However, the decompressed image (Y') is related to the original quantized image (Y_Q) by the following expression:

$$Y'[k, l] = Y_{QE}[k, l]\, Q_D[k, l] \tag{3}$$

Substituting equation (1) into equation (3) yields the following equation below:

$$Y'[k, l] = Y_{QE}[k, l]\, (S[k, l]\, Q_E[k, l]) \tag{4}$$

The variance of the decompressed image (V_Y) can then be expressed by

the following expression:

$$V_{Y'}[k, l] = Var\ (Y'[k, l]) = Var\ (S[k, l]\ Y_{QE}[k, l]\ Q_E[k, l]) \qquad (5)$$

Reducing this expression yields the following:

$$V_{Y'}[k, l] = S^2[k, l]\ V_Y[k, l] \qquad (6)$$

where V_Y represents the variance of the original uncompressed image.

Substituting equation (6) into equation (2) yields the following relationship between the scaling matrix S and the variances of the reference image (V*) and the original image (V_Y):

$$S[k, l]^2 = V^*[k, l] / V_Y[k, l], \qquad (7)$$

Therefore, the scaling matrix S can be used to boost the variance of the JPEG compressed image to that of the reference image by appropriate formation of the scaling matrix. This method is shown in a more generalized fashion in FIG. 4.

Preferred Embodiment of the Method

In FIG. 4, a method 62 of forming a scaled quantization table according to the invention is shown. The first step 64 is to generate a reference image. This reference image, in the preferred embodiment, embodies certain valued features or elements that the method seeks to preserve. In the preferred embodiment, these critical elements include highly readable text such as those typefaces having a serif font, e.g., Times Roman. The selection of the reference image is important because it is the frequency or energy characteristics of this image that the text

image sharpening method is intended to preserve. Because the method is statistical, the result can be improved by averaging over a number of typical images. Examples of such typical images are those using different fonts (e.g., Palatino and Devanagari), handwriting, short-hand, line drawings, schematics, bar codes, etc. These different images can further be categorized in a number of classes.

This generating step 64 is performed on a computer, typically using a word processor or desktop publishing application such as Adobe Illustrator or Microsoft Word. The image is entered into the computer and then rendered by the application into reference image data by the application. The reference image data will be in the appropriate color space, e.g., CIELAB or RGB, to allow the subsequent step to be performed. This process can be achieved by first rendering the image into an Adobe Postscript file and then rendered into a bit-mapped color source image data file using DisplayPostscript. Alternatively, other page description languages can be used to describe the image such as the PCL language by Hewlett Packard and then rendered into a bit map by an appropriate rendering program.

Once the reference image is generated and rendered into reference image data, the average energy of the reference image is determined in step 66. In the preferred embodiment, this step includes computing the variance matrix (V*) for the reference image data. The variance matrix, as is known in the art, statistically represents the frequency components or energy contained in the image. Unlike a scanned image, the reference image does not suffer from any of the inherent limitations of the color scanner because it is not compromised by the misregistration and MTF limitations of the scanner. Accordingly, the variance for the reference image retains the high-frequency energy that is critical to the visual quality of the reference image.

In step 68, a scanned image is scanned or selected from one or more stored pre-scanned images. This scanned image is one that suffers from the inherent limitation of the scanner. This scanned image can be any image, but in the preferred embodiment it is a scanned version of the reference image generated in step 64, or of the same type of image used to form an averaged reference image.

As in step 66, the average energy of the scanned image is then determined in step 70. The average energy again is represented by a variance matrix (V_Y) of the scanned image.

The variance matrix (V^*) of the reference image and the variance matrix (V_Y) of the scanned image are then used to compute the scaling matrix S in step 72. This step involves solving equation (7) above for each element in the scaling matrix S.

Finally, in step 74 the scaled version of the quantization table is calculated. This step is a simple element-by-element multiplication as represented by equation (1) above.

The use of the scale tables is seen clearly with reference to FIG. 5. In FIG. 5, a first set of Q tables 76 is provided to a JPEG compression engine 78, which compresses the image data in accordance with the JPEG compression standard. The compression engine 78 performs the quantization step using the Q tables 76. The compression engine 78 also performs the entropy encoding, as described above, using Huffman tables 80.

The Q tables 76 are then scaled by a scaler 82 using the method 62 described above with reference to FIG. 4. A JPEG format file 84 is then formed which includes a JPEG header 86, the scaled Q tables 88, the H tables 90 and the compressed image data 92. Thus, although the quantization was performed using the Q tables 76, the scaled Q tables are transmitted for use in the decompression process. This difference

between the two quantization tables, as represented by the scaling matrix, is what implements the text and image enhancing technique according to the invention. An advantage of the present invention is that no changes necessarily need to be made to the decompression engine to

5   implement the image enhancing technique of the invention. The decompression engine uses the scaled Q tables in the same way it would use unscaled Q tables. Thus, the prior art decompression engine of Fig. 3 can still decompress images compressed according to the invention. Moreover, it can do so with the improved image quality provided by the

10   invention. In addition, the scaled Q-tables can be precomputed and stored in memory so that the scaling step does not need to be performed real-time. This preferred embodiment of the invention, therefore, does not require scaler 82.

        Alternatively, the scaling can be performed on the decompression

15   side rather than on the compression side. A decompression engine 94 which implements the text and image enhancing technique according to this aspect of the invention is shown in FIG. 6. Decompression engine 94 could be used in conjunction with a prior art compression engine which does not implement the text and image enhancing technique according to

20   the invention.

        The decompression engine 94 includes a header extraction unit 96, which extracts the H tables and the Q tables. The H tables are used by an entropy decoder 98 to decode the compressed image data. The Q tables received by the decompression engine are then scaled by a scaler 100,

25   which scales the Q tables according to equation 7 above. The variance matrices (V^* and V_Y) are stored in the scaler for use in the scaling process. The scaled Q tables are then stored in a random access memory 102 for use by an inverse quantizer 104, which performs the inverse quantization step. Because the scaled Q tables are used in the inverse

quantization step the text and image enhancing technique according to the invention is implemented in this step. The inverse quantized image data is then transformed by IDCT 105 and rasterized by a block-to-raster translator 106. The result is text and image enhanced source image data.

5   Thus, the text and image enhancing technique can either be implemented in either the decompression side or the compression side, but not both. If the decompression engine 94 and compression engine 78 are used together, there must be some way to bypass the scaler 100 where the scaling is performed on the compression side. It is possible for the 10 JPEG header to include a tag field to indicate whether or not the scaling has been performed in the compression or, alternatively, whether it needs to be performed in the decompression. However, this would require the use of a JPEG application marker in the JPEG file. Accordingly, the preferred method is to scale the quantization tables on the compression 15 side and transmit the scaled quantization tables to the decompression engine. This would not necessitate any changes to the decompression engine or the JPEG standard.

Although the compression engine according to the invention is implemented in dedicated hardware as described hereinabove, 20 alternatively it can be implemented in software operating on a programmed computer having a microprocessor such as an Intel 80486 or Pentium or Hewlett Packard PA-RISC. In the latter case, the various tables, whether precomputed or computed real-time, are stored in the dynamic random access memory (DRAM) of the computer during the 25 compression and decompression processes and the various steps of the method are implemented by software processes or routines. In addition, there a numerous combinations of hardware and/or software that can be used to implement compression and/or decompression engines according to the invention depending on the desired performance and cost. The

combinations are too numerous to describe individually but those skilled in the art could implement such combinations based on the description found herein.

Preferred Embodiment of the Apparatus

A color facsimile machine 134 using the JPEG compression and decompression engines according to invention is shown in Fig. 7. The color fax machine 134 includes two related but independent components: a send component and a receive component. The send component includes a scanner 136 which receives a physical image document and which renders or transforms the physical image into color source image data. The color source image data is then passed to a corrections and transformation engine 138, which can correct for certain anomalies in the scanner and format the source color image data appropriately. This transformation can include transforming the color source image data from one color space, e.g., RGB, to another, e.g., CIELAB.

The corrected and transformed color source image data is then passed to a JPEG compression engine 140, which includes the two quantization tables (Q_E, Q_D) formed in accordance with the invention described herein. The first quantization table Q_E, in the preferred embodiment of the invention is formed in accordance with the method taught in our commonly-assigned application entitled METHOD FOR SELECTING JPEG QUANTIZATION TABLES FOR LOW BANDWIDTH APPLICATIONS, Ser. No.8/935,617, incorporated herein by reference. Alternatively, any customary JPEG quantization tables (e.g., K.1) can be used as the first quantization tables.

The JPEG compression engine compresses the source image data into compressed image data in the manner described above, which is then passed onto a G3/G4 encapsulation engine 142 along with the second

Q-table (Q_D). The encapsulation engine 142 performs the step of encapsulating the compressed image data in accordance with the amended T.30 Group 3 Facsimile Standard. Alternatively, instead of G3, the encapsulation could also be G4 for an equivalent machine operating

5 according to the T.30 Group 4 Facsimile Standard. The encapsulated data is then transmitted via a transmission means 144 over the limited bandwidth channel. In the preferred embodiments, this transmission means includes a modem (modulator), but can also include direct transmission circuitry.

10 On the receiving end, the color fax machine 134 includes a receiving means 146, which in the preferred embodiment uses a modem (demodulator) of the transmission means 144 or equivalent digital receive circuitry. The received compressed image data is then decoded by a G3/G4 decoding engine 148. The engine 148 decodes or unencapsulates

15 the compressed image in accordance with the applicable Group 3 or Group 4 facsimile standard. The decoded compressed image data is then decompressed by a JPEG decompression engine 150. In the preferred embodiment, this decompression engine is that shown in Fig. 6, which includes the Q-table scaler. Moreover, the decompression engine 150

20 includes means for determining whether the Q-tables were scaled in the compression process. As described above, this operation involves decoding the header file. Alternatively, a precomputed set of scaled Q-tables cold be stored in memory in the receive component and used in the decompression process by the engine 150. In a further embodiment, the

25 engine 150 could assume that the scaling was performed in the compression process and simply use the Q-tables as received.

The decompressed source image data is then passed to a corrections and transformations engine 152, which transforms the source image data into the color space required by a color printer 154 included in the color

fax 134. The color printer 154, which can use any printing technology such as electro-photographic, inkjet or impact, then reproduces the image. The reproduced color image, by using the scaled Q-tables formed in accordance with the invention, improves the text and image quality of the

5   image without requiring real-time processing of the image data. The text and image enhancing techniques described herein take advantage of the processing already required by the JPEG standard to implement the techniques. Thus, the overall cost of the product is not impacted.

The color facsimile machine 134 can include a plurality of different

10   scaled quantization tables for use during the compression and/or decompression process. These different scaled quantization tables can be precomputed, using the method described above, wherein each table is formed using a different reference image or an average of several images of a given type. These different reference images can be selected to

15   enhance different textual or image characteristics. The user can then select the particular scaled quantization table that matches his or her image so that the image enhancement technique can optimize for the user's image. In the preferred embodiment, the machine 134 includes a user actuable selection device to indicate the user's selection. The send

20   component of fax machine 134 would then select the appropriate scaled quantization table responsive to the user's selection.

The same method can be applied to a grayscale facsimile machine or when a color facsimile machine is used in grayscale mode. In this case only the luminance channel is used. As a result, when the term color is

25   used herein it refers to both color and grayscale.

The color facsimile machine 134 is but one example of an application for the JPEG compression and decompression engines of the invention. The quantization tables formed by the above described method can be used in any context that requires JPEG compression. These applications

are typically those that have limited bandwidth capability. An example of another such limited bandwidth application is a personal computer or work station. In these applications color images are displayed on the screen, which can be represented in a variety of different formats such as

5   Postscript. JPEG compression can be used to compress these color images to allow for more efficient transmission of the compressed image over the limited bandwidth channel.

    Having illustrated and described the principles of our invention in a preferred embodiment thereof, it should be readily apparent to those

10   skilled in the art that the invention can be modified in arrangement and detail without departing from such principles. For example, although the invention has been described in reference to the JPEG compression standard, the method is applicable to MPEG, H.261 or other compression standards as well. We claim all modifications coming within the spirit

15   and scope of the accompanying claims.

CLAIMS

1.   A machine for transmitting color images comprising:

means for rendering a color image into color source image data representing the color image;

5   a compression engine that compresses the source image data into compressed image data, the engine including:

a transforming means for converting the source image data into transformed image data,

means for storing a first multi-element quantization table

10   (Q_E);

a quantizer means for converting the transformed image data into quantized image data in accordance with elements contained in the first quantization table,

means for storing a second multi-element quantization table
related to but

15   (Q_D) nonidentical to the first quantization table Q_E,

an entropy encoder means for transforming the quantized image data into the encoded image data using an entropy table;

means for encapsulating the encoded image data, the second quantization table, and the entropy table to form an encapsulated data

20   file; and

means for transmitting the encapsulated data file.

2.   A machine for transmitting color images according to claim 1 further comprising a scaler for scaling the first quantization table (Q_E)

25   in accordance with a predetermined function to form the second quantization table (Q_D).

3.   A machine for transmitting color images according to claim 1 wherein the second quantization table is related to the first

quantization table in accordance with a predetermined function of the energy in a reference image and the energy in a scanned image.

4. A machine for transmitting color images according to claim 3 wherein the predetermined function is given by the following expression:

$$Q_D = S \times Q_B,$$

where S is a scaling matrix having each element S[k, l] formed according to the following expression:

$$S[k, l]^2 = V^*[k, l] / V_Y[k, l],$$

where V^* is a variance matrix of the reference image and V_Y is a variance matrix of the scanned image.

5. A machine for transmitting color images according to claim 1 wherein the means for rendering a color image into color source image data representing the color image includes:

a color scanner;

means for gamma correcting the color source image data; and

means for transforming the color source image data from a first color space to a second color space.

6. A machine for transmitting color images according to claim 1 wherein the means for rendering a color image into color source image data representing the color image includes computer software operable on a computer.

7. A machine for transmitting color images according to claim 1 further comprising:

means for receiving an encapsulated data packet;

means for decoding the encapsulated data packet to extract a header, received image data, a received quantization table (Q_R), and a received entropy table;

a decompression engine that decompresses the received image data into color source image data using the received quantization table and the received entropy table; and

means for reproducing a color image from the color source image data.

8. A machine for transmitting color images according to claim 7 further comprising a scaler for scaling the received quantization table into a scaled quantization table (Q_S), which is used by the decompression engine in lieu of the received quantization table.

9. A machine for transmitting color images according to claim 8 wherein the scaled quantization table is related to the received quantization table in accordance with a predetermined function of the energy in a reference image and the energy in a scanned image.

10. A machine for transmitting color images according to claim 9 wherein the predetermined function is given by the following expression:

$$Q_S = S \times Q_R,$$

where S is a scaling matrix having each element $S[k, l]$ formed according to the following expression:

$$S[k, l]^2 = V^*[k, l] / V_Y[k, l],$$

where V^* is a variance matrix of the reference image and V_Y is a variance matrix of the scanned image.

11.   A machine for transmitting color images according to claim 10 further comprising means for selectively scaling the received quantization table responsive to the header.

5   12.   A machine for transmitting color images according to claim 7 wherein the means for reproducing a color image from the color source image data includes computer software operable on a computer for displaying the color image on a computer display associated with the computer.

10

13.   A machine for transmitting color images according to claim 1 wherein the transforming means includes means for transforming the source image data into transformed image data using the discrete cosine transform (DCT).

15

14.   A method of compressing and transmitting images which produces decompressed images having improved text and image quality, the method comprising:

compressing a source image into compressed image data using a 20 first quantization table (Q_E);

forming a second quantization table (Q_D), wherein the second quantization table is related to the first quantization table in accordance with a predetermined function of the energy in a reference image and the energy in a scanned image;

25   transmitting the compressed image data; and

decompressing the compressed image data using the second quantization table Q_D.

15.   A method of compressing and transmitting images which

produces decompressed images having improved text and image quality according to claim 14 wherein the step forming a second quantization table includes scaling the first quantization in accordance with the predetermined function.

16.   A method of compressing and transmitting images which produces decompressed images having improved text and image quality according to claim 15 wherein the step scaling the first quantization in accordance with the predetermined function is performed prior to the transmitting step.

17.   A method of compressing and transmitting images which produces decompressed images having improved text and image quality according to claim 15 wherein the step scaling the first quantization in accordance with the predetermined function is performed subsequent to the transmitting step.

18.   A method of compressing and transmitting images which produces decompressed images having improved text and image quality according to claim 15 wherein the step forming a second quantization table includes forming a second quantization table (Q_D), where the second quantization table is related to the first quantization table according to the following expression:

$$Q_D = S \times Q_E,$$

where S is a scaling matrix having each element $S[k, l]$ formed according to the following expression:

$$S[k, l]^2 = V^*[k, l] / V_Y[k, l],$$

where V^* is a variance matrix of the reference image and V_Y is a variance matrix of the scanned image.

Sub
D_3

19.   A method according to claim 14 further comprising:

encapsulating the second quantization table Q_D with the compressed image data to form an encapsulated data file; and

5   transmitting the encapsulated data file.

20.   A method of compressing and transmitting images having improved text and image quality according to claim 14 wherein the step of forming a second quantization table (Q_D) includes:

10   selecting a target image; and

rendering the target image into an image file.

21.   A method of compressing and transmitting images having improved text and image quality according to claim 20 wherein the step

15   of selecting a target image includes selecting a target image having critical image elements that are critical to the quality of the image.

22.   A method of compressing and transmitting images having improved text and image quality according to claim 20 wherein the step

20   of selecting a target image includes selecting a target image having text.

23.   A method of compressing and transmitting images having improved text and image quality according to claim 20 wherein the step of selecting a target image includes selecting a target image having text

25   with a serif font.

24.   A method of compressing and transmitting images having improved text and image quality according to claim 14 wherein the scanned image is the reference image.

25. A method of forming a quantization table for use in compressing images in accordance with the JPEG compression standard, the method comprising:

selecting a first quantization table (Q_E); and

forming a second quantization table (Q_D), where the second quantization table is related to the first quantization table according to the following expression:

$$Q_D = S \times Q_E,$$

where S is a scaling matrix having each element $S[k, l]$ formed according to the following expression:

$$S[k, l]^2 = V^*[k, l] / V_Y[k, l],$$

where V^* is a variance matrix of a reference image and V_Y is a variance matrix of a scanned image.

26. A method of forming a quantization table according to claim 25 wherein the step of forming a second quantization table (Q_D) includes:

selecting the reference image;

rendering the reference image into reference image data; and

computing the variance matrix V^* for the reference image data.

27. A method of forming a quantization table according to claim 26 wherein the step of forming a second quantization table (Q_D) includes:

scanning the scanned image to produce scanned image data; and

computing the variance matrix V_Y for the scanned image data.

28. A method of forming a quantization table according to claim 27 wherein the step of scanning the scanned image to produce scanned image data includes scanning the reference image.

29.   A method of improving text and image quality of compressed images that are compressed using the JPEG compression standard, the method comprising:

selecting a reference image;

determining the energy content of the reference image;

selecting a scanned image;

determining the energy content of the scanned image;

selecting a first quantization table (Q_E);

scaling the first quantization table (Q_E) to form a second quantization table (Q_D) according to the ratio of the energy in the reference image to the energy content of the scanned image;

compressing a source image in accordance with the JPEG standard using the first quantization table (Q_E); and

decompressing the source image in accordance with the JPEG standard using the second quantization table Q_D whereby the decompressed image has improved image quality.

30.   A method of improving text and image quality according to claim 29 wherein the step of determining the energy content of the reference image includes determining a variance matrix (V*) of the reference image.

31.   A method of improving text and image quality according to claim 30 wherein the step of determining the energy content of the scanned image includes determining a variance matrix (V_Y) of the scanned image.

32.   A method of improving text and image quality according to

claim 31 wherein the step of scaling the first quantization table (Q_E) to form a second quantization table (Q_D) according to the ratio of the energy in the reference image to the energy content of the scanned image includes:

determining a scaling matrix (S), where each element S[k, l] being formed according to the following expression:

S[k, l]^2 = V*[k, l] / V$_Y$[k, l],

where V*[k, l] is a corresponding element in the variance matrix V* and V$_Y$[k, l] is the corresponding element in the variance matrix V$_Y$; and

scaling the first quantization table (Q_E) to form a second quantization table (Q_D) according to the following expression:

Q_D = S × Q_E.

33.   A method of improving text and image quality according to claim 29 further comprising:

encapsulating the second quantization table (Q_D) with the compressed image to form a JPEG file; and

transmitting the JPEG file over a limited bandwidth channel.

34.   A method of improving text and image quality according to claim 29 further comprising:

selecting a plurality of reference images;

determining the energy content of each reference image;

averaging the energy content of the plurality of reference images; and

scaling the first quantization table (Q_E) to form a second quantization table (Q_D) according to the ratio of the average energy in the reference images to the energy content of the scanned image.

35. A method of improving text and image quality according to claim 34 wherein the step of selecting a plurality of reference images includes selecting a plurality of reference images each reference image having a different graphical content.

5

36. A method of improving text and image quality according to claim 29 further comprising:

selecting a plurality of scanned images;

determining the energy content of each scanned image;

10   averaging the energy content of the plurality of scanned images; and

scaling the first quantization table (Q_E) to form a second quantization table (Q_D) according to the ratio of the energy in the reference image to the average energy content of the scanned images.

TEXT AND IMAGE SHARPENING OF JPEG COMPRESSED IMAGES IN THE FREQUENCY DOMAIN

ABSTRACT OF THE DISCLOSURE

The text and image enhancing technique according to the invention is integrated into the decoding or inverse quantization step that is necessarily required by the JPEG standard. The invention integrates the two by using two different quantization tables: a first quantization table (Q_E) for use in quantizing the image data during the compression step and a second quantization table used during the decode or inverse quantization during the decompression process. The second quantization table Q_D is related to the first quantization table according to a predetermined function of the energy in a reference image and the energy in a scanned image. The energy of the reference image lost during the scanning process, as represented by the energy in the scanned image, is restored during the decompression process by appropriately scaling the second quantization table according to the predetermined function. The difference between the two tables, in particular the ratio of the two tables, determines the amount of image enhancing that is done in the two steps. By integrating the image enhancing and inverse quantization steps the method does not require any additional computations than already required for the compression and decompression processes.

DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION	ATTORNEY DOCKET NO. 10940893

As a 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:

TEXT AND IMAGE SHARPENING OF JPEG COMPRESSED IMAGES IN THE FREQUENCY DOMAIN

the specification of which
   (X) is attached hereto. (Leave blank in response to Notice of Missing Parts)
   ( ) was filed on _____ as Application Serial No. _____
   ( ) was amended by the preliminary amendment filed with the original application papers.

I hereby state that I have reviewed and understood the contents of the above-identified specification, including the claims, as amended by any amendment(s) referred to above and that I have disclosed the best mode for carrying out the invention as of the effective filing date of this application. I acknowledge the duty to disclose all information which is material to patentability as defined in 37 CFR 1.56. If this is a continuation-in-part application, I acknowledge the duty to disclose all information known to me to be material to patentability as defined in 37 CFR 1.56 which became available between the filing date of the prior (priority) application and the National or PCT international filing date of this continuation-in-part application.

   ( ) In compliance with this duty there is attached an information disclosure statement 37 CFR 1.97.

Foreign Application(s) and/or Claim of Foreign Priority

I hereby claim foreign priority benefits under Title 35, United States Code Section 119 of any foreign application(s) for patent or inventor(s) certificate listed below and have also identified below any foreign application for patent or inventor(s) certificate having a filing date before that of the application on which priority is claimed:

COUNTRY	APPLICATION NUMBER	DATE FILED	PRIORITY CLAIMED UNDER 35 U.S.C. 119
			YES: ___ NO: ___
			YES: ___ NO: ___
			YES: ___ NO: ___

U. S. Priority Claim

I hereby claim the benefit under Title 35, United States Code, Section 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code Section 112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, Section 1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

APPLICATION SERIAL NUMBER	FILING DATE	STATUS (patented/pending/abandoned)

POWER OF ATTORNEY:

As a named inventor, I hereby appoint the attorney(s) and/or agent(s) listed below to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

C. Douglass Thomas	Edward Y. Wong	Herbert R. Schulze	Peter P. Tong
Reg. No. 32,947	Reg. No. 29,879	Reg. No. 30,682	Reg. No. 35,757

Send Correspondence to:	Direct Telephone Calls To:
Records Manager Legal Department, 20BO HEWLETT-PACKARD COMPANY P.O. Box 10301 Palo Alto, California 94303-0890	C. Douglass Thomas (415) 857-8129

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 issued thereon.

Full Name of Inventor: **Giordano Beretta**   / - OO   Citizenship: **Switzerland**

Residence/Post Office Address: **1760 Newell Road**

Palo Alto, California 94303

_____   March 23, 1995
Inventor's Signature   Date

Full Name of # 2 joint inventor: __Vasudev Bhaskaran__ *2 - 00*   Citizenship: __India__

Residence/Post Office Address: __375 Anna Avenue__

__Mountain View, California 94043__   *CA*

B. Vasudev
Inventor's Signature

__March 27, 1995__
Date

3-00

Full Name of # 3 joint inventor: __Konstantinos Konstantinides__   Citizenship: __Greece__

Residence/Post Office Address: __1508 Jacob Avenue__

__San Jose, California 95118__   *CA*

(signature)
Inventor's Signature

__3/23/95__
Date

Full Name of # 4 joint inventor: _____   Citizenship: _____

Residence/Post Office Address: _____

_____

Inventor's Signature

Date

Full Name of # 5 joint inventor: _____   Citizenship: _____

Residence/Post Office Address: _____

_____

Inventor's Signature

Date

Full Name of # 6 joint inventor: _____   Citizenship: _____

Residence/Post Office Address: _____

_____

Inventor's Signature

Date

Full Name of # 7 joint inventor: _____   Citizenship: _____

Residence/Post Office Address: _____

_____

Inventor's Signature

Date

Full Name of # 8 joint inventor: _____   Citizenship: _____

Residence/Post Office Address: _____

_____

Inventor's Signature

Date

(Use Next Page For Additional Inventor(s) Signature(s))   Page 2 of 2

Rev 09/93 (Form 3.02)

PRINT OF DRAWINGS
AS ORIGINALLY FILED

HP 1094893-1
Sheet 1 of 7

08/411369
940695

SOURCE IMAGE
DATA

10

RASTER TO
BLOCK
TRANSLATION — 12

14

FDCT — 16

18

QUANTIZATION — 20

26

Q TABLES — 24

22

ENTROPY
CODING — 28

32

H TABLES — 30

COMPRESSED
IMAGE DATA

FIG. 1
(PRIOR ART)

PRINT OF DRAWINGS
AS ORIGINALLY FILED

HP 1094893-1
Sheet 2 of 7

08/411369
94068

FIG. 2
(PRIOR ART)

PRINT OF DRAWINGS
AS ORIGINALLY FILED

HP 1094893-1
Sheet 3 of 7

08/411369
940695

COMPRESSED
IMAGE DATA

43

44

HEADER
EXTRACTION

52

ENTROPY
DECODING

50

H TABLES

46

56

INVERSE
QUANTIZATION

54

Q TABLES

48

59

IDCT

57

FIG. 3
(PRIOR ART)

60

BLOCK TO
RASTER
TRANSLATION

58

SOURCE IMAGE
DATA

00/411369

940695

```
  ┌─────────────┐ 64        ┌─────────────┐ 68
  │  GENERATE   │           │   SELECT    │
  │  REFERENCE  │           │   SCANNED   │
  │    IMAGE    │           │    IMAGE    │
  └─────────────┘           └─────────────┘
         │                         │
         ▼                         ▼
  ┌─────────────┐ 66        ┌─────────────┐ 70
  │  DETERMINE  │           │  DETERMINE  │
  │   AVERAGE   │           │   AVERAGE   │
  │   ENERGY    │           │   ENERGY    │
  └─────────────┘           └─────────────┘

         V*                        Vy
          \                       /
           \                     /
            \                   /
   62        ▼                 ▼
          ┌─────────────┐ 72
          │   COMPUTE   │
          │   SCALING   │
          │   MATRIX    │
          └─────────────┘
                 │
                 ▼  S
          ┌─────────────┐ 74
          │    SCALE    │
          │   Q TABLE   │
          └─────────────┘
```

FIG. 4

PRINT OF DRAWINGS
AS ORIGINALLY FILED

HP 1094893-1
Sheet 5 of 7

00/411369

940695

FIG. 5

84

86	88	90	92
JPEG HEADERS	SCALED Q TABLES	H TABLES	COMPRESSED IMAGE DATA

83 — SCALED Q TABLES

80 — H TABLES

82 — SCALER

76 — Q TABLES

78 — JPEG COMPRESSION ENGINE

08/411369

940695

COMPRESSED
IMAGE DATA

94

HEADER
EXTRACTION — 96

ENTROPY
DECODING — 98

H TABLE — 99

SCALER — 100

INVERSE
QUANTIZATION — 104

SCALED
Q TABLE — 102

IDCT — 105

BLOCK TO
RASTER
TRANSLATION — 106

FIG. 6

SOURCE IMAGE
DATA

PRINT OF DRAWINGS
AS ORIGINALLY FILED

HP 1094893-1
Sheet 7 of 7

08/411369

940695

IMAGE DOCUMENT

REPRODUCED IMAGE

SCANNER — 136

PRINTER — 154

CORRECTIONS &
TRANS-
FORMATIONS
ENGINE — 138

CORRECTIONS &
TRANS-
FORMATIONS
ENGINE — 152

JPEG
COMPRESSION
ENGINE — 140

JPEG
DECOMPRESSION
ENGINE — 150

G3/G4
ENCAPSULATION
ENGINE — 142

G3/G4
DECODING
ENGINE — 148

TRANSMISSION
MEANS — 144

RECEIVING
MEANS — 146

COMPRESSED
IMAGE DATA

COMPRESSED
IMAGE DATA

134

FIG. 7

08/411369

940695

SOURCE IMAGE
DATA

10

RASTER TO
BLOCK
TRANSLATION
12

FIG. 1
(PRIOR ART)

14

FDCT
16

18

QUANTIZATION
20

Q TABLES
24

26

22

ENTROPY
CODING
28

H TABLES
30

32

COMPRESSED
IMAGE DATA

08/411369
940695

FIG. 2
(PRIOR ART)

PRINT OF DRAWINGS
AS ORIGINALLY FILED

HP 1094893-1
Sheet 3 of 7

08/411369
940695

COMPRESSED
IMAGE DATA

43

HEADER
EXTRACTION   44

52

ENTROPY
DECODING   50   H TABLES   46

56

INVERSE
QUANTIZATION   54   Q TABLES   48

59

IDCT   57

60

BLOCK TO
RASTER
TRANSLATION   58

SOURCE IMAGE
DATA

FIG. 3
(PRIOR ART)

6/411369

940695

```
┌──────────────┐ 64        ┌──────────────┐ 68
│   GENERATE   │           │    SELECT    │
│  REFERENCE   │           │   SCANNED    │
│    IMAGE     │           │    IMAGE     │
└──────────────┘           └──────────────┘
        │                          │
        ▼                          ▼
┌──────────────┐ 66        ┌──────────────┐ 70
│  DETERMINE   │           │  DETERMINE   │
│   AVERAGE    │           │   AVERAGE    │
│    ENERGY    │           │    ENERGY    │
└──────────────┘           └──────────────┘
        │                          │
        V*                         Vy
         \                        /
          \                      /
           ▼                    ▼
        ┌──────────────┐ 72
        │   COMPUTE    │
        │   SCALING    │
        │   MATRIX     │
        └──────────────┘
               │
               S
               ▼
        ┌──────────────┐ 74
        │    SCALE     │
        │   Q TABLE    │
        └──────────────┘
```

62

FIG. 4

PRINT OF DRAWINGS
AS ORIGINALLY FILED

HP 1094893-1
Sheet 5 of 7

08/411369

940695

FIG. 5

08/411369

940695

COMPRESSED
IMAGE DATA

94

HEADER
EXTRACTION   96

ENTROPY
DECODING   98   H TABLE   99

SCALER   100

INVERSE
QUANTIZATION   104   SCALED
Q TABLE   102

IDCT   105

BLOCK TO
RASTER
TRANSLATION   106

FIG. 6

SOURCE IMAGE
DATA

PRINT OF DRAWINGS
AS ORIGINALLY FILED

HP 1094893-1
Sheet 7 of 7

08/411369
940695

IMAGE DOCUMENT

REPRODUCED IMAGE

SCANNER — 136

PRINTER — 154

CORRECTIONS & TRANS-FORMATIONS ENGINE — 138

CORRECTIONS & TRANS-FORMATIONS ENGINE — 152

JPEG COMPRESSION ENGINE — 140

JPEG DECOMPRESSION ENGINE — 150

G3/G4 ENCAPSULATION ENGINE — 142

G3/G4 DECODING ENGINE — 148

TRANSMISSION - MEANS — 144

RECEIVING MEANS — 146

COMPRESSED IMAGE DATA

COMPRESSED IMAGE DATA

134

FIG. 7

02/24/97

PATENT APPLICATION

ATTORNEY DOCKET NO. __1094893-1__

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Giordano Beretta et al.

Serial No.:   08/411,369

Examiner: B. Johnson

Filing Date: March 27, 1995

Group Art Unit: 2616

Title: TEXT AND IMAGE SHARPENING OF JPEG COMPRESSED IMAGES IN THE FREQUENCY DOMAIN

ASSISTANT COMMISSIONER FOR PATENTS
Washington, D.C. 20231

TRANSMITTAL LETTER FOR RESPONSE/AMENDMENT

Sir:

Transmitted herewith is/are the following in the above-identified application:

(X)   Response/Amendment   ( )   Petition to extend time to respond
( )   New fee as calculated below   ( )   Supplemental Declaration
(X)   No additional fee   (Address envelope to "Box Non-Fee Amendments")
( )   Other: _____

CLAIMS AS AMENDED BY OTHER THAN A SMALL ENTITY						
(1) FOR	(2) CLAIMS REMAINING AFTER AMENDMENT	(3) NUMBER EXTRA	(4) HIGHEST NUMBER PREVIOUSLY PAID FOR	(5) PRESENT EXTRA	(6) RATE	(7) ADDITIONAL FEES
TOTAL CLAIMS	35 MINUS		36	= 0	X $ 22	$ 0
INDEP. CLAIMS	4 MINUS		4	= 0	X $ 80	$ 0
[ ] FIRST PRESENTATION OF A MULTIPLE DEPENDENT CLAIM					+ $260	$ 0
EXTENSION FEE	1ST MONTH $110.00	2ND MONTH $390.00	3RD MONTH $930.00	4TH MONTH $1470.00		$ 0
				TOTAL ADDITIONAL FEE FOR THIS AMENDMENT		$ 0

Charge $____0____ to Deposit Account 08-2025. At any time during the pendency of this application, please charge any fees required or credit any overpayment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.19, 1.20 and 1.21. A duplicate copy of this sheet is enclosed.

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

Date of Deposit: *February 18, 1997*

Typed Name: **Linda A. Iimura**

Signature: _____

Respectfully submitted,

Giordano Beretta et al.

By _____

Pehr Jansson

Attorney/Agent for Applicant(s)
Reg. No.   35,759

Date: *February 18, 1997*

Telephone No.: (415) 857-7533

PATENT APPLICATION FEE DETERMINATION RECORD
Effective October 1, 1994

Application or Docket Number

08/411369

CLAIMS AS FILED - PART I

FOR	(Column 1) NUMBER FILED	(Column 2) NUMBER EXTRA	SMALL ENTITY RATE	FEE	OR	OTHER THAN SMALL ENTITY RATE	FEE
BASIC FEE				365.00	OR		730.00
TOTAL CLAIMS	36 minus 20 =	* 16	x$11=		OR	x$22=	352.00
INDEPENDENT CLAIMS	4 minus 3 =	* 1	x38=		OR	x76=	76.00
MULTIPLE DEPENDENT CLAIM PRESENT			+120=		OR	+240=	—
			TOTAL		OR	TOTAL	1158.00

* If the difference in column 1 is less than zero, enter "0" in column 2

CLAIMS AS AMENDED - PART II

AMENDMENT A

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT	(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	SMALL ENTITY RATE	ADDITIONAL FEE	OR	OTHER THAN SMALL ENTITY RATE	ADDITIONAL FEE
Total	*	Minus **	=	x$11=		OR	x$22=	
Independent	*	Minus ***	=	x38=		OR	x76=	
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				+120=		OR	+240=	
				TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT B

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT	(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	SMALL ENTITY RATE	ADDITIONAL FEE	OR	OTHER THAN SMALL ENTITY RATE	ADDITIONAL FEE
Total	*	Minus **	=	x$11=		OR	x$22=	
Independent	*	Minus ***	=	x38=		OR	x76=	
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				+120=		OR	+240=	
				TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT C

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT	(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	SMALL ENTITY RATE	ADDITIONAL FEE	OR	OTHER THAN SMALL ENTITY RATE	ADDITIONAL FEE
Total	*	Minus **	=	x$11=		OR	x$22=	
Independent	*	Minus ***	=	x38=		OR	x76=	
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				+120=		OR	+240=	
				TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20."
*** If the Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3."
The Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

FORM PTO-875
(Rev. 10/94)

Patent and Trademark Office, U.S. DEPARTMENT OF COMMERCE

PACE DATA ENTRY CODING SHEET

U.S. DEPARTMENT OF COMMERCE
Patent and Trademark Office

1ST EXAMINER	DATE 4/30
2ND EXAMINER	DATE

APPLICATION NUMBER 08/411369

TOTAL CLAIMS 36

INDEPENDENT CLAIMS 4

SMALL ENTITY? 0

TYPE APPL 1

FILING DATE
MONTH	DAY	YEAR
0 3	2 7	9 5

FILING FEE 1 1 5 8

FOREIGN LICENSE Y

SPECIAL HANDLING 0

GROUP ART UNIT 2 6 1 3

CLASS 3 8 2

SHEETS OF DRAWING 7

ATTORNEY DOCKET NUMBER 1 0 9 4 8 9 3

CONTINUITY DATA

CONT STATUS CODE

CODE

PARENT APPLICATION SERIAL NUMBER

PCT APPLICATION SERIAL NUMBER

P	C	T	/	/
P	C	T	/	/
P	C	T	/	/
P	C	T	/	/
P	C	T	/	/

PARENT PATENT NUMBER

PARENT FILING DATE
MONTH DAY YEAR

PCT/FOREIGN APPLICATION DATA

FOREIGN PRIORITY CLAIMED

COUNTRY CODE

PCT/FOREIGN APPLICATION SERIAL NUMBER

FOREIGN FILING DATE
MONTH DAY YEAR

1. **5,566,002**, Oct. 15, 1996, Image coding/decoding method and apparatus; Akihiro Shikakura, 358/433, 261.4, 427, 539 [IMAGE AVAILABLE]

2. **5,565,992**, Oct. 15, 1996, Image transmitting method; Masahiko Enari, 358/261.3, 433 [IMAGE AVAILABLE]

3. **5,563,649**, Oct. 8, 1996, System and method for transmitting video material; Kim V. W. Gould, et al., 348/17, 14 [IMAGE AVAILABLE]

4. **5,553,164**, Sep. 3, 1996, Method for compressing and extending an image by transforming orthogonally and encoding the image; Fumihiko Itagaki, 382/232 [IMAGE AVAILABLE]

5. **5,530,429**, Jun. 25, 1996, Electronic surveillance system; Dimitri V. Hablov, et al., 340/552; 342/192 [IMAGE AVAILABLE]

6. **5,515,105**, May 7, 1996, Video signal coder using variance controlled quantization; Jong-Tae Lim, 348/405 [IMAGE AVAILABLE]

7. **5,509,089**, Apr. 16, 1996, Method and system for encoding images using temporal filtering; Dhruva Ghoshal, 382/236; 348/415, 701; 382/261 [IMAGE AVAILABLE]

8. **5,508,942**, Apr. 16, 1996, Intra/inter decision rules for encoding and decoding video signals; Rohit Agarwal, ~~364/514R~~; ~~348/396~~ [IMAGE AVAILABLE]

9. **5,497,246**, Mar. 5, 1996, Image signal processing device; Nobuaki Abe, 358/426; 348/403, 404, 405; 358/430, 432; 382/248, 250, 251, 252 [IMAGE AVAILABLE]

10. **5,489,998**, Feb. 6, 1996, Color image processing apparatus having transmitter and receiver which perform color correction in accordance with a common standard color image and method for same; Osamu Yamada, et al., 358/523, 518 [IMAGE AVAILABLE]

11. **5,488,570**, Jan. 30, 1996, Encoding and decoding video signals using adaptive filter switching criteria; Rohit Agarwal, ~~364/514R~~; ~~348/396~~ [IMAGE AVAILABLE]

12. **5,473,704**, Dec. 5, 1995, Apparatus for substituting character data for image data using orthogonal conversion coefficients; Nobuaki Abe, 382/235; 345/194, 202; 358/433, 450; 382/250 [IMAGE AVAILABLE]

13. **5,465,164**, Nov. 7, 1995, Image processing method and device for the same; Susumu Sugiura, et al., 358/448; 348/384; 358/432, 458 [IMAGE AVAILABLE]

14. **5,426,512**, Jun. 20, 1995, Image data compression having minimum perceptual error; Andrew B. Watson, 358/426, 432, 433; 382/232 [IMAGE AVAILABLE]

15. **5,416,604**, May 16, 1995, Image compression method for bit-fixation and the apparatus therefor; Goo-man Park, 358/433; 348/384; 358/426 [IMAGE AVAILABLE]

16. **5,398,078**, Mar. 14, 1995, Method of detecting a motion vector in an image coding apparatus; Tadaaki Masuda, et al., 348/699, 413, 416 [IMAGE AVAILABLE]

17. ✓ **5,398,066**, Mar. 14, 1995, Method and apparatus for compression and decompression of digital color images; Eugenio Martinez-Uriegas, et al., 348/393, 394, 395 [IMAGE AVAILABLE]

18. **5,384,644**, Jan. 24, 1995, Image processing method and apparatus for encoding variable-length data; Taku Yamagami, 358/426; 348/384 [IMAGE AVAILABLE]

19. **5,260,808**, Nov. 9, 1993, Image processing apparatus; Akio Fujii, 358/458, 433 [IMAGE AVAILABLE]

20. **5,231,487**, Jul. 27, 1993, Storage of video signals; Terence R. Hurley, et al., 348/391, 424, 472; 386/21, 33, 40 [IMAGE AVAILABLE]

21. **4,953,214**, Aug. 28, 1990, Signal encoding and decoding method and device; Nobuyasu Takeguchi, et al., 395/2.39; 348/409; 395/2.95 [IMAGE AVAILABLE]

22. **4,776,030**, Oct. 4, 1988, Block quantizer for transform coding; Kou-Hu Tzou, 382/252; 358/432 [IMAGE AVAILABLE]

=>

382, 348, 358

Compression of quantization and (table # or matri. # # #)

1. **5,543,844**, Aug. 6, 1996, Method and apparatus for coding image data; Hideaki Mita, et al., 348/405, 27, 420 [IMAGE AVAILABLE]

2. **5,489,944**, Feb. 6, 1996, Encoding method and apparatus to determine quantization levels using energy characteristics in the DCT domain; Jae M. Jo, 348/405, 419 [IMAGE AVAILABLE]

3. **5,481,308**, Jan. 2, 1996, Method and apparatus for synthesizing subband video images; John Hartung, et al., 348/398 [IMAGE AVAILABLE]

1. **5,467,131**, Nov. 14, 1995, Method and apparatus for fast digital signal decoding; Vasudev Bhaskaran, et al., 348/384, 390 [IMAGE AVAILABLE]

2. **5,321,645**, Jun. 14, 1994, Simplified method and apparatus for loss signal compression; Konstantinos Konstantinides, et al., 364/852, 715.02 [IMAGE AVAILABLE]
=>

4. 5,263,099, Nov. 16, 1993, High speed window and level function
modification for real time video processing; James Kapcio, et al.,
382/131; 364/724.19; **382/260** [IMAGE AVAILABLE]

US-CL-CURRENT: **382/131**; 364/724.19; **382/260**

ABSTRACT:
A CT scanner (10) non-invasively examines a region of interest of a
patient to create an image representation that is stored in an image
memory (14). Each pixel of the image representation has a relatively
large number of bits of radiation intensity resolution, e.g. 14 bits or
16k levels, which is larger than the gray scale resolution of a
conventional video monitor (24), e.g. 8 bits or 256 levels. A look-up
table (38) digitally **filters** each pixel value with digital filter
values to reduce the number of levels of each pixel value to the number
of gray scale levels displayable by the video monitor. While the image is
being displayed, the operator selectively adjusts the digital filtering
to optimize the displayed image for the intended diagnostic purpose.
During the vertical flyback or other non-display periods of the video
monitor, a central processor (30) generates most significant bits of
addresses that read digital filter longwords from a filter memory (44). A
multiplexer (54) breaks each word into a plurality of filter values or
bytes which are serially conveyed to the digital **filter** look-up
table. A least significant bit address generator (62) generates the
least significant bits of the address for each filter value concurrently
with its conveyance to the look-up table. The multiplexer and least
significant bit address generator are clocked at a faster rate than the
central processor such that within one read clock pulse of the central
processor, a plurality of digital filter values are loaded into the
digital **filter** look-up **table**. In this manner, image filtering via
look-up table in real time is enhanced through an improved, faster
loading method.

HEWLETT-PACKARD COMPANY
Legal Department, 20BO
P. O. Box 10301
Palo Alto, California 94303-0890

PATENT APPLICATION

ATTORNEY DOCKET NO. ___1094893-1___

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Giordano Beretta et al.

Serial No.: 08/411,369

Examiner:

Filing Date: March 27, 1995

Group Art Unit: 2606

Title: TEXT AND IMAGE SHARPENING OF JPEG COMPRESSED IMAGES IN THE FREQUENCY DOMAIN

ASSISTANT COMMISSIONER FOR PATENTS
Washington, D.C. 20231

INFORMATION DISCLOSURE STATEMENT

Sir:

This Information Disclosure Statement is submitted:

(X) under 37 CFR 1.97 (b), or
(Within three months of filing national application; or date of entry of international application; or before mailing date of first office action on the merits; whichever occurs last)

( ) under 37 CFR 1.97 (c) together with either a:
( ) Certification under 37 CFR 1.97(e), or
( ) a $210.00 fee under 37 CFR 1.17(p), or
(After the CFR 1.97 (b) time period, but before final action or notice of allowance, whichever occurs first)

( ) under 37 CFR 1.97 (d) together with a:
( ) Certification under 37 CFR 1.97 (e), and
( ) a petition under 37 CFR 1.97 (d)(2)(ii), and
( ) a $130.00 petition fee set forth in 37 CFR 1.17 (i)(1).
(Filed after final action or notice of allowance, whichever occurs first, but before payment of the issue fee)

Please charge to Deposit Account **08-2025** the sum of __$ 0.00__. At any time during the pendency of this application, please charge any fees required or credit any overpayment to Deposit Account **08-2025** pursuant to 37 CFR 1.25.

(X) Applicant(s) submit herewith Form PTO 1449 - Information Disclosure Citation together with copies, of patents, publications or other information of which applicant(s) are aware, which applicant(s) believe(s) may be material to the examination of this application and for which there may be a duty to disclose in accordance with 37 CFR 1.56.

( ) A concise explanation of the relevance of foreign language patents, foreign language publications and other foreign language information listed on PTO Form 1449, as presently understood by the individuals(s) designated in 37 CFR 1.56 (c) most knowledgeable about the content is given on the attached sheet, or where a foreign language patent is cited in a search report or other action by a foreign patent office in a counterpart foreign application, an English language version of the search report or action which indicates the degree of relevance found by the foreign office is listed on form PTO 1449 and is enclosed herewith.

It is requested that the information disclosed herein be made of record in this application.

Respectfully submitted,

Giordano Beretta et al.

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

By _____

Pehr Jansson

Date of Deposit: **June 23, 1995**

Attorney/Agent for Applicant(s)
Reg. No.   **35,759**

Typed Name: **Linda A. Iimura**

Signature:

Date: **June 23, 1995**

Rev 04/95 (Form 3.05A)

Telephone No.:

FORM PTO-1449	ATTY. DOCKET NO.	SERIAL NO.
	1094893-1	08/411,369

LIST OF PATENTS AND PUBLICATIONS FOR
APPLICANT'S INFORMATION DISCLOSURE
STATEMENT

APPLICANT
Giordano Beretta et al.

FILING DATE	GROUP
03/27/95	

(Use several sheets if necessary)

REFERENCE DESIGNATION   U.S. PATENT DOCUMENTS

EXAMINER INITIAL		DOCUMENT NUMBER	DATE	NAME	CLASS	SUB CLASS
	AA	5,333,212	July 26, 1194	Adrianus Ligtenberg	382	56
	AB	5,063,608	Nov. 5, 1991	Shepard L. Siegel	382	56

OTHER REFERENCES (including Author, Title, Date, Pertinent Pages, etc.)

✓		AC	G. B. Beretta et al., "Experience with the New Color Facsimile Standard", ISCC Annual Meeting, April 23-25, 1995, P. 1-7.
/		AD	Albert J. Ahumada, Jr. et al., "Luminance-Model-Based DCT Quantization for Color Image Compression", Human Vision, Visual Processing, and Digital Display III, 1666, 365-374, SPIE, 1992.
✓		AE	Kenneth R. Alexander et al., "Spatial-Frequency Characteristics of Letter Identification", J. Opt. Soc. Am. A, 11,9,2375-2382, 1994.
✓		AF	Wen-Hsiung Chen et al., "Adaptive Coding of Monochrome and Color Images", IEEE Transactions on Communications, COM-25, 1285-1292, 1977.
✓		AG	Bowonkoon Chitprasert et al., Human Visual Weighted Progressive Image Transmission, IEEE Transactions on Communications, COM-38, 7, 1040-1044, 1990.
✓		AH	R. J. Clarke, "Spectral Responses of the Discrete Cosine and Walsh-Hadamard Transforms, IEE Proc., 130, Part F, 309-313, 1983.
✓		AI	K.K. De Valois et al., Color-Luminance Masking Interactions, Seeing Contour and Colour", J.J. Kulikowski, C.M. Dickinson and I.J. Murray Editors, Pergamon Press, Oxford, 1989.
✓		AJ	J. Raymond Edinger, Jr., "A Measure for Stairstepping in Digitized Text that Correlates with the Subjective Impression of Quality", IS&T's Tenth International Congress on Advances in Non-Impact Printing Technologies, 552-558, 1994.
✓		AK	Yasushi Hoshino et al., Applicability of a Standardized Discrete Cosine Transform Coding Method to Character Images", J. Electronic Imaging, 1, 3, 322-327, 1992.
✓		AL	Chansik Hwang et al., Human Visual System Weighted Progresseive Image Transmission Using Lapped Orthogonal Transform/Classified Vector Quantization", Optical Engineering, 32, 7, 1524-1530, 1993.
✓		AM	International Organization for Standardization: Information Technology - Digital Compression and Coding of Continuous-Tone Still Images - Part 1: Requirements and Guidelines, ISO/IEC IS 10918-1, October 20, 1992.
✓		AN	International Telecommunication Union: Amendments to ITU-T Rec. T.30 for Enabling Continuous-Tone Colour and Gray-Scale Modes for Group 3, COM 8-43-E, Question 5/8, March 1994.
✓		AO	International Telecommunication Union: Amendments to ITU-T Rec. T-4 to Enable Continuous-Time Colour and Gray-Scale Modes for Group 3, COM 8-44-E, Question5/8, March 1994.
✓		AP	Gordon E. Legge, "Reading: Effects of Contrast and Spatial Frequency", Applied Vision, OSA Technical Digest Series, 16, 90-93, 1989.
✓		AQ	Gordon E. Legge et al., Contrast Masking in Human Vision", J. Opt. Soc. Am., 70,12,1458-1471, 1980.

EXAMINER   DATE CONSIDERED 11/4/96

Rev 5/90 (Form 3.05)

FORM PTO-1449	ATTY. DOCKET NO.	SERIAL NO.
	1094893-1	08/411,369
LIST OF PATENTS AND PUBLICATIONS FOR APPLICANT'S INFORMATION DISCLOSURE STATEMENT	APPLICANT Giordano Beretta et al.	
	FILING DATE	GROUP
(Use several sheets if necessary)	03/27/95	2615

OTHER REFERENCES (Including Author, Title, Date, Pertinent Pages, etc.)

✓	(B)	AR	David L. McLaren et al., "Removal of Subjective Redundancy from DCT-Coded Images", IEE Proceedings-I, 138,5, 345-350, 1991.
✓	(B)	AS	I. Miyagawa et al., "Color-Facsimile System for Mixed-Color Documents", SID 94 Digest, 887-890, 1994.
✓	(B)	AT	Kathy T. Mullen, "The Contrast Sensitivity of Human Colour Vision to Red-Green and Blue-Yellow Chromatic Gratings", J. Physiol., 359, 381-400, 1985.
✓	(B)	AU	Daivd H. Parish et al., "Object Spatial Frequencies, Retinal Spatial Frequencies, Noise, and the Efficiency of Letter Discrimination", Vision Res., 31, 7/8, 1399-1415, 1991.
✓	(B)	AV	Denis G. Pelli et al., "Visual Factors in Letter Identification", IS&T's 47th Annual Conference/ICPS, p. 411, 1994.
✓	(B)	AW	Heidi A. Peterson et al., An Improved Detection Model for DCT Coefficient Quantization", Human Vision, Visual Processing, and Digital Display IV, 1913, 191-201, SPIE, 1993.
✓	(B)	AX	Ricardo L. de Queiroz et al., "Human Visual Sensitivity-Weighted Progressive Image Transmission Using the Lapped Orthogonal Transform", J. Electronic Imaging, 1, 3, 328-338, 1992.
✓	(B)	AY	Ricardo L. de Queiroz et al., Modulated Lapped Orthogonal Transforms in Image Coding", Digital Video Compression on Personal Computers: Algorithms and Technologies, 2187, 80-91, SPIE, 1993.
✓	(B)	AZ	Robert J. Safranek et al., "A Perceptually Tuned Sub-Band Image Coder with Image Dependent Quantization and Post-Quantization Data Compression", Proc. ICASSP 89, 3, 1945-1948, 1989.
✓	(B)	BA	Robet J. Safranek, "JPEG Compliant Encoder Utilizing Perceptually Based Quantization", Human Vision, Visual Processing, and Digital Display V, 1913, 117-126, SPIE, 1993.
✓	(B)	BB	Andrew B. Watson, "DCT Quantization Matrices Visually Optimized for Individual Images, Human Vision, Visual Processing, and Digital Display IV, 1913, 202-216, SPIE, 1993.
✓	(B)	BC	Andrew B. Watson et al., "Discrete Cosine Transform (DCT) Basis Function Visibility: Effects of Viewing Distance and Contrast Masking, Human Vision, Visual Processing, and Digital Display V, 2179, 99-108, SPIE, 1994.
		BD	

EXAMINER
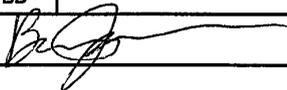
DATE CONSIDERED 11/4/96

Rev 5/90 (Form 3.05)

United States Patent [19]

Ligtenberg

[11] Patent Number: 5,333,212

[45] Date of Patent: Jul. 26, 1994

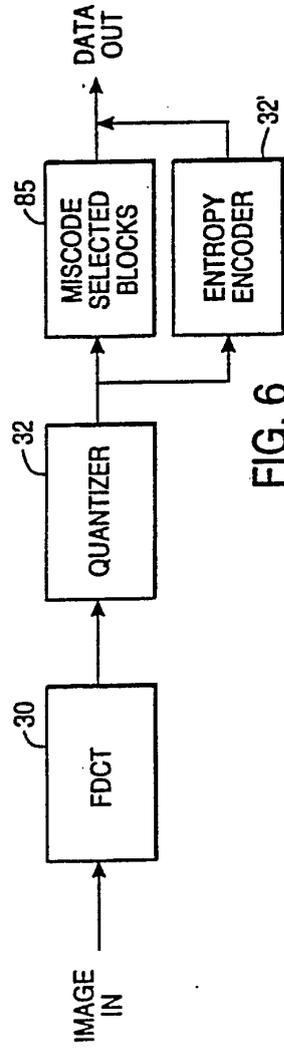[54] **IMAGE COMPRESSION TECHNIQUE WITH REGIONALLY SELECTIVE COMPRESSION RATIO**

[75] Inventor: Adrianus Ligtenberg, Palo Alto, Calif.

[73] Assignee: Storm Technology, Mountain View, Calif.

[21] Appl. No.: 978,312

[22] Filed: Nov. 17, 1992

Related U.S. Application Data

[63] Continuation of Ser. No. 664,256, Mar. 4, 1991, abandoned.

[51] Int. Cl.⁵ G06K 9/36; G06K 9/46; H04N 7/12

[52] U.S. Cl. 382/56; 348/390; 348/418

[58] Field of Search 358/133, 135, 433; 382/56; G06K 9/36, 9/46; H04N 1/04, 1/40

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,286,256	8/1981	Langdon, Jr. et al.	340/347
4,295,125	10/1981	Langdon, Jr.	340/347
4,463,342	7/1984	Langdon, Jr. et al.	340/347
4,467,317	8/1984	Langdon, Jr. et al.	340/347
4,633,490	12/1986	Goertzel et al.	375/122
4,652,856	3/1987	Mohiuddin et al.	340/347
4,691,329	9/1987	Juri et al.	382/56
4,698,689	10/1987	Tzou	358/433
4,797,741	1/1989	Sato et al.	382/56
4,805,030	2/1989	Tanaka	358/433
4,870,497	9/1989	Chamzas et al.	358/426
4,873,577	10/1989	Chamzas	358/426
4,903,317	2/1990	Nishihara et al.	382/56
4,922,273	5/1990	Yonekawa et al.	382/56
5,014,198	5/1991	Umemura	382/56

OTHER PUBLICATIONS

Mark R. Jaworski, "Imaging for Military Communications", Advanced Imaging Military Communications, May 1991, pp. 18, 21.
Connie Guglielmo, "PicturePress compresses in ratios up to 100-to-1", News MACWEEK, Nov. 6, 1990.

Primary Examiner—Jose L. Couso
Attorney, Agent, or Firm—Townsend and Townsend Khourie and Crew

[57] **ABSTRACT**

An enhancement to a standard lossy image compression technique wherein a single set of side information is provided to allow decompression of the compressed file. Certain portions of the image are selected (either by the user or automatically) for more compression than other portions of the image. A particular embodiment is implemented for use with the JPEG image compression technique. JPEG calls for subdividing the image into blocks, transforming the array of pixel values in each block according to a discrete cosine transform (DCT) so as to generate a plurality of coefficients, quantizing the coefficients for each block, and entropy encoding the quantized coefficients for each block. Techniques for increasing the compression ratio include subjecting each selected block to a low pass filtering operation prior to the transform, subjecting the coefficients for each selected block to a thresholding operation before the quantizing step, subjecting the coefficients for each selected block to a downward weighting operation before encoding them, or, where the entropy encoding uses Huffman codes, mapping coefficients to adjacent shorter codes.

31 Claims, 3 Drawing Sheets

Microfiche Appendix Included
(2 Microfiche, 163 Pages)

DCT-BASED COMPRESSION ENCODER — 15

8 X 8 BLOCKS

10

SOURCE IMAGE DATA

FDCT — 30

QUANTIZER — 32

QUANTIZATION TABLES — 37

ENTROPY ENCODER — 35

(HUFFMAN TABLES) — 40

HEADER | DATA — 17

COMPRESSED IMAGE DATA

DCT-BASED COMPRESSION DECODER — 20

ENTROPY DECODER — 42

(HUFFMAN TABLES)

DEQUANTIZER — 45

QUANTIZATION TABLES

IDCT — 47

22

RECONSTRUCTED IMAGE DATA

FIG.1 (PRIOR ART)

NUMBER OF BITS	0	1	2	3	...	9	10	
	EOB	S	L	S	...	S	L	0
		S	L	S	...	S	L	0
	...	...	...	...	...	...	...	...
	—	S	L	S	...	S	L	14
	ZRL	S	L	S	...	S	L	15
								NUMBER OF 0's

FIG. 7

60c
60a
60b

FIG. 2

IMAGE IN → FILTER SELECTED BLOCKS (70) → FDCT (30) → QUANTIZER (32) → ENTROPY ENCODER (35) → DATA OUT

FIG. 3

IMAGE IN → FDCT (30) → QUANTIZER (32) → THRESHOLD SELECTED BLOCKS (75) → ENTROPY ENCODER (35) → DATA OUT

FIG. 4

IMAGE IN → FDCT (30) → WEIGHT SELECTED BLOCKS (80) → QUANTIZER (32) → ENTROPY ENCODER (35) → DATA OUT

FIG. 5

IMAGE IN → FDCT (30) → QUANTIZER (32) → MISCODE SELECTED BLOCKS (85) → DATA OUT
ENTROPY ENCODER (32')

FIG. 6

1 2

IMAGE COMPRESSION TECHNIQUE WITH REGIONALLY SELECTIVE COMPRESSION RATIO

This is a continuation of application of Ser. No. 07/664,256 filed Mar. 4, 1991, now abondoned.

MICROFICHE APPENDIX - COPYRIGHT NOTICE

A two-fiche microfiche Appendix 1 containing source code for a software implementation of the JPEG encode, enhanced according to the present invention is included with this application. A portion of the disclosure of this patent document contains material which is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure as it appears in the Patent and Trademark Office patent file or records, but otherwise reserves all copyright rights whatsoever.

BACKGROUND OF THE INVENTION

The present invention relates generally to image compression and more specifically to a technique for differentially compressing portions of the image.

Emerging applications, such as full color (24-bit) desktop publishing, photovideotex, phototransmission, supported by 24-bit color scanners, monitors, printers, and cameras, need data reduction and standards to reach acceptable price performance levels.

One initial focus in the early 1980's, was on the use of photographic images within videotex systems. It was expected that such systems would eventually employ ISDN (64 Kbit/sec) lines for transmission and monitors as softcopy displays. The initial algorithm requirements and the evaluation procedures reflect the early focus. 720 by 576 pixels color images (CCIR 601 format) were selected as test material. Compression goals included good image quality around 1 bits/pixel and "progressive build-up", allowing an early recognition of the image at a lower quality. The image quality evaluation was tied to relatively inexpensive monitors.

In 1986 the Joint Photographic Expert Group (JPEG) was formed as a joint committee of ISO and the CCITT to develop and propose an efficient image compression standard. Soon more target applications were identified, broadening the scope of the standard. It needed to support a wide variety of pixel resolutions, color spaces, and transmission bandwidths. In addition the efficiency of implementation in both software and hardware became an important additional requirement.

Given these requirements, in a competition between twelve algorithms covering a wide spectrum of algorithms, three finalists were selected in June, 1987: block truncation approach, an interpolative spatial domain approach and a transform based method. From these three finalists, the Adaptive Discrete Cosine Transform (ADCT) was unanimously selected in early 1988 as having produced the best picture quality. Since then, a cooperative effort to refine, test and document the DCT-based method has been in progress.

This effort resulted in development of a three part structure, including the baseline system, the extended system, and independent lossless coding function. Technical agreement was reached in the October, 1989 Tokyo meeting, a draft specification was made publicly available in January, 1990, and an ISO Draft Proposed Standard (DPS) is expected for 1991.

The selected baseline algorithm is a lossy technique based on the discrete cosine transform (DCT), a uniform quantizer, and entropy encoding. The transform removes the data redundancy by concentrating most of the information in the first few transform coefficients. The quantizer controls the loss of information and the picture quality. The entropy encoding reduces the entropy of the signal. The JPEG proposed standard includes a baseline system, an extended system, and a separate lossless function. The baseline system represents the default communication mode, and each standard decoder is required to interpret data coded with the baseline system. The extended system provides additional features such as progressive build-up and arithmetic coding. These features can be used when implemented by both the encoder and the decoder.

SUMMARY OF THE INVENTION

The present invention is drawn to a flexible and effective image compression technique that provides maximum compression consistent with maintaining image quality in selected areas. The technique may be implemented for use with a standard image compression technique, and the compressed image file may be decompressed with any hardware or software that is compatible with the standard.

The invention operates in the context of a lossy image compression technique wherein a single set of side information is provided to allow decompression of the compressed file. According to the invention, certain portions of the image are selected (either by the user or automatically) for more compression than other portions of the image. The elimination of bits in regions selected for higher levels of compression is performed in a manner that has relatively minimal impact on visual quality. This selective or intelligent compression, using the bits where they are the most effective, allows the user to maximize image quality, compress to a fixed file size, or ensure an upper bound on the reconstruction error between the original and the decompressed image.

A particular embodiment of the invention is implemented for use in conjunction with the JPEG image compression technique. The JPEG proposed standard calls for subdividing the image into blocks, transforming the array of pixel values in each block according to a discrete cosine transform (DCT) so as to generate a plurality of coefficients, quantizing the coefficients for each block, and entropy encoding the quantized coefficients for each block. Within this context, one or more of a number of techniques for rejecting weak components are used to selectively reduce the number of bits in the compressed image for that block. These techniques include subjecting each selected block to a low pass filtering operation prior to the transform, subjecting the coefficients for each selected block to a thresholding operation before or after the quantizing step, subjecting the coefficients for each selected block to a downward weighting operation before encoding them, or, where the entropy encoding uses Huffman codes, mapping coefficients to adjacent shorter codes. These techniques tend to eliminate weak components or components having high spatial frequencies, and thus provide significant compression while maintaining visual quality.

A further understanding of the nature and advantages of the present invention may be realized by reference to

the remaining portions of the specification and the attached drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a high level block diagram showing the compression and decompression of a source image according to the JPEG standard;

FIG. 2 shows an image subdivided into regions for differential compression;

FIG. 3 is a block diagram showing an embodiment of the invention utilizing selective filtering;

FIG. 4 is a block diagram showing an embodiment of the invention utilizing selective thresholding;

FIG. 5 is a block diagram showing an embodiment of the invention utilizing selective weighting;

FIG. 6 is a block diagram showing an embodiment of the invention utilizing selective miscoding; and

FIG. 7 shows a specially constructed Huffman code table that allows controlled miscoding for differential compression.

BRIEF DESCRIPTION OF THE APPENDICES

Appendix 1 (microfiche-2 fiches) is a source code listing of a computer program for performing the image compression according to the present invention; and

Appendix 2 (paper copy) is a specification of the proposed JPEG standard.

DESCRIPTION OF SPECIFIC EMBODIMENTS

Introduction

As a threshold matter, it is useful to distinguish image compression and data compression. Data compression assumes that every bit of information in a file is important, so every bit is retained when the file is compressed. The algorithm simply stores the description of the data in a more efficient format. Because no data is lost during compression, data compression algorithms are called lossless. As a rule, these programs are capable of achieving compression ratio limits of roughly 8:1, with an average ratio of 2:1. Image compression algorithms, on the other hand must compress files at ratios from 10:1 to 100:1 in order to effectively solve the problem of huge graphic files. To achieve these higher compression rates, image compression algorithms must be lossy—they must assume that some of the data in an image file is unnecessary (or can be eliminated without affecting the perceived image quality). Since compression involves removing some of the image data from a file, compressed files don't retain all the quality of the original image (although the difference is visually indistinguishable at compression ratios up to 10:1).

The present invention is drawn to an image compression technique wherein regions of the image may be selected for relatively greater compression relative to other regions. Although increasing the compression ratio normally lowers the image quality, the invention utilizes techniques that have relatively little visual impact on the selected regions. In many cases, there are only localized portions whose high quality (information content) must be preserved.

There are a number of ways in which a color image can be broken into components. Standard monitors use the RGB characterization where R, G, and B are the red, green and blue components. Standard television broadcasting (NTSC) uses the YUV characterization where Y is the luminance component and U and V are the chrominance components (approximately red and blue). Printers use the CMYK characterization where

C, M, Y, and K are the cyan, magenta, yellow, and black components. The CCIR 601 standard describes a linear transformation between the RGB characterization and the YUV characterization,

The present invention is currently implemented as an enhancement of the existing JPEG (Joint Photographic Expert Group) image compression standard. Although the invention can be implemented in the context of other image compression techniques, the majority of the discussion below will be within the specific context of the proposed JPEG standard. A complete specification of the proposed standard has been filed with this application as an Appendix 2.

JPEG Overview

FIG. 1 is a high level block diagram illustrating the basic operations in the compression, transmission, and reconstruction of a source image. The source image is represented by one or more components, each of which includes an array of multi-bit pixels. A grayscale image would include a single component while a color image would include up to four components. The operations shown apply to each component.

Data representing a source image component 10 is communicated to a compression encoder 15 to provide compressed image data 17. This data may be stored as a file for subsequent retrieval and reconstruction, or it may be transmitted on some communication medium to a remote location for immediate or subsequent reconstruction. In any event, it is contemplated that the compressed image data will be communicated to a compression decoder 20 to provide reconstructed image data 22. Compression encoder 10 uses certain data structures for the compression, and relevant portions of these must be communicated as side information for use by compression decoder in the image reconstruction. The particular compression technique under discussion contemplates a single set of side information that applies to the entire image component.

In the proposed JPEG standard, compression encoder includes a forward discrete cosine transform (FDCT) stage 30, a quantizer 32, and an entropy encoder 35. The compression decoder includes an entropy decoder 42, a dequantizer 45, and an inverse discrete cosine transform (IDCT) stage 47. In the JPEG standard, the entropy encoder is lossless and use Huffman coding in the baseline system. As will be described below, certain embodiments of the present invention introduce lossiness into the entropy encoder. The side information includes a quantization table 37 used by quantizer 32, and, where entropy encoder 15 is a Huffman encoder, a set of Huffman code tables 40.

The image component is divided into blocks of 8 pixels by 8 pixels, and each block is separately processed. Positions within an 8×8 block are denoted by a double subscript, with the first subscript referring to the row and the second subscript referring to the column. Numbering is from left to right and from top to bottom so that the upper left corner is (00), the upper right corner is (07), the lower left corner is (70), and the lower right corner is (77).

The DCT stage reduces data redundancy, producing one DC coefficient and 63 AC coefficients for each block. The DCT equations for the forward and inverse transforms are as follows:

FDCT:

-continued

$$S_{vu} =$$

$$(C_u C_v/4) \sum_{x=0}^{7} \sum_{y=0}^{7} s_{yx} \cos((2x + 1)u\pi/16) \cos((2y + 1)v\pi/16)$$

IDCT:

$$s_{yx} =$$

$$(1/4) \sum_{u=0}^{7} \sum_{v=0}^{7} C_u C_v S_{vu} \cos((2x + 1)u\pi/16) \cos((2y + 1)v\pi/16)$$

where

S_{vu}=transform coefficient at (vu) in the coefficient table;

s_{yx}=pixel value at (yx) in the 8×8 block; and $C_u, C_v = 1/\sqrt{2}$ for u,v=0 and 1 otherwise.

As can be seen from the equation for the forward transform, S_{00} is the DC component, being proportional to the sum of all the pixel values. Larger values of u and v correspond to higher spatial frequency components. Since most images tend to be characterized by lower spatial frequencies, the coefficient table tends to have the largest values toward the upper left hand corner. While nominally lossless, the DCT results in a small loss due to the inherent inability to calculate the cosine terms with perfect accuracy.

The quantizer operates to reduce the information, and hence the image quality, by a known amount. The quantization is performed by dividing each coefficient in the 8×8 coefficient table by the corresponding entry in the 8×8 quantization table, and rounding the result to the nearest integer. This reduces the magnitude of the coefficients and increases the number of zero value coefficients. The step-size of the quantizer is determined by a "visibility threshold matrix". Step size is varied according to the coefficient location and can be tuned for each color component to optimize the quantization for human perception. A lossless quantizer would have all entries equal to 1, which would mean no quantization at all.

Next the difference between the current DC coefficient and the DC coefficient of the previous block is Huffman encoded to reduce statistical redundancy. The coding model for the AC coefficients rearranges them in a zig-zag pattern as follows:

0	1	5	6	14	15	27	28
2	4	7	13	16	26	29	42
3	8	12	17	25	30	41	43
9	11	18	24	31	40	44	53
10	19	23	32	39	45	52	54
20	22	33	38	46	51	55	60
21	34	37	47	50	56	59	61
35	36	48	49	57	58	62	63

When the quantized AC coefficients are ordered in the zig-zag pattern described above, the string of quantized coefficient values will be characterized by runs of 0's interrupted by non-zero values. Although the individual coefficients are 10-bit numbers, the actual coefficient value will typically be a number of fewer bits with leading bits that are 0's. A unique code is assigned to each possible combination of run length and number of bits. The code is thus characterized by an integer pair, namely run/size. The run length is constrained by the number of AC coefficients (63), but in order to keep the code tables more manageable, a maximum run of 15 is permitted. Combinations with a zero size are undefined, but 0/0 is assigned a code to designate end of block

(EOB) and 15/0 is assigned a code to specify a zero run length (ZRL) of 15 followed by one or more 0's.

The string of coefficient values is broken down into groups, each of which comprises a run of 0's followed by a non-zero value. The code for the particular run/size is placed in the data file, followed by the binary representation of the numerical value of the coefficient without leading 0's. Runs of more than 15 0's are handled by inserting the requisite number of ZRL codes. Decoding is accomplished by sensing a valid code, determining from it the number of data bits, and extracting the data value, which immediately follows the code.

As is typical for Huffman coding, short codes are chosen for the more probable events and longer codes are chosen for the less probable events. The codes are constrained so that no short code is the leading portion of a longer code.

The baseline system specifies two sets of typical Huffman tables, one intended for the luminance or achromatic (grayscale) component and one for the chromatic (color) components. Each set has two separate tables, one for the DC and one for the AC coefficients. Tables 1A and 1B show the code lengths (not the codes themselves) for these typical code tables. The tables are organized in rows corresponding to the number of 0's and columns corresponding to the number of bits in the values. As can be seen, the most frequent events, which are mapped to short codes, are concentrated in the upper left corner and along the top and left side edges. This reflects the fact that most of the quantized AC coefficients tend to be zero or small numbers (few bits).

The encoder of the baseline system may operate in 2 modes: a) one-pass encoding using default Huffman tables or custom, pre-calculated Huffman tables, or b) two-pass encoding, where during the first pass the encoder determines the optimal Huffman table specific for the image being encoded.

The JPEG syntax specifies that the baseline can process the 8-bit pixel data in either block interleaved or on a color component basis with no restrictions on the horizontal and vertical dimensions. In addition the baseline can handle at most four different color components and in block interleaved mode up to ten subsampled 8×8 blocks.

The extended system provides a set of additional capabilities, including progressive build-up and arithmetic coding, to meet the needs of applications requiring more than the baseline system functionality. These additional features are not required for all applications, and consequently, to minimize the cost of JPEG baseline compatibility, they are defined only in the Extended System. The extended system is a superset of the baseline system limits, by handling pixels of greater than 8 bits precision, additional data interleave schemes, and more color components.

Although the JPEG standard does not provide a priori compression ratios, there exist published quantization tables that give generally predictable levels of compression. To the extent that an image must be compressed by a defined amount (for example, to fit within a maximum file size), it is possible to perform an iterative procedure as follows. First, compress the image using an appropriate quantization table. Note the resultant size, and scale the quantization table accordingly. Compress the image with the scaled quantization table, and repeat the scaling and compression steps as necessary.

Variable Image Quality

While compressing an image approximately 10:1 typically results in little loss of perceptible visual quality, practical benefit typically requires compression ratios significantly greater than 10:1. However, such large compression ratios are likely to degrade the image, or at least portions thereof, to an unacceptable level. For example, text areas within the image might become unreadable.

The present invention addresses this dilemma by allowing different selected portions of an image to be compressed at different compression ratios so that portions that require extreme quality can be compressed at relatively low ratios while background portions and portions conveying little information can be compressed at higher ratios. The selection can be performed by the user, or in some instances can be done automatically, as for example if it is known ahead of time which portions of the image must be maintained with substantially no loss in visual quality.

FIG. 2 is a stylized view illustrating the selective compression according to the present invention and the type of file compression achievable thereby. A source image 60 is shown as having been subdivided into three regions, designated 60a, 60b, and 60c. These portions of the image are designated to receive different degrees of image compression. While the figure shows contiguous regions for the different compression ratios, it should be understood that the image can be subdivided in any way, depending on the regions of the image that are to be compressed at different ratios.

In the specific example, assume that region 60a includes fine detail, and thus requires excellent quality upon compression and decompression. In such an instance, a compression ratio of 5:1 is appropriate. Further assume that region 60b requires at most good quality, and can be compressed with a compression ratio of 20:1. Similarly, assume that region 60c contains very little detail, and can be compressed with a compression ration of 30:1. If each of regions 60a and 60b occupies ⅛ of the image area and region 60c occupies ¾ the overall or effective compression ratio is about 18:1. Typically, the regions that require the lowest level of compression will be relatively small, and therefore the overall compression ratio may be higher yet.

The procedure can be interactive with the user working at the screen (computer monitor) to select the regions and specify the relative quality levels. In a current implementation on an Apple Macintosh personal computer, the user can select regions by using a graphical interface such as the Quickdraw tools developed by Apple. The original image is on the screen, and using standard drawing tools such as the rectangular outline or the lasso tool, the user indicates with the mouse on the screen the area of interest. Depending on the action, more regions can be added to the selected region or can be deleted from the selected region. For the specific region the user determines a quality setting. All 8×8 pixel blocks of which a pixel is part of the outlined regions are marked. Those markers then are used as indicators how to process the blocks to the selected quality levels as described below.

The user is then able to see the result of the compression, both in visual terms (the compressed image is decompressed and displayed on the screen), and in numerical terms (the resultant compressed file size is

shown). If either aspect of the result is unsatisfactory, the user can modify the input parameters and try again.

An automatic mode is based on the final goal (e.g., fixed file size or bounded reconstruction error). The visually important reconstruction errors are detected and the quality is adapted until the quality criteria is reached. Most of those errors will occur around edges. If a fixed file size is required the quality level is adjusted continuously such that the total amount of compression bis varies within about 10% of the goal. The various techniques described below have numerical parameters that can be varied to fine tune the compression.

While it is possible to treat each region as a separate image, and apply a suitable image compression to that image, storing or transmitting the appropriate side information with the compressed data, the present invention takes a different approach. One or a number of various techniques, to be described below, are used for selectively removing more or less information from the different regions. The compression for all regions is characterized by a single set of side information, and this single set is stored or transmitted with the file. This saves some file space, and further avoids the need to send or store information regarding the way the image was segmented for variable quality. Moreover, and perhaps most important, the proposed JPEG standard contemplates a single set of side information, and provides no mechanism for piecing together segments of an image.

The following sections describe various embodiments of the invention, each directed to a particular technique for increasing the compression ratio of selected blocks. It should be understood that the techniques are applicable individually or in combination. Moreover, different individual techniques of combinations may be appropriate for different portions of the image.

Low Pass Filtering

FIG. 3 is a high level block diagram illustrating an embodiment of the invention for reducing the quality for selected blocks. In this embodiment, selected blocks are communicated to a low-pass filtering stage 70 prior to DCT stage 30.

A simple type of filter is a moving average filter, which removes high frequency spatial components by substituting for a given pixel value the average value of some number of neighboring pixels. The larger the number of neighboring pixels that are averaged, the greater the degree of filtering. It is preferred to use a rectangular filter that averages an array of M rows and N columns. In this regime, the filtering operation is described by the following equation:

$$s'_{yx} = \sum_{i=0}^{N} \sum_{j=0}^{M} s_{(y+i)(x+j)}/MN$$

Where s'_{yx} is the average value of the M×N block of pixels with the upper left corner being at (yx). Since this moving block average must be computed for each pixel, significant efficiencies can be achieved by recognizing that most of the computation has already been performed for a given average. For example, once the average for a given pixel position has been calculated, the average value for the pixel one row below it may be computed by taking the previous average value, subtracting the contribution from the previous row and

adding the contribution of the new row. This may be seen as follows:

$$s'_{yx} = s'_{(y-1)x} + \sum_{m=0}^{N} s_{(y+1)(x+m)}/N - \sum_{m=0}^{N} s_{(y-1)(x+m)}/N$$

A similar extension applies to horizontally adjacent pixels.

The above equations result in averaging the pixels where the center of the rectangle is diagonally offset below and to the right of the pixel being processed. It is preferred to use an odd number of pixels for each dimension of the filter rectangle, and to center the filter at the pixel. Accordingly, if the rectangle is considered to be $(2m+1) \times (2n+1)$, the expression for s'_{yx} becomes:

$$s'_{yx} = \sum_{i=-n}^{n} \sum_{j=-m}^{m} s_{(y+i)(x+j)}/((2n+1)(2m+1))$$

Since the filtering operation extends over block boundaries, it tends to smooth the transitions between regions that are being subjected to different filtering.

It is possible to apply different filters to different portions of the image. For example, four different quality levels can be established by no filter for regions of maximum quality, a 5×5 filter for regions of high quality, an 11×11 filter for regions of medium quality, and a 17×17 filter for regions of low quality. If more computing power is available, other types of linear or non-linear low pass filters can be used.

Thresholding and Downward Weighting of Coefficients

FIG. 4 is a high level block diagram illustrating another embodiment of the invention for reducing the quality for selected blocks. In this embodiment, the AC coefficients for the selected blocks are communicated to a thresholding stage 75 prior to entropy encoder 35. It is also possible to threshold before quantizing, but thresholding after is preferred since the quantized results include a visual weighting (larger divisors for higher frequencies).

A threshold level of 3 or 4 will tend to give high quality (say 16:1 compression) while a threshold level of 6 will give good quality (say 25:1 compression). In a current implementation, a single threshold is used for all the AC coefficients. A representative code sequence is as follows:

```
# define THRS 5
int S[8][8];          /* declare 8 by 8 array of integers */
void main(void)
{
    int index_v,index_u;
    /* initialize array to desired values here */
    /* test array against threshold (THRS) */
    for(index_v = 0;index_v < 8;index_v++)
    {
        for(index_u = 0;index_u < 8;index_u++)
        {
            if(index_v == 0 && index_u == 0) continue;
            if(S[index_v][index_u] < THRS)
                S[index_v][index_u] = 0;
        }
    }
}
```

The result of the thresholding operation is to set the AC coefficients having smaller absolute values to zero. It is noted that the quantizer performs a similar function in

that it sets to zero those coefficients that are below their respective quantization step sizes.

An extension of this basic technique is to apply different threshold values depending on the particular AC coefficient. For example, using larger threshold values for the higher frequency AC coefficients has an effect similar to low pass filtering.

FIG. 5 is a schematic flow diagram illustrating a quality reduction technique that can be used either in conjunction with or instead of the thresholding technique. In this embodiment, the AC coefficients are communicated to a downward weighting stage 80 prior to quantizer 32. While a single weighting coefficient could be used, the effect of low pass filtering can be achieved by using larger downward weighting factors for the higher frequency AC coefficients.

Huffman Miscoding for Shorter Code Words

FIG. 6 is a high level block diagram of another embodiment of the invention for reducing the quality for selected blocks. This embodiment contemplates formulating special Huffman code tables, suitable for providing lossless encoding, and selectively miscoding certain quantized coefficient values in the blocks where greater compression is desired. To this end, the lossless entropy encoder, designated 32', has an associated lossy entropy encoder 85.

FIG. 7 shows a specially constructed Huffman code table incorporating a technique for allowing controlled miscoding for those blocks where greater compression is desired. The table is organized in rows corresponding to the number of 0's and columns corresponding to the number of bits in the values. Zero-bit numbers for non-zero values do not occur and as a result no codes are defined for no bits. However, the special case of no 0's and no bits (i.e., 0/0) is defined as the end of block (EOB), and the special case of 15/0 is used to signify the run of 15 zeroes followed by a further run of at least one 0. Thus codes are assigned for EOB and ZRL, but not for the other 0-bit values.

The general procedure for assigning Huffman codes is to assign the shortest codes to the events with the highest probabilities and the longest codes to the events with the lowest probabilities. This embodiment of the invention contemplates creating a code table where adjacent pairs of entries are characterized by short and long codes, and for extra compression in selected blocks, values that would correctly be coded with the long code are miscoded with an adjacent short code.

The procedure for generating the special Huffman code table is as follows. Each position in the code table has a certain associated probability. For a given adjacent pair of positions, which are characterized by the same run length and consecutive odd and even integer sizes, the joint probability is determined and assigned to the odd size position. Based on these enhanced probabilities, the odd size positions in the pair will get a short code. A 16-bit code is assigned to the even size position in the pair. For example, for run/size =0/1, the sum of the probabilities for 0/1 and 0/25 is assigned to 0/1 and essentially zero probability is assigned to 0/2. Thus the code table will have the shortest codes in the odd columns and 16-bit codes in the even columns.

This is the code table that is sent as the side information, and it will provide lossless encoding of blocks of quantized coefficients. However, for those blocks where a higher compression ratio is desired, the entropy

encoding is made lossy by miscoding the values that require an even number of bits. Specifically, such a value will be assigned the code for an adjacent odd size and the value for the closest value with that code. For example, consider the possible values from 1 to 15. The value 1 is a 1-bit number, the values 2-3 are 2-bit bit numbers, the values 4-7 are 3-bit numbers, and the values 8-15 are 4-bit numbers. For the lossy miscoding, the value 2 will get the short code for a 1-bit value and the value 1, the value 3 will get the short code for a 3-bit value and the value 4, the values 8-11 will get the short code for a 3-bit value and the value 7, while the values 12-15 will get the short code for a 5-bit value and the value 16.

Current Implementation

The thresholding embodiment of the invention is implemented in software on a digital computer. Appendix 1 (© 1990, Storm Technology, Inc. is a set of microfiches containing a source code program for an image compression encoder according to the proposed JPEG standard, as extended by incorporation of this embodiment. The program is in the "C" language, well known to those of skill in the art. The program was written to run on an Apple Macintosh personal computer, although it will be apparent to those of skill in the art that a wide variety of programming languages and hardware configurations could readily be used based on this disclosure without departing from the scope of the invention.

Conclusion

In conclusion, it can be seen that the present invention provides a simple and effective approach to extending an image compression standard to provide truly customized compression. The invention allows maximum compression consistent with preserving detail in critical areas of the image.

While the above is a complete description of various embodiments of the invention, including a software implementation of one of the embodiments, various modifications, alternatives, and equivalents may be used. For example, a software implementation as disclosed provides a relatively inexpensive solution to the problem. A faster (say by a factor of 10) solution may be provided by use of a hardware accelerator in the computer. A preferred implementation of the hardware accelerator includes one or more programmed digital signal processors (DSP's). While special purpose hardware could be designed for such an accelerator board, it is believed that a DSP based system provides greater flexibility for upgrades. Performance does not come free of cost, and a hardware implementation is likely to cost 5 times as much as the software implementation.

Therefore, the above description and illustrations should not be taken as limiting the scope of the invention which is defined by the appended claims.

TABLE 1A

Huffman Code Lengths for Luminance AC Coefficients

BITS IN VALUE

	0	1	2	3	4	5	6	7	8	9	10
0	4*	2	2	3	4	5	7	8	10	16	
N 1	—	4	5	7	9	11	16				
U 2	—	5	8	10	12	16					
M 3	—	6	9	12	16						
B 4	—	6	10	16							
E 5	—	7	11	16							
R 6	—	7	12	16							
O 7	—	8	12	16							
F 8	—	9	15	16							
9	—	9	16								
Z 10	—	9	16								
E 11	—	10	16								
R 12	—	10	16								
O 13	—	11	16								
14	—	16									
S 15	11**	16									

All codes in this region are 16 bits long.

*EOB
**ZRL

TABLE 1B

Huffman Code Lengths for Chrominance AC Coefficients

BITS IN VALUE

	0	1	2	3	4	5	6	7	8	9	10
0	2*	2	3	4	5	5	6	7	9	10	12
1	—	4	6	8	9	11	12	16			
2	—	5	8	10	12	15	16				
3	—	5	8	10	12	16					
4	—	6	9	16							
5	—	6	10	16							
6	—	7	11	16							
7	—	7	11	16							
8	—	8	16								
9	—	9	16								
10	—	9	16								
11	—	9	16								
12	—	9	16								
13	—	11	16								
14	—	14	16								
15	10**	15	16								

All codes in this region are 16 bits long.

*EOB
**ZRL

What is claimed is:

1. A method of lossy image compression comprising the steps of:

accepting an image into a digital processor;

using the digital processor to subdivide a component of the image into a plurality of blocks;

using the digital processor to select a subset containing fewer than the plurality of blocks for preferentially greater compression;

using the digital processor to encode each block of the plurality according to a defined compression regime while modifying in a particular way at least a portion of the compression regime as applied to each block of the subset, the compression regime, as modified in the particular way, being applied only to the blocks of the subset;

using the digital processor to provide a single set of side information, all of which applies to every one of the encoded blocks of the plurality notwithstanding the fact that during said step of using the digital processor to encode each block of the plurality, at least a portion of the compression regime

was modified as applied to each block of the subset; and

using the digital processor to produce compressed image data comprising the encoded blocks and the single set of side information.

2. The method of claim 1 in which the step of using the digital processor to select a subset of the blocks for preferentially greater compression is carried out interactively with a user and incorporates accepting into the digital processor information from the user regarding which blocks are to be included in the subset.

3. The method of claim 1 in which the step of using the digital processor to select a subset of the blocks for preferentially greater compression is carried out by the digital processor without user interaction.

4. The method of claim 1 in which the single set of side information comprises a quantization table and a Huffman code table.

5. The method of claim 1 in which the encoded quantized coefficients and the single set of side information conform to the JPEG standard.

6. The method of claim 1 wherein the compressed image data is transmitted as a data stream without being stored as a data file.

7. The method of claim 1, and further comprising the steps of:

using the digital processor to select an additional different subset containing fewer than the plurality of blocks for a different degree of preferentially greater compression; and

within the step of using the digital processor to encode each block of the plurality, modifying in a different particular way at least a portion of the compression regime as applied to each block of the additional different subset.

8. A method for compressing an image component comprising the steps of:

accepting the image component into a digital processor;

using the digital processor to subdivide the image component into a plurality of blocks, each block comprising an array of pixel values;

using the digital processor to select a subset containing fewer than the plurality of blocks for preferentially greater compression;

using the digital processor to subject each block of the subset to a low pass filtering operation;

using the digital processor to transform the array of pixel values in each block so as to generate a plurality of coefficients for each block;

using the digital processor to quantize the coefficients for each block;

using the digital processor to encode the quantized coefficients for each block;

using the digital processor to provide a single set of side information, all of which applies to every one of the encoded blocks of the plurality; and

using the digital processor to produce compressed image data comprising the encoded quantized coefficients for each block and the single set of side information.

9. The method of claim 8 in which the encoded quantized coefficients and the single set of side information conform to the JPEG standard.

10. A method for compressing an image component comprising the steps of:

accepting the image component into a digital processor;

using the digital processor to subdivide the image component into a plurality of blocks, each block comprising an array of pixel values;

using the digital processor to select a subset containing fewer than the plurality of blocks for preferentially greater compression;

using the digital processor to transform the array of pixel values in each block so as to generate a plurality of coefficients for each block;

using the digital processor to quantize the coefficients for each block;

using the digital processor to encode the quantized coefficients for each block;

using the digital processor to subject the absolute values of the coefficients for each block of the subset to a thresholding operation between the transforming and encoding steps for that block, the thresholding operation causing each coefficient having an absolute value less than a particular value to be set to zero;

using the digital processor to provide a single set of side information, all of which applies to every one of the encoded blocks of the plurality; and

using the digital processor to produce compressed image data comprising encoded quantized coefficients for each block and the single set of side information.

11. The method of claim 10 wherein the thresholding operation for each block in the subset is performed after the quantizing step for that block.

12. The method of claim 10 wherein the thresholding operation for each block in the subset is characterized by different threshold levels for at least two coefficients.

13. The method of claim 10 in which the encoded quantized coefficients and the single set of side information conform to the JPEG standard.

14. A method for compressing an image component comprising the steps of:

accepting the image component into a digital processor;

using the digital processor to subdivide the image component into a plurality of blocks, each block comprising an array of pixel values;

using the digital processor to select a subset containing fewer than the plurality of blocks for preferentially greater compression;

using the digital processor to transform the array of pixel values in each block so as to generate a plurality of coefficients for each block;

using the digital processor to quantize the coefficients for each block;

using the digital processor to encode the quantized coefficients for each block;

using the digital processor to subject the coefficients for each block of the subset to a downward weighting operation between the transforming and encoding steps for that block;

using the digital processor to provide a single set of side information, all of which applies to every one of the encoded blocks of the plurality; and

using the digital processor to produce compressed image data comprising the encoded quantized coefficients for each block and the single set of side information.

15. The method of claim 14 wherein the weighting operation for each block in the subset is performed before the quantizing step for that block.

16. The method of claim 14 wherein the weighting operation for each block in the subset is characterized by different weighting factors for at least two coefficients.

17. The method of claim 14 in which the encoded quantized coefficients and the single set of side information conform to the JPEG standard.

18. A method for compressing an image component comprising the steps of:
accepting the image component into a digital processor;
using the digital processor to subdivide the image component into a plurality of blocks, each block comprising an array of pixel values;
using the digital processor to select a subset containing fewer than the plurality of blocks for preferentially greater compression;
using the digital processor to transform the array of pixel values in each block so as to generate a plurality of coefficients for each block;
using the digital processor to quantize the coefficients for each block;
using the digital processor to generate a Huffman code table in which long and short codes are interspersed;
after the step of using the digital processor to generate a Huffman code table, using the digital processor to encode the quantized coefficients for each block;
during the step of using the digital processor to encode the quantized coefficients, for each block of the subset using the digital processor to miscode a given quantized coefficient by selecting an adjacent Huffman code rather than the Huffman code that is appropriate for the given quantized coefficient if the adjacent Huffman code is shorter than the Huffman code that is appropriate for the given quantized coefficient;
using the digital processor to provide a single set of side information, all of which applies to every one of the encoded blocks of the plurality; and
using the digital processor to produce compressed image data comprising the encoded quantized coefficients for each block and the single set of side information.

19. The method of claim 18, and further comprising the step, performed for each block in the subset in the event that there are more that one adjacent shorter code, of selecting the adjacent code that corresponds to the value that is closest to the value of the coefficient before the coefficient was subjected to the quantizing step.

20. The method of claim 18 in which the encoded quantized coefficients and the single set of side information conform to the JPEG standard.

21. A method for compressing an image component comprising the steps of:
accepting the image component into a digital processor;
using the digital processor to subdivide the image component into a plurality of blocks, each block comprising an array of pixel values;
using the digital processor to transform the array of pixel values in each block so as to generate a plurality of coefficients for each block;
using the digital processor to quantize the coefficients for each block;

using the digital processor to encode the quantized coefficients for each block;
using the digital processor to subject the absolute values of the coefficients for each block to a thresholding operation between the transforming and encoding steps for that block, the thresholding operation causing each coefficient having an absolute value less than a particular value to be set to zero;
using the digital processor to provide a single set of side information, all of which applies to every one of the encoded blocks of the plurality; and
using the digital processor to produce compressed image data comprising the encoded quantized coefficients for each block and the single set of side information.

22. The method of claim 21 wherein the thresholding operation for each block is performed after the quantizing step for that block.

23. The method of claim 21 wherein the thresholding operation for each block is characterized by different threshold levels for at least two coefficients.

24. A method for compressing an image component comprising the steps of:
accepting the image component into a digital processor;
using the digital processor to subdivide the image component into a plurality of blocks, each block comprising an array of pixel values;
using the digital processor to transform the array of pixel values in each block so as to generate a plurality of coefficients for each block;
using the digital processor to quantize the coefficients for each block;
using the digital processor to encode the quantized coefficients for each block;
using the digital processor to subject the coefficients for each block to a downward weighting operation between the transforming and encoding steps for that block;
using the digital processor to provide a single set of side information, all of which applies to every one of the encoded blocks of the plurality; and
using the digital processor to produce compressed image data comprising the encoded quantized coefficients for each block and the single set of side information.

25. The method of claim 24 wherein the weighting operation for each block is performed before the quantizing step for that block.

26. A method for compressing an image component comprising the steps of:
accepting the image component into a digital processor;
using the digital processor to subdivide the image component into a plurality of blocks, each block comprising an array of pixel values;
using the digital processor to transform the array of pixel values in each block so as to generate a plurality of coefficients for each block;
using the digital processor to quantize the coefficients for each block;
using the digital processor to generate a Huffman code table in which long and short codes are interspersed;
after the step of using the digital processor to generate a Huffman code table, using the digital proces-

sor to encode the quantized coefficients for each block;

during the step of using the digital processor to encode the quantized coefficients, for each block using the digital processor to miscode a given quantized coefficient by selecting an adjacent Huffman code rather than the Huffman code that is appropriate for the given quantized coefficient if the adjacent Huffman code is shorter than the Huffman code that is appropriate for the given quantized coefficient;

using the digital processor to provide a single set of side information, all of which applies to every one of the encoded blocks of the plurality; and

using the digital processor to produce compressed image data comprising the encoded quantized coefficients for each block and the single set of side information.

27. The method of claim 26, and further comprising the step, performed for each block in the event that there are more than one adjacent shorter code, of selecting the adjacent code that corresponds to the value that is closest to the value of the coefficient before the coefficient was subjected to the quantizing step.

28. A method of lossy image compression and decompression, operating in the context of a defined compression regime and a defined decompression regime, the method comprising the steps of:

accepting a first image into a digital processor;

using the digital processor to subdivide a component of the first image into a plurality of blocks;

using the digital processor to select a subset containing fewer that the plurality of blocks for preferentially greater compression;

using the digital processor to encode each block of the plurality according to the defined compression regime while modifying in a particular way at least a portion of the compression regime as applied to each block of the subset, the compression regime, as modified in the particular way, being applied only to the blocks of the subset;

using the digital processor to provide a single set of side information, all of which applies to every one of the encoded blocks of the plurality notwithstanding the fact that during said step of using the digital processor to encode each block of the plurality, at least a portion of the compression regime was modified as applied to each block of the subset;

using the digital processor to produce a compressed image data comprising the encoded blocks and the single set of side information;

decompressing the compressed image data according to the defined image decompression regime, using the single set of side information, to produce a second image; and

displaying the second image.

29. The method of claim 28 wherein the step of decompressing the compressed image data is performed using the same digital processor.

30. The method of claim 28 wherein the step of decompressing the compressed image data is performed using a different digital processor than the first-mentioned digital processor.

31. The method of claim 1 wherein the compressed image data is stored as a data file.

* * * * *

[54] ADAPTIVE ZONAL CODER

[75] Inventor: Shepard L. Siegel, Derry, N.H.

[73] Assignee: Datacube Inc., Peabody, Mass.

[21] Appl. No.: 431,548

[22] Filed: Nov. 3, 1989

[51] Int. Cl.^{5} G06K 9/00
[52] U.S. Cl. 382/56; 383/50;
358/133
[58] Field of Search 382/56, 49, 27, 50,
382/52; 358/133, 261.1, 261.3, 432, 135, 136

[56] References Cited

U.S. PATENT DOCUMENTS

4,012,715	3/1977	Essenmacher	382/52
4,032,775	11/1981	Widergren et al.	358/136
4,442,454	4/1984	Powell	358/167
4,553,165	11/1985	Bayer	358/167
4,725,885	2/1988	Gonzales et al.	358/135
4,774,574	9/1988	Daly et al.	358/133
4,780,761	10/1988	Daly et al.	358/133
4,807,029	2/1989	Tanaka	358/133
4,821,119	4/1989	Gharavi	358/136
4,860,217	8/1989	Sasaki et al.	364/518
4,861,112	8/1989	Nichols et al.	358/400
4,894,713	1/1990	Delogne et al.	358/133
4,908,862	3/1990	Kaneko et al.	358/135

[57] ABSTRACT

The invention provides a method and apparatus for performing image data compression. Initially, a two-dimensional block of image data in a spatial domain is transformed by a transform coder, resulting in a two-dimensional array of activity coefficients in a frequency domain. The array is then serialized and quantized, yielding a one-dimensional array of coefficients with a leading coefficient and an original trailing coefficient. Next, a portion of the array is selected by choosing a new trailing coefficient based on a setable ratio of the energy of the new short array to the energy of the original array. Lastly, an end-of-block symbol is appended after the new trailing coefficient, and the selected portion of the array is encoded using an entropy coder. The invention allows image data to enjoy enhanced compression with good image fidelity, and allows image fidelity to degrade gracefully in trade for higher degrees of image compression.

9 Claims, 2 Drawing Sheets

IMAGE DATA

TRANSFORM CODER → CONTINUOUS VALUES → QUANTIZER → DISCRETE VALUES → ENTROPY CODER → BIT STREAM

FIG. 1

FIG. 2A

| 1 | 5 | 2 | 3 | 6 | 9 | 13 | 10 | 7 | 4 | 8 | 11 | 14 | 15 | 12 | 16 |

FIG. 2B

| 1 | 5 | 2 | 3 | 6 | 9 | 13 | 10 | 7 | 4 | ☒ |

FIG. 2C

FIG. 3

1
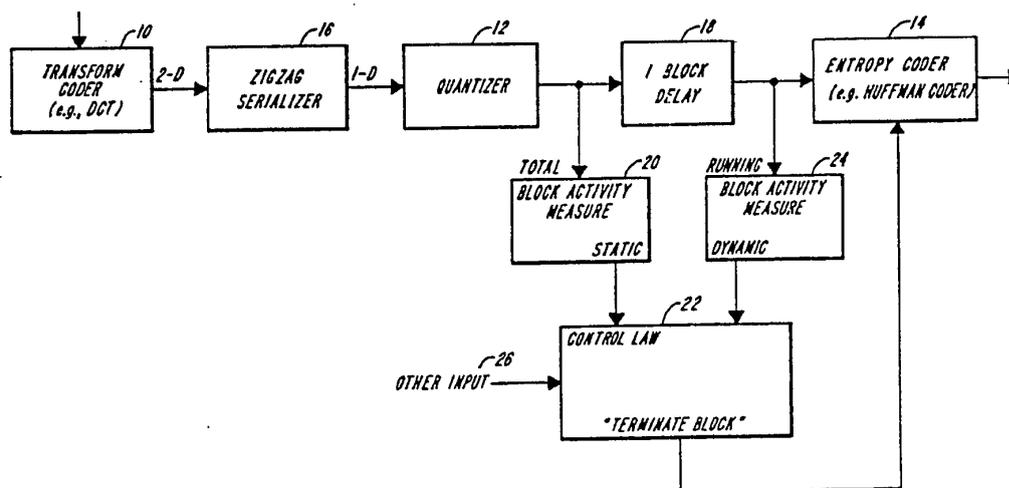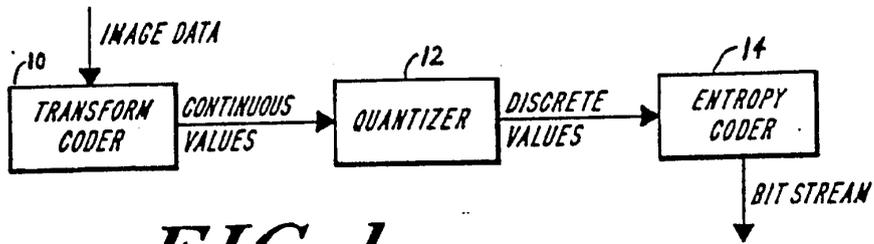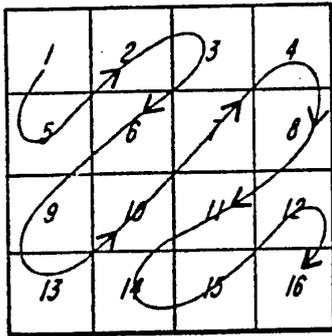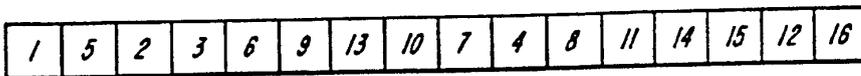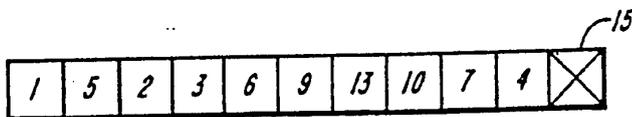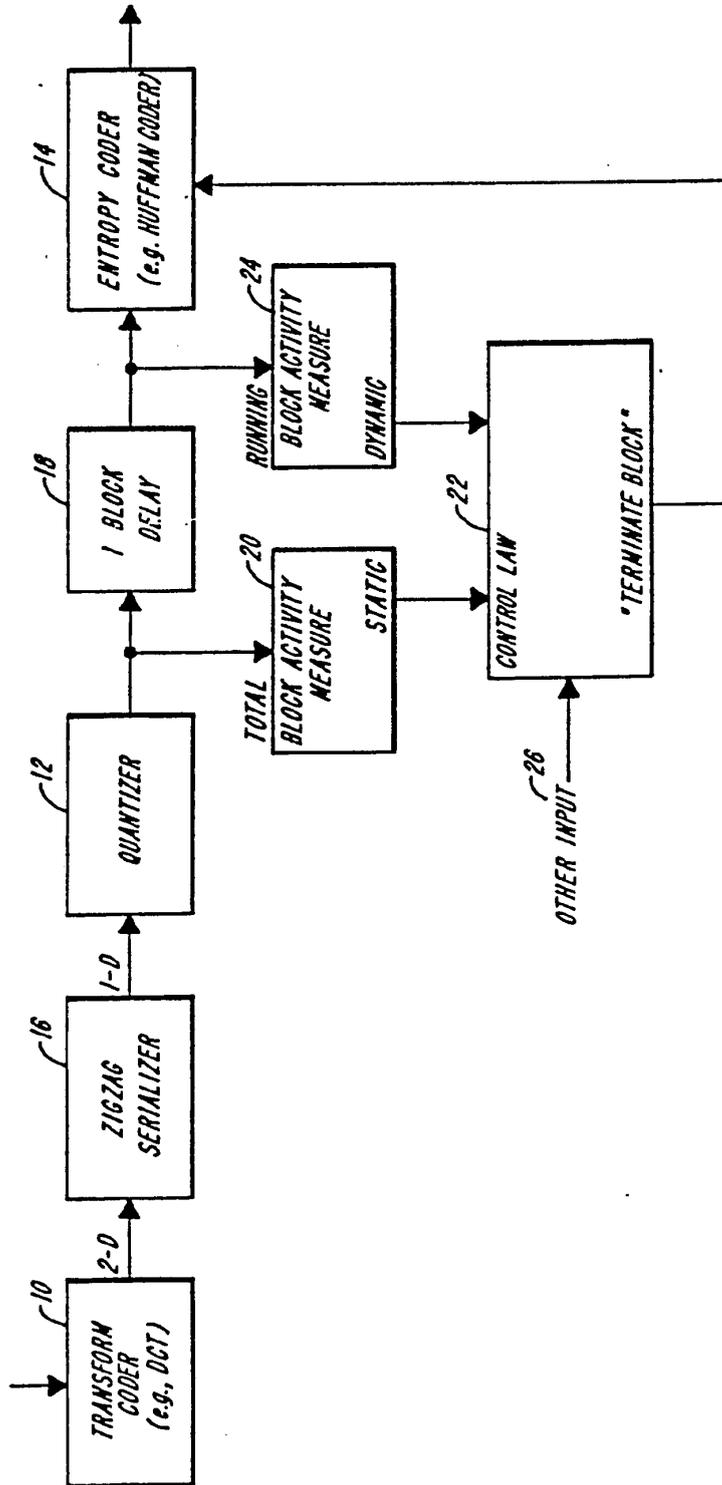
ADAPTIVE ZONAL CODER

FIELD OF THE INVENTION

This invention relates to image data compression and more particularly to adaptive techniques for transform codinq of image data for transmission or storaqe.

BACKGROUND OF THE INVENTION

Many communication systems require the transmission and/or storage of image data. Any technique that minimizes the amount of information required to encode a given image is highly desireable. Transform coding represents a major class of such methods. Meaningful images usually contain a high degree of long-range structure, i.e., they have high inter-element correlations. In transform coding, an image with high inter-element correlations is converted into a collection of uncorrelated coefficients, thereby removing most of the redundant information in the original image. Early transform techniques employed a Fourier-type transform. Current approaches use a discrete cosine transform (DCT) or a Hadamard transform, both of which provide relatively higher coding efficiency.

Commonly, an image to be encoded is partitioned into a plurality of image blocks, and each block is divided into an 8×8 or 16×16 square array. The resulting blocks are encoded one after another. A typical image compression pipeline includes a transform coder, a quantizer, and an entropy coder. A quantizer is used to map a coefficient, falling within a range of continuous values, into a member of a set of discrete quantized values, and an entropy coder uses statistical properties of the information to compress it, i.e., to express the same message using fewer binary digits. For example, a Huffman coder relates the most frequent incoming data symbols to the shortest outgoing codes, and the least frequent incoming data symbols to the longest outgoing codes. After the transform coder transforms an image block into a collection of uncorrelated coefficients, the quantizer provides the corresponding series of discrete values to the entropy coder. The resulting stream of binary digits is then either transmitted or stored.

Although it is possible to retain all the coefficients that result from transform coding an image, it is neither necessary nor desirable. Due to the nature of the human visual system, certain coefficients can be omitted without degrading perceived image quality. By retaining only visually important coefficients, an image with acceptable fidelity can be recovered using significantly less information than provided by the transform coding process.

For a typical two-dimensional source image, the transform operation results in a redistribution of image energy into a set of relatively few low-order coefficients that can be arranged in a two-dimensional array. Visually significant coefficients are usually found in the upper left-hand corner of the array.

According to a known method, referred to as 'zonal coding', only those coefficients lying within a specified zone of the array, e.g., the upper left quarter of the array, are retained. Although significant data compression can be achieved using this method, it is inadequate because picture fidelity is reduced for scenes that contain significant high frequency components.

In an alternative approach, referred to as 'adaptive transform coding', a block activity measure is used to choose an optimum quantizer. Although this method

2

achieves gains in efficiency by providing a degree of quantization precision appropriate to the activity level of a given block, the additional information needed to specify the state of the quantizer must be transmitted along with the usual coefficient data.

SUMMARY OF THE INVENTION

The invention provides a method for image data compression that includes the following steps: transforming a two-dimensional block of image data in a spatial domain to data in a frequency domain. The transformed image data is represented as a two-dimensional array of activity coefficients. The two dimensional array is then serialized, resulting in a one-dimensional array of coefficients with a leading coefficient and an original trailing coefficient. The array is then quantized. Next, a portion of the array is selected beginning with the leading coefficient and ending with a new trailing coefficient that is closer to the leading coefficient than the original trailing coefficient. Lastly, an end-of-block symbol is appended after said new trailing coefficient, and the selected portion of the array is encoded using an entropy coder.

In a preferred embodiment, the portion is selected by first measuring the total activity of the array. A measure of activity of successive sub-portions of the array is determined on-the-fly, and added to a running total. Each new value of the running total is compared to the total activity of the array. Based on this comparison, and in a further preferred embodiment, on a setable level of image quality, a new trailing coefficient is designated, and only the portion of the array bounded by the leading coefficient and the new trailing coefficient is presented for encoding.

Another general feature of the invention is apparatus for inclusion in an image data compression pipeline, the pipeline including a transform coder, a serializer, a quantizer and an entropy coder. The apparatus includes a memory connected to said quantizer and said entropy coder. The memory has an input and an output adapted to store and delay at least a block of transformed, serialized and quantized image data, and is adapted to provide an entropy coder with successive sub-portions of said block of image data. The apparatus also includes a first measurer connected to the quantizer for measuring the total activity of the block of image data, and a second measurer connected to the output of the memory for measuring the activity of successive sub-portions of the image data and computing a running total of the activity. A control law unit is connected to the first and second measurers, and to an entropy coder, and is adapted to compare the running total provided by the second measurer to a level based in part on the total activity provided by the first measurer, and sends an end-of-block symbol to the entropy encoder. The entropy coder is adapted to encode only a portion of the block of image data based on when it receives an end-of-block symbol.

Thus, it is not necessary to transmit additional information along with each block for indicating, e.g., its activity level, or run length. The end-of-block symbol provides an image data receiver with sufficient information for parsing an incoming bit stream back into blocks of image data.

BRIEF DESCRIPTION OF THE DRAWINGS .

The invention will be more fully understood by reading the following detailed description, in conjunction with the accompanying drawings, in which:

FIG. 1 is a block diagram of a typical image compression pipeline;

FIG. 2A is a representation of a 4×4 array of coefficients;

FIG. 2B is a representation of a 1×16 array of coefficients corresponding to the array of FIG. 2A;

FIG. 2C is a representation of a truncated array of coefficients corresponding to the array of FIG. 2B with an appended end-of-block symbol; and

FIG. 3 is a block diagram of the image compression system of the invention.

DETAILED DESCRIPTION OF THE INVENTION

With reference to FIG. 1, a typical image compression pipeline includes a transform coder 10, such as a discrete cosine transform coder (DCT), as disclosed in Ahmed et al., "Discrete Cosine Transform", IEEE Trans on Computers, Jan 1974, pp 90–93; a quantizer 12, such as a linear quantizer, as described in Wintz, "Transform Picture Coding", section III, Proc. IEEE, vol. 60, pp 809–820; and an entropy coder 14, such as a Huffman coder, as discussed in Huffman, "A Method for the Construction of Minimum-Redundancy Codes", Proc. IRE 40, No. 9, 1098–1101. After the transform coder 10 transforms image data into a collection of uncorrelated coefficients, the quantizer 12 maps each coefficient, selected from a range of continuous values, into a member of a set of discrete quantized values. These quantized values are then encoded by the entropy coder 14. The resulting stream of binary digits is then either transmitted or stored.

Typically, image data is two-dimensional. Accordingly, the information provided by the transform coder 10 is presented in the form of a two-dimensional array. In a preferred embodiment, a serializer, such as a zigzag serializer, is used to covert the two-dimensional array of continuous values into a one-dimensional array of continuous values. The one-dimensional array is then quantized to yield a one-dimensional array of discrete values. For example, a zigzag serializer operating on an 4×4 array of integers as shown in FIG. 2A would produce a 1×16 array of integers as shown in FIG. 2B.

In a preferred embodiment, a zigzag serializer 16 is included in the image compression pipeline between the transform coder 10 and the quantizer 12, as shown in FIG. 3. After serializing the two-dimensional array provided by the transform coder 10, the quantizer maps the resulting one-dimensional array of continuous values into a one-dimensional array of quantized values. The array is then held for one block processing interval in a one-block delay memory unit 18, while an identical copy of the array is measured by a total block activity measure module 20. The module 20 computes the summation of the square (or the absolute value) of each element in the one-dimensional array, providing a value that represents the 'activity' or 'energy' of the image represented by the array to a control law module 22.

The zigzag serializer 16 places the low-order coefficients that result from, for example, a discrete cosine transform, at the beginning of an array, and the higher-order terms at the end. If there are sufficient low-order terms, the higher-order terms are of less importance

and, due to the nature of the human visual response, may be omitted without perceived image degradation. For example, FIG. 2C shows the one-dimensional array of FIG. 2B after truncation of its six highest order coefficients. Since it is common for the activity of a block to be found mostly in its lower-order coefficients, transmitting only these coefficients results in substantially greater compression of image data. However, unlike the known case of non-adaptive zonal coding, a block with significant activity in its higher-order coefficients will be transmitted in a manner that allows most of these coefficients to be included in the transmitted block, resulting in a received image of superior fidelity. Furthermore, it is not necessary to transmit additional information along with each block for indicating, e.g., its activity level. Instead, the end-of-block symbol 15 allows an image data receiver to know how to parse an incoming bit stream back into blocks of image data.

After dwelling in the memory unit 18 for one processing interval, the one-dimensional array is again measured for activity. A running block activity measure module 24 computes a running total of the squares or the absolute values of the elements in the array as it enters the module 24. A new value of the running total is provided to the control law module 22 as each additional element of the array enters the module 24. Also, as each element of the array enters the module 24, an identical element enters the entropy coder 14. The newly arriving array elements are held in a memory register within the entropy coder 14 until a terminate-block signal, representing an end-of-block character, is provided by the control law module 22. A terminate-block signal may be generated after an entire array has entered the entropy encoder 14. Alternatively, the control law module may generate a terminate-block signal after a predetermined amount of activity has been measured by module 24. In this case, even if the entire array has not yet entered the entropy coder 14, the resulting partial array is transmitted, with an end-of-block symbol appended at the end of the partial array.

The control law module 22 compares the total block activity of the current array, provided by module 20, with the running block activity measure, such as the running sum of squares or absolute values that is progressively generated by the module 24. In a preferred embodiment, a ratio or difference circuit is included in the control law module 22 to generate a measure of the percentage of current block activity. The comparison of the total block activity measure with the running block activity measure indicates how much 'activity' has entered the entropy coder 14. A desired level of image fidelity can be set using an additional input 26.

For a given level, the number of coefficients transmitted will vary, depending on the distribution of activity among the low, middle, and high order coefficients. If the activity of the block is concentrated in its lower order terms, it will be necessary to send fewer coefficients than if a large fraction of the blocks activity is found in the higher order coefficients. If the channel used for transmission becomes overloaded, image fidelity can be decreased in trade for increased data compression.

The image data compression method of the invention can be used to encode any grayscale image or representation of the difference between two images.

Other modifications and implementations will occur to those skilled in the art without departing from the spirit and the scope of the invention as claimed. Ac-

cordingly, the above description is not intended to limit the invention except as indicated in the following claims.

What is claimed is:

1. A method for image data compression comprising the steps of:

transforming a two-dimensional block of image data in a spatial domain to resulting data in a frequency domain represented as a two-dimensional array of activity coefficients;

serializing said entire two-dimensional array of coefficients, yielding a one-dimensional array of coefficients with a leading coefficient and an original trailing coefficient;

quantizing said one-dimensional array of coefficients;

selecting a portion of said one-dimensional array of coefficients beginning with said leading coefficient and ending with a new trailing coefficient that is closer to the leading coefficient than said original trailing coefficient, wherein said portion is selected by a process comprising the steps of:

obtaining a measure of total activity of said one-dimensional array of coefficients,

determining in real time a measure of activity of successive sub-portions of said one-dimensional array, and adding said measure of activity to a running total,

performing a comparison in real time of said running total to a setable level based on said measure of total activity, and

designating a new trailing coefficient according to said comparison;

appending an end-of-block symbol after said new trailing coefficient; and

encoding only said portion with an entropy coder.

2. The method of claim 1, wherein said sub-portions are single coefficients.

3. The method of claim 1, wherein said setable level is also based on a degree of desired image quality.

4. Apparatus for image data compression comprising:

a transform coder for transforming a two-dimensional block of image data in a spatial domain to resulting data in a frequency domain represented as a two-dimensional array of activity coefficients;

a serializer for serializing said entire two-dimensional array of coefficients, yielding a one-dimensional array of coefficients with a leading coefficient and an original trailing coefficient;

a quantizer for quantizing said one-dimensional array of coefficients;

a selector for selecting a portion of said one-dimensional array of coefficients beginning with said leading coefficient and ending with a new trailing coefficient that is closer to the leading coefficient than said original trailing coefficient, said selector including:

a measurer for measuring total activity of said one-dimensional array of coefficients,

a measurer for obtaining a measure of activity in real time of successive sub-portions of said one-dimensional array, and for adding said measure of activity to a running total,

a comparer for performing a comparison in real time of said running total to a setable level based on said measure of total activity, and

a designator for designating a new trailing coefficient according to said comparison;

an appendor for appending an end-of-block symbol after said new trailing coefficient; and

an entropy encoder for encoding only said portion.

5. The apparatus of claim 4, wherein said sub-portions are single coefficients.

6. Apparatus for inclusion in an image data compression pipeline that includes a transform coder, a serializer, a quantizer and an entropy coder comprising:

a memory connected to said quantizer and said entropy coder with an input and an output adapted to store and delay at least a block of transformed, serialized and quantized image data, and adapted to provide an entropy coder with successive sub-portions of said block of image data;

a first measurer connected to said quantizer for measuring the total activity of said block of image data;

a second measurer connected to said output of said memory for measuring the activity of successive sub-portions of said image data and computing a running total of said activity;

a control law unit connected to the first and second measurers, and to an entropy coder, adapted to compare said running total provided by said second measurer to a level based in part on said total activity provided by said first measurer, and sending an end-of-block signal to said entropy encoder;

wherein said entropy coder is adapted to encode only a portion of said block of image data based on when it receives an end-of-block signal.

7. A method for image data compression comprising the steps of:

transforming a two-dimensional block of image data from a spatial domain to a frequency domain, wherein said resulting block of image data in the frequency domain is represented by a two-dimensional array of activity coefficients;

serializing said entire two-dimensional array of coefficients, yielding a one-dimensional array of coefficients;

quantizing said one-dimensional array of coefficients;

selecting a visually significant portion of said one-dimensional array of coefficients, said step of selecting including the steps of:

obtaining a measure of total activity of the one-dimensional array of coefficients,

determining in real time a running total of the activity of successive portions of said one-dimensional array as they enter the entropy coder,

performing in real time a comparison of successive values of said running total to a setable level based on said measure of total activity of the one-dimensional array of coefficients, and

demarking a trailing end of said one-dimensional array according to said comparison;

appending a symbol at a trailing end of the selected portion for indicating the end of said portion; and

encoding said portion with an entropy coder.

8. Apparatus for use in an image processing pipeline including a transform coder, a serializer, and a quantizer, said apparatus comprising:

first means for obtaining a measure of total activity of a two-dimensional block of image data represented as a one-dimensional array of quantized activity coefficients;

second means for determining in real time a running total of successive activity coefficients of said one-

7

dimensional array as said activity coefficients are received by an entropy coder;

third means connected to said first means, said second means and said entropy coder for performing a comparison of said running total to a settable level based on said measure of total activity, and for demarking an activity coefficient of said one-

8

dimensional array as a trailing end of said one-dimensional array according to said comparison.

9. The apparatus of claim 8, further comprising memory means connected to said entropy encoder for allowing a block of image data to dwell momentarily while said first means measures its total activity.

* * * * *

10

15

20

25

30

35

40

45

50

55

60

65

Human Visual Weighted Progressive Image Transmission

BOWONKOON CHITPRASERT, MEMBER, IEEE, AND K. R. RAO, SENIOR MEMBER, IEEE

Abstract—A progressive image transmission scheme which combines transform coding with the human visual system (HVS) model is developed. The adaptive transform coding of Chen and Smith [4] is utilized to classify an image into four equally populated subblocks based on their ac energies. The modulation transfer function (MTF) of the HVS model is obtained experimentally, based on processing a number of test images. A simplified technique for incorporating the MTF into the discrete cosine transform (DCT) domain is utilized. In the hierarchical image buildup, the image is first reconstructed from the dc coefficients of all subblocks. Further transmission hierarchy of transform coefficients and consequent image buildup are dependent on their HVS weighted variances. The HVS weighted reconstructed images are compared to the ones without any weighting at several stages. The HVS weighted progressive image transmission results in perceptually higher quality images compared to the unweighted scheme.

INTRODUCTION

PROGRESSIVE transmission of images involves an approximate reconstruction of an image whose fidelity is built up gradually until the viewer decides either to abort the transmission sequence or allow further reconstruction. The applications are in transmitting images over low-bandwidth channels such as telephone lines. If an image is sent in its original form in a raster scan fashion, transmission of the entire image can take several minutes, instead of which the image quality is built up hierarchically have been developed. The objective in all these schemes is to develop the significant features of an image at an early stage so that a viewer can interactively respond. Other considerations such as the complexity of the technique and robustness to channel noise play significant roles in the merits of a scheme.

Applications of progressive transmission include teleconferencing, telebrowsing, medical diagnostic imaging, videotex, security services, electronic shopping in mail order, and access to large databases. Telebrowsing or videobrowsing is the system in which the recipient wishes to browse through remotely stored images and quickly abort transmission of the unwanted ones as soon as they are recognized. Progressive transmission can play an important role in a picture archiving and communication system (PACS) whose functions are transmitting, storing, processing, and displaying radiological image data such as computer tomography (CT) and magnetic resonance imaging (MRI).

Progressive transmission of images can be classified into two categories as spatial or pel domain and transform or spectral domain. The latter has the advantage of information packing, and the image buildup can be achieved adaptively based on the significance of the transform coefficients. Furthermore, the selection of coefficients

can be altered by HVS weighting in the spectral domain. By either weighting or rearranging the transmission sequence of transform coefficients by the human visual sensitivity, the image buildup can be perceptually pleasing, even at earlier stages.

Several schemes for the transform progressive transmission have been proposed. Ngan [2] proposed such a technique in which five schemes with different transmission sequences were simulated. Elnahas et al. [3] applied the adaptive transform coding scheme of Chen and Smith [4] to hierarchical image buildup. The role of the HVS in transform coding is described in the next section. The proposed scheme is described next, followed by the simulation results.

THE ROLE OF HVS IN TRANSFORM CODING

The HVS has been incorporated in transform coding of images by several researchers. Mannos and Sakrison's work [1] may be the first major breakthrough in image coding incorporating the HVS. Using the assumption that the HVS is isotropic, they modeled the HVS as a nonlinear point transformation followed by the MTF of the form

$$H(f) = a(b + cf) \exp(-cf)^d \qquad (1)$$

where f is the radial frequency in cycles/degree (CPD) of the visual angle subtended and a, b, c, and d are constants. After carrying out extensive experiments, they have arrived at the following HVS model:

$$H(f) = 2.6(0.192 + 0.114f) \exp(-(0.114f)^{1.1}). \qquad (2)$$

This has a peak at $f = 8$ cycles/°.

Earlier research in transform coding incorporating the HVS applied the DFT to the entire frame [1], [5]. This was followed by applying DCT to small subblocks [6]–[10]. Most researchers (except [8], [10]) used Mannos and Sakrison's transfer function (2) as the weighting function. The nonlinear function is ignored in [6], [9], [10]. Both preprocessing and postprocessing are required for all of the schemes.

Recently, new MTF's have been proposed for use with the DCT. Nill [11] proposed a multiplicative function $A(f)$ which is multiplied by the following MTF:

$$H(f) = (0.2 + 0.45f) \exp(-0.18f). \qquad (3)$$

This function has a peak value at $f = 5$ CPD. No nonlinear function is used in this work by assuming that the low-contrast images are of interest. No image processing results were reported in [11].

Ngan et al. [12] used Nill's multiplicative function $A(f)$ with their MTF, which has a peak at $f = 3$ CPD. This function is given by

$$H(f) = (0.31 + 0.69f) \exp(-0.29f). \qquad (4)$$

After multiplying $H(f)$ by $A(f)$, the resulting function has a peak at $f = 4$ CPD. Using a zig-zag scanning sequence for the DCT coefficients, they achieved an acceptable reconstructed image at bit rates of 0.2–0.3 b/pel, depending on the images.

The MTF's of Mannos and Sakrison [1], Nill [11], and Ngan [12] are shown in Fig. 1. Note that the latter two functions are obtained after multiplying (3) and (4) by $A(f)$. In general, incorporating the HVS models improves the coding scheme. So does the proposed

---
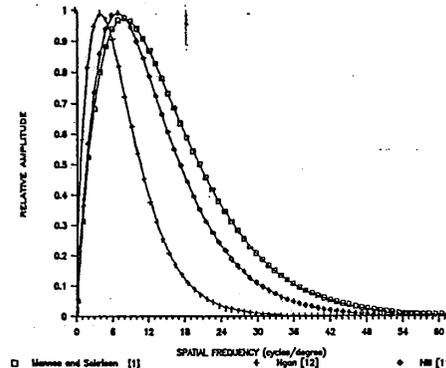
Fig. 1. Comparison of the MTF's.

progressive transmission scheme presented in the next section. The proposed scheme incorporates the HVS model in a way different from the schemes developed so far.

THE PROPOSED TRANSMISSION SCHEME

The proposed scheme (Fig. 2) is basically the modified version of Chen and Smith's adaptive coding [4]. The digitized image consisting of p pels/line and q lines/frame, each with 8 b resolution (256 levels), is divided into subblocks of size $N \times N$ so that the total number of blocks is B where $B = p \times q/N^2$. In this research, $N = 8$ is selected. Intensity values in each subblock are then transformed to the "brightness" values by a point nonlinear transformation. By eliminating the intensity-brightness transformation and the subsequent forward DCT, the system can be simplified. These "brightness" values are transformed into the spectral domain by means of the DCT. The transform coefficients are multiplied by the selected MTF at the corresponding frequencies. These weighted coefficient subblocks are sorted according to their ac energy levels into K classes, the typical value of K being 4 or 8. Four classes are used throughout this research. The ensemble average variance matrix of each class is then calculated. Details of the calculation of subblock classification and the average variances can be found in [4]. The transmission hierarchy of coefficients is determined from the magnitudes of the variances. The variances of all the classes are rearranged in a descending order. The obtained order specifies the class and the positions of the coefficients in the subblocks. The set of matrices of this order is called the transmission sequence map. The coefficients at the specified locations of the subblock are sent from the specified class from left to right and top to bottom. The classification map and the transmission sequence array are sent to the receiver as overhead information before transmitting the transform coefficients. The subblocks of the original image are transformed by the 2-D DCT. These unweighted coefficients are quantized and rearranged according to the transmission sequence. These coefficients may be stored in the central database and can be readily transmitted to the receiver.

Quantization

For simplicity, the dc coefficients are scaled by $N = 8$ and rounded off to the nearest integers in the range between 0 and 255. This scaling makes the maximum possible dc coefficient to be in the limit of maximum possible intensity so that there is no clipping for the dc coefficients. For all the ac coefficients, a nonuniform 4 b Laplacian

Fig. 2. Block diagram of the proposed progressive transmission system. (a) Transformer. (b) Receiver. In (a), the dashed line and elimination of the top two blocks on the left simplify the overall system.

quantizer is used. The coefficients are normalized by their standard deviations. The purpose is to use a single unit variance quantizer for all the coefficients. Thus, it is necessary to send all the standard deviation matrices for all the classes to the receiver prior to transmission of the coefficients. Although there are some methods to estimate the variances at the receiver from the received data, they are not used here because the standard deviation matrices represent a small amount of side information.

Overhead Information

The classification map is an array representing the number of subblocks B in an image. $\log_2 K$ bits are required for each subblock to
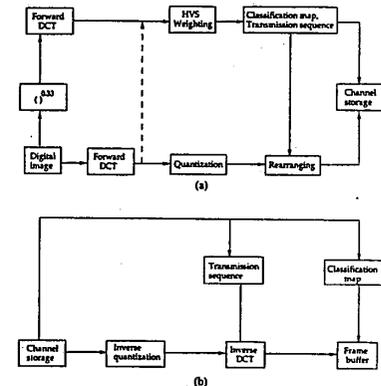
Fig. 3. The original test image: "Girl."

represent K classes. Thus, the overhead bits for the classification are

$$b_{oct} = B \log_2 K \text{ bits.}$$

In the transmission sequence array of $K(N^2 - 1)$ ac coefficients (excluding the dc coefficient position), $\log_2 K$ bits are required to identify the class, and $2(\log_2 N)$ bits for the location of each coefficient in the subblock. The overhead bits for the transmission sequence are

$$b_{ot} = K(N^2 - 1)(\log_2 K + 2 \log_2 N).$$

The real number representation for the K standard deviation matrices requires 32 b (four 8-b bytes) for $N^2 - 1$ entries (excluding the dc coefficient entry). Thus, the overhead bits for the standard deviation are

$$b_{os} = 32K(N^2 - 1).$$

For a $p = q = 512$ image and $N = 8$, the overhead information for four classes consists of
classification bits $= 4096 \times 2 = 8192$ b
transmission sequence bits $= 4 \times 63 \times (2 + 6) = 2016$ b
standard deviation bits $= 32 \times 4 \times 63 = 8064$ b.
For $K = 4$, the total number of overhead bits is 18 272 b or, equivalently, 0.07 b/pel. Using a channel rate of 1200 b/s, the total time required is 15.23 s. These numbers seem to be big. But in the implementation, only the classification map is needed first; the others can be sent one entry each just before the sequence of coefficients at that position. Thus, the total overhead before receiving the first ac coefficient is $8192 + 6 + 32 = 8230$ b or, equivalently, 0.03 b/pel or 6.86 s over a 1200 b/s channel.

THE SIMULATION RESULTS

The four monochrome test images used for simulation are "Girl" (512×512) (Fig. 3), "Baboon" (504×512), "Martha" (480×512), and "Boats" (384×512). Results for "Girl" only are shown. The simulation results based on the two methods described below are compared. The first one is the proposed method, which will be called the HVS weighted method (Fig. 2). The second method is the adaptive coding of Chen and Smith [4], with the average variance used as the transmission sequence. This method will be called the unweighted method. The schematic diagram of this method is basically the lower path of Fig. 2(a).

Using the three functions described in (2), (3), and (4) in the proposed method of incorporating the HVS, the results are found to be quite different from one another. Using the MTF of (2) results in the image with blocking artifacts. The image is built up mainly in the areas of high-activity classes (class 1 or 2) in the early stages, leaving the low-activity areas untouched. The results are not quite different from the unweighted method. Changing to the MTF of (3) leads to better results than those of (2), but still are unpleasant. For the MTF of (4), the intermediate stage images become more blurred, but the blocking artifacts disappear rapidly in the early stages. This MTF yields rather pleasant images for the "Girl" and "Martha," but are rather blurred and noisy for the "Boats" and "Baboon." This has led to the extensive experiments for finding an optimal MTF that yields good results for all the test images. This

0.4942	1.0000	0.7023	0.3814	0.1856	0.0849	0.0374	0.0160
1.0000	0.4549	0.3085	0.1706	0.0845	0.0392	0.0174	0.0075
0.7023	0.3085	0.2139	0.1244	0.0645	0.0311	0.0142	0.0063
0.3814	0.1706	0.1244	0.0771	0.0425	0.0215	0.0103	0.0047
0.1856	0.0845	0.0645	0.0425	0.0246	0.0133	0.0067	0.0032
0.0849	0.0392	0.0311	0.0215	0.0133	0.0075	0.0040	0.0020
0.0374	0.0174	0.0142	0.0103	0.0067	0.0040	0.0022	0.0011
0.0160	0.0075	0.0063	0.0047	0.0032	0.0020	0.0011	0.0006

Fig. 4. The proposed weighting function $H_F(k, m)$. As the dc coefficient is transmitted during the first stage, the corresponding weighting function is not considered in the transmission hierarchy.

MTF, which has a peak at $f = 3.75$ CPD, is given by

$$H(f) = 2.46(0.1 + 0.25f) \exp(-0.25f). \quad (5)$$

A simple convolution-multiplication property for the DCT has been derived by the authors [13]. Assuming that the subscripts c and F denote DCT and DFT sequences, this property, for the one-dimensional case, can be expressed as follows:
If

$$Y_c(k) = \frac{X_c(k)H_F(k)}{C^2(k)}, \quad k = -N, -N+1, \cdots, N-1 \quad (6)$$

where

$$C(k) = \frac{1}{\sqrt{2}} \quad \text{for } k = 0$$

$$= 1 \quad \text{for } k = -N, -N+1, \cdots, -1, 1, 2, \cdots, N-1 \quad (7)$$

then, the corresponding spatial sequence $y(n)$, the IDCT of (6), is

$$y(n) = [x_s(n)^*h(n)] + X_c(0)H_F(0)/\sqrt{N},$$
$$n = -N, -N+1, \cdots, N-1 \quad (8)$$

where

$$x_s(n) = x(n), \quad n = 0, 1, \cdots, N-1$$
$$= x(-1-n), \quad n = -1, -2, \cdots, -N. \quad (9)$$

This result is utilized here. The 2-D weighting function $H_F(k, m)$ is mapped to $H_F(f)$ as follows:

$$H_F(k, m) = \frac{H_F(f)}{C^2(k)C^2(m)}, \quad k, m = 0, 1, 2, \cdots, N-1. \quad (10)$$

Also, for the isotropic model of the MTF,

$$f = \frac{\sqrt{k^2 + m^2}}{2N} f_s \text{ CPD.} \quad (11)$$

Using the sampling density f_s of 64 pels/° (see the Appendix for details), for $N = 8$, $k, m = 0, 1, \cdots, 7$, (10) and (11) are substituted in (5) to obtain the weighting function shown in Fig. 4. The transmission sequences for "Girl" are shown in Figs. 5 and 6. Only the first seven stages of the ac coefficient transmission sequence are shown to highlight the zones of coefficients in each class at the early stages. One stage is defined when the average of one coefficient per subblock is transmitted. It can be seen that more ac coefficients of the lower activity classes of the HVS weighted methods are sent, and also are sent earlier than those of the unweighted method. (This is true for all the test images.)

Since the first stage of reconstruction is from the dc coefficients of every subblock, the reconstructed images for both the weighted and unweighted methods are identical. The reconstructed images at stages 3, 5, and 7 are shown in Figs. 7-9. It can be seen that

CLASS 1					
0	1	3	7	13	22
2	4	6	11	20	
9	8	10	14	23	
17	18	19	24		
CLASS 2					
0	5	15	25		
12	16	28			
26	27				
CLASS 3					
0	21				

Fig. 5. The ac coefficient transmission sequence map of the image "Girl," unweighted method. (Only the first seven stages are shown.)

CLASS 1				
0	1	3	9	24
2	5	10	21	
7	14	18		
16				
CLASS 2				
0	4	8	17	
6	12	20		
13	22			
28				
CLASS 3				
0	11	19		
15	26			
25				
CLASS 4				
0	23			
27				

Fig. 6. The ac coefficient transmission sequence map of the image "Girl," HVS weighted method. (Only the first seven stages are shown.)

Fig. 8. Stage 5 of reconstruction of the "Girl" image. Unweighted method (upper) and HVS weighted method (lower). Bit rate: 3/8 b/pel.

Fig. 9. Stage 7 of reconstruction of the "Girl" image. Unweighted method (upper) and HVS weighted method (lower). Bit rate: 1/2 b/pel.

Fig. 7. Stage 3 of reconstruction of the "Girl" image. Unweighted method (upper) and HVS weighted method (lower). Bit rate: 1/4 b/pel.

the early stages (stage 3), the differences between the two methods are not obvious. Almost the same sharp edges are shown. In the later stages (stage 5), it can be seen that the low-activity areas are building up in the HVS weighted method, while the busy areas in the unweighted method are still building up. For the most crucial image (the "Boats" image), this buildup is in the form of low-energy details (strings, braces, etc.) which have not shown up yet in the unweighted method. In the later stages (stage 7), the two methods yield similar images, except that the HVS weighted method yields few more details in the low-activity areas. It can be seen that the results from the HVS weighted method are more pleasant than those of the unweighted method at the same stage (bit rate), and also are acceptable at the bit rate of 1/4 or 3/8 b/pel, depending on the images.

The comparison of normalized signal-to-noise ratio (NSNR) for the first eight stages is shown in Table I. The bit rate for the overhead information is not included in this table. The NSNR of the HVS weighted method is slightly less than that of the unweighted method at each stage. This confirms that the MSE measure does not correlate well with the subjective measure.

SUMMARY AND CONCLUSIONS

By utilizing a simplified method of incorporating the HVS model in the DCT domain, an adaptive progressive image transmission scheme has been developed. An MTF based on various test images is proposed. This function retains the sharp edges in the high-activity areas and smooths the noise in the low-activity regions. As the effect of the HVS scheme is in the classification and transmission hierarchy rather than modifying the transform coefficients, the receiver

TABLE I
COMPARISON OF OBJECTIVE PERFORMANCE BETWEEN THE UNWEIGHTED AND THE HVS WEIGHTED METHODS. IMAGE: "GIRL."

Stage	Bit/Pel	NSNR (dB) Unweighted	NSNR (dB) HVS weighted
1	1/8	18.0048	18.0048
2	3/16	22.6271	22.2156
3	1/4	24.2178	23.5851
4	5/16	25.6004	24.7014
5	3/8	26.5104	25.3960
6	7/16	27.2467	26.0718
7	1/2	27.8759	26.9039
8	9/16	28.4079	27.0471

has to perform only the IDCT, i.e., no additional postprocessing is required. The HVS weighted method results in perceptually more pleasing images, and leads to acceptable images at low bit rates. A better quantization scheme may be used to improve the performance of the system.

Using one MTF for all the subblocks or for all the classes may not be the best way to incorporate the HVS in the transmission scheme. The human eye adapts to the local activity, as well as to the contrast. This leads to using an adaptive filter or different filters for different areas of an image. Designing and evaluating these filters may be an interesting area of research. Other research directions may be extending the HVS weighting to color images.

APPENDIX

This Appendix describes the sampling of the MTF. Once the decision to use the Fourier domain MTF in conjunction with the DCT of the brightness has been made [see (6) and (10)], sampling the Fourier domain MTF has to be undertaken. The frequency variable f in cycles/degree of the MTF needs to be changed to the normalized spatial frequency f_n in cycles/sample or cycles/pel. This requires a conversion factor f_s as follows:

$$f(\text{cycles/degree}) = f_n(\text{cycles/pel})f_s(\text{pels/degree}) \quad (A1)$$

where f_n, the corresponding frequency in the DCT domain, is

$$f_n = k/2N, \quad k = 0, 1, 2, \cdots, N - 1 \quad (A2)$$

since the first ac coefficient of the DCT is one-half cycle (in contrast to that of the DFT). The f_s, called the sampling density by Ericsson [9], depends upon the viewing distance. According to Mannos and Sakrison [1], "...the simulation program must operate on images defined on a discrete raster of 512×512 picture elements...the simulated images were viewed at a distance at which 65 pels subtended 1 degree of vision...." In this research, f_s is chosen to be 64 pels/° because this number is divisible by $2N$. By substituting $k = 7$, $N = 8$, and $f_s = 64$ into (A1) and (A2), the maximum visual spatial frequency is 28 cycles/° for the 8×8 subblock.

REFERENCES

[1] J. L. Mannos and D. J. Sakrison, "The effect of a visual fidelity criterion on the encoding of images," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 525-536, July 1974.

[2] K. N. Ngan, "Image display techniques using the cosine transform," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 173-177, Feb. 1984.

[3] S. E. Elnahas *et al.*, "Progressive transmission of digital diagnostic images," *Appl. Digital Image Processing VIII, Proc. SPIE*, vol. 575, pp. 48-55, Aug. 1985.

[4] W. H. Chen and C. H. Smith, "Adaptive coding of monochrome and color images," *IEEE Trans. Commun.*, vol. COM-25, pp. 1285-1292, Nov. 1977.

[5] E. L. Hall, *Computer Image Processing and Recognition*. New York: Academic, 1979.

[6] J. B. O'Neal, Jr. and T. R. Natarajan, "Coding isotropic images," *IEEE Trans. Inform. Theory*, vol. IT-23, pp. 679-707, Nov. 1977.

[7] N. C. Griswold, "Perceptual coding in the cosine transform domain," *Opt. Eng.*, vol. 19, pp. 306-311, May-June 1980.

[8] K. H. Tzou, T. R. Hsing, and J. G. Dunham, "Applications of physiological human visual system model to image compression," *Proc. SPIE*, vol. 504, pp. 419-424, 1984.

[9] S. Ericsson, "Frequency weighted interframe hybrid coding," Rep. TRITA-TTT-8401, Telecommun. Theory, Royal Inst. Technol., Stockholm, Sweden, Jan. 1984.

[10] H. Lohscheller, "A subjectively adapted image communication system," *IEEE Trans. Commun.*, vol. COM-32, pp. 1316-1322, Dec. 1984.

[11] N. B. Nill, "A visual model weighted cosine transform for image compression and quality assessment," *IEEE Trans. Commun.*, vol. COM-33, pp. 551-557, June 1985.

[12] K. N. Ngan, K. S. Leong, and H. Singh, "Cosine transform coding incorporating human visual system model," presented at SPIE Fiber '86, Cambridge, MA, Sept. 1986, pp. 165-171.

[13] B. Chitprasert and K. R. Rao, "Discrete cosine transform filtering," presented at the 22nd Asilomar Conf. Signals, Syst., Comput., Pacific Grove, CA, Oct. 1988.

★

Bowonkoon Chitprasert (S'83-M'87) was born in Aranprathed, Prachinburi, Thailand, on November 8, 1954. He received the B. Eng. degree in electrical engineering from Chulalongkorn University, Bangkok, Thailand, in 1976 and the M.S. and Ph.D. degrees in electrical engineering from the University of Texas, Arlington, in 1983 and 1987, respectively.

From 1976 to 1980 he was employed with the Electricity Generating Authority of Thailand, responsible for installations and maintenance of large power transformers and switch gears. He was a Research Assistant and a Teaching Assistant at the Department of Electrical Engineering, University of Texas, Arlington, from 1982 to 1987. In March 1988 he joined Compression Laboratories, Inc., San Jose, CA, to work on algorithm development for the coding of video signals. His current research interests are in the areas of low bit-rate coding and compression of HDTV signals.

★

K. R. Rao (M'67-SM'73) received the B.E. degree from the University of Madras, Madras, India, in 1952, the M.S.E.E. and M.S.N.E. degrees from the University of Florida, Gainesville, in 1959 and 1960, respectively, and the Ph.D. degree in electrical engineering from the University of New Mexico, Albuquerque, in 1966.

Since 1966 he has been with the University of Texas, Arlington, where he is currently a Professor of Electrical Engineering. He has published extensively in reviewed technical journals in the areas of discrete transforms and digital image processing. With two other researchers he introduced the discrete cosine transform in 1974, which has since become very popular in digital image coding. he organized and conducted short courses and conferences on thermoelectric energy conversion between 1970 and 1988. He is coauthor of three books, *Orthogonal Transform for Digital Signal Processing* (Springer-Verlag, 1975), *Fast Transforms: Algorithms, Analyses and Applications* (Academic Press, 1982), and *Discrete Cosine Transform* (Academic Press, 1990). He has edited the benchmark volume, *Discrete Transforms and Their Applications* (Van Nostrand Reinhold, 1984), and has coedited another, *Teleconferencing*.

Evaluation of a Partial-Band Jammer with Gaussian-Shaped Spectrum Against FH/MFSK

HYUCK M. KWON, MEMBER, IEEE, LEONARD E. MILLER, MEMBER, IEEE, AND JHONG S. LEE, SENIOR MEMBER, IEEE

Abstract—It is often assumed in system analyses that a partial-band noise jammer against a frequency-hopped M-ary frequency shift keying (FH/MFSK) communication system has a rectangular spectrum. The rectangular spectrum is, of course, unrealizable, and so it is of interest to consider more realistic spectrums, such as those where a jammer uses a bandpass filter before amplifying the jamming noise. In this paper, a Gaussian-shaped filter is used to represent a class of bandpass filters. Clearly, this is a more realistic shape than the ideal rectangular shape. In addition, it can be analyzed easily by a reasonable approximation, i.e., a Gaussian-shaped spectrum that is constant over each hopped M-ary signaling band. Numerical results indicate that such a Gaussian-shaped partial-band noise jammer has nearly the same effects as an ideal rectangular-shaped partial-band noise jammer with an equivalent bandwidth.
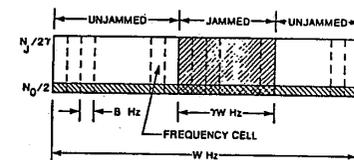
I. INTRODUCTION

PARTIAL-BAND noise jamming against a frequency-hopped M-ary frequency-shift keying (FH/MFSK) system has often been assumed to be implemented by a two-level power spectral density (PSD) [1]-[5]. According to this jamming scheme, the jammer distributes his total available power J watts uniformly over a fraction γ, $0 < \gamma \le 1$ of the total spread-spectrum system bandwidth W Hz, as shown in Fig. 1. Given that a rectangular spectrum is, of course, unrealizable, it is of interest to consider more realistic spectrums such as those where a jammer uses some sort of bandpass filter before amplifying the jamming noise. The Gaussian-shaped spectrum shown in Fig. 2 is thus used to represent the class of realistic spectrums and, in addition, it is chosen for simplicity of analysis. For the analysis, we can reasonably approximate the Gaussian-shaped spectrum as constant over each hopped M-ary signaling band because, in practice, the system band and the jammed frequency band are much greater than the M-ary signaling band.

The measure of comparison used here is the worst case bit-error probability of FH/MFSK for equal-power partial-band noise jammers. In Section II, a detailed analysis is given of the bit-error probability of L hops/symbol FH/MFSK, $L \ge 1$ against partial-band noise jamming plus additive white Gaussian noise (AWGN). In Section III, numerical results are shown for $M = 2$ with $L = 1$, 2 hops/symbol, and are compared to those for a rectangular spectrum.

II. ANALYSIS OF PARTIAL-BAND JAMMING WITH GAUSSIAN-SHAPED PSD

We consider an L hops/symbol FH/MFSK communication system with an "adaptive gain control (AGC)" receiver [3] which is summarized in the following paragraph.

Fig. 1. AWGN plus ideal partial-band jamming power spectral density with two levels.

Fig. 2. AWGN plus realizable partial-band jamming power spectral density with Gaussian-shape.

MFSK modulation conveys information by transmitting one of the $M = 2^K$ symbols every T_s seconds where the symbol consists of K data bits. The M-ary symbol is conveyed to the receiver by selecting one of M frequencies to be transmitted. We assume that the frequencies are spaced evenly across a contiguous band with the spacing B to obtain orthogonal signals. Thus, the bandwidth of the M-ary cluster will be MB. In order to mitigate the effect of the assumed jamming, the signal is further subjected to a spread-spectrum modulation in the form of frequency hopping. We consider an L hops/symbol frequency hopping scheme, in which one symbol duration T_s seconds is broken into L independent transmissions of T_s/L seconds. When the total system bandwidth is W, there are $N_h = W/B - M + 1$ hopping locations available; if $W/B \gg M$, then we may take the approximation $N_h \approx N = W/B$. At the FH/MFSK receiver, the incoming signal is dehopped by mixing it with a synchronized replica of the hopping oscillator. The dehopped signal is then applied to a bank of M bandpass filters, each of bandwidth B, centered at the M possible signaling frequencies. The output of each filter is processed by a square-law envelope detector. Each squared envelope is sampled once every T_s/L seconds. The sampled detector outputs are normalized by the received noise or noise plus jamming power [7]. The purpose of the normalization is to prevent jammed hops from dominating the symbol decision process. For the case of multihops/symbol, $L > 1$, the normalized L outputs of each channel are linearly summed to form the receiver decision statistics. Then, the receiver selects the bandpass channel with the largest output and decides that the trans-

Spectral response of the discrete cosine and Walsh-Hadamard transforms

R.J. Clarke, B.Tech., M.Sc., Mem. I.E.E.E., C. Eng., M.I.E.E.

Indexing term: Image processing

Abstract: It is becoming increasingly popular, when considering transform coding schemes for images, to attempt to take the spatial frequency response of the human observer into account by performing a classical one- or two-dimensional filtering (pre-emphasis) operation upon the coefficients of the Fourier transformed image, and it would be advantageous if such a procedure could be carried out with more commonly used image transforms, notably the discrete cosine and Walsh-Hadamard transforms. It is demonstrated here that, notwithstanding the theoretical difficulties associated with the convolution/multiplication operation where the discrete cosine transform is concerned, its spectral response and the nature of the appropriate filtering characteristic are such that an operation of the above mentioned kind may be carried out, and the benefits of psychovisual coding obtained. On the other hand, the results obtained in the case of the Walsh-Hadamard transform show that it is unlikely that its performance will be found satisfactory in such an application.

1   Introduction

Transform image coding schemes have been extensively treated in the research literature for some years now, and have in many cases reached a high level of sophistication [1,2]. It has become increasingly realised, however, that improved systems will result if the visual response of the observer is taken into account, and in a transform coding system this implies recognition of the fact that the eye is more sensitive to certain spatial frequencies than to others [1] (the logarithmic point nonlinearity in the amplitude response of the eye is not considered here). If we consider a Fourier coded image, it is evident that those spectral components to which the eye is most sensitive should be coded more accurately than the others, and it is logical to use the measured optical spatial response curve to weight the transform coefficients prior to quantisation, bit allocation and coding. Such a procedure has been carried out by Hall [3,4] with encouraging results, and the now more frequent use of other, more convenient, transforms has naturally brought with it the question of whether the transform coefficient sets which they produce are equally amenable to a filtering operation of this nature. This paper examines the spectral response of the discrete cosine and Walsh-Hadamard transforms in this context.

2   Spatial frequency response of the eye

Determination of the response of the eye to sine-wave contrast gratings has been the subject of considerable research. A discussion of several results in this area is given by Mannos and Sakrison [5], who were concerned to develop a suitable weighting criterion on the basis of which numerical distortion measures might be correlated with subjective (perceived) image degradation. Their function

$$A(f) = 2.6(0.0192 + 0.114f) \exp(-(0.114f)^{1.1}) \qquad (1)$$

where f is the spatial frequency in cycles per degree, is shown in Fig. 1 and is the one used by Hall in his previously quoted work. As far as its use in psychovisual coding is concerned, the image is first transformed into a set of weakly correlated coefficients, and the coefficient array is then pre-emphasised by the weighting function before bit allocation and coding, a complementary stage of de-emphasis taking place at the receiver, and the use of the Fourier transform in such an application has already been referred to. The difficulty with

the use of other transforms for this purpose is that of the appearance of spurious energy components at arbitrary locations within the coefficient set. There are thus two necessary conditions to be fulfilled if psychovisual processing of the kind referred to in this paper is to be successful with a given transform. First, the major part, at least, of the energy residing in a given spatial frequency component within the data must appear in, if not one, then in only a small set of adjacent transform coefficients. Secondly, the optimum weighting function must be sufficiently broad, i.e. a sufficiently slowly varying function of spatial frequency, so that, if necessary, two or three adjacent coefficients may receive the same degree of pre-emphasis without unduly affecting the result. Fig. 1, although drawn to a somewhat deceptive vertical scale, shows this to be the case, for the lower and upper half-power points of the response lie at approximately 2 and 20 cycles/degree, respectively, and the equivalent Q value of the corresponding bandpass filter is about one half. A resonable degree of energy 'spreading' in the transform coefficients will not therefore interfere with the application of pre-emphasis. The first
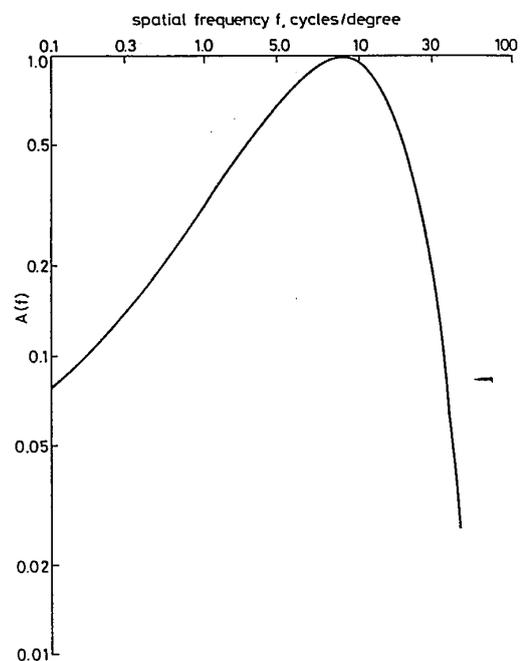
Fig. 1   Optical response weighting function corresponding to eqn. 1

, of the necessary conditions will be considered in a subsequent Section.

3 Problem

In recent years the Fourier transform has, to a certain extent, fallen into disfavour for image transform coding. Despite the fact that the conjugate symmetry of its components when transforming real data means that there are no more distinct coefficient values than if it were a real transform, it still requires complex number operations (the processing and storage of two arrays), and its rate of convergence is also low [6]. Workers in the field have therefore turned to real transforms such as the discrete cosine and Walsh-Hadamard transforms (DCT and WHT), the former being extremely effective for the purpose, the latter being very easy to implement. The difficulty with both the DCT and WHT (and other similar transforms) is that the spectral distribution is not one of true frequency components as is the case with the discrete Fourier transform (DFT). Furthermore, the convolution/multiplication property of the latter transform does not hold for the DCT and WHT (although the relationship may, in the case of the DCT, be expressed as a double convolution [7]). Theoretical aspects of the spectral properties of the WHT are discussed by Ohnsorg [8]. Griswold [9] has attacked the problem by defining an energy distribution in the cosine domain which falls off monotonically with increasing coefficient order and depends on the slope of the autocorrelation function of the image data. In this paper an alternative approach is taken, which is based upon the observed properties of the transform basis vectors. Consider the basis set shown in Fig. 2, which consists of the 16 basis vectors of the DCT
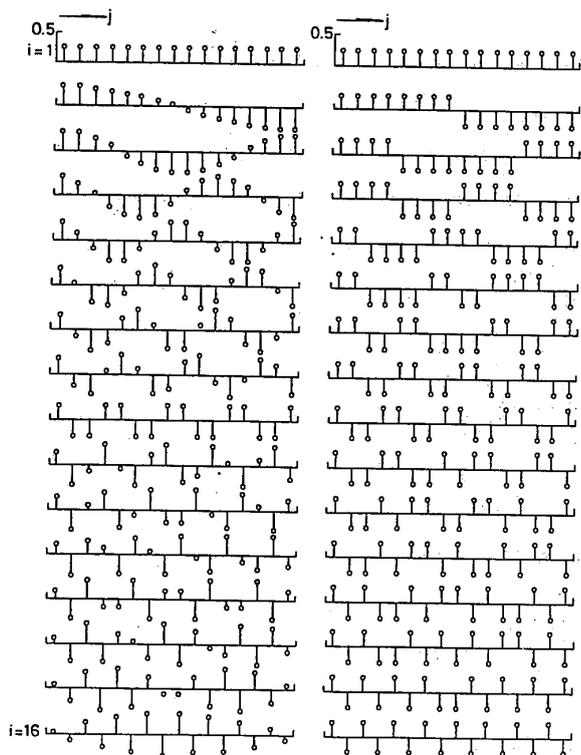
Fig. 2 *Basis vectors of the 16-point DCT*
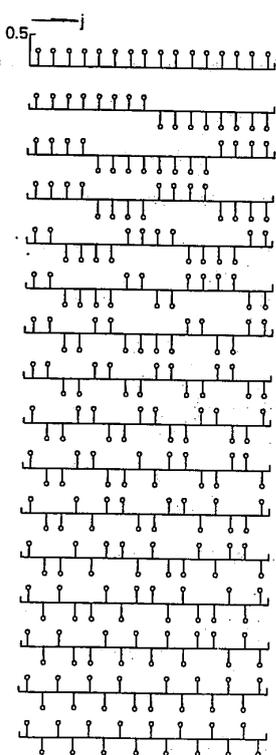
Fig. 3 *Basis vectors of the 16-point WHT*

matrix of order 16:

$$B(i,j) = C \cos((i-1)(j-1/2)\pi/N) \qquad N = 16$$

$$i, j = 1, \ldots, 16$$

where $C = (2/N)^{1/2}$, unless $i = 1$ when $C = N^{-1/2}$. It is apparent that the form of the vectors is that of a set of cosinusoidal components of gradually increasing frequency[†] sampled at 16 points which cover a complete halfcycle at the lowest nonzero frequency. Thus, for example, basis vector 7 is the result of taking 16 samples equidistantly spaced over seven halfcycles of a cosinusoid, starting one half a sample interval from the origin. Again, basis vector 15 shows the construction well: it is the result of sampling fifteen halfcycles of a cosinusoid at 16 points, and the low-frequency 'beat' pattern can clearly be seen. Since the transform operates by correlating the members of the orthogonal basis set with the data vector, it is apparent that, on an intuitive level, the various transform coefficients obtained should take on significant values when detail of an equivalent spatial frequency is present within the data vector. The problem remains as the effect of the phase of the spectral component within the data. In the case of the Fourier transform, of course, this is taken care of by variations in the relative magnitudes of real and imaginary components of each transform coefficient, the overall amplitude $(\text{Re}(\cdot)^2 + \text{Im}(\cdot)^2)^{1/2}$ remaining constant as the phase relation alters. With a real transform there is, naturally, no imaginary component, and yet the transform coefficient set must retain all the information within the original data including the relative phase of any sinusoidal component, for otherwise it would not be possible to reconstruct the data vector exactly by inverse transformation. The manner in which this is accomplished is demonstrated by the following experiment.

4 Experimental details

The experiment is, in essence, very simple. The object is to determine whether, irrespective of arbitrary phase relationships, the presence of spatial frequency detail in an image data vector is reflected unambiguously, in some way, in the set of transform coefficients of the discrete cosine and Walsh-Hadamard transforms (the order-16 basis vectors for which are shown in Fig. 3). In the latter case, incidentally, because the 'shapes' of the basis vectors are rectangular rather than cosinusoidal, the results for sine-wave data may, a priori, be expected to be poorer than those obtained with the DCT. The data is the descrete sine wave

$$D(j) = \sin((j-1)k\pi/N) \qquad N = 16$$

$$j = 1, \ldots, 32$$

$$k = 1, \ldots, 15$$

k determines the frequency of the sine wave: thus, when $k = 1$ the data take the form of 32 samples within one complete cycle, 16 of which (one halfwave at the lowest frequency) will form the 16 data samples taken for the 16-point transform. At the upper frequency end of the range, $k = 15$ will result in 15 halfcycles of the total data length of 15 cycles acting as the data vector to be transformed. Thus for each value of k the data vector consists of a discrete (sampled) sine wave

[†]It is conventional to use the term 'sequency' as the counterpart of 'frequency' in connection with waveforms having unequally spaced zero crossings. Since we are concerned with the transform domain response to spatial *frequency* components, however, to avoid confusion that terminology is retained here.

having one of 15 different values of spatial frequency. To cover all possible phase relations between data and transform, for each value of k the data are moved sequentially through 16 steps, covering all possible values of relative phase shift (except at the lowest frequency of all, where a shift of 16 steps corresponds to a half sine wave of data only; in this case the values for the other halfcycle will be those for the first but with the sign reversed). The result is a set of 15 arrays, each for a different frequency of data (there is, of course, no result for $k = 0$) and each containing, for that frequency the results of the transform operation

$$[Y_k(m, i)] = [B(i, j)] [D_k(j + m)] \qquad (4)$$

$$k = 1, \ldots, 15$$

$$m = 0, \ldots, 15$$

$$i, j = 1, \ldots, 16$$

where $B(i, j)$ is the basis matrix, $D_k(j + m)$ is the (sequentially shifted) column data vector of frequency k, and the 16 rows of Y_k contain the 16 spectral coefficients for frequency k for all values of the phase shift index m.
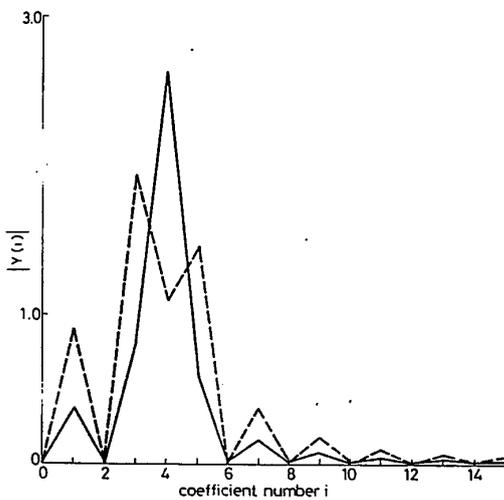
Fig. 4 *Magnitude of the spectral response Y(i) for the DCT as a function of transform coefficient number*

Data consist of four halfcycles of $D(j)$
— — — best-case spurious response
— · — · — worst-case spurious response

5 Results

5.1 Discrete cosine transform

The complete set of results is too extensive to present in total here. What will be done is to discuss the main characteristics of the results and show maximum and minimum responses for each spectral coefficient. First, whatever the frequency and phase shift chosen, the energy in coefficients far from that corresponding to a given selected input frequency is always small. This is clearly illustrated in Figs. 4 and 5, which show the magnitude of the spectral response $Y(i)$ to two arbitrarily selected input components consisting of 16 data samples covering four and 11 halfcycles, respectively. In each case two curves are plotted (discrete sample points being joined simply to allow clarity of presentation): one corresponding to the worst (highest) spurious response, the other

to the best (lowest). Results for all possible values of phase shift then fall between the two ordinate values given for each coefficient.

In all cases the response shows alternate zeros except for the component corresponding to the equivalent input frequency, where the response is generally substantial. Again, there are always three dominant coefficients and, as the phase shift in the input data changes, the relative magnitudes and signs of these three coefficients change. This is illustrated in Fig. 6, where the input data comprise four halfcycles of a sine wave and the amplitudes of the three neighbouring coefficients $Y(3)$, $Y(4)$ and $Y(5)$ are plotted as a function of data phase shift. It can be seen that it would be possible to determine, if desired, the original data phase relationship from the relative magnitudes and phases of the spectral components. The fact that there are three significant components suggests that it will be advantageous to determine coefficient energy within a window covering three adjacent components,
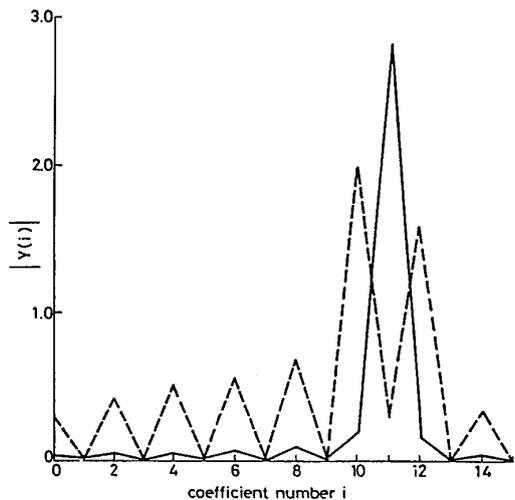
Fig. 5 *As Fig. 4, with the data consisting of eleven halfcycles of D(j)*

——— best-case spurious response
— — — worst-case spurious response
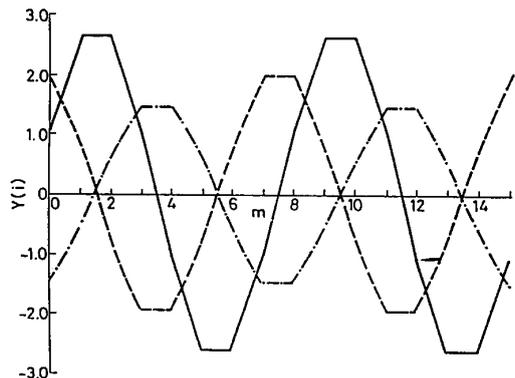
Fig. 6 *Magnitudes and signs of three adjacent transform components Y(3), Y(4) and Y(5) for the DCT as a function of the relative phase shift between input data and transform block*

Data consist of four halfcycles of $D(j)$
m = discrete phase shift index
——— $Y(4)$
— — — $Y(3)$
— · — · — $Y(5)$

$$[Y^2(i)]_3 = \tfrac{1}{3}(Y^2(i-1) + Y^2(i) + Y^2(i+1)) \qquad (5)$$

averaged in such a way. Again best- and worst-case results are shown, and it can be seen that the spectral distribution in the transform domain does indeed mirror that of the input data, and it is therefore justifiable to assume that most of the spectral energy at a given spatial frequency resides in three
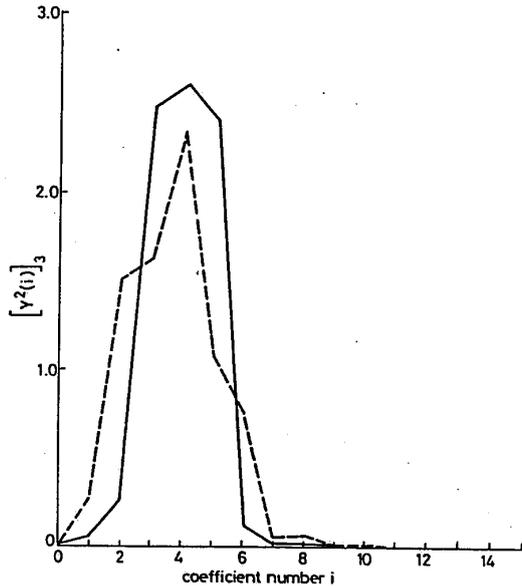
Fig. 7 *Magnitude-squared spectral response over a window of length three corresponding to the results of Fig. 4*

——— best-case spurious response
– – – worst-case spurious response
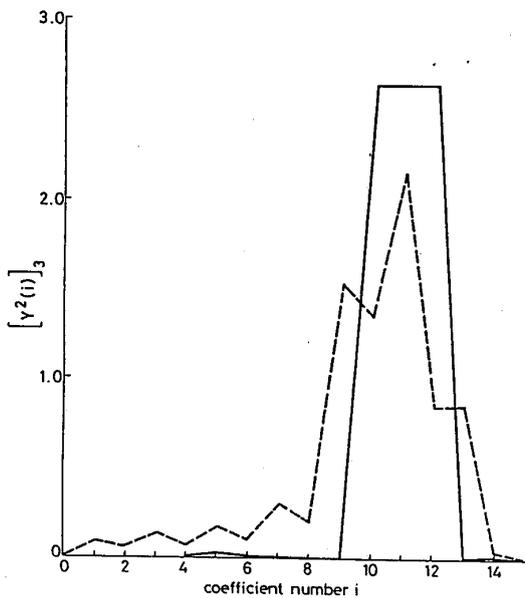
Fig. 8 *As Fig. 7, but corresponding to the results of the Fig. 5*

——— best-case spurious response
– – – worst-case spurious response

312

multiplication of the spectral transform distribution by appropriate weighting characteristic will carry out the desired pre-emphasis previously referred to in Section 2, since in all cases the spectral response to an input signal of a given spatial frequency and arbitrary phase is only significant within very restricted region in the transform domain.

The above experiment was repeated using a discrete cosine transform of order eight. Similar results were obtained, but the order of such a transform is too low to allow much discrimination between components, and it is therefore suggested that a transform of length 16 is the smallest that can usefully be used in such an application.

5.2 Walsh-Hadamard transform

The encouraging results obtained above in the case of the 16 point DCT suggest attempting the process of pre-emphasis using the Walsh-Hadamard transform, which of course has the advantage of easy implementation, and Figs. 9–12 present exactly the same details for the WHT as Figs. 4, 5, 7 and 8 do for the DCT. In this case it can be seen that the response is substantially more diffuse, with two distinct sets of signifi-
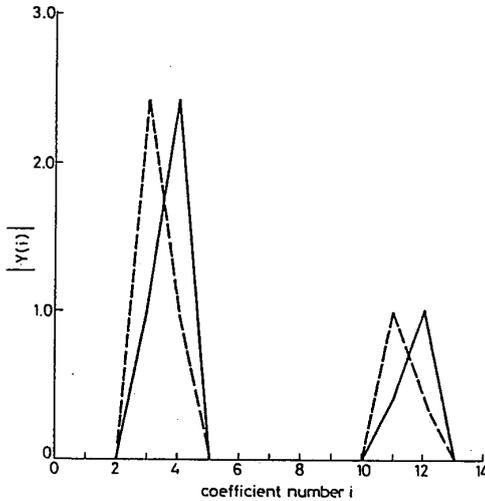
Fig. 9 *As Fig. 4, but using the WHT*

——— best-case spurious response
– – – worst-case spurious response
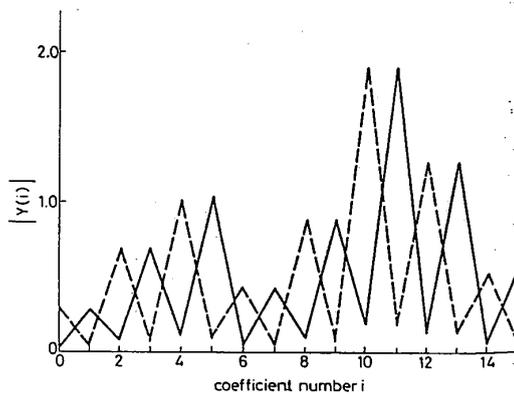
Fig. 10 *As Fig. 5, but using the WHT*

——— best-case spurious response
– – – worst-case spurious response

...ant components when the data consist of four halfcycles per block length, and a similar result for the case where the data consist of 11 halfcycles, with the addition of finite components at all other locations also. Results obtained using an eight-point WHT were similar, and had the additional disadvantage of reduced frequency discrimination.
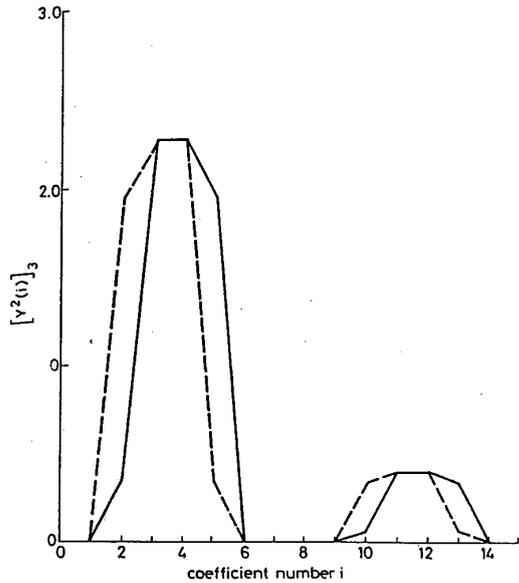
Fig. 11 As Fig. 7, but using the WHT

———— best-case spurious response
– – – – worst-case spurious response

6   Conclusions

It has been demonstrated that detail of a given spatial frequency in image data, irrespective of phase angle, resides almost totally within three adjacent coefficients in the transform domain when a 16-point DCT is applied. In the context of psychovisual coding, the implication is that appropriate weighting of the DCT coefficients will allow improved coding efficiency in the manner already reported by Hall in the case of the DFT. A similar result obtains when a DCT of length

eight is used, but with much reduced frequency discrimination. Results for the WHT of either length are significantly poorer, and the presence of substantial spurious components at unrelated points within the transform coefficient set indicates that this transform is unsuitable for the purpose.
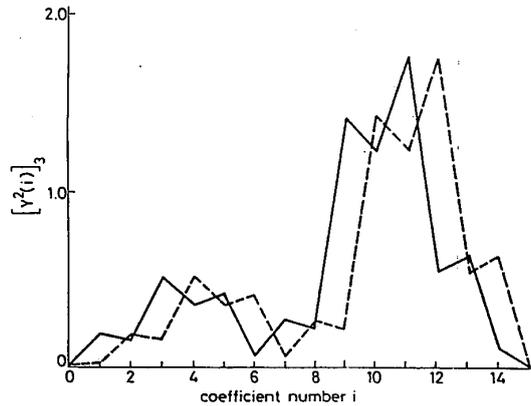
Fig. 12 As Fig. 8, but using the WHT

———— best-case spurious response
– – – – worst-case spurious response

7   References

1 HALL, E.L.: 'Computer image processing and recognition' (Academic Press, 1979)
2 MEIRI, A.Z., and YUDILEVICH, E.: 'A pinned sine transform image coder', IEEE Trans., 1981, COM-29, pp. 1728–1735
3 HALL, C.F., and ANDREWS, H.C.: 'Digital colour image compression in a perceptual space'. Proc. SPIE, 1978, 149, pp. 182–188
4 HALL, C.F.: 'Perceptual coding in the Fourier transform domain'. NTC convention record, 1980. pp. 36.1.1–36.1.7
5 MANNOS, J., and SAKRISON, D.: 'The effects of a visual fidelity criterion on the encoding of images', IEEE Trans., 1974, IT-20, pp. 525–536
6 PRATT, W.K.: 'Digital image processing' (Wiley, 1978)
7 CHEN, W., and FRALICK, S.: 'Image enhancement using cosine transform filtering', in WILDE, C.O., and BARRETT, E.: 'Symposium on image science mathematics, Nov. 1976. pp. 186–192
8 OHNSORG, F.R.: 'Spectral modes of the Walsh-Hadamard transform', IEEE Trans., 1971, EMC-13, pp.55–59 (Proceedings of 1971 symposium on Walsh functions)
9 GRISWOLD, N.C.: 'Perceptual coding in the cosine transform domain'. Opt. Eng., 1980, 19, pp. 306–311

Color-Luminance Masking Interactions

K. K. De Valois and E. Switkes

University of California, Berkeley, CA 94720, USA

Visual systems in all animals are sensitive to changes both in the wavelength of light and in its intensity. Systems which exhibit *color vision* can detect wavelength changes irrespective of associated changes in effective intensity. The ability to dissociate wavelength changes and intensity changes allows an organism to compile two separate spatial maps of the world. This provides a greater amount of information, but it also raises the possibility of conflict, since those maps are not always identical. For example, when a directional light source produces luminance shadows across a surface of uniform chromaticity, a luminance map and a color map of the scene would suggest different sets of contours with the possibility of correspondingly different physical realities. One early problem faced by the visual system is the resolution of such conflicts.

A second problem arises from the fact that because of the massive amount of information to be passed through a very limited channel—the optic nerve—most cells must carry information about both chromatic differences and luminance differences. Thus, although in theory one can treat chromatic and luminance information independently, separate physiological mechanisms are not found at these early levels. How the visual system decodes two different types of information carried by the same substrate is an intriguing problem.

In order to understand pattern vision, one must ultimately be able to characterize the way in which the color visual system processes spatial information, in much the same manner as spatial processing by the luminance system has been previously studied. To predict visual system responses to real-world scenes, it is also necessary to consider the interactions of the chromatic- and luminance-analyzing pathways. Finally, to understand visual behavior, it must ultimately be related to observed cell characteristics at various levels in the visual system.

Early attempts to characterize the spatial information transfer characteristics of the color vision system consisted primarily of measures of the spatial contrast sensitivity function (CSF). Such measures (e.g. van der Horst and Bouman, 1969; Kelly, 1983; Mullen, 1985) have generally found that the color CSF differs from that seen with luminance in two respects: the high-spatial-frequency cut-off occurs at lower absolute spatial frequencies, and there is no low-frequency attenuation. Thus, the chromatic CSF is low-pass, not band-pass. These differences can be modeled in terms of the receptive-field organization of ganglion and/or lateral geniculate chromatically-opponent cells, which show center-

282

surround antagonism for luminance differences and center-surround synergism for chromatic changes (De Valois and De Valois, 1975; Ingling and Martinez, 1983).

Although the shapes of the overall contrast sensitivity functions fit a simple model relating psychophysical sensitivity to receptive field (RF) characteristics and numbers of cells at the geniculate level, they do not necessarily yield much information about the tuning function of the individual cortical subunits which they presumably also reflect. In the case of luminance, for example, there is considerable evidence that the CSF reflects the activity of many more narrowly tuned mechanisms (e.g. Blakemore and Campbell, 1969; Legge and Foley, 1980; De Valois, Albrecht and Thorell, 1982). We have investigated the spatial frequency tuning of mechanisms sensitive to purely chromatic contrast using both adaptation and masking paradigms (De Valois and Switkes, 1983; Switkes, Bradley and De Valois, 1988; Bradley, Switkes and De Valois, 1988). Our data demonstrate that for color, as for luminance, there are multiple channels, each of which is more narrowly tuned for spatial frequency than is the overall CSF. Adapting to an isoluminant chromatic grating of frequency f produces a transient loss in contrast sensitivity which is centered at f and which falls off in both directions (Bradley, Switkes and De Valois, 1988). When compared to the corresponding data for luminance grating adaptation, such functions appear somewhat more broadly tuned, with the primary difference—at least at moderate spatial frequencies—being a more gradual low-frequency fall-off. Although the mechanism of contrast adaptation is not clear (Ohzawa, Sclar and Freeman, 1982; Albrecht, Farrar and Hamilton, 1984), it is commonly presumed to reflect the tuning characteristics of cells in V1. If this is so, then the generally broader adaptation effects, particularly with respect to the low-frequency attenuation, are quite consistent with reports of broader bandwidths, on average, for V1 cells responding to isoluminant chromatic patterns. Many (though certainly not all) V1 cells appear to have low-pass characteristics for chromatic patterns, but band-pass characteristics for luminance patterns (Thorell, De Valois and Albrecht, 1984). Such cells could be responsible for the low-pass chromatic CSF, despite the presence of many band-pass chromatic mechanisms. Alternatively, the low-pass color CSF may simply reflect a relatively greater number and/or more sensitive cortical band-pass mechanisms tuned to low spatial frequencies.

Using a simultaneous masking paradigm we also find similar bandpass tuning for chromatic and for luminance patterns (De Valois and Switkes, 1983; Switkes, Bradley and De Valois, 1988). A high-contrast, isoluminant red–green mask grating, for example, will significantly reduce contrast sensitivity for a red–green grating of similar spatial frequency, while having little or no effect upon the detectability of a red–green grating of a very different spatial frequency. (Fig. 1, *vide infra*, presents results from a similar experiment using blue–yellow chromatic gratings.) Selective masking, like selective adaptation, indicates the presence of mechanisms which are more narrowly tuned than the overall CSF for isoluminant color gratings.

K. K. DE VALOIS AND E. SWITKES
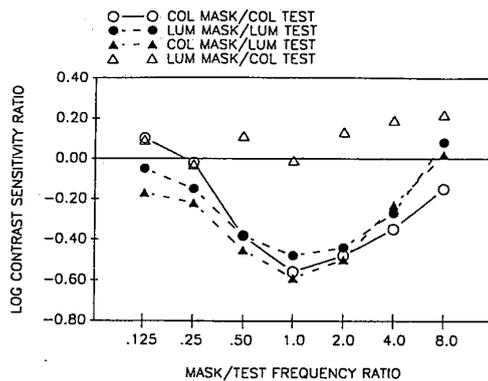
FIG. 1. Spatial frequency tuning of simultaneous masking when both mask and test gratings are isoluminant blue–yellow gratings (open circles, solid lines), both are luminance gratings (filled circles, dashed lines), mask is blue–yellow and test is luminance (filled triangles, dot–dash lines), or mask is luminance and test is blue–yellow (open triangles, dotted lines). The change in contrast sensitivity [log(masked contrast sensitivity/unmasked contrast sensitivity)] is plotted as a function of mask/test spatial frequency ratio. Negative values indicate reduced contrast sensitivity—i.e. masking. Mask contrasts were chosen to yield about .5 log unit of masking when mask and test were identical. The same mask contrasts were used when mask and test differed.

In the case of luminance, there is also substantial evidence to indicate that there exist mechanisms which are tuned in the second spatial dimension, as well. For example, grating adaptation effects are maximal when the test pattern has the same orientation as does the adaptation pattern (Gilinsky, 1968; Blakemore and Campbell, 1969). The loss in contrast sensitivity falls off as the orientation difference increases. Such orientation selectivity is consistent with a cortical locus for pattern-specific adaptation, given that narrow orientation tuning is first seen in cells in V1 (Hubel and Wiesel, 1962).

We have examined the orientation selectivity of the visual mechanisms which respond to isoluminant chromatic patterns using both masking and adaptation paradigms. As is observed for luminance patterns, we find orientation-selective effects for isoluminant chromatic gratings in both masking and adaptation experiments (De Valois, Webster and Switkes, 1984; Bradley, Switkes and De Valois, 1988). For example, either adaptation or masking with a vertically-oriented grating produces greater effects upon the detection of a similar, vertically oriented test grating than upon the detection of an oblique grating. An oblique grating, in turn, is somewhat more affected than is a horizontally-oriented grating. There is, in each case, clear and marked tuning for the orientation of an isoluminant chromatic pattern, although it appears in general somewhat broader than that found for luminance patterns in comparable experiments. This is of particular significance for two reasons. First, the extent to which a useful map of

mask and test
1 are luminance
ninance (filled
open triangles,
vity/unmasked
atio. Negative
were chosen to
he same mask

licate that there
on, as well. For
pattern has the
Blakemore and
the orientation
a cortical locus
ing is first seen

1anisms which
ind adaptation
ation-selective
nd adaptation
witkes and De
1 a vertically-
ilar, vertically
g. An oblique
tally-oriented
entation of an
what broader
ts. This is of
useful map of

the visual world can be developed from pure color information depends upon the presence of spatial tuning in two dimensions. If there were no orientation selectivity for isoluminant chromatic patterns, the usefulness of color vision for pattern perception would be limited indeed. Secondly, although the orientation selectivity of V1 cells for luminance patterns is well known (e.g. Hubel and Wiesel, 1962; De Valois, Yund and Hepler, 1982), the existence of orientation tuning for pure color patterns is much less well established. Our psychophysical results (and others, see Elsner, 1978) imply that significant orientation specificity for isoluminant chromatic patterns must exist at some level.

We have also examined the contrast dependence of simultaneous masking and adaptation in the chromatic domain. When both mask and test patterns have the same spatial frequency and orientation, and when both correspond to luminance variations, a characteristic masking function is seen (Legge and Foley, 1980). Masks of very low contrasts—either subthreshold or slightly suprathreshold—facilitate the detection of similar test patterns. As the mask contrast increases, however, increasingly more test contrast is required for detection. Such masking is often taken to reflect fundamental contrast-response characteristics of the mechanisms involved in processing luminance contrast.

We have examined contrast-masking functions for isoluminant chromatic patterns and find them to be extraordinarily similar to those seen for luminance (Switkes, Bradley and De Valois, 1988). At very low mask contrasts the detection of a superimposed identical test grating is facilitated. Above the contrast threshold for the mask, test detectability drops in a monotonic fashion with increasing mask contrast. The transition between these two portions of the masking function is smooth.

We have compared the contrast-masking function for color with that for luminance by measuring both functions for the same subjects, on the same equipment, using patterns with the same space- and time-averaged luminance and chromaticity. When the results are plotted using a metric in which mask contrast is quantified in terms of multiples of threshold contrast, the color and luminance contrast masking functions are, on the average, virtually identical. There is not only a superficial similarity, but a close quantitative correspondence.

In the case of pattern-specific adaptation, color and luminance also show a similar contrast dependency (Bradley, Switkes and De Valois, 1988). For both, increasing the contrast of the adaptation pattern yields a corresponding increase in the magnitude of the post-adaptation threshold. Although the contrast-adaptation function is monotonic for both color and luminance, the correspondence of functions is not as strong as in the masking paradigm, with the slope of the chromatic function being somewhat steeper.

Thus, based on both adaptation and simultaneous masking experiments, we conclude that the chromatic contrast sensitivity function, like the luminance contrast sensitivity function, reflects the existence of multiple submechanisms which are more narrowly tuned for orientation and for spatial frequency than the overall CSF. Although the tuning of these subunits does not appear to be

286   K. K. DE VALOIS AND E. SWITKES

extremely narrow—in fact, perhaps somewhat broader than that for luminance—it is marked in both dimensions and is sufficient to support the construction of a useful map of the visual world based upon solely chromatic variations. The contrast-response functions of the underlying mechanisms appear to be similar, suggesting that not only spatial information but also contrast information is transmitted in a very similar manner for both color and luminance. While spatial acuity based solely upon chromatic information is somewhat poorer than that based upon luminance differences, it is clearly sufficient to support useful pattern vision. Given such an ability, it would be at best uneconomical not to use it. Thus, a real understanding of spatial vision must depend upon understanding spatial color vision, as well as spatial luminance vision.

We have also studied the interactions of chromatic and luminance contrasts when both are simultaneously present. Again, we have used both simultaneous masking and adaptation paradigms. We find a marked asymmetry between the masking of chromatic contrast by luminance contrast and masking of luminance contrast by chromatic contrast (De Valois and Switkes, 1983; Switkes, Bradley and De Valois, 1988). If the mask varies in luminance and the test pattern is an isoluminant chromatic grating, the direction of the masking effect will depend upon the contrast of the mask—as is the case for luminance–luminance masking or color–color masking. For low mask contrasts, contrast sensitivity will be increased, and for very high mask contrasts, contrast sensitivity will decrease. The similarity is only qualitative, however. The range of contrasts over which facilitation occurs when the mask is luminance and the test is chromatic is quite different. For this direction of cross-masking, significant facilitation is not seen until the mask reaches its own threshold, and it continues until the mask contrast is roughly 16 to 32 times threshold. When mask and test are either both color or both luminance, facilitation is only seen up to contrasts of perhaps 4 to 6 times threshold, and it first appears when the mask itself is well below detection threshold. When mask contrast exceeds the facilitatory region, luminance masks begin to impede the detection of a chromatic test, as they do a luminance test.

If the mask is an isoluminant chromatic grating and the test a luminance grating, however, the results are very different. At no mask contrast is there facilitation. As long as the mask contrast is below threshold there is no discernible effect. Once it exceeds its own threshold, however, it begins to impede the detection of a luminance test pattern.

The masking interactions between chromatic and luminance gratings differ in another respect from those seen within either domain. If a luminance test is superimposed upon a luminance mask of identical spatial frequency, the detectability of the test will, of course, depend upon the relative phase at which the two gratings are added. When test and mask are in phase, the task becomes one of contrast increment detection. When they are 180° out of phase, the task is decrement detection. These might well be similar (for example, when the mask contrast is significantly above its threshold—Kulikowski, 1976). However, when the phase relationship is neither, the resulting change in contrast will be lower.

Thus, for si
change in c
for either e?

The masl
however, ar
added. It :
insensitivit
due to an u
the "chrom
and if it wei
luminance
relative test
indicated.

Our orig
Switkes, 19
chromatic s
green phos|
blue–yellov
yellow refl
produced v
from those

CIE coor
point of thi
experiment
luminance
a tritanopic
confusion a
only by virt
any pair of
and L cone
the sparse i
1981), spat
axis must b
S-cone sysi
detected so
different rr
luminance
from the o
Eisner and
underlying
by the sarr
visual syst(
due simpl)
function, a

uminance—
ruction of a
ations. The
be similar,
ormation is
while spatial
r than that
:ful pattern
se it. Thus,
ling spatial

: contrasts
iultaneous
:tween the
luminance
s, Bradley
ttern is an
ill depend
: masking
y will be
decrease.
'er which
c is quite
not seen
: contrast
i color or
> 6 times
ietection
:e masks
: test.
minance
is there
cernible
iede the

liffer in
: test is
detect-
the two
one of
task is
: mask
, when
lower.

Thus, for superposition of a test grating of a given contrast, the detection of a net change in contrast should be more difficult for intermediate relative phases than for either extreme. We have verified that this is so.

The masking of luminance by color and the masking of color by luminance, however, are both independent of the relative phase in which the two patterns are added. It appears not to matter how the mask and test are aligned. This insensitivity to relative phase makes it unlikely that the observed interactions are due to an undetected luminance artifact in the putative isoluminant gratings. If the "chromatic" gratings in fact contained significant luminance components, and if it were these components which were responsible for the interactions with luminance gratings, then those interactions should be dependent upon the relative test/mask phase. Since they are not, such an explanation seems contra-indicated.

Our original reports of asymmetric color/luminance masking (De Valois and Switkes, 1983; Switkes and De Valois, 1983) described experiments in which the chromatic stimuli lay along a red–green axis determined by the monitor's red and green phosphors. More recently, we have used other chromatic axes, including a blue–yellow axis in which the blue was determined by the blue phosphor and yellow reflected the best approximation to a unique yellow which could be produced with our equipment. Results with the blue–yellow axis did not differ from those found with the red–green axis. Summary data are shown in Fig. 1.

CIE coordinates of one axis of particular interest are shown in Fig. 2. The center point of this axis is that of Illuminant C, a standard white. In this, as in all our experiments, the space-and time-averaged chromaticity did not vary. Thus, the luminance patterns were white–black. The chromatic axis, in this case, lay along a tritanopic confusion line. Variations in chromaticity which lie along a tritanopic confusion axis and which have no associated luminance change can be detected *only* by virtue of their differential effect on the short-wave sensitive (S) cones. For any pair of luminance-matched points along such an axis, the absorption by the M and L cones is invariant. This is of particular interest for two reasons. First, given the sparse retinal distribution of S cones (De Monasterio, Schein and McCrane, 1981), spatial acuity for a pattern produced by variations along such a chromatic axis must be poor. The evolutionary history and the vulnerability to injury of the S-cone system also seem to differ from those of the M and L cones. Patterns detected solely by variations in S cone activity could be processed in a significantly different manner. Secondly, there is considerable evidence to suggest that the luminance signal—if indeed there exists a single "luminance" signal—derives from the output of M and L cones but has little or no input from S cones (e.g. Eisner and MacLeod, 1980). This is important in a consideration of possible underlying mechanisms, since chromatic and luminance information are carried by the same physiological substrates throughout at least the early levels of the visual system. If the interactions we observe between color and luminance were due simply to the sharing of a common pathway with a compressive response function, and if the S cones in fact do not contribute substantially to the luminance

288   K. K. DE VALOIS AND E. SWITKES
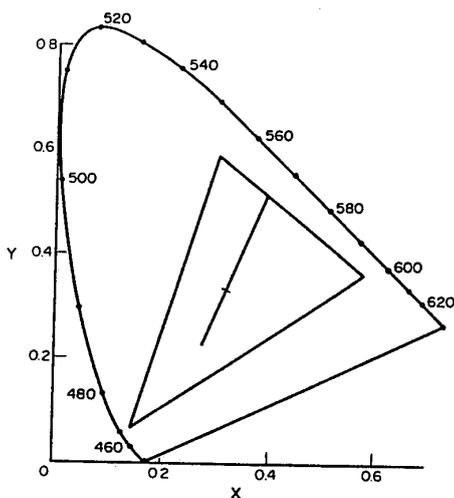
FIG. 2. Tritanopic confusion axis modulated around Illuminant C. Equiluminant modulation along such an axis can be detected only by the differential absorption produced in the S cones. The triangle defines the color space which can be realized with the monitor used (19" Mitsubishi color monitor).
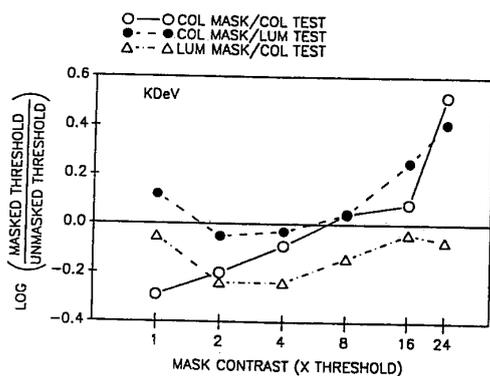
FIG. 3. Increment contrast thresholds [log(masked threshold/unmasked threshold)] plotted as a function of normalized mask contrast (mask contrast/threshold contrast). All stimuli (both masks and tests) were 1 c/deg gratings. The chromatic patterns ("color") were isoluminant gratings in which chromaticity was modulated along the tritanopic confusion axis shown in Fig. 2. The luminance patterns had the same space-averaged luminance (8 ft. L.) and chromaticity (that of Illuminant C). Here, negative values illustrate facilitation—the test is more detectable in the presence of the mask—and positive values indicate masking.

pathway, th
confusion ax
those observ
experiment o

When bo
confusion a×
similar to th
similar to t
variations. V
triangles, d
for interactic
facilitation f
and test is a
similar to th
mask contra
threshold, it
tions betwee
and a lumin
axes and lun
result from
function, if
luminance p
using a red-
concluded tl
masks at ar
pathway.

Finally, v
by luminan
tions. Spec
chromatic a
ing set of ob
respond to
retinal or ge
center inpu
surround be
chromatic c
spatial tunir
luminance
absent for p

Different
in cortical a
chromatic g
frequency 1
significant r

pathway, then the interactions between a pattern varying along a tritanopic confusion axis and a luminance pattern might be expected to be minimal, unlike those observed with other chromatic axes. Figure 3 illustrates the results of an experiment designed to test this possibility.

When both mask and test patterns are gratings varying along a tritanopic confusion axis (open circles, solid lines), the contrast-masking function is very similar to that found for other chromatic axes we have utilized. It is also very similar to that found with mask and test patterns produced by luminance variations. When the test is a tritan pattern and the mask is a luminance pattern (triangles, dash–dot lines), however, the results are again similar to those found for interactions between other chromatic axes and luminance. There is significant facilitation for low to moderate mask contrasts. When the mask is a tritan pattern and test is a luminance pattern (filled circles, dashed lines), the results are again similar to those observed with other chromatic axes. At low (i.e. subthreshold) mask contrasts there is no significant effect. When the mask is well above its own threshold, it impedes the detectability of the luminance test. Thus, the interactions between an isoluminant pattern which lies along a tritanopic confusion axis and a luminance pattern are very similar to those seen between other chromatic axes and luminance patterns. This suggests that these interactions do not simply result from the sharing of a common pathway with a compressive response function, if indeed S cones do not make a significant contribution to the luminance pathway. These results are consistent with our earlier masking studies using a red–green axis (Switkes, Bradley and De Valois, 1988), in which we concluded that the lack of facilitation of the detection of luminance tests by color masks at *any* contrast eliminates masking originating from an early common pathway.

Finally, we have compared the *appearance* of gratings which are defined solely by luminance variations and those which are defined solely by chromatic variations. Specifically, we have compared the apparent spatial frequencies of chromatic and luminance gratings. This comparison is motivated by an interesting set of observations relating to the spatial frequency tuning of single cells which respond to both chromatic and luminance patterns. If one considers a simple retinal or geniculate chromatic opponent cell, with one cone type providing the center input and another the surround input, it can be shown that center and surround behave synergistically for luminance changes, but antagonistically for chromatic changes (De Valois and De Valois, 1975). This results in different spatial tuning functions for color and for luminance within individual cells, with luminance patterns showing some low spatial frequency attenuation which is absent for purely chromatic patterns.

Different spatial tuning functions have also been reported for individual cells in cortical area V1. Although most cells which respond to either isoluminant chromatic gratings or isochromatic luminance gratings have very similar spatial frequency peaks and bandwidths (Thorell, De Valois and Albrecht, 1984), a significant minority differ. When they do so, the difference is virtually always in

the same direction: the peak spatial frequency for chromatic gratings is lower than the peak spatial frequency for luminance gratings, and the spatial bandwidth for color is broader than that for luminance.

Consider the problem associated with the interpretation of signals from such cells which carry both chromatic and luminance information. Suppose that one's estimate of the spatial frequency of a given grating is based upon either the identification of the single most responsive cell or upon some population average of all the cells responding to the pattern. If a given cell carries only a single frequency label—and it is difficult to imagine that the case is otherwise—a serious problem arises. For a cell which is tuned identically for color and for luminance gratings, the signal would be unambiguous. Chromatic and luminance gratings of the same veridical spatial frequency would match perceptually in spatial frequency. A maximum response would appropriately signal the same spatial frequency irrespective of the type of contrast (chromatic or luminance) which produced it. If, however, the tuning for color and for luminance is not identical, then that signal alone cannot be unambiguously interpreted. A maximum response might be produced by, say, a 2 c/deg luminance grating but by a 1.8 c/deg chromatic grating. If a subject's perception or identification of the stimulus spatial frequency were based solely upon the activity of such a cell, then a 2 c/deg luminance grating should appear to have the same spatial frequency as a 1.8 c/deg chromatic grating. Thus, to match the apparent spatial frequency of a 2 c/deg luminance grating, a subject would have to set a chromatic grating to a lower real frequency.

Averaging over the activity of a large population of cells does not solve the problem. Since a large proportion of V1 cells are identically tuned for chromatic and luminance spatial frequencies, the magnitude of the expected mismatch would be reduced. However, since the distribution of chromatic-luminance tuning functions is strongly skewed (in the direction of lower peak frequencies for color, higher peak frequencies for luminance, Thorell, De Valois and Albrecht, 1984), the matching bias would still be expected. To match the apparent spatial frequency of a luminance grating of 2 c/deg, a subject should set a chromatic grating to a lower veridical spatial frequency.

The same prediction still arises if one considers the different spatial frequency bandwidths of many cells for chromatic and luminance gratings. When these differ, the filter functions for chromatic gratings tend to be low-pass, while those for luminance gratings (in the same cells) are bandpass. Averaging across such cells will yield an estimate of chromatic spatial frequency which is higher than the veridical frequency if each cell simply carries a single frequency label reflecting its peak luminance frequency.

If the spatial frequency tuning characteristics of indivual cells were irrelevant to the perception of the spatial frequency of a grating, one might assume that estimation of a grating's spatial frequency would be based upon something like a point-to-point representation of the pattern. In that case, a judgment of spatial frequency could be made by, for example, estimating the cortical distance

between t
should le
luminance
occur in ei
We hav
frequency
gratings. '
observers
with no f
also collec
temporal 1
answer wh
of interme
frequency
subject is
chromatic
Figure
frequencie
grating of
luminance
somewhat

Fig. 4.
appear
referen
filled
percep
grating

ngs is lower than
al bandwidth for

gnals from such
ppose that one's
upon either the
)ulation average
es only a single
:wise—a serious
d for luminance
ance gratings of
ially in spatial
1e same spatial
ninance) which
is *not* identical,
. A maximum
iting but by a
ification of the
uch a cell, then
frequency as a
:equency of a 2
iting to a lower

not solve the
for chromatic
ted mismatch
tic-luminance
requencies for
and Albrecht,
parent spatial
t a chromatic

:ial frequency
When these
;, while those
g across such
gher than the
l reflecting its

:re irrelevant
assume that
iething like a
:nt of spatial
ical distance

between the representations of zero crossings. Any such purely spatial model should lead to the prediction of veridical matches between chromatic and luminance gratings. There is no basis upon which to assume that a bias would occur in either direction.

We have tested these predictions by having subjects match the apparent spatial frequency of isoluminant chromatic gratings and isochromatic luminance gratings. We used a two-alternative spatial forced-choice procedure in which observers indicate which grating appears to have the higher spatial frequency (with no feedback) to find which spatial frequencies appear the same. We have also collected significant amounts of data using both method of adjustment and temporal forced-choice procedures. All procedures yield the same answer—an answer which is contrary to all of the predictions described. A chromatic grating of intermediate spatial frequency *appears* to be lower—not higher—in spatial frequency than a luminance grating of the same veridical frequency. If, then, a subject is asked to match a 2 c/deg luminance grating, s/he will do so with a chromatic grating which is *higher* in real frequency.

Figure 4 presents averaged data from two subjects. At intermediate spatial frequencies, a luminance reference grating is always matched by a chromatic test grating of a higher veridical frequency. A chromatic reference requires a luminance test of a lower real frequency. The magnitude of the mismatch falls off somewhat at higher spatial frequencies and appears to be maximal around 2 c/deg.
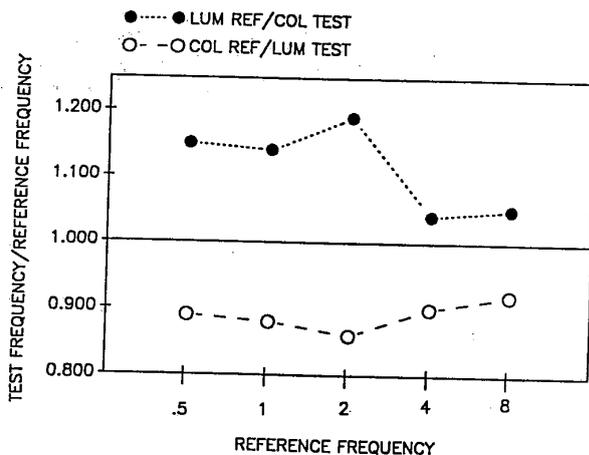
FIG. 4. Relationship of a reference spatial frequency to the test spatial frequency which appears to match it perceptually, plotted as a function of the reference frequency. When the reference (standard) is a luminance grating and the test is an isoluminant chromatic grating (filled circles, dotted lines), the test must be set to a higher *real* frequency to produce a perceptual match. When the reference is a chromatic grating and the test is a luminance grating, the test must be set to a lower veridical frequency. Both reference and test gratings were set to equal multiples of their respective threshold contrasts.

292   K. K. DE VALOIS AND E. SWITKES

The data shown here were collected using a chromatic pattern in which the chromatic axis produced constant absorption by the S cones. Similar results, however, are found when the chromatic grating lies along either a tritanopic confusion axis or a red–green axis determined by the monitor's red and green phosphors. When two different chromatic axes are reference and test gratings, respectively, the matches are made accurately, as they are for either color or luminance when test and reference are the same. The data shown here were collected using equal multiples of threshold contrast for reference and test. The direction and the magnitude of the mismatch do not vary as contrast is scaled up or down over a range of from 2 to 20 times threshold, as long as the two patterns are matched on that metric.

Data such as these suggest that simple models relating pattern identification to the tuning characteristics of individual cells—or, indeed, to the tuning characteristics of groups of cells—will be insufficient. The fact that the matches are not veridical also suggests that a model based upon a point-to-point spatial representation will fail, as well. It is clear that our elementary notions of the manner in which the activity of neurons in the visual system are interpreted are at best inadequate.

In summary, we have shown that there exist two-dimensional spatially-bandpass mechanisms which subserve the detection and analysis of patterns based purely upon chromatic contrast, that when both chromatic and luminance contrast coexist within a single pattern their interactions are significant and complex, and that primitive models relating the activity of single visual cells to the perception of even simple pattern attributes are inaccurate.

References

Albrecht, D. G., Farrar, S. B. and Hamilton, D. B. (1984) Spatial contrast adaptation characteristics of neurones recorded in the cat's visual cortex. *J. Physiol.* 347, 713–739.

Blakemore, C. and Campbell, F. W. (1969) On the existence of neurones in the human visual system selectively sensitive to the orientation and size of retinal images. *J. Physiol.* 203, 237–260.

Bradley, A., Switkes, E. and De Valois, K. K. (1988) Orientation and spatial frequency selectivity of adaptation to color and luminance gratings. *Vision Res.* 28, 841–856.

De Monasterio, F. M., Schein, S. J. and McCrane, E. P. (1981) Staining of blue-sensitive cones of the macaque retina by a fluorescent dye. *Science* 213, 1278–1281.

De Valois, R. L. and De Valois K. K. (1975) Neural coding of color. In: *Handbook of Perception V* (Carterette, E. C. and Friedman, M. P., eds.). pp. 117–166. New York: Academic Press.

De Valois, K. K. and Switkes, E. (1983) Simultaneous masking interactions between chromatic and luminance gratings. *J. Opt. Soc. Amer.* 73, 11–18.

De Valois, K. K., Webster, M. and Switkes, E. (1984) Orientation selectivity for luminance and color patterns. *Invest. Ophthalmol. Visual Sci.* (Suppl.) 25, 232.

De Valois, R. L., Albrecht, D. G. and Thorell, L. G. (1982) Spatial frequency selectivity of cells in macaque visual cortex. *Vision Res.* 22, 545–559.

De Valois, R. L., Yund, E. W. and Hepler, N. (1982) The orientation and direction selectivity of cells in macaque visual cortex. *Vision Res.* 22, 531–544.

Eisner, A. and MacLeod, D. I. A. (1980) Blue-sensitive cones do not contribute to luminance. *J. Opt. Soc. Amer.* 70, 121–123.

Eisner, A. (1978) Hue difference contours can be used in processing orientation information. *Percep. and Psychophys.* 24, 451–456.

Gilinsky, A. S. (1968) Orientation-specific effects of patterns of adapting light on visual acuity. *J. Opt. Soc. Amer.* 58, 13–18.

Hubel, D. H. and
architecture in
Ingling, C. R., Jr. at
Vision (Mollon.
Kelly, D. H. (1983
J. Opt. Soc. An
Kulikowski, J. J. (1
1419–1431.
Legge, G. E. and F
1458–1471.
Mullen, K. T. (1985
chromatic grati
Ohzawa, I., Sclar, C
Nature 298, 587
Switkes, E., Bradley
among chromat
Switkes, E. and De
In: Colour Visi
Press.
Thorell, L. D., De V
pure color and I
van der Horst, G. J
J. Opt. Soc. Am

n which the
ailar results,
a tritanopic
:d and green
est gratings,
her color or
n here were
nd test. The
: is scaled up
two patterns

atification to
ning charac-
:ches are not
l representa-
ner in which
adequate.
al spatially-
of patterns
l luminance
lificant and
l cells to the

characteristics

ι visual system
:37–260.
γ selectivity of

re cones of the

f Perception V
c Press.
:hromatic and

ιnce and color

rity of cells in

:tivity of cells

ιance. J. Opt.

ition. Percep.

:uity. J. Opt.

Hubel, D. H. and Wiesel, T. N. (1962) Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol.* **160**, 106–154.

Ingling, C. R., Jr. and Martinez, E. (1983) The spatiochromatic signal of the r–g channel. In: *Colour Vision* (Mollon, J. D. and Sharpe, L. T., eds.), pp.433–444. London: Academic Press.

Kelly, D. H. (1983) Spatiotemporal variation of chromatic and achromatic contrast thresholds. *J. Opt. Soc. Amer.* **73**, 742–750.

Kulikowski, J. J. (1976) Contrast constancy and the linearity of contrast sensation. *Vision Res.* **16**, 1419–1431.

Legge, G. E. and Foley, J. M. (1980) Contrast masking of human vision. *J. Opt. Soc. Amer.* **70**, 1458–1471.

Mullen, K. T. (1985) The contrast sensitivity of human colour vision for red–green and blue–yellow chromatic gratings. *J. Physiol.* **359**, 381–400.

Ohzawa, I., Sclar, G. and Freeman, R. D. (1982) Contrast gain control in the cat's visual cortex. *Nature* **298**, 5871–5872.

Switkes, E., Bradley, A. and De Valois, K. K. (1988) Contrast dependence of masking interactions among chromatic and luminance gratings. *J. Opt. Soc. Amer.* **A5**, 1149–1162.

Switkes, E. and De Valois, K. K. (1983) Luminance and chromaticity interactions in spatial vision. In: *Colour Vision* (Mollon, J. D. and Sharpe, L. T., eds.), pp. 465–470. London: Academic Press.

Thorell, L. D., De Valois, R. L. and Albrecht, D. G. (1984) Spatial mapping of monkey V1 cells with pure color and luminance stimuli. *Vision Res.* **24**, 751–769.

van der Horst, G. J. C. and Bouman, M. A. (1969) Spatiotemporal chromaticity discrimination. *J. Opt. Soc. Amer.* **59**, 1482–1488.

A Measure for Stairstepping in Digitized Text that Correlates with the Subjective Impression of Quality

J. Raymond Edinger, Jr.
Eastman Kodak Company, Rochester, New York

Abstract

The electrophotographic printer/copier industry is being driven to produce digital equipment with ever higher addressability and bits per pixel. One reason for this drive is to minimize the phenomenon of stairstepping in the diagonal strokes of text. To assess improvements in text image quality, it is important to be able to measure the level of stairstepping to ascertain whether any changes are visually meaningful. This paper describes a measure of stairstepping that has been shown to correlate with the subjective impression of text quality for a variety of levels of addressability and bits per pixel. The assessment method has the distinct advantage of not requiring special targets but is simply measured from the letter "o."

Introduction

Raggedness, in general, refers to geometric distortion of an edge from its nominal position so that the edge appears wavy or jagged rather than smooth and straight. It is also referred to as tangential edge profile. Stairstepping refers to the special case of raggedness that occurs in diagonal lines printed by a digital system. For example, stairstepping can often be seen in the diagonal stroke in an upper case N (see example illustrated in Figure 2b). Its visual impact usually decreases with increased writing addressability or dots per inch.

Raggedness measurement[1-6] is usually limited to noise along intrack and crosstrack edges with less regard to assessment of stairstepping. Indeed, until recently, there had been little need to quantify stairstepping since stairstepping was easily predicted; that is, it was simply a function of dpi (dots per inch) and the angle of the edge relative to the direction of the writer. With the advent of multiple bits per pixel and other enhancements, however, the visual impact of stairstepping has become less predictable. Thus, a metric for stairstepping that correlates with the subjective impression of text quality is desirable. Since stairstepping is a function of an edge's angle, evaluating edges or lines of a *standard angle* would be required to consistently benchmark stairstepping across products and processes. In fact, an approach often used to measure raggedness (e.g., comparing the location of points along an edge to the ideal edge location as defined by linear regression) has been successfully applied to evaluate stairstepping in lines at a given angle.[1] But since one could not count on having a stroke at the "standard" angle in any given body of text, a more universal measure was desirable; i.e., one that would not require a special target.

Mr. Allan Kaplan, at Eastman Kodak Company, suggested evaluating raggedness by examining the diameter of o's and their departure from roundness. Unfortunately, o's are usually not perfect circles to start with. To satisfactorily apply the technique to an o's shape, a priori knowledge of the o's ideal shape would be needed, or only o's that were supposed to be perfect circles could be used. Both solutions would prevent the measure from being truly universal.

Evaluating raggedness by measuring the length of the perimeter around the letter o has been successfully applied for comparison of ragged o's to perfectly smooth o's.[7] This approach, however, does not work for comparing edge quality of digitized text made at one level of dpi to that made at another level. Close consideration reveals that if a digitized o is created by square pixels, its perimeter is equal to four times the average diameter of the o, *regardless of pixel size*. Hence, the perimeter of an ideal 120 dpi o, for example, is exactly the same as one created at 600 dpi. Naturally, toner clumping, digitization errors, round pixels, etc., tend to invalidate this postulate, but a measure that does not show differences under "ideal" conditions could hardly be sensitive enough to show differences in more practical cases.

It was noted that noncircularity of o's (viz., an oval instead of a circle) is not visually objectionable (unless the o is grossly distorted) and, indeed, most o's are intentionally noncircular. What might make an o appear objectionable, then, would be roughness around the edge and not the fact that it is not a circle. For this reason, Mr. Kaplan's suggestion of examining diameters was pursued. The approach selected, then, was to measure the *instantaneous change* in diameter as one traveled around the o's perimeter.

Development of the Algorithm

To effect the stairstepping measure, an image analyzer is used to capture the o's. From the analyzer intercept maps, diameters are calculated for sixteen different

HUAWEI EX. 1116 - 104/714

angles taken every 11.25 degrees around the o. Figure 1 shows a schematic of the first nine angles measured. Typically, image analyzers provide the Feret's diameter of objects. For o's with significant indentations in the perimeter, results reported using Feret's diameters may tend to smooth the apparent raggedness. This is because Feret's diameter is similar to measuring with a caliper; i.e., it misses relatively small indentations and bases the resulting diameter on peaks. Hence to assure that the true diameter around the o's perimeter would be obtained, o diameters were calculated off-line based on intercept maps produced by the image analyzer. An intercept map is simply a bit map based on the thresholded binary image and shows only pixels at transitions between on and off.

For a 12-point lower case o, measuring every 11.25 degrees gives an assessment of the o's edge about every 150 micrometers along its perimeter. The measurement window is about 8 micrometers wide, allowing resolution to pixels significantly smaller than created by a 600 dpi writer.

Pictured in Figures 2a-2f are photomicrographs (approx. 16X) of an assortment of o's with various levels of raggedness. N's from the same conditions are also pictured to show the associated stairstepping in the diagonals. Figures 3a-3f are graphs of the diameters of the same o's versus measurement angle. These o's, along with six others, were ranked by two persons for subjective impression of quality. The o's were then captured by the image analyzer and stairstepping values calculated as described below. Table I compares the quality rank by the two judges with that given by the objective measurement. The Spearman's rank correlation coefficient between the two methods is 0.965, which indicates good agreement between the judges' ranking and rank by the objective measure. (A rank correlation value of +1 indicates complete agreement; -1 indicates complete disagreement; and 0 indicates no relationship.)

As mentioned above, it was reasoned that instantaneous changes in an o's diameter are more important than the more subtle differences resulting from o's being ovals rather than perfect circles. Compare the o and graph in Figures 2a and 3a with those pictured in Figures 2b and 3b. Although the o in Figure 2a is far from being a circle, its diameter changes smoothly as shown in the accompanying graph of diameter versus measurement angle (Figure 3a). The judges ranked the o in Figure 2a as the second best in the series compared to rank 11 for the one shown in Figure 2b (the lower the rank, the higher the quality).

Table II lists the raw data for the diameter-versus-angle function along with an analysis of the function's changing slope for the o shown in Figure 2a. The second derivative of a 4th degree polynomial fitted to the diameter-versus-angle data is shown in Figure 4 to help illustrate the analysis. The magnitudes of slope changes used to quantify stairstepping, however, are not taken from the second derivative function, nor is a regression fit to the raw data used. Either of these would tend to smooth out the raggedness. Instead, the actual value for the

change in slope that is closest to the inflection point is used for the assessment.

For the function in Figure 3a, there are two inflection points, denoted A and B. The absolute values of change in the function's slope around the inflection points is 8 and 29 units for A and B, respectively. The slope actually changes larger amounts at other points: for example, it changes 62 units at point C and 39 units at the function's nadir. But the changes in slope at C and the nadir are not associated with inflection points and so would not likely give a visual impression of a step in the o's perimeter. No step is visibly apparent at the points represented by inflections A and B either, but the analysis to follow will show these changes in slope to be rather inconsiderable when compared with samples that *do* show significant steps along the perimeter, e.g., Figure 2b.

The o in Figure 2b was made with a digital electrophotographic system operating at 300 dots per inch. The eleven inflection points in the diameter-versus-angle function are labeled on the graph in Figure 3b. The differences in slope around the inflection points range from 11 to 158 units. The largest (158) corresponds with the "notches" at 4 and 10 o'clock and the second largest (146) corresponds to the "notches" at about 2 and 8 o'clock (see Figure 2b). From this example and the one illustrated in Figure 2a, it seems that slope changes, besides requiring a change in sign, must change greater than about 40 units to produce a visually apparent step in the o's perimeter.

Another example of inflection points marked by significant changes in slope is given in Figures 2c and 3c. This o was made with a 120 dpi writer. The diameter-versus-angle function has nine inflection points with differences in slope ranging from 27 to 534 units. Clearly, this o has significant steps along its perimeter, which are also readily apparent in the diameter-versus-angle function. Once again, readily visible steps in the perimeter correspond to slope changes associated with an inflection point plus a change in magnitude greater than 40.

And finally, an example is given in Figures 2d and 3d of an o that has ten inflection points, all with slope differences that are minute.[8] They range from only 3 to 10 units. The diameter-versus-angle function is almost flat and appears smooth, as does the dot's perimeter. Thus again, slope changes less than about 40 units are probably of little visual consequence. (This target is a dot etched in chrome on glass. The variations in diameter, which run to about +/- 5 micrometers, are primarily due to the limited resolution of the image analyzer resulting in errors in digitization of the image.)

To summarize, raggedness in the perimeter of o's is most apparent when two conditions are met: (1) the point represents an inflection point in the slope-versus-diameter function, and (2) the change in slope is greater than 40 units.

From these examples it is evident that, with increasing differences in slope around an inflection point, the visual impact of raggedness becomes significantly

greater. The algorithm, then, was configured as the square root of the sum of the squares of the magnitudes of the differences in slope associated with each inflection point of the diameter-versus-angle function.

Correlation of the Objective Measure with Subjective Scaling

To further confirm correlation of the algorithm with the subjective impression of text quality, a series of about seventy prints of text imagery made at various combinations of dots per inch and bits per pixel was obtained. These prints, which were copies of a typical business letter using 12-point Times Roman font, had been previously rated for text quality by over thirty judges. An interval subjective quality scale based on this subjective evaluation of the prints had been determined. The stairstepping in each print was measured on three to six o's at various locations in the body of the print's text.

Prior to looking for correlation between the new algorithm and subjective scaling, a check was made to see if the algorithm correlated with the presence of *measurable* stairstepping using a "conventional" raggedness evaluation technique. The relative stairstepping of the text in each print was measured on the slash ("/") character using a technique based on tangential profile of edges.[4] Figure 5 shows the resulting correlation. Next, correlation between this measure of stairstepping (based on the "/") and subjective scaling for quality was sought. The data in Figure 6 confirm that the objective stairstepping measure based on the "/" correlates with the subjective interval scale of text quality. This indicates that the survey judges were responding, to a significant degree, to stairstepping.

Based on the correlation in Figure 6, a perfectly satisfactory measure of stairstepping could be had with a conventional raggedness measure using the "/." As pointed out earlier, however, one can not rely on having a "/" in test imagery, and hence the need existed to develop a universal target for measuring stairstepping. It now remained to be seen if the new stairstepping measure based on o's correlates with the survey's subjective scale. The data are plotted in Figure 7. The correlation coefficient (r-squared) is about 0.65, which, for seventy pairs of data, indicates significant correlation.

The noise in Figure 7 is not atypical of results from subjective surveys. In addition to the usual noise associated with subjective testing, it was found that the raggedness of the o's, themselves, varies substantially within a page. The standard deviation of measured stairstepping in o's across a page of text averaged about 40 units. Thus it is important to take a statistically appropriate sampling of o's.

Conclusion

Analyzing variations in diameter around a 10- to 12-point letter o, with little need for regard to font,[9] correlates with the subjective impression of stairstepping in digital text. The measure has been demonstrated to rank with the subjective impression of quality for a variety of fonts and levels of stairstepping. Compare the N and o in Figure 2b with those in Figure 2e. The stairstepping measure gives a value of 291 for the o in Figure 2b compared to only 165 for the one in Figure 2e. The stairstepping difference, although easily seen in a photomicrograph, is subtle at normal viewing magnifications. Yet, it is small differences like this that provide discrimination between good and excellent digital text, between a page of text that looks crisp and uniform and one that looks second-rate. And being able to *quantify* reductions in stairstepping (e.g., from 291 to 165 units) is important.

The industry is being driven to produce digital copier and printer systems with ever higher addressability or multiple bits per pixel, or both. Minimizing stairstepping is one reason for this drive. The technique described in this paper has been shown to correlate well with both the actual stairstepping measured in survey stimuli and with the subjective impressions of text quality.

Acknowledgement

References and Notes

1. Crawford, J. L., C. D. Elzinga, and R. Yudico, "Print Quality Measurements for High Speed Electrophotographic Printers," *IBM J. Res. Dev.*, 28:276 (1984).
2. Dainty, J. C., D. R. Lehmbeck, and R. Triplett, "Modeling of Edge Noise in Particulate Images," *J. Imag. Technol.*, 11:131-136 (1985).
3. Dvorak, C. A., and J. R. Hamerly, "Just-Noticeable Differences for Text Quality Components," *J. Appl. Photogr. Eng.*, 9:97-100 (1983).
4. Edinger, J. R., "The Image Analyzer - A Tool for the Evaluation of Electrophotographic Text Quality," *J. Imag. Sci.*, 31:177-183 (1987).
5. Hamerly, J. R. and R. M. Springer, "Raggedness of Edges," *J. Opt. Soc. Am.*, 71:285-288 (1981).
6. Tanaka, Y. and T. Abe, "Quantitative Analysis of Print Quality Features," *J. Imag. Technol.*, 13:202-207 (1987).
7. Jansson, L., "Print Quality," *Byte*, 199-207, September 1987.
8. The fact that this o is really a dot is of no consequence since only the outside diameter is considered.
9. Although the measure appears to be relatively insensitive to font, it is suggested that italic o's be avoided.

HUAWEI EX. 1116 - 106/714

Table I. Rank by Judges Correlated with Rank by Objective Stairstepping Measure

Image Type	Judges' Rank	Objective Rank	Objective Measure
* Chrome dot	1	1	22
* Chrome O	2	2	30
Phototypeset paper	3	3	34
* Offset on coated stock	4	4	63
Offset on card stock	5	5	101
Offset on bond paper	8	6	129
Simulated typewriter	7	7	149
* 300 dpi, enhanced	6	8	165
300 dpi, sample A	9	9	175
* 300 dpi, 1 bit 11	10	291	291
300 dpi, sample B	10	11	294
* 120 dpi	12	12	959

* Illustrated in Figure 2.

Table II. Diameter Versus Angle Analysis for Chrome O (Figures 2a and 3a)

Angle	Diameter	Delta D	Change in Slope	
0	2157			
		21		
11.25	2136		-112	
		133		
22.5	2003		-69	
		202		
33.75	1801		+8	inflection point A
		194		
45	1607		+36	
		158		
56.25	1449		+39	
		119		
67.5	1330		+62	point C
		57		
78.75	1273		+42	
		15		
90	1258		+39	nadir
		-24		
101.25	1282		+58	
		-63		
112.5	1345		+29	
		-121		
123.75	1466		+63	
		-150		
135	1616		-29	inflection point B
		-213		
146.25	1829		-70	
		-184		
157.5	2013			
		-114		
168.75	2127			

RMS of absolute slope changes around infection point is: $(8^2 + 29^2)^{1/2} = 30$.
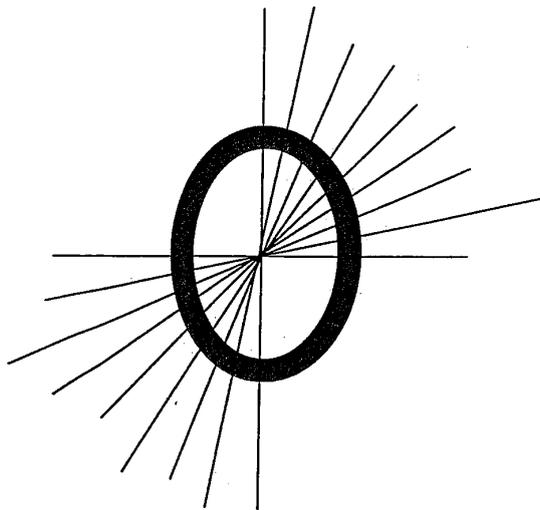
Figure 1. Schematic of first nine diameter measures
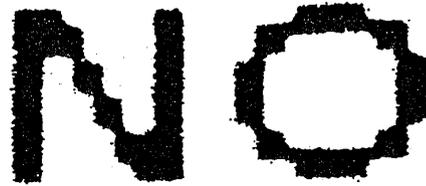
Figure 2c. 120 dpi: Stairstepping = 959
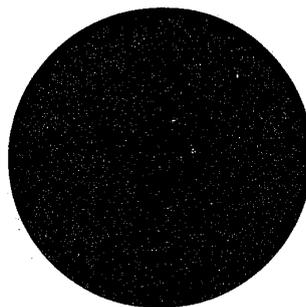
Figure 2d. Etched chrome dot on glass: Stairstepping = 22
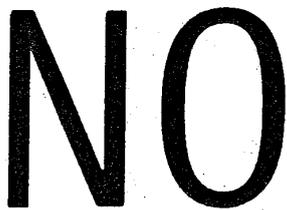
Figure 2a. Etched chrome on glass: Stairstepping = 30

Figure 2e. 300 dpi, enhanced: Stairstepping = 165

Figure 2b. 300 dpi, 1 bit: Stairstepping = 291

Figure 2f. Offset lithography: Stairstepping = 63
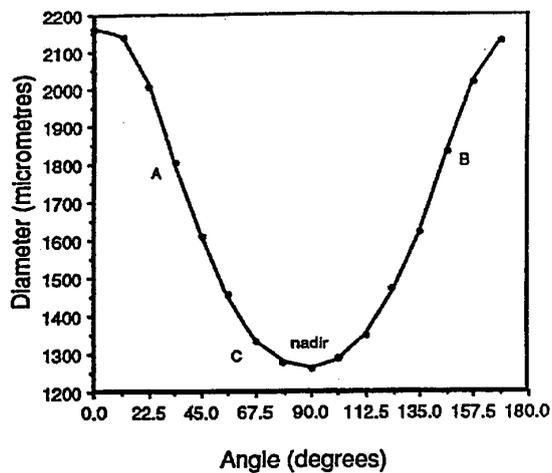
Figure 3a. Upper case O etched in chrome
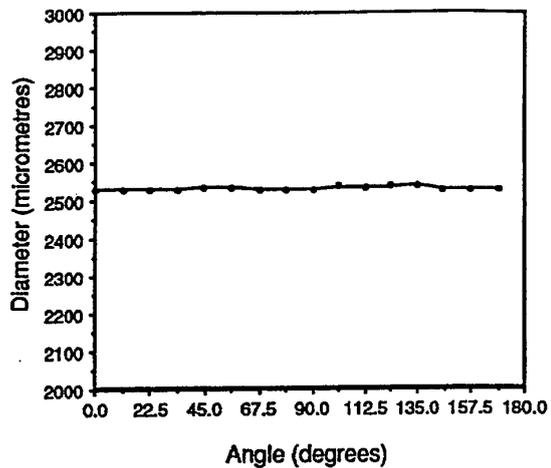
Figure 3d. 2.5 mm chrome dot on glass
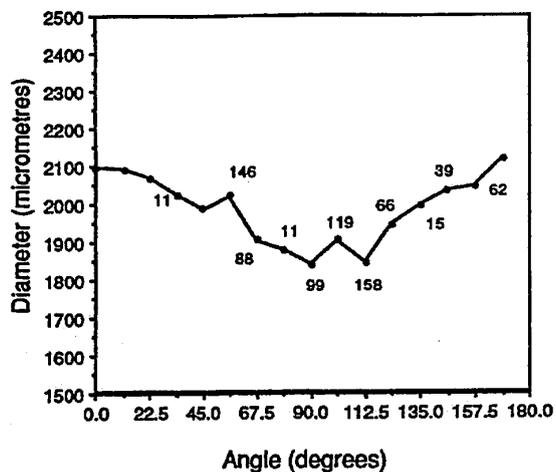
Figure 3b. Letter "o" at 300 dpi, 1 bit
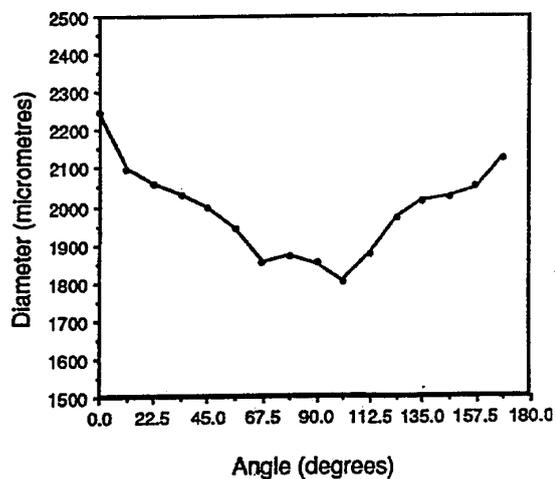
Figure 3e. Letter "o" at 300 dpi, enhanced
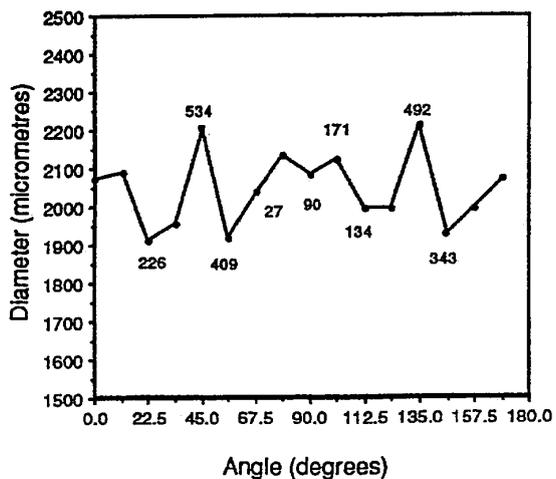
Figure 3c. Letter "o" at 120 dpi, 1 bit
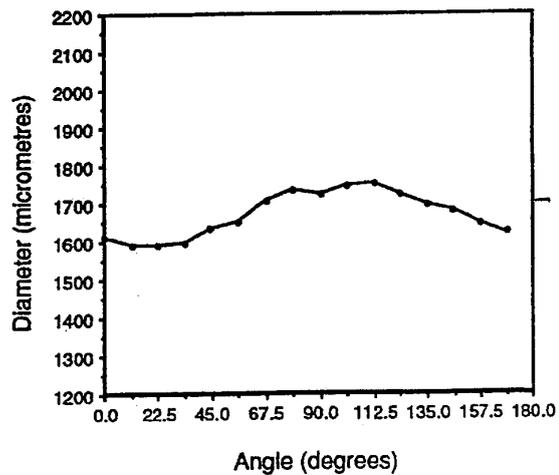
Figure 3f. Offset lithograph "o" on coated stock
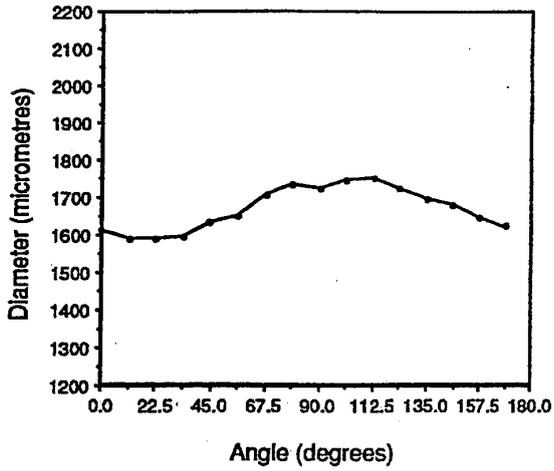
Figure 4. Slope analysis of diameter-versus-angle function for
chrome o
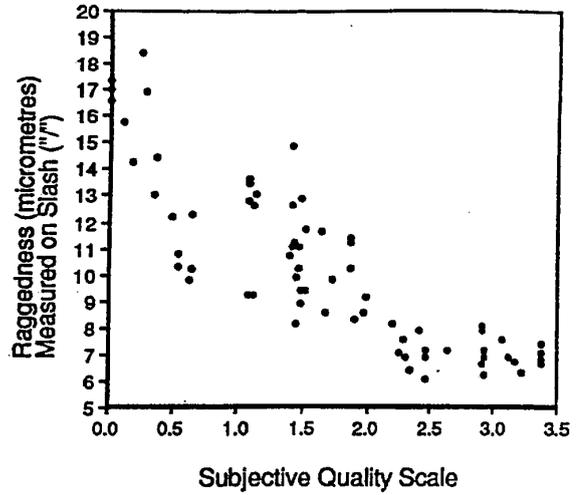
Figure 6. Raggedness measure correlates with subjective
scaling
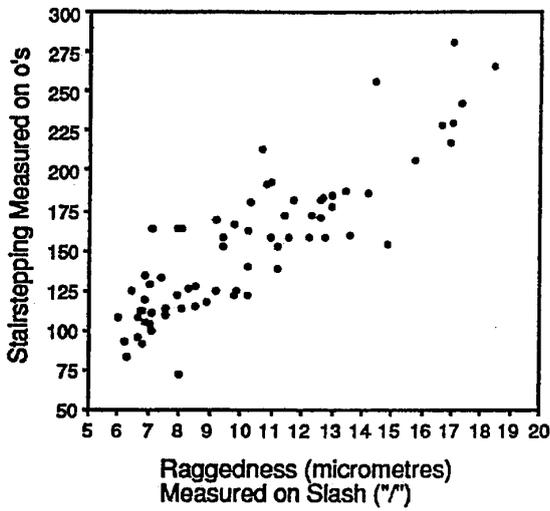
Figure 5. Stairstepping measured using o's correlates with
raggedness
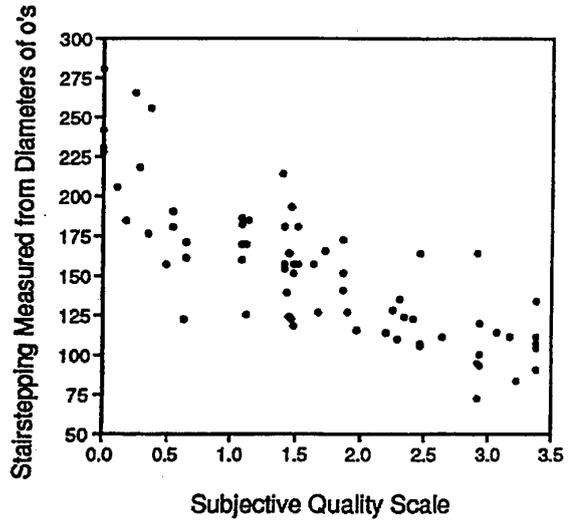
Figure 7. Measuring o's correlates with subjective impression
of stairstepping

Applicability of a standardized discrete cosine transform coding method to character images

Yasushi Hoshino
Dai Ando
Suguru Higashino
Nippon Telegraph and Telephone Corporation
NTT Human Interface Laboratories
Yokosuka, Kanagawa, Japan

Masaki Tsukamoto
Nagoya University
Nagoya, Aichi, Japan

Abstract. *The Joint Photographic Experts Group (JPEG) baseline system, which is scheduled to be standardized in 1992, is applied to character images, and the characteristics of the application are investigated. The JPEG system is suitable for continuous-tone images, however, continuous-tone images are usually accompanied by characters. The image quality of characters is investigated on various magnitudes of quantization tables and the deterioration mechanisms are discussed. A method of image quality improvement that is accomplished by density transformation after decoding is proposed and its effects are confirmed.*

1 Introduction

Recently, progress in LSI technologies has made practical applications of digital image processing possible. The amount of digitized image information is extremely large, so processes such as data compression are necessary for communication and storage. For bilevel images, several coding methods are standardized and facsimile communications are used conveniently and widely throughout the world. However, for continuous-tone images, a standardized coding method has not been determined.

The Joint Photographic Experts Group (JPEG) will standardize a discrete cosine transform (DCT)-based coding method as a baseline system for continuous-tone color and monochrome images in 1992. The DCT coding method is very effective for continuous-tone images, and high-quality compression characteristics have been obtained by the optimization of quantization table parameters.[1] Based on the

progress of Integrated Service Digital Network (ISDN) technology, the coding technologies,[2] continuous-tone printing technologies,[3] and various color image handling become easy and color image systems such as color facsimile are expected to begin to grow rapidly.

Usually, documents consist of characters, figures, tables, and images. Even if the main part of a document is an image, it is often accompanied by characters, for example, an explanation of the figure. A coding method for binary images has been used widely, but it is not fit for continuous-tone images. On the other hand, the DCT coding method is good for continuous-tone images, but its application characteristics to binary images such as characters is not well understood. Therefore, it is important to comprehend the applicability of the DCT coding method to character images.

This paper investigates the applicability of the DCT coding method to character images. First, the DCT coding method standardized by JPEG is reviewed briefly. The results of DCT coding application to character images are discussed and an improvement in the application is proposed.

2 Method of JPEG DCT Coding

Coding methods for continuous-tone images have been investigated by JPEG for international standards.[4–6] The JPEG algorithm is classified into three systems: the baseline system, the extended system, and the independent function. The baseline and extended systems are DCT irreversible coding, while the independent function is spatial differential pulse code modulation (DPCM) reversible coding. The baseline system is indispensable and fundamental, so we proceed with discussions of the baseline system in this paper. The coding process is shown in Fig. 1. DCT is carried
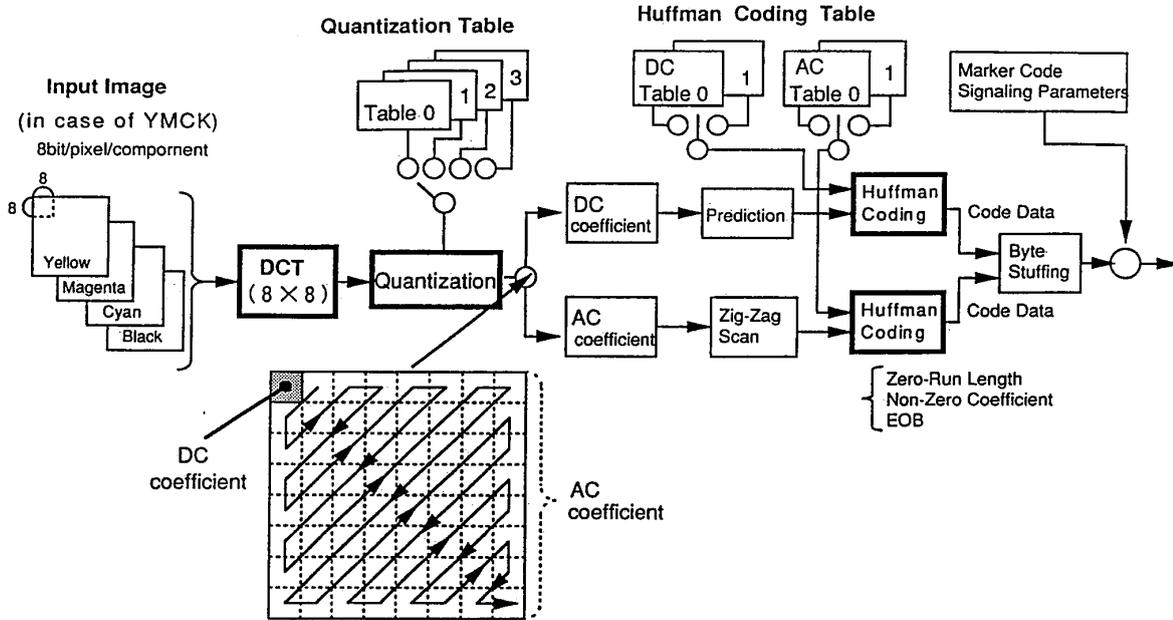
Fig. 1 Coding process by JPEG baseline system.

2	7	10	16	24	40	51	61
7	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Fig. 2 Example of quantization table.

out on respective 8 × 8 pixel matrices from digitized pixel data matrix. The DCT process is carried out on 8 × 8 pixel matrices $f(i,j)$ as follows:

$$F(u,v) = \frac{c(u)c(v)}{4} \sum_{j=0}^{7} \sum_{k=0}^{7} f(j,k)$$

$$\times \cos\left[\frac{(2j+1)u\pi}{16}\right] \cos\left[\frac{(2k+1)v\pi}{16}\right] ,$$

where

$$c(w) = \begin{cases} 1/\sqrt{2} & \text{for } w=0 \\ 1 & \text{for } w=1,2,3,4,5,6,7 . \end{cases}$$

The results of DCT are quantized by the quantization table as shown in Fig. 2. The quantization table is not the default table, instead we used the table suggested in Ref. 1. The quantized matrix is zig-zag scanned and Huffman coding is applied. Decoding is carried out by the reverse process of coding. The compression ratio is controlled by the magnitude of the quantization table components, and the adjustment is usually done by multiplying by a constant value, which is called the *scaling factor*. For the same compression ratio, the quality of the image can also be influenced by the value of each individual quantization table component.[1] When images are color, compared with monochrome images, higher compression can be accomplished by using color transformation and subsampling of color component techniques.

3 Application to Bilevel Images

DCT coding is applied to character and continuous-tone images, respectively. Figure 3 shows the results of DCT coding and decoding with various quantization tables. The signal-to-noise ratio (SNR) and compression ratio are arranged in Table 1. According to the increase in the scaling factor, compression became effective; on the other hand, image quality gradually deteriorates. In character images, at the bit rate of 1.32 bits/pixel as shown in Fig. 3(b), the deterioration is not perceived. At the rate of 0.79 to 0.40, as shown in Figs. 3(c), 3(d), and 3(e), noise arises around the character, and at the rate of 0.27, as shown in Fig. 3(f), characters are fairly blurred. In continuous-tone images, at the compression rate of 0.62 to 0.32, as shown in Figs. 3(b) and 3(c), the deterioration is not perceived; at the rate of 0.21, as shown in Fig. 3(d), edges are blurred slightly; at the rate of 0.17, as shown in Fig. 3(e), the matrix of the DCT unit is perceived; and at the rate of 0.13, as shown in Fig. 3(f), the matrix becomes an eyesore and false contours are perceived.

(a) (b) (c)

(d) (e) (f)

(a) (b) (c)

(d) (e) (f)

Fig. 3 Example of JPEG algorithm applications to character and continuous-tone images; magnifications of quantization table: (a) original, (b) scaling factor 1.0, (c) 2.7, (d) 6.0, (e) 10.0, and (f) 35.0.

Table 1 SNR and bit rate versus scaling factor

scaling factor	character image		continuous-tone image	
	SNR (dB)	bit rate (bits/pixel)	SNR (dB)	bit rate (bits/pixel)
1.0	30.6	1.32	34.4	0.62
2.7	27.3	0.79	32.2	0.32
6.0	24.6	0.53	30.5	0.21
10.0	22.7	0.40	29.5	0.17
35.0	20.3	0.27	27.3	0.13
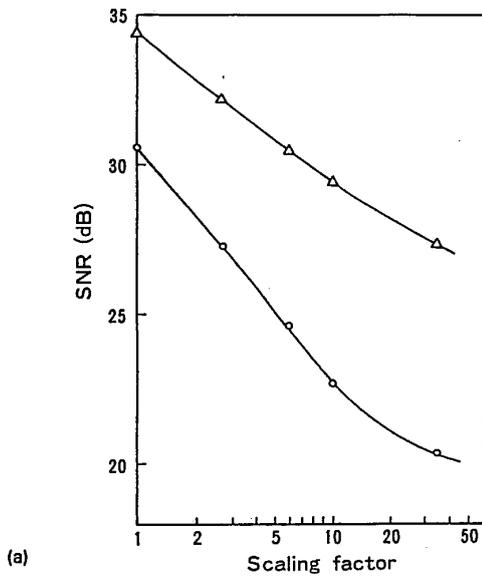
(a)
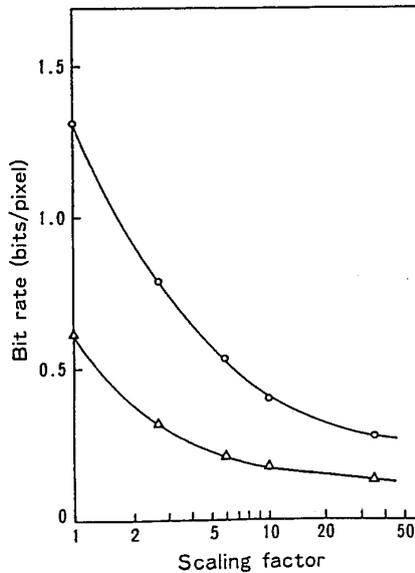
(b)

Fig. 4 (a) Relation of SNR versus scaling factor and (b) relation of bit rate versus scaling factor. △ = continuous-tone image; ○ = character image.
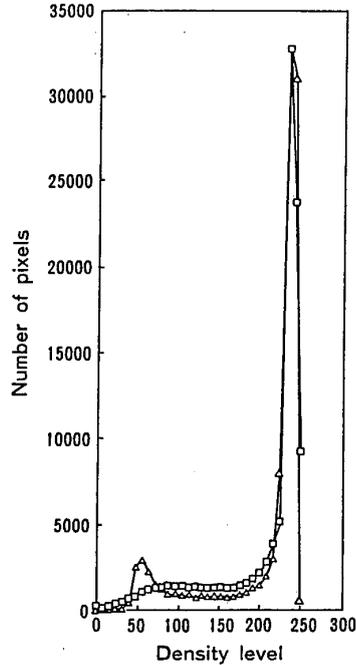
Fig. 5 Histogram of the density level of character images. △ = original; □ = after JPEG process.

Figure 4 shows the relationship of SNR and bit rate versus the scaling factor of the quantization table. As the scaling factor of the table increases, compression becomes effective, however, the SNR decreases. The compression ratio and SNR of character images are lower than that of continuous-tone images. The difference between these images is due to the difference in the amounts of high spatial frequency components. On the whole, character images have steep changes in density levels.

Figure 5 shows the histogram of the density levels of character images. There are two peaks in the histogram. The large peak represents the background and the small peak represents the character images. After coding and decoding, the histogram of the original becomes blunt. Changes from this processing are investigated on the idealized case: The right side of an 8×8 matrix is background and the left side of the matrix is image. Figure 6 shows the results of the changes. As the compression table increases, deviations from the original grow larger. The deviations in the background level are considered to deteriorate the image quality because the deviations near the character images are felt to be noisy.

4 Proposal of Improvement and Discussion

A trial to improve character image quality is carried out by means of a simple treatment after decoding. We assume that the region is recognized to be a character or continuous-tone image, because various recognition methods have been studied and can be utilized. It has been suggested that the deterioration in character images is mainly due to deviations
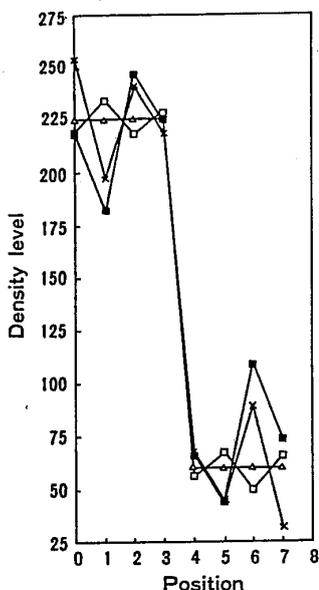
Fig. 6 Changes of edges in JPEG process. △ = original; □ = scaling factor 2.7; X = 10.0; ■ = 35.0.
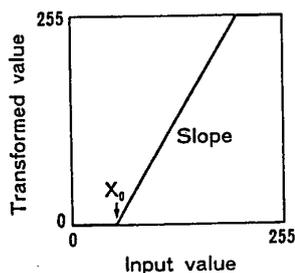
Fig. 7 Method of transformation characterized by x_0 (crossing point) and slope.

Fig. 8 Example of transformed images: slope, rising point; (a) 1.67, 28; (b) 1.67, 55; and (c) 1.67, 77.

near edges after the coding and decoding processing (Fig. 6). Therefore, we try to improve the quality by compressing the deviations.

The compressing method is done by transformation to high contrast. The method of transformation is shown in Fig. 7. Both of the slopes and starting points are varied widely with many combinations in Fig. 7, and the transformations on the decoded character images are carried out. Some transformed examples are shown in Fig. 8. Seven professional experts on image quality have evaluated the samples and selected the best one. The sample selected as the best by most persons is Fig. 8(b) and the transformation condition is the slope 1.67 and the crossing point of the *x* axis 51. Compared with the untreated sample of Fig. 3(e), we can see that the image quality of the sample is improved by this treatment.

Modified Huffman (MH) coding, which is used in G3 fax for bilevel images, is applied to the same image of the preceding discussion. The result of MH coding and the

decoding process is shown in Fig. 9. The quality is near that of Fig. 8(b). The bit rate is 0.23 bits/pixel, whose value is about ½ of the DCT coding, 0.4 bits/pixel. The difference is considered to arise from the reason that the DCT is intended for continuous-tone images and one dot needs several bits. However, MH coding cannot be applied to continuous-tone images; if applied, image quality is extremely deteriorated. The characteristics of DCT coding are very suitable to continuous-tone images and the bit rate is controllable by the magnitude of the quantization table.

When the document is a mixture of continuous-tone and character images, two ways of processing can be considered.

つて開催される主管庁会議と、

この意見は、また、電信規即

まま世界の国際通信の活動方向

TTの活動は、つねに時代の

TTの料金は、どのようにす

および大陸間通話を半自動化し

ルで直面する問題について、具体

が直面する問題について、

とができない場合が多い。この

Fig. 9 Result of MH process.

One way is that two different processes can be applied after the documents have been discriminated into continuous-tone and character images at the first step; another way is to apply single processing of DCT coding to the images. The former way is efficient in terms of the amount of data, but the processing is complex. The latter way is simple in terms of processing, but the data compression process is not as efficient in some cases. Therefore, DCT coding is a possibility that can be widely used in the case of mixed documents because of processing simplicity.

5 Summary

The DCT coding method, which is standardized for continuous-tone images by the JPEG, is applied to character images and its applicability is investigated. The relationship of the scaling factor of the quantization table versus SNR and bit rate is examined, and the mechanisms of image quality deterioration are comprehended. A method of image quality improvement by density transformation after decoding is proposed and its effects are confirmed.

References

1. S. Higashino, K. Yanaka, and Y. Hoshino, "A quantization table design based on spectral analysis for DCT full color image coding," in IS&T's 44th Annual Conf. Advance Printing of Paper Summaries, pp. 484–487 (1991).
2. D. Ando, H. Higashino, K. Yanaka, and Y. Hoshino, "Optimization of standardized DCT coding method for full-color high-resolution still images," presented at IS&T's 7th International Congress on Advances in Non-Impact Printing Technologies, Portland, Oregon, October 1991.
3. M. Omodani, T. Tanaka, Y. Hoshino, and M. Ohta, "Continuous-tone photographic image reproduction using ion flow printing process," *J. Imaging Technol.* 16, 153–157 (1990).
4. W. B. Pennebaker, "JPEG Technical Specification," ISO/IEC JTC1/SC2/WG8 JPEG-8-R8 (1990).
5. G. K. Wallace, "The JPEG still picture compression standard," *Comm. ACM* 34, 31–45 (1991).
6. M. Rabbani, and P. W. Jones, *Digital Image Compression Techniques,* Vol. TT7, pp. 113ff, SPIE Press, Bellingham, Washington (1991).

Yasushi Hoshino received his BS, MS, and Dr degrees from Toyko University in 1970, 1972, and 1984, respectively. He joined Electrical Communications Laboratories, NTT, in 1972 and engaged in the research and development of nonimpact printing technologies. He developed a high-speed electrophotographic laser printer and the most compact electrophotographic printer by using a light-emitting diode array. He also developed a color electrophotographic printer that uses elliptical laser beam scanning. He is now studying a novel electrophotographic printing technology and high-quality color printing by means of ionographic technology. His research interests are fundamentals and applications of electrophotographic technologies, electrostatics, charge retention phenomena, charge carrier transport mechanisms in resistive semiconductors, and image processing technologies. He has received international awards from the United States and China.

Dai Ando received the BE degree from Tohoku University, Sendai, Japan, in 1989. Since 1989, he has engaged in research and development of digital image coding at the Human Interface Laboratories, NTT, Japan.
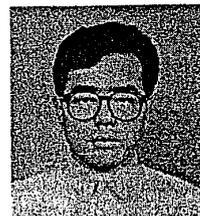
Suguru Higashino received the BE from Science University of Tokyo in 1986, and the ME from Tokyo Institute of Technology in 1988. In 1988 he joined Human Interface Laboratories, NTT, where he has been engaged in research of still-picture coding. Since 1991, he has been engaged in development of a picture filing system in the Integrated Communications Systems Sector of NTT.

Masaki Tsukamoto received his BS degree from Nagoya University in 1992. He is now studying chemistry in a graduate course at Nagoya University.

HUAWEI EX. 1116 - 116/714

Human visual system weighted progressive image transmission using lapped orthogonal transform/classified vector quantization

Chansik Hwang
Kyungpook National University
Department of Electronics
Taegu, Korea 702-701

Suresh Venkatraman
Array Microsystems
1420 Quail Lake Loop
Colorado Springs, Colorado 80906

K. R. Rao
University of Texas at Arlington
Electrical Engineering Department
Box 19016
Arlington, Texas 76019

Abstract. A progressive image transmission (PIT) scheme based on the classified transform vector quantization (CVQ) technique using the lapped orthogonal transform (LOT) and human visual system (HVS) weighting is proposed. Conventional block transform coding of images using the discrete cosine transform (DCT) produces, in general, undesirable blocking artifacts at low bit rates. Here image blocks are transformed using the LOT and classified into four classes based on their structural properties and further subdivided adaptively into subvectors depending on the LOT coefficient statistics with HVS weighting to improve the reconstructed image quality by adaptive bit allocation. The subvectors are vector quantized and transmitted progressively. Coding tests using computer simulations show that the LOT/CVQ-based PIT of images is an effective coding scheme. The results are also compared with those obtained using PIT/DCTVQ. The LOT/CVQ-based PIT reduces the blocking artifact significantly.

Subject terms: visual communication; transform image coding; vector quantization; human visual sensitivity.

Optical Engineering 32(7), 1524–1530 (July 1993).

1 Introduction

An effective image data compression technique to reduce the bit rate for transmission or data storage while maintaining an acceptable image quality is essential for such applications as video teleconferencing, high-definition television (HDTV) transmission, facsimile transmission, etc. Numerous bandwidth compression techniques have been developed for this purpose, such as differential pulse code modulation (DPCM), transform coding[1] (TC), hybrid coding, and adaptive versions of these techniques with new image-processing methods.[2] In these methods, block TC is used to convert statistically dependent or correlated pixels into independent or uncorrelated coefficients. Among several block transforms, it is known[3] that the discrete cosine transform (DCT) approaches the statistically optimal transform, the Karhunen-Loeve transform (KLT), for highly correlated images. Vector quantization (VQ) of coefficients[4-6] helps reach the rate distortion bound of the source. Various techniques that apply VQ to DCT

coefficients of still frame images or interframe prediction errors of motion video have been investigated. These techniques, however, can result in block artifacts at low bit rates. Several techniques have been developed to reduce or eliminate blocking artifacts. One effective technique[7-9] is to apply the lapped orthogonal transform (LOT), whose basis functions extend beyond the traditional image block boundaries. Besides reducing the block structure, LOT also has some interesting filtering properties. Also, LOT is a real transform having a fast algorithm.[8,9] Being a separable transform, extension to multiple dimensions is straightforward. In this paper, an image compression scheme for progressive image transmission (PIT) using LOT/VQ is proposed to take advantage of these properties. Classification[10-12] in the LOT domain is also introduced to adapt to the directional and structural properties within the blocks in the spatial domain. Each of the classified 8×8 blocks in the LOT domain is adaptively partitioned into a number of subvectors of smaller dimensions according to their variances and human visual system (HVS) weighting.[13-15] The subvectors for each class are progressively transmitted in several stages until the final required image quality is achieved. This technique allows an approximate image to be built quickly and the details to be transmitted progressively through several passes.[16] This is useful in interactive visual communications, where a viewer

can decide to build up an image with high quality based on a crude version or pass on to another image in a large database of images stored in compressed form.

2 PIT Using LOT/CVQ

2.1 Classified VQ Based on LOT (LOT/CVQ)

A classification scheme is used in the LOT domain to classify blocks into perceptually distinct classes according to the directional activity in them. Blocks from different activity classes are then adaptively partitioned into a number of subvectors of small dimensions according to their variance distribution and HVS for low-frequency components. The coding scheme is depicted in Fig. 1.

Although several activity measures have been proposed for transform block classification,[17] the relationship between directional activity in the spatial domain and that in the transform domain is exploited simply in the scheme used here. The blocks transformed using the LOT are classified into four classes: one low-activity class and three activity classes corresponding to the edge orientation in the spatial domain—horizontal, vertical, and diagonal. The blocks are classified based on three activity indices calculated using the coefficients in the region shown in Fig. 2. These regions correspond to horizontal, vertical, and diagonal edges or structure in a block in the spatial domain. If the indices are below a certain threshold, the block is classified as a low-activity block.

Because the variances of the transform coefficients vary widely, it would be inefficient to use the same quantizer for all the coefficients. Hence, for each class, the 8×8 transform blocks are partitioned into a number of subvectors of small dimensions according to their variances. In our scheme, the blocks belonging to the high-activity classes are partitioned into seven subvectors, and those of the low-activity class are partitioned into three subvectors. The number of subvectors for the activity classes have been chosen so as to include as many high-frequency coefficients as possible. Because the energy in the high-frequency coefficients is small for the low-activity class, only three subvectors containing the lower frequency coefficients have been chosen. Variance distributions for the four classes and one possible corresponding subvector construction based on blocks from four test images are shown in Figs. 3 and 4, respectively.

In Fig. 4, blank squares imply that the corresponding LOT coefficients are discarded. As shown in Figs. 3 and 4, the selection of LOT coefficients to construct subvectors is not straightforward, especially among high-frequency components. Even if a coefficient has high variance, its significance to the human visual system may be small. Hence, an improvement in the reconstructed image quality based on subjective criteria can be expected by partitioning the subvectors adaptively based on the significance of transform coefficients to the human visual system (HVS). Several schemes for the modulation transfer function (MTF) of the HVS model have been proposed.[13-15] Chitprasert and Rao[14] proposed an MTF for use with DCT that yields good results for PIT. The HVS weighting matrix is shown in Fig. 5. The new variance distribution after HVS weighting (Fig. 5) and the corresponding subvector construction are shown in Fig. 6.

One of the advantages of partitioning the blocks into subvectors is that bits can be assigned efficiently. Vectors with large variances can be assigned more bits and those with
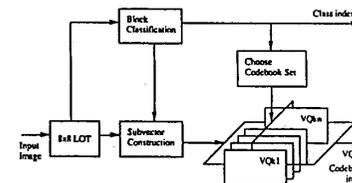

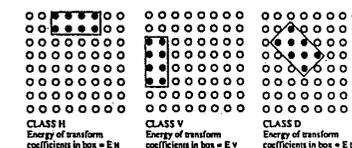**Fig. 1** LOT/CVQ coding scheme.

Fig. 2 Regions of reference for classification of blocks into different activity classes.

smaller variances can be assigned fewer bits. Table 1 shows a bit assignment for the subvector construction based on Fig. 7 for an average bit rate of 0.7 bit/pixel.

In Table 1, k is the subvector dimension and bits indicate the codevector size, i.e., the code book size $2^{\# \text{ of bits}}$. The dc coefficient for all classes is uniformly quantized to 8 bits. Code books are generated for each subvector from a training sequence that is sufficiently long and contains most of the features found in naturally generated images.

2.2 Progressive Image Transmission

Progressive image transmission is receiving attention for application in interactive image communication over restricted data rate channels.[11,16] In progressive image transmission, the least information necessary to represent each block is transmitted quickly with as few bits as possible. On the receiver's request, the image can be progressively improved with further transmission in several stages until the required quality is achieved. Because our LOT/CVQ scheme contains a hierarchical multistage structure, it allows progressive image transmission. The coefficients corresponding to the dc value of each class are scalar quantized and transmitted first. Each successive stage of subvectors of ac coefficients is utilized in reconstructing the successive approximation.

3 Simulation Results

We have applied the PIT using LOT/CVQ to encode monochrome images of 512×512 pixels with 256 gray levels. The original image is first divided into 8×8 blocks and transformed using the LOT. These blocks are classified using the scheme described earlier and partitioned into subvectors. The class information for each block is included as an overhead in the coded bitstream. This overhead information indicates to the decoder which code book set to use and also in what

CLASS H

213867	22890	7299	3155	1522	469	659	185
17071	3259	3497	1213	1232	311	635	123
2248	371	656	343	380	171	161	81
1133	361	366	204	198	160	108	50
848	156	222	99	119	67	71	35
886	105	173	73	97	42	61	31
540	64	109	47	66	37	41	20
557	65	113	40	60	25	36	15

CLASS V

158524	4277	1733	565	634	147	383	69
21787	2983	1587	423	603	120	365	67
7966	2068	902	331	291	114	151	51
4711	1494	627	296	193	109	86	48
2434	1036	579	247	166	85	74	45
1824	719	425	230	141	82	72	41
1123	458	349	179	134	79	72	37
1318	381	303	152	122	62	51	31

CLASS D

160966	9269	1072	857	384	263	163	129
9101	6699	2926	753	416	238	137	123
1147	3104	2982	1471	584	308	149	120
932	924	1660	1176	568	301	156	101
428	546	661	744	382	220	114	81
347	319	375	353	258	178	115	85
228	245	244	203	170	132	86	67
203	215	192	153	130	124	80	56

CLASS L

291441	1079	117	101	33	33	16	16
563	250	82	53	29	19	13	11
84	77	56	37	26	19	12	10
81	53	43	31	23	63	19	11
45	37	30	23	18	18	11	9
36	26	24	18	16	17	13	10
26	21	18	14	13	15	13	9
23	18	14	12	11	11	8	6

Fig. 3 Variance distribution of the four classes based on four test images in the LOT domain.
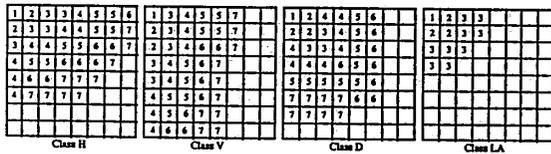
Fig. 4 Subvector configuration corresponding to variances shown in Fig. 3. Coefficients with same digits belong to the same subvector.

0.494	1.000	0.702	0.381	0.186	0.085	0.037	0.016
1.000	0.455	0.308	0.171	0.083	0.039	0.017	0.008
0.702	0.308	0.241	0.124	0.065	0.031	0.014	0.006
0.381	0.171	0.124	0.077	0.042	0.022	0.010	0.005
0.186	0.083	0.065	0.042	0.025	0.013	0.007	0.003
0.085	0.039	0.031	0.022	0.013	0.008	0.004	0.002
0.037	0.017	0.014	0.010	0.007	0.004	0.002	0.001
0.016	0.008	0.006	0.005	0.003	0.002	0.001	0.001

Fig. 5 HVS weighting matrix.[14]

Table 1 Bit allocation for subvectors of each class for a bit rate of 0.7 bit/pixel.

Sub Vector	Class H		Class V		Class D		Class LA	
	k	bits	k	bits	k	bits	k	bits
1	1	8	1	8	1	8	1	8
2	2	10	1	8	2	10	2	10
3	2	8	2	8	3	9	7	8
4	3	8	3	8	5	9		
5	6	9	4	8	6	8		
6	6	8	5	8	9	9		
7	7	8	10	9	9	8		

way to combine the subvectors to reconstruct an 8×8 block. For classification into four classes, the overhead information is 4096 log$_2$4 = 8192 bits. This corresponds to an overhead bit rate of 0.331 bit/pixel. Code books are generated by a Linde-Buzo-Gray (LBG) algorithm using four different training images: "baboon," "boat," "lady-flqwer," and "Martha." The bit allocation for each of the subvectors and the total number of bits depend on the target output bit rate. A similar coding scheme using the DCT was introduced by Nam and Rao.[12] This was extended to LOT/CVQ in Ref. 18.

The coding of images was done at a target average bit rate of 0.7 bit/pixel (including overhead information) using DCT/CVQ and LOT/CVQ, and the results of the two simulations were compared at each progressive stage. Figure 7 shows the original "Lena" image and Fig. 8 shows the progressive approximations at each stage and the error images at 0.5 bit/pixel (stage 3) for comparison between the two schemes. It can be clearly seen that the block artifacts are significantly reduced and the subjective quality of the reconstructed image is superior in the case of LOT/HVS/CVQ scheme, especially at low bit rates. Normalized SNR is used for evaluation of subjective performance:

$$SNR = 10 \log_{10} \frac{\sum_i \sum_j x_{ij}^2}{\sum_i \sum_j (x_{ij} - \hat{x}_{k,ij})^2} \text{ dB} \quad k = 0, 1, \dots,$$

where x_{ij} and $\hat{x}_{k,ij}$ represent the (i,j)'th element of the original image and the k'th approximation, respectively. The tabulated results (Table 2) show that the LOT/HVS/CVQ exhibits an improvement at each progressive stage over DCT/HVS/CVQ.

Figure 9 shows reconstructed and error images at 0.5 bit/pixel (stage 3) without applying HVS weighting in subvector construction. Table 2 shows that the SNR for LOT/HVS/CVQ is lower than that for LOT/CVQ, but the subjective image quality is better, as can be seen from the images in Figs. 8 and 9, because the sharp discontinuous edges that result from compression are eliminated by HVS weighting.

CLASS H

CLASS D

CLASS V

CLASS L

Fig. 6 HVS weighted variance distribution and proposed subvector configuration.

Table 2 Bit rates and SNR values for DCT/HVS/CVQ, LOT/HVS/CVQ, and LOT/CVQ at each progressive stage.

k	Bit rate (bpp)	SNR (dB)		
		DCT/HVS/CVQ	LOT/HVS/CVQ	LOT/CVQ
1	0.16	17.9	18.7	18.7
2	0.31	21.0	22.1	22.1
3	0.43	23.0	23.8	23.8
4	0.50	24.8	25.8	25.8
5	0.57	25.5	26.6	26.6
6	0.63	25.7	26.8	26.8
7	0.70	25.8	26.8	26.8

Fig. 7 Original "Lena" image.

4 Conclusions

A progressive image transmission scheme using a classified transform coding using LOT and VQ (LOT/CVQ) is developed and compared with the DCT/CVQ scheme. For VQ, subvector partitioning was performed adaptively based on the significance of transform coefficients to HVS. The simulation results, both subjective and objective, show that LOT/

CVQ reduces block artifacts, and additionally LOT/HVS/CVQ results in subjectively improved reconstructed images. Application of HVS to the transform coefficients resulted in an efficient subvector construction. Also more bits were allocated to coefficients with larger HVS weights. As seen from the images in Fig. 9 and from Table 2, the reconstructed approximations rapidly converge to a good quality both subjectively and objectively. This scheme is particularly well suited for interactive image communications over low-bandwidth channels. An artifact that is a result of the LOT can been seen in images coded for very low bit rates in the form of ringing. This is not perceivable at bit rates of 0.5 bit/pixel and higher. This artifact is much less annoying than the blocking artifacts of DCT coding.

Fig. 8 Continued.

Fig. 8 Progressive stages and error images at 0.7 bit/pixel: (a) the zeroth stage using DCT/HVS/CVQ at 0.16 bit/pixel, (b) the zeroth stage using LOT/HVS/CVQ at 0.16 bit/pixel, (c) the first stage using DCT/HVS/CVQ at 0.31 bit/pixel, (d) the first stage using LOT/HVS/CVQ at 0.31 bit/pixel, (e) the second stage using DCT/HVS/CVQ at 0.43 bit/pixel, (f) the second stage using LOT/HVS/CVQ at 0.43 bit/pixel, (g) the third stage using DCT/HVS/CVQ at 0.50 bit/pixel, (h) the third stage using LOT/HVS/CVQ at 0.50 bit/pixel, (i) the error image (amplified by 5) of DCT/HVS/CVQ at 0.50 bit/pixel, (j) the error image (amplified by 5) of LOT/HVS/CVQ at 0.50 bit/pixel, (k) the sixth stage using DCT/HVS/CVQ at 0.70 bit/pixel, and (l) the sixth stage using LOT/HVS/CVQ at 0.70 bit/pixel.

References

1. P. A. Wintz, "Transform coding," Proc. IEEE 60, 809–823 (July 1972).
2. A. K. Jain, "Image data compression: a review," Proc. IEEE 69, 349–389 (Mar. 1981).
3. K. R. Rao and P. Yip, Discrete Cosine Transform: Algorithms, Advantages and Applications, Academic Press, San Diego (1990).
4. Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," IEEE Trans. Commun. 28, 84–95 (Jan. 1984).
5. A. Gersho, "On the structure of vector quantizer," IEEE Trans. Inf. Theory IT-28, 157–166 (Mar. 1982).
6. R. M. Gray, "Vector quantization," IEEE Acoust. Speech Signal Process. Magazine 1, 4–29 (Apr. 1984).
7. P. M. Cassereau, D. H. Staelin, and G. De Jager, "Encoding of images based on a lapped orthogonal transform," IEEE Trans. Commun. 37, 189–193 (Feb. 1989).
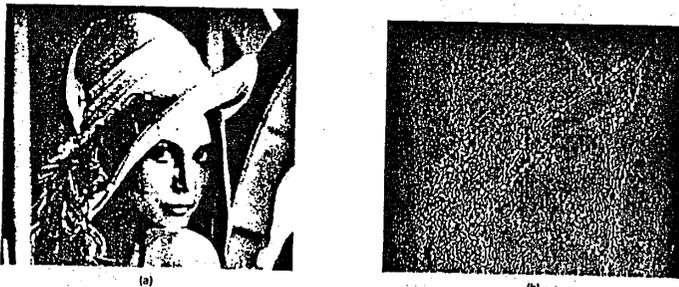
(a)   (b)

Fig. 9 Reconstructed and error images at 0.5 bit/pixel (stage 3) without HVS weighting in subvector configuration (a) the third stage using LOT/CVQ at 0.5 bit/pixel (b) the error image (amplified by 5) at 0.5 bit/pixel.

8. H. S. Malvar and D. H. Staelin, "The LOT: transform coding without blocking effects," *IEEE Trans. Acoust. Speech Signal Process.* 37, 553–559 (Apr. 1989).
9. H. S. Malvar, "Lapped transforms for efficient transform/subband coding," *IEEE Trans. Acoust. Speech Signal Process.* 38, 969–978 (June 1990).
10. B. Ramamurthy and A. Gersho, "Classified vector quantization of images," *IEEE Trans. Commun.* 34, 1105–1115 (Nov. 1986).
11. Y. S. Ho and A. Gersho, "Classified transform coding of images using vector quantization," *Proc. ICASSP' 89*, pp. 1890–1893, Glasgow, Scotland (May 1989).
12. J. Y. Nam and K. R. Rao, "Image coding based on two-channel conjugate VQ," *J. Vis. Commun. Image Process.* 2, 289–298 (Sep. 1991).
13. N. B. Nill, "A visual model weighted cosine transform for image compression and quality assessment," *IEEE Trans. Commun.* COM-33, 551–557 (June 1985).
14. B. Chitprasert and K. R. Rao, "Human visual weighted progressive image transmission," *IEEE Trans. Commun.* COM-38, 1040–1044 (July 1990).
15. K. N. Ngan, K. S. Leong, and H. Singh, "Adaptive discrete cosine transform coding in perceptual domain," *IEEE Trans. Acoust. Speech Signal Process.* 37, 1743–1750 (Nov. 1989).
16. L. Wang and M. Goldberg, "Progressive image transmission by transform coefficient residual error quantization," *IEEE Trans. Commun.* COM-36, 75–87 (Jan. 1988).
17. M. Breeuwer, "Transform coding of images using directionally adaptive VQ," in *Proc. ICASSP 88*, pp. 788–791, New York (Apr. 1988).
18. S. Venkatraman, J. Y. Nam, and K. R. Rao, "Image coding based on classified lapped orthogonal transform—vector quantization," *Proc. SPIE* 1818, 500–511 (1992).

Chanaik Hwang received the MS degree in electrical engineering from the Korea Advanced Institute of Science and Technology, Seoul, Korea, in 1979. Since 1979 he has been with the Department of Electronics, Kyungpook National University, Taegu, Korea, where he is currently an associate professor. He was visiting professor at the University of Texas, Arlington, in 1991 for one year. His current research interests are image data compression and high-definition television.

Suresh Venkatraman received the BE degree in electronics and communications engineering from University Visvesvaraya College of Engineering, Bangalore University, India, in 1990 and the MS degree in electrical engineering from the University of Texas at Arlington (UTA) in 1992. He was a research assistant at UTA from 1991 to 1992, where his research interests included DSP applications, discrete transforms, video signal processing and compression, and high-resolution video coding. He is currently with Array Microsystems, Colorado Springs, Colorado, where his current areas of work include video signal processing and video compression.

K. R. Rao received the PhD degree in electrical engineering from the University of New Mexico, Albuquerque, in 1966. Since 1966, he has been with the University of Texas at Arlington (UTA) where he is currently a professor of electrical engineering. He, along with two other researchers, introduced the discrete cosine transform in 1974, which has since become very popular in digital signal processing. He has coauthored the books *Orthogonal Transforms for Digital Signal Processing* (Springer-Verlag, 1975) (translated into Chinese and Russian) and *Fast Transforms: Analyses and Applications* (Academic Press, 1982). He has edited a benchmark volume, *Discrete Transforms and Their Applications* (Van Nostrand Reinhold, 1985). He is coauthor of a chapter entitled "Discrete Transforms" for a handbook on *Digital Signal Processing*, edited by D. F. Elliott (Academic Press, 1987). He is coauthor of the book *Discrete Cosine Transform*, published by Academic Press (1990), which was translated into Japanese (June 1992) by Ohmsha publishers. He has coedited a benchmark volume *Teleconferencing* (Van Nostrand Reinhold, 1985). He has published extensively in the areas of digital image coding and discrete transforms and has conducted workshops worldwide on emerging international standards for video codecs.

Image sequence correspondence via a Hopfield neural network

Jyh-Yeong Chang
National Chiao Tung University
Department of Control Engineering
1001 Ta-Hsueh Road
Hsinchu 300, Taiwan

Shin-Wen Lee
Industrial Technology Research Institute
Advanced Technology Center
Computer and Communication Research Laboratory
195 Chung-Hsing Road Section 4
Chutung 310, Taiwan

Mong-Fong Horng
National Chiao Tung University
Department of Control Engineering
1001 Ta-Hsueh Road
Hsinchu 300, Taiwan

Abstract. A neural network approach to finding trajectories of feature points in a monocular image sequence is proposed. In conventional methods, this problem is formulated as an optimization problem and solved using heuristic algorithms. The problem usually involves lengthy computations, making it computationally difficult. We apply the Hopfield neural network to image sequence correspondence. The design and development of the Lyapunov function for this problem are discussed in detail. Furthermore, the neural-network-based image correspondence scheme is extended to the case of successive image frames, in which some feature points are allowed to be occluded. Examples and simulation results are presented to illustrate the design process and the convergence characteristics of the proposed neural network. By using the massive parallel-processing power of neural networks, a real-time and accurate solution can be obtained.

Subject terms: visual communication; image sequence correspondence; Hopfield neural model; image tracking; neural network structure.

Optical Engineering 32(7), 1531–1538 (July 1993).

1 Introduction

Finding trajectories of moving objects in a monocular image sequence is a vital technique in the field of motion analysis. A major step in finding such trajectories is to identify images of the same physical point in a sequence of frames; this step is usually called the correspondence problem. This technique, which successively refines the structure of the object as more frames are acquired, frees us from assumptions of rigidity and relies on natural assumptions about motion characteristics. Many correspondence algorithms[1-4] have been introduced in the past few years. Most research for establishing correspondence[2,3] uses only two frames of a sequence to solve this problem. However, two-frame algorithms require assumptions about the nature of the objects. Recently, a correspondence algorithm based on a sequence of frames was proposed by Sethi and Jain.[1] In their method, smoothness of motion is used to relax the need for assumptions about the rigidity of the object. The path coherence function is used to formulate the correspondence problem as an optimization problem. For instance, if we are given a sequence of m frames that have n feature points each, then basically there will be $C(n^m,n)$ solutions of combinations, i.e., n trajectories from

n^m possible trajectories. After comparing measurements of motion smoothness and velocity continuity calculated from the defined coherence functions for each solution, optimal n trajectories can be obtained. However, a large amount of computation time is usually necessary to identify the optimal solution with n true paths from among all legal solutions. To expand the computation power applied to this problem, we shall introduce a Hopfield neural network algorithm[5,6] for finding trajectories of feature points in a monocular image sequence. The parallel nature of the neural network matches nicely the high-speed computation requirements of the image correspondence problem.

The Hopfield neural network has been used in a wide range of applications. Most engineering optimization problems, including image processing algorithms, can be solved using the minimum energy searching property of the Hopfield model.[7-12] When the neural network is used to solve an optimization problem, the problem is usually formulated as minimization of a constrained cost function, where all the constraints on the solution can be explicitly incorporated into the cost function. For example, an image restoration technique[10] has been developed where the cost function is minimized by the Hopfield network. Using the Hopfield network to match the subgraph features for object recognition[11] has also been discussed. Recently, the use of a Hopfield network for the feature point matching of left and right images[12] has also been reported. Note that this stereo vision

INTERNATIONAL TELECOMMUNICATION UNION

TELECOMMUNICATION
STANDARDIZATION SECTOR

STUDY PERIOD 1993 - 1996

Question: 5/8

STUDY GROUP 8 - CONTRIBUTION 43

Source: Q.5/8 Rapporteur

Title: Amendments to ITU-T Rec. T.30 for enabling continuous-tone colour and gray-scale modes for Group 3.

Abstract

The proposed draft Annex E to Rec. T.30 enables the transfer of continuous-tone colour and gray-scale images as an option to Group 3 facsimile. In this Annex, the protocol elements necessary to send continuous-tone image data coded in accordance with Rec. T.81, Digital compression and coding of continuous-tone still images, are added to Rec. T.30. In addition, amendments to the main body of Rec. T.30 that refer to this Annex are made.

* Contact person - Name:Mr. Daniel T. Lee, Hewlett Packard, USA   Tel.:+1 415 857 3946

REFERENCE TEXT IS ITU-T Rec. T.30 (1993) UNLESS STATED OTHERWISE

Proposed Amendments to Recommendation T.30:

1. Table 2/T.30 and Table C.1/T.30 are amended to include the following entries:

Bit No.	DIS/DTC	DCS
68	JPEG coding	JPEG coding
69	Full colour mode	Full colour mode
70	Always set to zero	Default Huffman tables
71	12 bits/pel/component	12 bits/pel/component
72	Extend field	Extend field
73	No subsampling (1:1:1)	No subsampling (1:1:1)
74	Custom illuminant	Custom illuminant
75	Custom gamut range	Custom gamut range
76	Reserved for future use	Reserved for future use
77	Reserved for future use	Reserved for future use
78	Reserved for future use	Reserved for future use
79	Reserved for future use	Reserved for future use
80	Extend field	Extend field

2: Append the following to "Notes to Table 2/T.30":

22   The optional continuous-tone colour mode and gray-scale mode protocols are described in Annex E to T.30. If bit 68 in the DIS/DTC frame is set to one, indicating JPEG mode capability, then bit 15 and bit 27 in the DIS/DTC frame are also set to one. Bit 15 indicates 200x200 pels/25.4 mm resolution capability, which is basic for colour facsimile. Bit 27 indicates error correction mode capability, which is mandatory for colour facsimile. Bits 69 to 75 are relevant only if bit 68 is set to 1 (JPEG mode). See definitions of parameters in Annex E paragraphs 5.1.1-5.1.7.

3. Append the following Annex E to T.30:

Annex E

Procedure for the G3 document facsimile transmission of

continuous-tone colour images

(This annex forms an integral part of this Recommendation)

E.1 Introduction

This Annex describes the additions to ITU-T Rec. T.30 to enable the transmission of continuous-tone (multi-level) colour and gray-scale images for Group 3 facsimile mode of operation.

The objective is to enable the efficient transmission of high quality, full colour and gray-scale images over the general switched telephone network and other networks. The images are normally obtained by scanning the original sources with scanners of 200 pels/25.4 mm or higher, and bit depths of eight bits per picture element per colour component or higher. The original sources are typically colour or gray-scale photographs or hard copies from high quality printing systems.

The method specified here performs well on full colour images, but for transmission of multi-colour images such as business graphics, other methods may be more efficient. Two such methods would be the transmission of images using ITU-T Rec. T.434, Binary File Transfer, and ITU-T Rec. T.82, (JBIG encoding). This Annex does not address the encoding of multi-colour images. This topic is left for further study.

The encoding methodology for continuous-tone (multi-level) images is based on the JPEG (ITU-T Rec. T.81 | ISO/IEC 10918-1) image encoding standard. The JPEG image coding method includes both a lossy mode and a lossless mode of encoding. This Annex adopts the lossy mode of encoding which is based on the Discrete Cosine Transform.

The representation of colour image data is based on ITU-T Rec. T.42. It adopts a device-independent colour space representation, the CIELAB space, that allows unambiguous exchange of colour information.

This Annex explains the procedure for negotiation of the capabilities for transmission of continuous-tone colour and gray-scale images. It specifies the definitions and the specifications of new entries to the Facsimile Information Field of the DIS/DTC and DCS frames of Rec. T.30.

Information is specified pertaining to image digitisation resolution (in bits/pel), sampling ratio of colour components, JPEG capability, colour capability, and image data scaling that is subject to negotiation in the pre-message phase of the Rec. T.30 protocol.

This Annex does not address the semantics and syntax of the actual encoding of the continuous-tone colour and gray-scale images. That information is included in Annex I to Rec. T.4. —

The use of error correction mode (ECM) for error free transmission is mandatory in the procedure described by this Annex. Under the error correction mode of transmission, the JPEG encoded image data are embedded in the Facsimile Coded Data (FCD) part of the HDLC (High Level Data Link Control) transmission frames specified by Annex A of Rec. T.30.

The technical features of encoding and decoding the continuous-tone colour and gray-scale image data are described in Annex I to Rec. T.4. It describes two modes of image encoding (lossy gray-scale and lossy colour) which are defined using Rec. T.81.

E.2   Definitions

CIELAB	CIE 1976 (L* a* b*) space. A colour space defined by the CIE (Commission Internationale de l'Éclairage), having approximately equal visually perceptible difference between equally spaced points throughout the space. The three components are L*, or Lightness, and a* and b* in chrominance.
JPEG	Joint Photographic Experts Group, and also shorthand for the encoding method, described in Rec. T.81, which was defined by this group.
Baseline JPEG	A particular eight-bit sequential Discrete Cosine Transform (DCT) - based encoding and decoding process specified in Rec. T.81.
Quantisation table	A set of 64 values used to quantise the DCT coefficients in baseline JPEG.
Huffman table	A set of variable length codes required in a Huffman encoder and a Huffman decoder.

E.3   Normative references

-    ITU-T Rec. T.81 | ISO/IEC 10918-1, Information Technology -- Digital compression and coding of continuous-tone still images, Part 1: Requirements and guidelines. (Commonly referred to as JPEG standard).

-    ITU-T Rec. T.42, Continuous-tone colour representation method for facsimile.

-    ITU-T Rec. T.4, Standardisation of Group 3 facsimile apparatus for document transmission.

E.4   Negotiation procedure

The negotiation to transmit and receive JPEG encoded continuous-tone colour and gray-scale images under the Group 3 facsimile protocol is invoked through the setting of the bits in the DIS/DTC and DCS frames during the pre-message procedure (Phase B) of the Rec. T.30 protocol.

The first capability to be established between the calling unit and the called unit is to indicate whether JPEG mode is available. Then the second capability to be established is whether full colour mode is available.

Thirdly, a means is provided to indicate to the called unit that the Huffman tables are the default tables. The transmission of Huffman tables is mandatory.

In addition to these three characteristics, the following four capabilities that pertain to mandatory or optional capabilities are exchanged.

Table E.1/T.30

Mandatory and optional capabilities

Mandatory	Optional
8 bits/pel/component	12 bits/pel/component
4:1:1 Chrominance subsampling	No subsampling (1:1:1)
CIE Standard Illuminant D50	Custom illuminant
Default gamut range	Custom gamut range

E.5   New entries to DIS/DTC and DCS frames

One additional octet to the DIS/DTC and DCS frames is defined in this Annex. The new octet is to occupy bits 68 to bits 75.

The definitions, excerpted from Table 2/T.30 and Table C.1/T.30, are as follows:

Bit No.	DIS/DTC	DCS
68	JPEG coding	JPEG coding
69	Full colour mode	Full colour mode
70	Always set to zero	Default Huffman tables
71	12 bits/pel/component	12 bits/pel/component
72	Extend field	Extend field
73	No subsampling (1:1:1)	No subsampling (1:1:1)
74	Custom illuminant	Custom illuminant
75	Custom gamut range	Custom gamut range

E.5.1 Definitions of new entries to DIS/DTC and DCS frame

E.5. 1.1 Capability to enable JPEG

Bit 68 is called "Capability to enable JPEG."

In a DIS/DTC frame, setting bit 68 to 1 indicates that the called unit's JPEG mode is available and can decode continuous-tone image data (8 bits/component or more). Setting bit 68 to 0 indicates that the called unit's JPEG mode is not available and it cannot decode JPEG encoded data.

In a DCS frame, setting bit 68 to 1 indicates that the calling unit's JPEG mode is used and JPEG encoded image data are sent. Setting bit 68 to 0 indicates that the JPEG mode is not used and image is not encoded using JPEG.

E.5. 1.2 Capability to enable colour

Bit 69 is called "Capability to enable colour."

In a DIS/DTC frame, setting bit 69 to 1 indicates that the called unit has full colour capability. It can accept full colour image data in CIELAB space. Setting bit 69 to 0 indicates that the called unit has gray-scale mode only, that is, it accepts only the lightness component (the L* component) in the CIELAB representation.

In a DCS frame, setting bit 69 to 1 indicates that the calling unit sends image in full colour representation in the CIELAB space. Setting bit 69 to 0 indicates that the calling unit sends only the lightness component (the L* component) in the CIELAB representation.

Note - If bit 68 = 1 and bit 69 = 0, the continuous-tone image data have no colour component. The image data are called gray-scale or black and white gray-scale images. Continuous-tone full colour image capability is enabled only when bits 68 and 69 are both set to one.

E.5. 1.3 Indication of default Huffman table

Bit 70 is called "Indication of default Huffman tables."

The transmission of Huffman tables is mandatory. A means is provided to indicate to the called unit that the Huffman tables are the default tables. Default tables are specified only for the default image intensity resolution (8 bits/pel/component). The default Huffman tables are to be determined (for example, Tables K.3 - K.6 in Annex K of Rec. T.81).

In a DIS/DTC frame, bit 70 is not used and is set to zero.

In a DCS frame, setting bit 70 to 0 indicates that the calling unit does not identify the Huffman tables that it uses to encode the image data as the default tables. Setting bit 70 to 1 indicates that the calling unit identifies the Huffman tables that it uses to encode the image data as the default tables.

E.5. 1.4 Image intensity resolution

Bit 71 is called "Image intensity resolution."

In a DIS/DTC frame, setting bit 71 to 0 indicates that the called unit can only accept image data that are digitised to 8 bits/pel/component. Setting bit 71 to 1 indicates that the called unit can also accept image data that are digitised to 12 bits/pel/component.

In a DCS frame, setting bit 71 to 0 indicates that the calling unit's image data are digitised to 8 bits/pel/component. Setting bit 71 to 1 indicates that the calling unit's image data are digitised to 12 bits/pel/component.

E.5. 1.5 Chrominance subsampling ratio

Bit 73 is called "Chrominance subsampling ratio."

In a DIS/DTC frame, setting bit 73 to 0 indicates that the called unit expects a 4:1:1 subsampling ratio of the chrominance components in the image data; the a* and b* components in the CIELAB colour space representation are subsampled four times to one against the L* (Lightness) component. The details are described in Rec. T.4 Annex I. Setting bit 73 to 1 indicates that the called unit, as an option, accepts no subsampling in the chrominance components in the image data.

In a DCS frame, setting bit 73 to 0 indicates that the called unit uses a 4:1:1 subsampling ratio of the a* and b* components in the image data. Setting bit 73 to 1 indicates that the called unit does no subsampling.

E.5. 1.6 Illuminant

Bit 74 is called "Illuminant."

In a DIS/DTC frame, setting bit 74 to 0 indicates that the called unit expects that the CIE Standard Illuminant D50 is used in the colour image data as specified in Rec. T.42. Setting bit 74 to 1 indicates that the called unit can also accept other illuminant types beside the D50 illuminant. The specification of illuminant is embedded into the JPEG syntax as described in Annex I to Rec. T. 4.

In a DCS frame, setting bit 74 to 0 indicates that the calling unit uses the D50 illuminant in the colour image data representation as specified in Rec. T.42. Setting bit 74 to 1 indicates that another type of illuminant is used, the specification of which is embedded into the JPEG syntax as described in Annex I to Rec. T.4.

E.5. 1.7 Gamut range

Bit 75 is called "Gamut range."

In a DIS/DTC frame, setting bit 75 to 0 indicates that the called unit expects that the colour image data are represented using the default gamut range as specified in Rec. T.42. Setting bit 75 to 1 indicates that the called unit can also accept other gamut ranges, the specification of which is embedded into the JPEG syntax as described in Annex I to Rec. T.4.

In a DCS frame, setting bit 75 to 0 indicates that the calling unit uses the default gamut range as specified in Rec. T.42. Setting bit 75 to 1 indicates that the calling unit uses a different gamut range, the specification of which is embedded into the JPEG syntax as described in Annex I to Rec. T.4.

ITU - Telecommunication Standardization Sector

STUDY GROUP 8 and its Working Parties

Temporary Document 3 123

Original: English

Geneva, 21-30 June 1994

Question: 5/8

Source: EDITOR

Title: EDITORIAL CHANGES TO COM 8-43-E, DRAFT RECOMMENDATION T.30 ANNEX E

The following editorial changes are made to the Draft Recommendation T.30 Annex E, distributed as COM 8-43-E:

Page 2. Replace the word in the amended Table 2/T.30 as shown:

Bit No.	DIS/DTC	DCS
70	Always set to zero	~~Default~~Preferred Huffman tables

Page 4, Section E.4. Replace the word in the third paragraph as shown:

Thirdly, a means is provided to indicate to the called unit that the Huffman tables are the ~~default~~preferred tables. The transmission of Huffman tables is mandatory.

Page 5, Section E.5. Replace the word in the Table as shown:

Bit No.	DIS/DTC	DCS
70	Always set to zero	~~Default~~Preferred Huffman tables

Contact: Daniel T. Lee
Tel: +1 415 857 3946
Fax: +1 415 857 8526

Page 6, Section E.5.1.3. Modify the Section as shown:

E.5. 1.3 Indication of ~~default~~preferred Huffman table

Bit 70 is called "Indication of ~~default~~preferred Huffman tables."

The transmission of Huffman tables is mandatory. A means is provided to indicate to the called unit that the Huffman tables are the ~~default~~preferred tables. ~~Default~~Preferred tables are specified only for the default image intensity resolution (8 bits/pel/component). The ~~default~~preferred Huffman tables are ~~to be determined (for example,~~ Tables K.3 - K.6 in Annex K of Rec. T.81~~)~~.

In a DIS/DTC frame, bit 70 is not used and is set to zero.

In a DCS frame, setting bit 70 to 0 indicates that the calling unit does not identify the Huffman tables that it uses to encode the image data as the ~~default~~preferred tables. Setting bit 70 to 1 indicates that the calling unit identifies the Huffman tables that it uses to encode the image data as the ~~default~~preferred tables.

INTERNATIONAL TELECOMMUNICATION UNION

TELECOMMUNICATION
STANDARDIZATION SECTOR

STUDY PERIOD 1993 - 1996

Question: 5/8

STUDY GROUP 8 - CONTRIBUTION 44

Source: Q.5/8 Rapporteur

Title: Amendments to ITU-T Rec. T.4 to enable continuous-tone colour and gray-scale modes for Group 3.

Abstract

The proposed draft Annex I to Rec. T.4 enables the transfer of continuous-tone colour and gray-scale images as an option to Group 3 facsimile. In this Annex, continuous-tone image data coded in accordance with Rec. T.81, Digital compression and coding of continuous-tone still images, are added to Rec. T.4. The data are defined based on the colour space representation specified in Rec. T.42. In addition, amendments to the main body of Rec. T.4 that refer to this Annex are made.

* Contact person - Name:Mr Daniel T Lee Hewlett Packard USA   Tel.:+1 415 857 3946

REFERENCE TEXT: ITU-T Rec. T.4 (1988).

Proposed amendments to Rec. T.4

1. Append the following sentences to Rec. T.4 section 2.1:

Interworking between equipment with A5/A6 and A4 facilities, and between equipment with combinations of these facilities is shown in Annex C.

Optionally, continuous-tone and colour images may be transmitted using Group 3 facsimile apparatus as described in Annex I. A subset of the dimensions listed above, namely those having vertical resolutions of 7.7 lines/mm and 15.4 lines/mm, may be used with the procedure in Annex I. A vertical resolution of 3.85 lines/mm is not supported by Annex I.

2. Add the following section to the end of Rec. T.4:

12   Continuous-tone colour and gray-scale modes

Continuous-tone colour and gray-scale modes are optional features of Group 3 which enables transmission of colour or gray-scale images. These modes are specified in Annex I.

3. Add the following Annex I to Rec. T.4:

- 3 -
COM 8-44-E

Annex I

Optional continuous-tone colour mode for Group 3

(This Annex forms an integral part of this Recommendation)

I.1 Introduction

This Annex specifies the technical features of continuous-tone colour and gray-scale modes for Group 3 facsimile. Continuous-tone and colour modes are optional features of Group 3 facsimile which enables gray-scale or colour image transfer.

The method for image encoding is based upon the ITU-T Rec. T.81 (JPEG), Digital compression and coding of continuous-tone still images, and ITU-T Rec. T.42, which specifies the colour space representation.

The methods for image transfer applied to Group 3 facsimile are a subset of Rec. T.81, consistent with the Recommendation.

The description of colour components and colourimetry for colour data is included in Rec. T.42.

Together with ITU-T Rec. T.30 Annex E, this Annex provides specification of the telecommunication protocol and coding for transmission of continuous-tone colour and gray-scale images via Group 3 facsimile service.

I.2 Definitions

The definitions contained in Rec. T.4, Rec. T.30, Rec. T.81 and Rec. T.42 apply unless explicitly amended.

CIELAB	CIE 1976 (L* a* b*) space. A colour space defined by the CIE (Commission Internationale de l'Éclairage), having approximately equal visually perceptible difference between equispaced points throughout the space. The three components are L*, or Lightness, and a* and b* in chrominance.
JPEG	Joint Photographic Experts Group, and also shorthand for the encoding method, described in Rec. T.81, which was defined by this group.
Baseline JPEG	A particular eight-bit sequential Discrete Cosine Transform (DCT) - based encoding and decoding process specified in Rec. T.81.
Quantisation table	A set of 64 values used to quantise the DCT coefficients in baseline JPEG.
Huffman table	A set of variable length codes required in a Huffman encoder and a Huffman decoder.

I.3 Normative references

- CIE Publication No. 15.2, Colourimetry, 2nd. Ed. (1986).
- ITU-T Rec. T.81 | ISO/IEC 10918-1, Information Technology -- Digital compression and coding of continuous-tone still images, Part 1: Requirements and guidelines. (Commonly referred to as JPEG standard).
- ITU-T Rec. T. 42, Continuous-tone colour representation method for facsimile.
- ITU-T Rec. T. 30, Procedures for document facsimile transmission in the general switched telephone network.

I.4   Definition of different multi-level image transfer modes

The following different multi-level image transfer modes are defined:

Lossy gray-scale mode   (LGM)

Lossy colour mode   (LCM)

Lossless gray-scale mode (LLGM)

Lossless colour mode   (LLCM)

At this time, only LGM and LCM are described. LLGM and LLCM, while available within the coding methods described in Rec. T.81, are for further study.

I.4.1 Lossy gray-scale mode

Lossy gray-scale mode provides the user of Group 3 equipment with a means to transfer images with more than one bit/pel of monochrome image data. The method is not information conserving, and the amount of lossiness is determined by the quantisation tables described in Rec. T.81. The appearances of the gray-scale levels are defined by the Lightness (L*) component of CIELAB space.

I.4.2 Lossy colour mode

Lossy colour mode provides the user of Group 3 equipment with a means to transfer images with more than one bit/pel of image data in each of three colour components. The colour components are explicitly defined in Rec. T.42, and consist of CIELAB lightness and chrominance variables. The method is not information conserving and the amount of lossiness is determined by the quantisation tables described in Rec. T.81.

I.5 Coding of the image description

Sufficient image description is specified within the headers of Rec.T.81 Annex B, Compressed data format, to decode the image data. Other information, such as aspect ratio, orientation, and color space are defined uniquely by the application. In addition, some information required to establish the availability of this service is transmitted as specified in Rec. T.30 Annex E. Specifically, the transfer of JPEG-coded data, the use of gray-scale or colour data, and the use of 8 or 12 bits/component/pel data is negotiated and specified in the DIS/DTC and DCS frames as stated in Rec. T.30 Annex E.

I.5.1 Lossy gray-scale mode

The image description coding for gray-scale mode is accomplished by parameters specifying JPEG coding of a gray-scale image as specified in Rec. T.30 Annex E, as well as by specification of a single component as the number-of-components, Nf, in the Frame Header. The JPEG syntax is more thoroughly described in I.6.

I.5.2 Lossy colour mode

The image description coding for colour mode is accomplished by parameters specifying JPEG coding of a colour image and spatial resolution as specified in Rec. T.30 Annex E, as well as by specification of three components as the number-of-components, Nf, in the Frame Header. The colour data are block-interleaved, as specified in Rec. T.81. In addition, the JPEG subsampling factors and correspondence of quantisation tables to colour components are specified within the Frame Header, as detailed in Rec. T.81.

I.6 Data format

I.6.1 Overview

The JPEG-encoded image data consist of a series of markers, parameters, and scan data that specify the image coding parameters, image size, bit-resolution, and entropy-encoded block-interleaved data.

The data stream is encoded for facsimile transfer using the error correction mode (ECM) specified in Rec. T.30 Annex A. Pad characters (X'00', the null character, or X'20', the 'space' character) are added after EOI within the last ECM frame of the page to complete the last frame, in alignment with Rec. T.4 Annex A.

I.6.2 JPEG data structure

The JPEG data structure for this application has the following elements, as specified by Rec. T.81 Annex B: Parameters, markers, and entropy-encoded data segments. Parameters and markers are often organised into marker segments. Parameters are integers of length 1/2, 1, or 2 octets. Markers are assigned two-octet codes, an X'FF' octet followed by an octet not equal to X'00' or X'FF'.

The markers used in this application are characterised as follows:

> (1) The encoder shall insert these markers, and the decoder shall be able to carry out a corresponding process upon these marker segments:

> > SOI, APP1, DQT, DHT, SOF0, SOS, EOI.

> (2) The encoder may insert these markers without negotiation, and the decoder shall be able to carry out a corresponding process upon these marker segments:

> > DRI, RSTn, DNL.

> (3) The encoder may insert this marker without negotiation, and the decoder shall skip these marker segments and continue the decoding process:

> > COM, APPn (n not 1).

> (4) The encoder may insert this marker when the decoder has the ability to carry out a process corresponding to this marker segment (negotiation is necessary). If used, it replaces SOF0 in the data stream:

> > SOF1.

The definitions of the markers are precise, and given in detail in Rec. T.81 Annex B, except for the APPn markers. For example, SOI is a two-octet word X'FFC0', in hexadecimal notation. APPn markers are undefined markers provided within Rec. T.81 to facilitate the adaptation of the Recommendation to particular applications. Group 3 colour facsimile is one such application. The APPn markers are defined in section I.6.5 - I.6.8.

The DNL marker is a JPEG option that is critical to the function of this coding method in machines that do not pre-scan the image. When the number of lines, Y, in the frame header is set to value 0, the number of lines in the frame remains open until defined by the DNL marker at the end of the scan. If the scanning terminates early, the DNL marker can also be used to reset the Y value to a smaller value.

L6.2.1 Example of JPEG data structure for a 4:1:1 subsampled colour image

SOI	(start of image marker)
APP1, Lp	(application marker one, marker segment length)
Api	(application data octets: "G3FAX"-X'00', X'07CA' (version), X'00C8' (200 dpi))
(COM, Lc, Cmi)	(comment marker, marker segment length, comment octets (up to 255))
DHT, Lh	(define Huffman table marker, Huffman table length definition)
Tc, Th	(table class Tc = 0 for DC, destination identifier Th = 0 for L*)
Li, Vij	(number of codes for each of the 16 allowed code lengths, code values)
Tc, Th	(table class Tc = 1 for AC, destination identifier T h= 0 for L*)
Li, Vij	(number of codes for each of the 16 allowed code lengths, code values)
Tc, Th	(table class Tc = 0 for DC, destination identifier Th = 1 for a*, b*)
Li, Vij	(number of codes for each of the 16 allowed code lengths, code values)
Tc, Th	(table class Tc = 1 for AC, destination identifier Th = 1 for a*, b*)
Li, Vij	(number of codes for each of the 16 allowed code lengths, code values)
DQT, Lq	(define quantisation table marker, quantisation table length definition)
Pq, Tq	(element precision Pq = 0 for 8-bit, destination identifier Tq = 0 for lightness)
Qk	(64 quantisation table elements for quantisation table 0 (lightness))
Pq, Tq	(element precision Pq = 0 for 8-bit, destination identifier Tq = 1 for chrominance)
Qk	(64 quantisation table elements for quantisation table 0 (chrominance))
(DRI, Lr, Ri)	(define restart interval marker, marker segment length, restart interval in MCUs)
SOF0, Lf	(Start of frame marker for default 8-bit Huffman coded DCT, frame header length)
P, Y, X	(sample precision P = 8, number of lines Y, number of samples per line X)
Nf	(number of image components Nf = 3 for colour)
C1	(component identifier C1 = 0 for L* component)
H1, V1	(horizontal and vertical sampling factors: H1 = 2, V1 = 2 for L* in colour 4:1:1)
Tq1	(quantisation table selector: Tq1 = 0)
C2	(component identifier C2 = 1 for a* component)
H2, V2	(horizontal and vertical sampling factors: H2 = 1, V2 = 1 for a* in colour 4:1:1)
Tq2	(quantisation table selector: Tq2 = 1)
C3	(component identifier C3 = 2 for b* component)
H3, V3	(horizontal and vertical sampling factors: H3 = 1, V3 = 1 for b* in colour 4:1:1)
Tq3	(quantisation table selector: Tq3 = 1)
SOS, Ls, Ns	(Start of scan marker, scan header length, number of components Ns = 3 for colour)
Cs1	(scan component selector Cs1 = 0 for L*)
Td1, Ta1	(DC entropy coding table selector Td1 = 0, AC table selector Ta1 = 0 for L*)
Cs2	(scan component selector Cs2 = 1 for a*)
Td2, Ta2	(DC entropy coding table selector Td2 = 1, AC table selector Ta2 = 1 for a*)
Cs3	(scan component selector Cs3= 2 for b*)
Td3, Ta3	(DC entropy coding table selector Td3 = 1, AC table selector Ta3 = 1 for b*)
Ss, Se	(Ss = 0 for sequential DCT, Se = 63 for sequential DCT)
Ah, Al	(Ah = 0 for sequential DCT, Al = 0 for sequential DCT)
Scan data	(compressed image data)
(with RSTn)	(restart marker between image data segments, with n = 0-7 repeating in sequence)
(DNL, Ld, Y)	(define number of lines marker, marker segment length, number of lines)
EOI	(end of image marker)

Notes - Parentheses around a marker indicate the marker is classified to (2), (3), or (4). All indented lines are single or multiple parameters.

The Huffman tables can be identified as the default Huffman tables during negotiation as described in Rec. T.30 Annex E . The default Huffman tables are to be determined (for example, Tables K.3 - K.6 in Annex K of Rec. T.81).

I.6.2.2 Scan data structure

The scan data consist of block interleaved L*, a*, and b* data. Blocks are entropy-encoded DCT-transformed 8x8 arrays of image data from a single image component. The L*, a* and b* components are assigned indices zero, one, and two respectively in the frame header. When a gray-scale image is transmitted, only the L* component is represented in the data structure. The number of image components is either one (for a gray-scale image) or three (for a colour image).

The data are block-interleaved when a colour image is transmitted, and only one scan is contained within the image data. The blocks are organised in minimum coding units (MCU) such that an MCU contains a minimum integral number of all image components. The interleaving has the following form in the default (4:1:1) subsampling case, as defined in Rec. T.81 Annex A.2.3. In this case an MCU consists of four blocks of L* data, one block of a* data, and one block of b* data. The data are ordered L*, L*, L*, L*, a*, b* in the MCU. The four L* blocks proceed in the same scan order as the page: left to right and top to bottom. Therefore the L* blocks are transmitted first upper left, then upper right, then lower left, then lower right.

I.6.3 Subsampling method

The default (4:1:1) subsampling is specified as a four coefficient (tap) filter with coefficients (1/4, 1/4, 1/4, 1/4). Thus a* and b* are computed from non-subsampled data by averaging the four values of chrominance at the lightness locations. The location of the subsampled chrominance pixels is shown in the following illustration.

```
X    X  | X    X
    O   |     O
X    X  | X    X
--------+--------
X    X  | X    X
    O   |     O        X  represents lightness pel center
X    X  | X    X       O  represents chrominance pel center
```
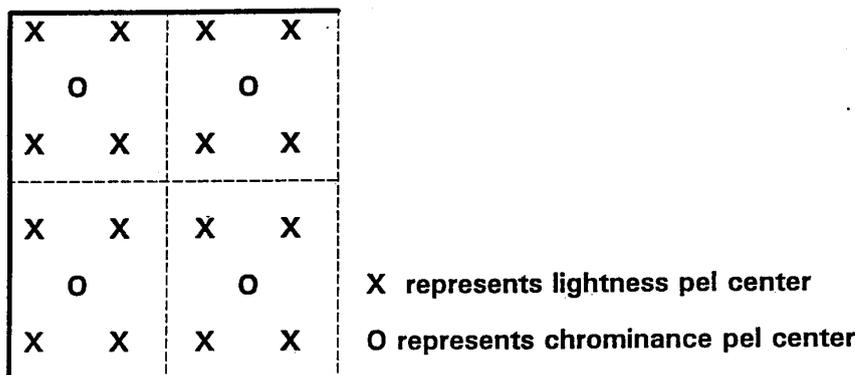
Figure I.1   Position of lightness and chrominance samples (4:1:1 subsampling) within the MCU's. Each small square represents an MCU.

I.6.4 Colour representation using the default gamut range

The following colour representation is in alignment with Rec. T.42.

Colour data is represented using the CIELAB space. CIELAB colour data are acquired under a particular illuminant, and computed from spectral or colourimetric data using a particular white point. The basic illuminant is CIE Standard Illuminant D50. The white point is the perfectly diffuse reflector associated with the D50 illuminant. In CIE XYZ colour space, this white point is specified as $X_0 = 96.422$, $Y_0 = 100.000$, $Z_0 = 82.521$. Optional illuminants are for further study. The default range of CIELAB data which may be coded in eight bits/pel/component is (to the nearest integer):

L* = [0, 100]

a* = [-85, 85]

b* = [-75, 125].

The default representations for encoding real CIELAB data as eight bit integers are:

$$L = (L*) * (255/100)$$

$$a = (a*) * (255/170) + 128$$

$$b = (b*) * (255/200) + 96,$$

where L, a, and b represent eight bit integers, and L*, a*, and b* represent real numbers. Rounding to the nearest integer is performed. If L, a, or b fall outside the range [0, 255], they are truncated to 0 or 255 as appropriate.

The default representations for encoding real CIELAB data as twelve bit integers are:

$$L = (L*) * (4095/100)$$

$$a = (a*) * (4095/170) + 2048$$

$$b = (b*) * (4095/200) + 1536,$$

where L, a, and b represent the twelve bit integers, and L*, a*, and b* represent the continuous numbers. Rounding to the nearest integer is performed. If L, a, or b fall outside the range [0, 4095], they are truncated to 0 or 4095 as appropriate.

I.6.5 Definition of the APPn markers for continuous-tone G3FAX:

The application marker APP1 initiates identification of the image as a G3FAX application and defines the spatial resolution and subsampling. This marker directly follows the SOI marker. The data format is as follows:

X'FFE1' (APP1) , length, FAX identifier, version, spatial resolution.

The above terms are defined as follows:

Length:	(Two octets) Total APP1 field octet count including the octet count itself, but excluding the APP1 marker.
FAX identifier:	(Six octets) X'47', X'33', X'46', X'41', X'58', X'00'. This X'00'-terminated string "G3FAX" uniquely identifies this APP1 marker.
Version:	(Two octets) X'07CA'. This string specifies the year of approval of the standard, for identification in the case of future revision (for example, 1994).
Spatial resolution :	(Two octets) Lightness pixel density in pels/25.4 mm. The basic value is 200. Allowed values are 200, 300, and 400 pels/25.4 mm, with square (or equivalent) pels.

Note - the functional equivalence of inch-based and mm-based resolutions is maintained. For example, the 200x200 pel/25.4 mm and 8/7.7 line/mm resolutions are equivalent.

An example of the string including the SOI and APP1 codes for a baseline JPEG encoded 1994 G3FAX application at 200 pels/25.4 mm:

X'FFD8', X'FFE1', X'000C', X'47', X'33', X'46', X'41', X'58', X'00', X'07CA', X'00C8'.

I.6.6 FAX option identifier: G3FAX1 for gamut range.

X'FFE1' (APP1), length, G3FAX option identifier, gamut range data.

The above terms are defined as follows:

Length:	(Two octets) Total APP1 field octet count including the octet count itself, but excluding the APP1 marker.
FAX identifier:	(Six octets) X'47', X'33', X'46', X'41', X'58', X'01'. This X'01'-terminated string "G3FAX" uniquely identifies this APP1 marker as containing FAX information about optional gamut range data. (The FAX option identifiers are referred to as G3FAX1 - G3FAX255, meaning the octet-terminated string, "G3FAX", X'nn').
Gamut range data:	(Twelve octets) The data field contains six two-octet signed integers. For example: X'0064' represents 100. The calculation from a real value L* to an eight bit value, L, is made as follows:

$$L = (255/Q) * L* + P,$$

where the first integer of the first pair, P, contains the offset of the zero point in L* in the eight most significant bits. The second integer of the first pair, Q, contains the span of the gamut range in L*. Rounding to the nearest integer is performed. The second pair contains offset and range values for a*. The third pair contains offset and range values for b*. If the image is gray-scale (L* only), the field still contains six integers, but the last four are ignored.

Note - This representation is in accord with Rec. T.42. When the twelve bits/pel/component option is used, the range and offset are represented as above in eight bits. These represent the eight most significant bits of the zero-padded twelve-bit number in the offset, and the eight-bit integer range data as above. Appropriately higher precision calculation should be used.

For example, the gamut range L* = [0, 100], a* = [-85, 85], and b* = [-75, 125] would be selected by the code:

X'FFE1', X'0014', X'47', X'33', X'46', X'41', X'58', X'01', X'0000', X'0064', X'0080', X'00AA', X'0060', X'00C8'.

I.6.7 FAX option identifier: G3FAX2 for illuminant data

X'FFE1' (APP1), length, G3FAX option identifier, illuminant data. This option is for further study with the exception of the default case; the specification of the default illuminant, CIE Illuminant D50, may be added for information.

Length:	(Two octets) Total APP1 field octet count including the octet count itself, but excluding the APP1 marker.
FAX identifier:	(Six octets) X'47', X'33', X'46', X'41', X'58', X'02'. This X'02'-terminated string "G3FAX" uniquely identifies this APP1 marker as containing optional illuminant data.
Illuminant data:	(Four octets) The data consist of a four octet code identifying the illuminant. In the case of a standard illuminant, the four octets are one of the following:

CIE Illuminant D50:	X'00', X'44', X'35', X'30'
CIE Illuminant D65:	X'00', X'44', X'36', X'35'
CIE Illuminant D75:	X'00', X'44', X'37', X'35'
CIE Illuminant SA:	X'00', X'00', X'53', X'41'
CIE Illuminant SC:	X'00', X'00', X'53', X'43'

CIE Illuminant F2:   X'00', X'00', X'46', X'32'

CIE Illuminant F7:   X'00', X'00', X'46', X'37'

CIE Illuminant F11:   X'00', X'46', X'31', X'31'

In the case of a colour temperature alone, the four octets consist of the string "CT", followed by the temperature of the source in degrees Kelvin represented by an unsigned two-octet integer. For example, a 7500° K illuminant is indicated by the code:

X'FFE1', X'000C', X'47', X'33', X'46', X'41', X'58', X'02', X'43', X'54', X'1D4C'.

I.6.8 Future option identifiers: G3FAX3 to G3FAX255

In addition to the G3FAX1 and G3FAX2 identifiers used for specifying optional parameters, the identifiers from G3FAX3 to G3FAX255 are reserved for future use.

Geneva, 21-30 June 199~

Question: 5/8

Source: EDITOR

Title: EDITORIAL CHANGES TO COM 8-44-E, DRAFT RECOMMENDATION T.4 ANNEX I

The following editorial changes are made to the Draft Recommendation T.4 Annex I, distributed as COM 8 44-E:

Page 2, Correct the section numbering as shown:

~~12~~13   Continuous-tone colour and gray-scale mode

Page 3, Section I.1. Correct the spelling in the fourth paragraph as shown:

The description of colour components and ~~colourimetry~~colorimetry for colour data is included in Rec. T.42.

Page 3, Section I.3. Correct the spelling in the first reference as shown:

-   CIE Publication No. 15.2, ~~Colourimetry~~Colorimetry, 2nd. Ed. (1986).

Page 5, Section I.6.2. Replace the word in the third paragraph as shown:

The definitions of the markers are precise, and given in detail in Rec. T.81 Annex B, except for the APPn markers. For example, SOI is a two-octet word ~~X'FFC0'~~X'FFD8', in hexadecimal notation.

Page 6, Section I.6.2.1 Modify the text in the first several lines as shown:

SOI   (start of image marker)

APP1, Lp   (application marker one, marker segment length)
     Api   (application data octets: "G3FAX"-X'00', X'07CA' (version), X'00C8' (200 dpi))

(APP1, Lp)   (application marker one, marker segment length)
     Api   (application data octets: "G3FAX"-X'01' (gamut range option), X'0000', X'0064', X'0080', X'00A0', X'0060', X'00C8', (gamut range values))

(COM, Lc, Cmi) (comment marker, marker segment length, comment octets ~~(up to 255)~~)

Contact: Daniel T. Lee
             Tel: +1 415 857 ...
             Fax: +1 415 857 ...

READING: EFFECTS OF CONTRAST AND SPATIAL FREQUENCY[1]

Gordon E. Legge

University of Minnesota, Minneapolis, MN 55455.

Introduction

Reading is a complex everyday task. Successful reading requires high-speed visual information processing. For several years, my colleagues and I have been studying visual factors in reading with two major goals in mind: to understand the roles played by sensory mechanisms in reading and to understand how visual impairment affects reading. In a typical study, we examine the effect of an important text variable (e.g. contrast) on reading by people with normal vision. Taking the normal data as a bench mark, we try to explain abnormalities in the performance of low-vision subjects.

In this talk, I will discuss the effects of contrast and spatial-frequency filtering on reading. After reviewing the empirical findings, I will comment on the likely sensory basis for the effects and point to the relevance of the findings for low vision.

Methods

Our primary measure of reading performance is *reading speed* in words/minute. We use two procedures to evaluate a subject's maximum reading speed. In the first, a line of text drifts smoothly across the screen of a TV monitor. The subject reads the text aloud as it drifts by. If no errors are made, the drift rate is increased on the next trial. This process continues until the subject begins to make errors. Because there is a sharp transition from error-free reading to a drift rate with many errors, this technique yields a reproducible and well-defined measure of reading performance. In our second procedure, a sentence appears on the TV screen in a static display for a timed period. The subject reads through the sentence as rapidly as possible. If the subject completes the sentence, the exposure duration is reduced. Eventually, the subject cannot complete the sentence and reading speed is computed as the number of words read divided by the exposure time.

Reading rates obtained with these two procedures are highly correlated across subjects. Normal subjects can read the static text slightly faster than the drifting text but the reverse is true, on average, for subjects with low vision.

How Does Luminance Contrast Affect Reading Speed?

Normal vision is quite tolerant to contrast loss (Legge, Rubin & Luebker, 1987). A tenfold reduction from maximum contrast[2] results in only a slight decrease in reading speed (less than a factor of two.) A further reduction in text contrast results in a much sharper decline in reading rate. Curves of reading speed as a function of contrast for different character sizes superimpose when text contrast is normalized by threshold contrasts for the letters.

This pattern of results can be related to psychophysical models of contrast coding. Such models typically contain a contrast-transfer function that relates visual response to stimulus contrast. The transfer function is compressive at high contrasts and can account for the tolerance of reading to contrast change. The transfer function is accelerating at low contrast and can account for the strong dependence of reading on contrast at low levels.

High contrast is critical for many people with low vision. Their reading speeds decline with any reduction from maximum contrast. Rubin & Legge (1989) showed that a subgroup of subjects with low vision--those with cloudy ocular media but no retinal involvement--can be characterized as "contrast attenuators." Their performance is normal, apart from a scaling of text contrast that is equal to their decrease from normal in letter-contrast sensitivity.

How Does Color Contrast Affect Reading Speed?

We have measured reading speed for text conveyed by color contrast (e.g. red letters on a green background) rather than the more usual luminance contrast (Parish, Legge & Luebker, 1989). Colors were matched for luminance by flicker photometry. Color contrast was varied by mixing different proportions of red and green (or yellow and blue) in the text and background. Color contrast was defined as the luminance contrast of the component colors making up the stimulus.

When color contrast is high, reading rates match the highest values found for luminance contrast. As with luminance, reading performance first declines slowly with reduction in color contrast, and then more rapidly at low levels. Reading rates for luminance and color contrast can be compared if both are expressed as multiples of threshold contrast. When this is done, curves of reading speed vs. contrast superimpose for luminance and color. This finding holds for both normal and low vision.

These results show that reading can rely equally well on information conveyed by luminance or color contrast. The mechanisms coding this information must be quite

---

[2]Text contrast is defined as $(C-B)/(C+B)$ where C and B are the luminances of the characters and the background.

similar and possibly share a common neural pathway.

How Does Character Size Affect Reading Speed?

Reading rates are highest on a plateau extending from about $.3^{\circ}$ to 2° (Legge *et al.*, 1985a). There is a sharp decline in reading speed for smaller characters, and a more gradual decline for larger ones. Of relevance to low vision, it is important to note that normal vision can sustain functionally useful reading rates for enormous characters (e.g. 70 words/minute for 24° characters).

We have studied character-size effects in low vision (Legge, *et al.*, 1985b). Prescription of appropriate magnification in reading aids relies on identification of the range of character sizes for which reading is fastest. For a subgroup of low-vision subjects--those with large central scotomas--reading performance benefits from ever-increasing character size.

It is possible to define a *contrast sensitivity function* (CSF) for reading. This is done by finding the contrast required to produce a threshold reading rate (say 35 words/minute) at each of many character sizes. These threshold contrasts (or their reciprocals) are plotted as a function of the fundamental frequency[3] of the characters. Such plots are qualitatively similar in shape to sine-wave grating CSFs. When appropriate stimulus conditions are compared, there is a striking quantitative similarity as well.

These results suggest that character size effects in reading are related to differences in contrast sensitivity across spatial frequency. Apparently, mechanisms that limit spatiotemporal contrast sensitivity play a role in reading.

How do Low- and High-Pass Spatial Filtering Affect Reading Speed?

We measured reading rates for text that was low-pass spatial-frequency filtered (Legge *et al.*, 1985a). Reading speed was quite tolerant to bandwidth reduction (blur). Performance was unaffected except when the bandwidth extended less than one octave above the fundamental frequency of the characters. The *critical bandwidth* of one octave was constant across a wide range of character sizes. The fact that rapid reading requires just one octave in spatial frequency implies that only one spatial-frequency channel is required for reading.

While blurry letters may be highly legible, what about high-pass-filtered text? A traditional view of reading holds that important information is conveyed by coarse word-shape information. This information might be carried by low spatial frequencies, below those carrying letter information (Morris, 1988). If word-shape information plays a role in the dynamics of reading, performance should be impaired if

---

[3]The *fundamental frequency* of characters in text is equal to the reciprocal of character size in degrees.

text is high-pass filtered. Recent measurements in my lab indicate that this is not the case. There is little difference in reading speed for unfiltered and high-pass filtered text.

Conclusions

Psychophysical techniques can be used to study the role of vision in reading. The dependence of reading speed on important text variables such as contrast, character size and spatial-frequency content can be related to known characteristics of sensory mechanisms. An understanding of the role of vision in normal reading enables us to understand reading deficits in people with low vision.

References

egge G. E., Pelli D. G., Rubin G. S. and Schleske M. M. (1985a) Psychophysics of reading. I. Normal vision. *Vision Res. 25*, 239-252.

egge G. E., Rubin G. S. and Luebker A. (1987) Psychophysics of reading. V. The role of contrast in normal vision. *Vision Res. 27*, 1165-1171.

egge G. E., Rubin G. S., Pelli D. G. and Schleske M. M. (1985b) Psychophysics of reading. II. Low vision. *Vision Res. 25*, 253-266.

iorris R. (1988) Image processing aspects of type. *Proc. EP88 Conf. on Text Processing*, Cambridge University Press.

arish D.H., Legge G.E. and Luebker A. (1989) Reading and color contrast. Annual Meeting of the Association for Research in Vision and Ophthalmology, Sarasota.

ubin G. S. and Legge G. E. (1989) Psychophysics of reading. VI. The role of contrast in low vision. *Vision Res., 29*, 79-91.

93

their judgments on temporal properties of the stimuli, rather than other phenomenal aspects.

[19]R. M. Boynton, W. R. Bush, and J. M. Enoch, "Rapid changes in foveal sensitivity resulting from direct and indirect adapting stimuli," J. Opt. Am. Soc. Am. 44, 56–60 (1954).

[20]The R measure is equivalent to the traditional measure β (response bias) when the a priori probabilities of signal trial and noise trial presentation are equal. R was chosen over β because accurate estimates of β could not always be derived on the basis of 200 experimental trials. See W. A. Yost, "A forced-choice adaptive procedure for measuring auditory thresholds in children," Behav. Res. Methods Instrum. 10, 671–677 (1978) for a similar use of log R.

[21]L. G. Standing and P. C. Dodwell, "Retroactive contour enhancement: A new visual storage effect," Quart. J. Exp. Psych. 24, 21–29 (1972).

[22]E. Donchin and D. Lindsley, "Retroactive brightness enhancement with brief paired flashes of light," Vision Res. 5, 59–70 (1965).

[23]E. Donchin and D. Lindsley, "Visually evoked response correlates of perceptual masking and enhancement," EEG Clin. Neurophysiol. 19, 325–335 (1965).

[24]E. Pulos, J. E. Raymond, and W. Makous, "Transient sensitization by a contrast flash," Vision Res. 20, 281–288 (1980).

[25]E. Pulos and W. Makous, "Sensitization following offset of an annulus," J. Opt. Soc. Am. 69, 1452(A) (1979).

[26]R. W. Bowen, K. A. Markell, V. Pappageorge, and F. Alfano, "Temporal resolution during transient light and dark adaptation," Invest. Ophthalmol. Visual Sci. Suppl. 18, 248–249 (1979).

[27]K. Dunlap, "The shortest perceptible time interval between two flashes of light," Psychol. Rev. 22, 226–250 (1915).

[28]S. H. Bartley and F. W. Wilkinson, "Certain factors in producing complexity of responses to a single pulse of light," J. Psychol. 4, 299–306 (1953).

[29]R. M. Boynton, "Discrimination of homogeneous double pulses of light," in Visual Psychophysics, Vol. VII/4, Handbook of Sensory Physiology, edited by D. Jameson and L. Hurvich (Springer-Verlag, Berlin, 1972), Chap. 9.

[30]R. M. Springer, J. A. Deutsch, and G. Stanley, "Double flashes from single pulses of light," Percept. Psychophys. 18, 398–400 (1975).

[31]G. M. Shickman, "Brief illumination and visual temporal resolving power," Dissertation, Harvard University, 1960 (unpublished).

[32]K. J. W. Craik, "The effect of adaptation on differential brightness discrimination," J. Physiol. 92, 406–421 (1938).

Contrast masking in human vision

Gordon E. Legge

Department of Psychology, University of Minnesota, Minneapolis, Minnesota 55455

John M. Foley

Department of Psychology, University of California, Santa Barbara, California 93106

(Received 12 February 1980)

Contrast masking was studied psychophysically. A two-alternative forced-choice procedure was used to measure contrast thresholds for 2.0 cpd sine-wave gratings in the presence of masking sine-wave gratings. Thresholds were measured for 11 masker contrasts spanning three log units, and seven masker frequencies ranging ± one octave from the signal frequency. Corresponding measurements were made for gratings with horizontal widths of 0.75° (narrow fields) and 6.0° (wide fields). For high contrast maskers at all frequencies, signal thresholds were related to masking contrast by power functions with exponents near 0.6. For a range of low masking contrasts, signal thresholds were reduced by the masker. For the wide fields, high contrast masking tuning functions peaked at the signal frequency, were slightly asymmetric, and had approximately invariant half-maximum frequencies that lie 3/4 octave below and 1 octave above the signal frequency. The corresponding low contrast tuning functions exhibited peak threshold reduction at the signal frequency, with half-minimum frequencies at roughly ± 0.25 octaves. For the narrow fields, the masking tuning functions were much broader at both low and high masking contrasts. A masking model is presented that encompasses contrast detection, discrimination, and masking phenomena. Central constructs of the model include a linear spatial frequency filter, a nonlinear transducer, and a process of spatial pooling that acts at low contrasts only.

INTRODUCTION

The term masking is used commonly to refer to any destructive interaction or interference among transient stimuli that are closely coupled in space or time.[1] Adaptation is often distinguished from masking, but the distinction is not very precise. An adapting stimulus onset always precedes the test stimulus and an adapting stimulus usually has a relatively long duration. The effect of masking may be a decrease in brightness, errors in recognition, or a failure to detect. This paper is concerned only with the effect of one stimulus on the detectability of another where the stimuli are coincident in space and simultaneous in time.

The effect of one stimulus on the detectability of another need not be to decrease detectability. Indeed, several studies have shown that a low contrast masker increases the detectability of a signal.[2-4] This effect is variously called negative masking, facilitation, or the pedestal effect. This paper is concerned with this effect as well as the more common masking effect.

In the experiments, an observer was required to discriminate the superposition a + b of two sine-wave grating stimuli a and b from grating b presented alone. Grating b is termed the masker. Its contrast remained fixed throughout a measurement. Grating a is termed the signal. Its contrast was varied to find its threshold. This masking procedure includes, as special cases, contrast detection and contrast discrimination. In contrast detection, the masking contrast is 0. In contrast discrimination, masker and signal gratings have the same frequency and phase, and differ only in contrast.

0).
; offset of an
and F. Alf
ark adaptati
9 (1979).
ral between
l.
rs in produ
," J. Psychol
louble puls
book of Sen
Springer-Ve
oie flashes h
78-400 (197
poral resolv
npublished]
itial brightn

he detect
negative
s paper is
common

discrimi.
ig stimuli
is termed
ut a mea-
trast was
includes,
icrimina.
is 0. In
have the
contrast.

1458

trast discrimination is sometimes referred to as contrast increment detection. Our use of the term masking and our paradigm for studying it are very similar to those commonly used in the study of auditory frequency analysis.[5]

Contrast masking has been used to study the spatial frequency selectivity of pattern vision by measuring signal thresholds at one spatial frequency in the presence of masking patterns of other spatial frequencies.[3,6-10] For high contrast maskers, and signals at medium and high spatial frequencies, these studies have found that signal threshold elevation is maximal when the masker and signal have the same frequency. Threshold elevation decreases regularly as the masking frequency departs from the signal frequency. The functions relating signal threshold elevation to masker frequency may be termed *spread of masking functions* or *masking tuning functions*.

It has been suggested that the bandwidths of these tuning functions provide estimates for spatial frequency channel bandwidths.[7] These estimates are in rough agreement with tuning functions obtained in measurements of the spatial frequency adaptation effect.[11,12] However, there is considerable variability in the form of tuning functions, both within and between methods, even at similar spatial frequencies. Widths at half-maximum range from 2 octaves[10] in a masking experiment, to 0.62 octaves[13] in an adaptation experiment. Estimates of the spatial frequency channel bandwidth from measurements of the detectability of complex gratings consisting of two sinusoidal components have yielded bandwidths as narrow as 0.4 octaves[14] or less than 1.0 cycle per degree (cpd) in the range 5 to 20 cpd.[15] However, it has been pointed out that the channel bandwidths derived from the detection of complex gratings depend very much on one's model of detection.[16-18] The same is true of masking and adaptation. It is not possible to relate the frequency selectivity of the masking or adaptation tuning functions to properties of underlying spatial frequency channels without invoking models of contrast masking or adaptation. To date, there exists no model of contrast masking. Moreover, there is evidence suggesting that adaptation may be determined in part by the inhibition of one channel by another,[19,20] although the need for this additional complication has been challenged.[21] Consequently, at this point, the spatial frequency channel bandwidth underlying frequency selectivity in masking or adaptation is still undetermined.

In the present study, masked thresholds were measured for a 10 cpd signal over a 2 octave range of masker frequencies and a 3 log unit range of masker contrasts. If the masking tuning function is to be taken as indicative of the relative spatial frequency sensitivity of the detecting channel, then, at minimum, this function ought to have the same form independent of masker contrast. One objective of the study was to determine whether contrast masking possesses this property. It does not. However, the results may be interpreted in the context of a model that includes a spatial frequency channel whose sensitivity function is independent of contrast and peaks at the signal frequency. This model is an extension of the nonlinear transducer model of Foley and Legge for contrast detection and discrimination.[4]

The second purpose of the study was to investigate properties of contrast masking under conditions of restricted field

size. It has been observed that masking tuning functions become broader (on a log frequency scale) for low signal frequencies.[9,10] The broadening of the masking tuning functions bears some similarity to peak shift and/or broadening of spatial frequency adaptation tuning functions at low spatial frequencies.[12,22] Comparison of these studies suggests a loose covariation between bandwidth estimates and field size. This covariation raises the possibility that the broadening of the tuning functions at low spatial frequencies may be related, not so much to inherent properties of low spatial frequency channels, but to the restricted number of cycles in the stimulus. Hence, corresponding masking measurements were made with stimuli that subtended either 6° or 0.75° horizontally. This stimulus width variation was found to have a large effect, but the model developed to account for the effects of varying contrast accounts for this result as well.

METHOD

Apparatus
Vertical sine-wave gratings were presented on a CRT display by Z-axis modulation.[23] The display, designed and constructed at the Physiological Laboratory, Cambridge, had a P31 phosphor. The constant mean luminance was 200 cd/m². The raster was derived from a 100 Hz horizontal sweep and 100 kHz vertical triangle wave.

The relation between grating contrast and Z-axis voltage and frequency was measured with a narrow slit and UDT 80X Opto-meter. As the grating pattern drifted behind the stationary slit, the Opto-meter obtained 256 luminance samples per cycle. The resulting sequence of values was Fourier analyzed. By this means the harmonic composition of the modulation was obtained for a given Z-axis voltage and frequency. During the experiments, all contrasts were kept within the range for which the amplitudes of harmonic distortion products were less than 3% of the fundamental. Luminance modulation was horizontally restricted to a region symmetric about the center of the screen. The remainder of the screen was maintained at the constant mean luminance level without modulation. This mode of presentation was produced by passing the Z-axis signal through an electronic switch. The onset and period of switch closure were controlled by logic pulses from the CRT circuitry and corresponded to that portion of the display sweep for which luminance modulation was desired.

Voltage waveforms, corresponding to masker and signal gratings, were derived from two function generators. Their outputs were electronically added and applied to the switch prior to Z-axis input. As a result, CRT luminance modulation consisted of the superposition of masker and signal gratings. When the masker and signal had the same spatial frequency, both were derived from the same function generator.

A DEC PDP-8 computer controlled stimulus durations, and contrast levels with D/A converters, dB attenuators, and analog multipliers. The computer sequenced stimulus presentations and collected data.

Procedure
Observers viewed the display binocularly, with natural pupils, at a distance of 114 cm. The screen subtended 10°

horizontally and 6° vertically, with a white cardboard surround. For the wide field masking, modulation was restricted horizontally to 6°, symmetric about a central fixation mark. For narrow field masking, modulation was restricted horizontally to 0.75°, symmetric about the same fixation mark. When signals were presented, their onsets and offsets were simultaneous with those of the masker.

Psychophysical threshold estimates were obtained with a version of the two-alternative forced-choice staircase procedure.[24] The observer began by adjusting signal grating contrast to a value just above threshold by turning a hand-held logarithmic attenuator. The observer was then given a block of self-initiated trials. Each trial consisted of two 200 ms exposures, marked by auditory tones, and separated by 750 ms. Only one exposure contained the signal grating, but both exposures contained the masking grating. The observer identified the signal interval by pressing one of two keys. A correct choice was followed by a tone. Three correct choices at one contrast level were followed by a constant decrement in contrast, and one incorrect choice was followed by an increment. The mean of the first six contrast peaks and valleys in the resulting sequence was taken as an estimate of the 0.79 proportion correct contrast level.[24] Typically, a block consisted of about 35 trials. For each condition, the *threshold measure* was the geometric mean of the threshold estimates from four to six blocks. The error bars in Figs. 2 and 4 correspond to ± one standard error of the mean.

Contrast thresholds were obtained in two masking experiments:

(1) Wide Field Masking: Signals and maskers subtended 6° by 6°. Contrast thresholds for sine-wave grating signals at 2.0 cpd were measured as a function of masking contrast for seven masking frequencies. The masking frequencies, arranged to lie θ, approximately ±¼, ±½, and ±1 octave from the signal frequency were: 1.0, 1.4, 1.7, 2.0, 2.4, 2.8, and 4.0 cpd. For each masking frequency, the 11 masking contrasts (percent contrasts) were: 0.05, 0.10, 0.20, 0.40, 0.80, 1.6, 3.2, 6.4, 12.8, 25.6, and 51.2%.

An experimental run, lasting about an hour, was devoted to a single masking frequency. Twelve signal threshold estimates were obtained, one in the absence of masking (detection threshold), and one for each of the 11 masking contrasts. Masking contrasts were presented in either ascending or descending order. No consistent differences could be observed in the resulting threshold estimates, suggesting that neither observer fatigue nor cumulative pattern adaptation played a role.

Three observers participated in the experiments, although time constraints prevented one observer from completing all conditions. An observer's participation in a given condition involved two runs through the set of contrasts, one in ascending and one in descending order. Except in Fig. 1, the data points are always the geometric means of either four or six threshold estimates, obtained from two or three observers. For a given observer, the order of masking frequencies was randomized across runs.

(2) Narrow Field Masking: Signals and maskers subtended 0.75° horizontally by 6.0° vertically, symmetrically arranged about the center of the screen. The remainder of

the screen was maintained at the same mean luminance level. Signal gratings at 2.0 cpd were truncated to 1.5 periods, consisting of a central bright half-cycle, flanked on either side by dark half-cycles.

The same set of masking frequencies and contrasts was used in the narrow field case, except that the lowest masking contrast of 0.05% was omitted. Procedural details were like those for wide field masking. Runs of narrow and wide field masking were interleaved.

Observers
There were three observers, all practiced with the methods and stimuli. WWL is a female, and SH a male, both in their mid twenties. JMF is a male in his late thirties. Throughout the experiments, observers were optically corrected.

RESULTS AND DISCUSSION

Individual data
At the signal frequency of 2.0 cpd, the three observers' threshold contrasts (percent contrasts) for the 6.0° and 0.75° gratings were, respectively: JMF, 0.28 and 0.51; WWL, 0.31 and 0.49; SH, 0.28 and 0.50. These values are geometric means of 7 to 16 separate detection threshold estimates, collected on different days in different runs.

Figure 1 shows data separately for the three observers when the masker frequency was 2.8 cpd and the gratings were 0.75° wide. Each point is the geometric mean of two threshold estimates. The important properties of the masking data will be discussed in the next two subsections. Note here, however, that the scatter of points across individuals is typical of data collected in all masking conditions. No systematic individual differences in sensitivity or shape of the masking curves were apparent across conditions. Accordingly, data were combined across observers. The remaining figures show combined data.
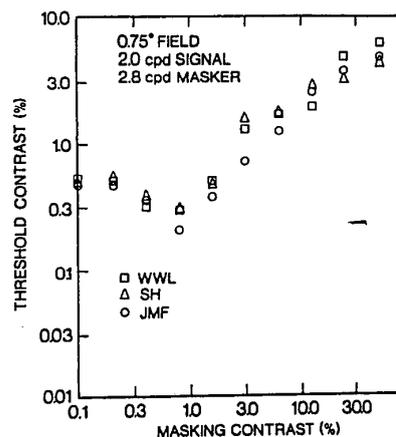

FIG. 1. Contrast masking data for three observers. Data for three observers are shown separately for one masking frequency. Each point is the geometric mean of two threshold estimates. The scatter of points is typical of results in other conditions. In subsequent figures, points represent data averaged across observers.
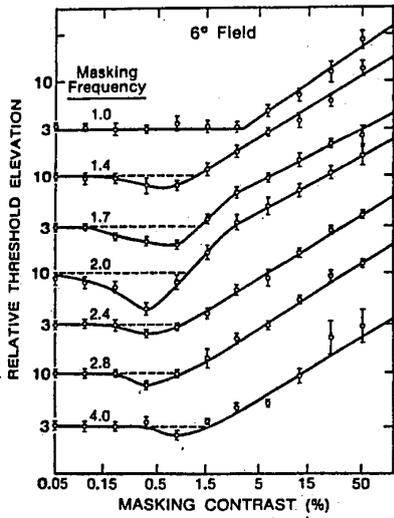
Gordon E. Legge and John M. Foley   1460

FIG. 2. Signal threshold as a function of masking contrast: wide fields. Signals and maskers subtended 6° by 6°. Contrast thresholds for 2.0 cpd sine-wave gratings are plotted as a function of masker grating contrast for seven masker spatial frequencies. To facilitate display, the sets of data points have been vertically displaced and sequenced in order of masking frequency. For each set of data, the ordinate gives threshold contrast in arbitrary units re unmasked threshold level indicated by the horizontal dashed lines. Data points are the geometric means of four to six threshold estimates from three observers. Threshold estimates were obtained from blocks of forced-choice trials. Error bars represent ±1 standard error. Straight lines have been fit to the data for masking contrasts in the range 3.2% to 51.2%. Their slopes are given in Table I. Smooth curves have been drawn through the remaining data within a set.

Wide field masking

In the wide field masking experiment, sinusoidal luminance modulation of the screen covered a horizontal extent of 10°.

Figure 2 presents signal thresholds as a function of the contrast of maskers. The signals were always 2.0 cpd sine-wave gratings. There were seven masking frequencies, one for each of the sets of data in the figure. Both signals and maskers were in cosine phase with the fixation point. Both were presented simultaneously for 200 ms. The data points are geometric means of four to six threshold estimates from three observers (see METHOD). There were 11 masking contrasts for each masking frequency, ranging over more than 3 log units from 0.05% to 51.2%. In Fig. 2, the ordinate is relative threshold elevation. It should be interpreted as follows. For each set of data, the horizontal dashed line represents the unmasked detection threshold in arbitrary contrast units. For a given set of data, all plotted thresholds should be taken relative to this level. For example, the curve parametrized by the masking frequency 1.0 cpd has an unmasked signal threshold of 3.0 arbitrary contrast units. A datum plotted at 9.0 represents a masked threshold greater by a factor of 3 than the unmasked threshold.

There is a range of very low masking contrasts for which the masker has little or no effect upon signal threshold. This range is frequency dependent, being lower for maskers whose frequencies approach more closely to the signal frequency of 2.0 cpd.

There is a range of high masking contrasts for which the signal thresholds at all masking frequencies appear to lie along straight lines in these double logarithmic coordinates. Moreover, because the slopes of these lines are similar, they appear to be roughly parallel. Best-fitting straight lines (method of least squares) have been computed over the contrast range from 3.2% to 51.2%. They are drawn as the straight line portions of the solid curves through the data in Fig. 2. The slopes of these straight lines are given in Table I. For wide field masking, the slopes range from 0.525 to 0.711 with a mean value of 0.620. There appear to be no systematic variations of slope with frequency. To a good approximation, the slopes at the seven masking frequencies may be taken as equal, and the high contrast portions of the masking functions in Fig. 2 as parallel. Parallel plots in double logarithmic coordinates mean that the corresponding functions are scaled versions of one another. The straight lines through the high contrast portions of the masking functions in Fig. 2 are roughly parallel, but they are not coincident. For masking frequencies more and more remote from the signal frequency of 2.0 cpd, corresponding straight lines are increasingly shifted to the right, representing a decrease in the effectiveness of masking. Denoting signal contrast threshold by C_t, masking contrast by C, and taking the power function exponent to be 0.62, the high contrast masking functions take the form

$$C_t = [k(f)C]^{0.62}, \qquad (1)$$

where $k(f)$ is a frequency-dependent scaling factor. The values of $k(f)$ were computed using a least-square criterion to find the best-fitting straight lines with slopes 0.62 through the high contrast data in Fig. 2. Values of $k(f)$ are presented in Table I.

Values of $k(f)$, normalized by the peak value at 2.0 cpd, are plotted as the filled circles in Fig. 8(a) (see below). Best-fitting straight lines have been drawn through the points to the right and left of the peak for purposes of interpolation and extrapolation later. $k(f)$ may be termed a sensitivity function. It represents the sensitivity of signal detection at 2.0 cpd to maskers of different frequencies. The values plotted in Fig. 8(a) are related reciprocally to masker contrasts required to produce a criterion signal threshold contrast. In this respect, the sensitivity function of Fig. 8(a) is analogous to an action spectrum. It corresponds to the "equivalent contrast

TABLE I. Exponents of the power functions relating signal thresholds to masking contrast.

Masking frequency	Wide field Exponent[a]	Narrow field Exponent[b]	"sensitivity function" $k(f)$
1.0	0.711	0.788	0.0025
1.4	0.610	0.567	0.0059
1.7	0.525	0.526	0.0072
2.0	0.558	0.672	0.0152
2.4	0.651	0.501	0.0093
2.8	0.673	0.577	0.0089
4.0	0.615	0.769	0.0050
Mean	0.620	0.627	

[a] Based on slope of best-fitting straight line in log-log coordinates in the range 3.2%–51.2% masking contrast.
[b] Based on slope of best-fitting straight line in log-log coordinates in the range 6.4%–51.2% masking contrast.

transformation" introduced by Blakemore and Nachmias.[25] In the masking model to be presented below, this sensitivity function represents the frequency selectivity of a linear, spatial frequency filter (channel). Notice that the sensitivity function is broader to the right of its peak than to the left. The half-maximum frequencies lie near $\frac{1}{4}$ octave below the peak and slightly more than $\frac{1}{2}$ octave above the peak. The overall bandwidth between half-maximum frequencies is therefore about 0.75 octaves. Blakemore et al.[26] found a contrast-invariant channel bandwidth of 0.75 octaves from grating adaptation studies using the equivalent contrast transformation.

Pantle[8] used binocular viewing and forced-choice methods to measure thresholds for sine-wave gratings presented for 1.7 s upon a continuously present, slowly drifting background grating. Frequencies of the background gratings varied from those of the signal gratings by factors of 1, 2, 3, and 5. Over several conditions of masker and signal frequencies, and contrast ranges, there was evidence for power-law relations between signal threshold and background contrast. The log-log slopes obtained by Pantle are not inconsistent with the conclusion reached here, that masking functions in which maskers differ from signals by as much as an octave in frequency are scaled versions of the same power law, for a range of high masking contrasts.

When the masker and signal both have spatial frequency 2.0 cpd, the observer's task is one of contrast discrimination. Table I indicates that the best-fitting straight line through the contrast discrimination data of Fig. 2 has a slope of 0.558, well below the value of 1.0 predicted by Weber's law. Legge,[10] using the same forced-choice procedure, luminance and monocular viewing, has measured contrast discrimination functions of this sort at several spatial frequencies. At 1.0, 4.0, and 16.0 cpd, the corresponding slopes were found to be 0.62, 0.67, and 0.70, respectively. (At the very low spatial frequency of 0.25 cpd, the slope was only 0.28.) Several other investigators have measured the contrast discrimination function using a variety of stimulus conditions and procedures. Some have found Weber's law behavior.[27] Others find that contrast discrimination thresholds increase more slowly than Weber's law predicts.[2,8,28–30] Still others find Weber's law behavior under some conditions and departures from it under others.[31–33] For a discussion of possible reasons for these discrepant findings, see Legge.[10]

Now consider the nonmonotonicity of the masking functions in Fig. 2. For masking contrasts between about 0.05% and 0.8%, signal thresholds are lower than thresholds measured in the absence of masking. For instance, for a 2.0 cpd masker having contrast 0.4%, the signal threshold is about two-and-one-half times lower than the detection threshold. In such cases, the masker may be said to *facilitate* signal detection. This phenomenon has been termed the *pedestal effect.* The *pedestal effect* also occurs for the discrimination of luminance increments.[34,35] It has been observed elsewhere for grating contrast discrimination,[2,3,10,27,36] and has been discussed at some length by Foley and Legge.[4] The pedestal effect of Fig. 2 is frequency selective. It is readily apparent when the masker and signal have the same frequency, but is greatly diminished when the masker and signal differ by ± 0.5 octaves. Unlike high contrast portions of the masking functions of Fig. 2, the frequency-dependent shape of the pedestal

effect means that the low contrast portions of the masking functions are *not* parallel. Hence, the low contrast portions of the masking functions are *not* simply scaled versions of one another.

The facilitation effect should be distinguished from "subthreshold summation.[37]" In the subthreshold summation paradigm, the observer adjusts the contrast C_1 of a superimposed pair of stimuli $C_1 + C_2$ until the combination reaches threshold. In the forced-choice discrimination paradigm of this paper, the masker C_1 is presented in both intervals of a trial, and the signal C_2 is presented in one interval only. The observer must discriminate between stimuli C_1 and $C_1 + C_2$. The two procedures would not, in general, be expected to yield the same result. In any case, a simple "summation to threshold" model cannot account for the facilitation effect, because signal thresholds are reduced by masking contrasts well above the unmasked threshold, and because the sum of masker contrast and threshold contrast is not constant when the masker's contrast is below the detection threshold.

The frequency selectivity of masking can be examined more carefully in Fig. 3. Data points, representing signal thresholds, have been replotted from Fig. 2 as a function of masking frequency. The masking contrasts parametrize these *masking tuning functions.* The ordinate is relative threshold elevation, the ratio of masked to unmasked signal threshold. The ten symbols represent masked thresholds obtained with ten masking contrasts. Values for 0.8% masking contrast have been omitted for the sake of clarity. An ordinate value of 1.0 means that there is no effect of masking. For very low contrast, there is no effect of masking. For masking contrasts in the range from about 0.1% to 0.8%, there is a *facilitation* or *pedestal* effect with very *narrow* frequency selectivity. As the masking contrasts increase beyond 0.8%, the tuning functions turn inside out and become the more familiar threshold elevation functions. These have medium band-
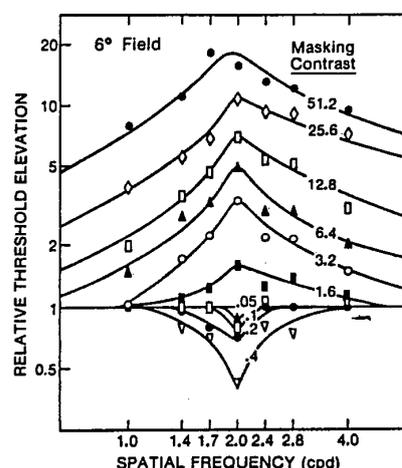

FIG. 3. Masking tuning functions for different masking contrasts: 6° fields. Signals and maskers subtended 6° by 6°. The data in Fig. 2 have been replotted as relative threshold elevation of 2.0 cpd sine-wave grating signals as a function of the spatial frequency of masking gratings. Smooth curves have been drawn through the sets of data. The ten sets of data are for different masking contrasts, as indicated. Data at 0.8% masking contrast have been omitted for clarity. An ordinate value of 1.0 indicates that masking has no effect.
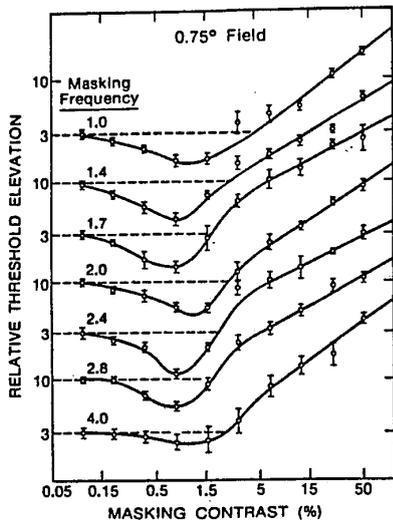
Gordon E. Legge and John M. Foley   1462

FIG. 4. Signal threshold as a function of masking contrast: narrow fields. The signals and maskers subtended 0.75° horizontally by 6° vertically. The seven sets of data are analogous to those of Fig. 3, except that straight lines were fit to the data in the range of 6.4% to 51.2% masking contrast.

widths and are more broadly tuned than the pedestal effect. The high contrast tuning functions are asymmetric, being slightly broader above their peaks than below. This asymmetry in masking tuning functions has been noted elsewhere.[7,8,38,39]

Some studies have found that masking contrasts that increase signal threshold when they are close in frequency to the signal act to reduce signal threshold when masking frequency is about ⅓ the signal frequency.[3,29,38] This phenomenon has been called "remote facilitation." In our experiments, no masker frequency is one-third the signal frequency. However, facilitation is apparent at and near the signal frequency when the masker has a contrast of 0.8% or less. These observations suggest that facilitation is proximal at low contrasts, but becomes remote at high contrasts.

Can a single empirical measure be used to characterize the bandwidth of spatial frequency masking? To the extent that the high contrast portions of the masking functions in Fig. 2 are scaled versions of one another, estimates of the half-maximum bandwidth frequencies in Fig. 3 will be invariant over a range of high masking contrasts. The half-maximum frequencies so obtained lie roughly one octave (or slightly less) above the signal frequency, and about ¾ of an octave below it.

However, such an empirical bandwidth estimate is certainly inadequate for description of the frequency selectivity of the pedestal effect. Its half-minimum frequencies lie roughly ±¼ octave from the signal frequency. This very narrow tuning of the pedestal effect in contrast masking has been noted earlier.[36]

Certainly no single empirical measure can characterize the frequency selectivity of masking over the full range of contrasts used in this study. Nevertheless, the masking model, to be presented below, postulates the existence of a linear filter whose (channel) bandwidth is invariant with contrast.

Narrow field masking

In the narrow field masking experiment, sinusoidal luminance modulation was confined to a horizontal extent of 0.75°. The remainder of the screen was maintained at a mean luminance level of 200 cd/m².

Figure 4 presents 2.0 cpd signal thresholds as a function of masker contrast, for seven masking frequencies. The conventions for plotting the data are like those for Fig. 2. The signals and maskers were always truncated sine-wave gratings in cosine phase with a central fixation mark.

Once again, there is a range of high masking contrasts for which the sets of data are well approximated by straight lines in the double logarithmic coordinates. The lines are approximately parallel and have slopes well below 1.0. (See Table I.) The mean slope is 0.627, very close to the value of 0.620 for wide field masking.

For a range of low masking contrasts, the pedestal effect is evident at all masking frequencies.

In Fig. 5, data points from Fig. 4 are replotted as signal thresholds versus masking frequency. The conventions for plotting the data are like those in Fig. 3. The nine symbols represent masked thresholds obtained with nine masking contrasts. Values for 1.6% masking contrast have been omitted for the sake of clarity. Smooth curves have been drawn through the data at each masking frequency. For masking contrasts below about 2 to 3%, there is a general facilitation effect that appears to extend across the range of masking frequencies used. This broad tuning of the pedestal effect is very different from the extremely narrow tuning of the pedestal effect in wide field masking (see Fig. 3). At higher masking contrasts, the tuning functions turn inside out so that the presence of masking elevates signal thresholds. At high masking contrasts, these tuning functions become more broadly tuned than those in Fig. 3 for wide field masking. There appears to be a tendency for the peak threshold elevation to occur for masking frequencies above the signal frequency. Analogous effects have been observed under rather
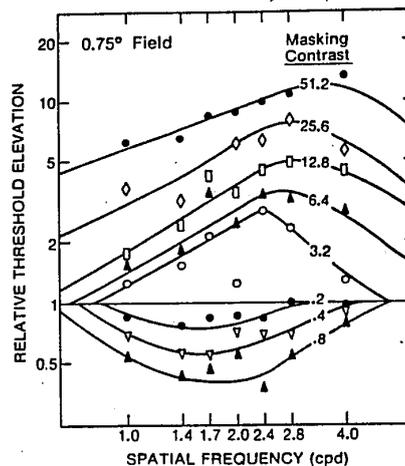

FIG. 5. Masking tuning functions for different masking contrasts: narrow fields. Signals and maskers subtended 0.75° horizontally by 6° vertically. Other details as in Fig. 3. Data at 1.6% masking contrast have been omitted for clarity.

HUAWEI EX. 1116 - 151/714

different spatial frequency masking conditions. Legge[9] observed that masking tuning functions for signals at 0.375 and 0.75 cpd were broader than those at higher signal frequencies, and tended to have their peaks shifted to masking frequencies above the signal frequency. The tuning functions were obtained with gratings subtending 10°. Legge[10] observed similar effects for monocular masking tuning functions at 0.125 and 0.25 cpd, using a 13° field. Legge[9] hypothesized that characteristics of masking tuning functions at low signal frequencies might reflect properties of a transient mechanism, having low-pass spatial frequency sensitivity, coexisting with a set of band-pass sustained mechanisms. The results of the present study, however, suggest that the broadening of masking tuning functions may not be confined to low signal frequencies below 1.0 cpd. Instead, it may be that broadening is related to a reduced number of cycles in the stimulus gratings. The number of cycles and the spatial frequency of signals covary for stimuli of fixed angular subtense. Further research will be required to unravel the separate dependences of the spread of masking upon signal frequency and field subtense.

Implicit in the data of Figs. 2 and 4 is an important difference between wide field and narrow field masking. In Fig. 6, contrast discrimination thresholds (both signal and masker at 2.0 cpd) have been replotted as a function of masker contrast for the wide field condition (●) and for the narrow field condition (*). The two theoretical curves are discussed below. For masking contrasts above about 6.4%, there is very little difference between wide field and narrow field thresholds. For low masking contrasts, below about 0.8%, wide field thresholds fall well below narrow field thresholds. Similarly, contrast detection thresholds (see Individual data) are lower in the wide field condition than in the narrow field condition. A notable peculiarity is that for a narrow range of intermediate contrasts, the narrow field thresholds actually drop below the wide field thresholds.
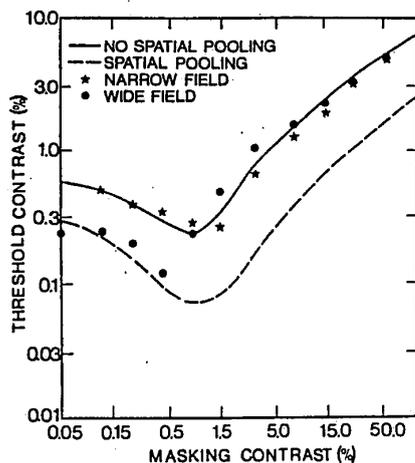
FIG. 6. Contrast discrimination: data and model. Contrast discrimination thresholds, for which both signal and masker are 2.0 cpd sine-wave gratings, have been replotted from Figs. 2 and 4 as a function of contrast. (●) wide field; (*) narrow field. The two curves are derived from the masking model. ——— prediction based on output of a single detector, no spatial pooling. - - - model predictions include spatial pooling over 6°.

The decrease in contrast detection threshold with an increase in the number of grating cycles has been well documented.[36,40-43] The new observation here is that a similar improvement in sensitivity occurs for contrast discrimination at very low contrasts as field size is increased, but that contrast discrimination is relatively insensitive to field size at high contrasts.

CONTRAST MASKING MODEL

There exists no quantitative model of spatial frequency masking. The model proposed here is an extension of the nonlinear transducer model of contrast detection and discrimination of Foley and Legge.[4] The masking model is comprehensive insofar as it treats processes of detection, discrimination, and masking within a single theoretical framework. The development owes much to the earlier work of Nachmias and Sansbury,[2] and Stromeyer and Klein.[3] The model bears similarities to detection and discrimination models in audition (see, e.g., McGill and Goldberg[44] and Hall and Sondhi.[45]) The model is presented in the spirit of a first effort to account for a broad and complex body of masking data. It accounts for most of the diverse results of this paper.

The masking model postulates that detection of a 2.0 cpd sine-wave grating signal is accomplished by an ensemble of spatially localized detectors, differing only in their positions. Each detector's response is determined by three processes—a linear filter characterized by a *spatial frequency sensitivity function* $k'(f)$, a *nonlinear transducer* F, and *additive, zero-mean, constant-variance Gaussian noise*. The model will be fully characterized by specifying: (i) the function $k'(f)$; (ii) the functional F; and (iii) the "decision rules" by which detector outputs are combined to determine a response in the forced-choice procedure. Figure 7 is a schematic representation of the model. In the following subsections, $k'(f)$ is identified with the sensitivity function plotted in Fig. 8(a), and the nonlinear transducer F is shown to have the form given in Fig. 8(b). The model's decision rules incorporate a form of spatial pooling at low contrasts, but not at high contrasts.

Linear filter

Recent theoretical treatments have attributed phenomena of spatial frequency selectivity in vision to hypothetical spatial weighting functions or "receptive fields" associated with spatially localized detectors.[3,36,46-48] Legge[36] has modeled grating detection by an ensemble of such detectors, distributed across the visual field. The ensemble, all members of which have the same spatial weighting function, was termed a *spatial frequency channel*. The spatial frequency selectivity associated with the channel is characterized by the Fourier transform of the spatial weighting function. For a detector centered at position x_0 in visual coordinates, having spatial weighting function $S(x)$, the output r_0, associated with stimulus waveform $L(x)$, is

$$r_0[L(x)] \propto \int_{-\infty}^{\infty} L(x)S(x - x_0)dx.$$

This convolution is linear, and constitutes the *linear filter* of the masking model. For a sine-wave grating of frequency f in cosine phase with the origin of coordinates, the stimulus
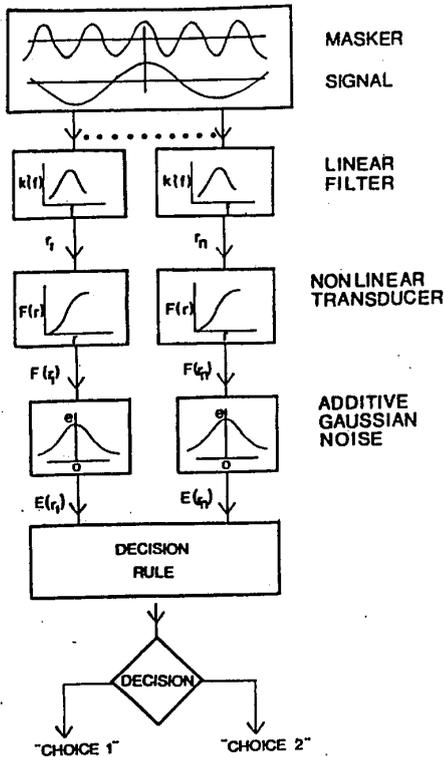
Gordon E. Legge and John M. Foley   1464

FIG. 7. Schematic representation of the masking model.

waveform $L(x)$ is

$$L(x) = L_0(1 + C\cos2\pi f x), \qquad (3)$$

where L_0 is mean luminance and C is the grating contrast. If the detector is assumed to be insensitive to mean luminance,[49] its dependence on the grating frequency f is found from Eqs. (2) and (3) to be

$$r_0(f) \propto C \int_{-\infty}^{\infty} \cos2\pi f x S(x - x_0)dx. \qquad (4)$$

Assuming that $S(x)$ is center symmetric and of finite extent,[50] we obtain

$$r_0(f) \propto C\cos2\pi f x_0 \left[\int_{-\infty}^{\infty} \cos2\pi f x S(x)dx \right] \qquad (5)$$

$$= C\cos2\pi f x_0 k'(f)$$

where the bracketed quantity has been rewritten as $k'(f)$ and is the Fourier transform of the symmetric function $S(x)$. Equation (5) indicates that the output of the linear filter, resulting from a stimulus grating of frequency f, is proportional to the product of three factors—grating contrast C, the value of the sensitivity function $k'(f)$ at f, and a phase-sensitive factor $\cos2\pi f x_0$ that depends on the position x_0 of the detector.

The filter output for a detector located at the fixation point, $x_0 = 0$, is simply $Ck'(f)$. The contrast discrimination data in Fig. 6 suggest that the effects of spatial pooling are largely absent at high contrasts. Frequency selectivity of masking

at high contrast may, therefore, predominantly reflect the filtering characteristics of detectors located at or very near the fixation point. It has already been shown that, for a range of high contrasts, the masking functions of Fig. 2 may be approximated by scaled versions of one another. This frequency-dependent scaling is characterized in a contrast-invariant manner by the sensitivity function of Fig. 8(a). In the context of the masking model, this spatial frequency sensitivity function can be identified as the function $k'(f)$ associated with the linear filter. The filter attenuates signal inputs to the detectors in accordance with Eq. (5).

(b)

(a)
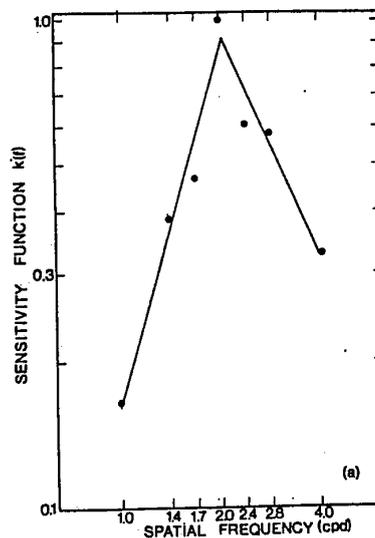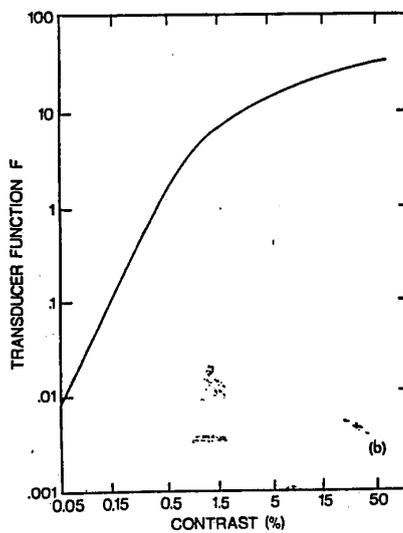
FIG. 8. (a) Spatial frequency sensitivity function used in the masking model. The scale factors $k(f)$ (Table I) obtained from high contrast masking are plotted in normalized form as a function of frequency. Straight lines have been fit through the points to the left and right of 2.0 cpd for purposes of interpolation and extrapolation. (b) Nonlinear transducer used in the masking model. F is the nonlinear transducer of the masking model: $F(r) = (a_1|r|^{2.4})/(|r|^2 + a_2^2)$ where $a_1 = 45$ and $a_2 = 0.0075$. F is plotted for the case in which $r = C$—detector located at x_0 responding to a 2.0 cpd sine-wave grating.

Gordon E. Legge and John M. Foley   1465

More generally, the spectral representations of stimuli of finite width will be continuous functions of frequency. The output of the linear filter for a detector centered at x_0 generalizes from Eq. (5) to[51]

$$r_0 \propto \int_{-\infty}^{\infty} C(f)k'(f)\cos2\pi f x_0 df, \qquad (6)$$

where $C(f)$ is the contrast spectrum of the stimulus. The spectral spread of the stimulus becomes important when modeling the narrow field results for which gratings were truncated to widths of 0.75°. The fitted straight lines to the points of Fig. 8(a) approximate $k'(f)$. This approximation is used in the numerical evaluation of the integral in Eq. (6).

It is known that contrast sensitivity decreases with increasing retinal eccentricity.[52-54] Effects of retinal inhomogeneity were probably slight in the current experiments. The wide field gratings extended only 3° on either side of the fixation point. Nevertheless, in the masking model, the filter output of each detector was multiplied by the factor $\exp(-0.116|x_0|)$. This function, taken from Wilson and Giese,[54] describes the decline in sensitivity with retinal eccentricity for 2.0 cpd gratings. Inclusion of this factor in the masking model had only slight effects upon its performance.

Nonlinear transducer

In the masking model, the output of the linear filter associated with a detector is subjected to a nonlinear transformation, denoted F. If the output of the linear filter for a detector at position x_0 is r_0, the corresponding output of the nonlinear transducer is $F(r_0)$.

Foley and Legge[4] used contrast detection data to infer a nonlinear transducer of the form

$$F = aC^n, \qquad (7)$$

where a and n are constants, and C is the grating contrast. They found values of the exponent n that ranged from 2.11 to 3.04, depending on observer and spatial frequency. A positively accelerating nonlinear transducer of this type accounts for contrast detection and discrimination data over a range of low contrasts. With increasing contrast C, fixed increments in transducer output are associated with ever decreasing increments in transducer inputs. This property of the nonlinear transducer can be used to model the facilitation that occurs with low contrast discrimination. However, the nonlinear transducer of Eq. (7) will not work at high contrasts, because it predicts that threshold will continue to decrease with increasing masking contrast. The choice of nonlinear transducer to use in the masking model is[55]

$$F = a_1|r|^{2.4}/(|r|^2 + a_2^2), \qquad (8)$$

where r is the input to the nonlinear transducer, and a_1 and a_2 are constants. This equation is plotted in Fig. 8(b) in log-log coordinates, with values of the constants computed as discussed below. For small inputs $|r| < a_2$, the transducer function reduces to $F = (a_1/a_2^2)|r|^{2.4}$. This form of the transducer is compatible with the transducer inferred by Foley and Legge,[4] Eq. (7). For large inputs $|r| > a_2$, the transducer has the form $F = a_1|r|^{0.4}$. At high contrasts, the transducer is compressive, but nonsaturating. In the compressive region, greater and greater input increments are required to achieve

equal output increments. In the case of contrast discrimination, the empirical consequence is that signal threshold increase with increasing contrast.

Notice that the constant a_2 in Eq. (8) determines the range of inputs r that lie in the positively accelerating and compressive regions of the nonlinearity. If a_2 were 0, there would be no positively accelerated portion of the nonlinearity and consequently, no dip in the masking functions of Figs. 2 and 4. a_1 is simply a constant of proportionality.

Noise process

Observers do not always give the same response in identical forced-choice trials. To account for this fact, it is assumed that noise is added to the output of the nonlinear transducer. The noise is Gaussian, zero-mean, with constant, unit variance,[57] and is uncorrelated with the noise in other detectors. The inclusion of noise means that the output of a detector must be regarded as a random variable. Let the output of detector, associated with a linear filter response r, be $E(r)$:

$$E(r) = F(r) + e. \qquad (9)$$

F is the nonlinear transducer; e is the additive, zero-mean, unit-variance Gaussian noise; the mean of the random variable $E(r)$ is $F(r)$, and its constant standard deviation is 1.0.

Decision process

An observer's response in a forced-choice trial depends on a decision rule. The decision rule may apply to the output of a single detector or may be based on the outputs of a set of independent detectors. The way in which outputs from different detectors are combined to produce a decision will now be discussed.

In a forced-choice trial, the observer is presented with one interval containing the signal-plus-masker and another interval containing the masker alone. Consider a detector centered at x_i. If its linear filter responses to signal and masker are r_{s_i} and r_{m_i}, respectively, its outputs in the two intervals of the trial are $E(r_{s_i} + r_{m_i})$ and $E(r_{m_i})$. For every detector, it is assumed that each observation interval is associated with only one value of the output variable E. These values are stochastically independent.

First, consider the case in which an observer's decision is based upon the output of a single detector. Let the observer's choice be governed by the following decision rule:

DECISION RULE: When the decision in a forced-choice trial is based upon the output of a single detector, choose the interval in which the value of the output

variable E is greater. (10)

This rule implies that the observer computes the difference $E(r_s + r_m) - E(r_m)$ for the detector. If the signal-plus-masker interval produces the larger output, his choice will be correct. If the interval containing the masker alone produces the larger value of E, his choice will be incorrect.

Now, consider the case of many detectors. Data of Fig. 6 suggest that field size is much more important to contrast discrimination at low contrasts than at high contrasts. Apparently, at low contrasts, observers can base decisions on information collected over a wide portion of the visual field.

...

have been several investigations of spatial pooling associated with contrast detection.[16,17,36,58–62] The most popular model has been termed *probability summation*. Probability summation has been well developed for contrast detection. However, it is not applicable, without considerable refinement, to contrast discrimination or masking.

As an alternative model of spatial pooling that is applicable to contrast detection, discrimination, and masking, the decision rule for one detector may be extended to many detectors, as follows:

DECISION RULE: When the decision in a forced-choice trial is based upon the outputs of many detectors, identify the detector whose outputs in the two intervals have the greatest difference. Choose the interval in which this detector's output is greater. (11)

This rule implies that the observer computes the difference $E(r_{s_i} + r_{m_i}) - E(r_{m_i})$ for each detector, and then bases his decision upon the detector for which this difference has the largest magnitude.

Many decision rules are possible that include versions of spatial pooling. Several decision rules, including linear and quadratic summation across detector outputs, were studied in attempts to model the masking data of this paper. None were found to be as adequate as the decision rule given here, although the differences were often small.

The contrast discrimination data of Fig. 6 suggest that the effects of spatial pooling are primarily confined to low contrasts.[63] For purposes of modeling, the following decision rules were adopted: (i) At low masking contrasts, the forced-choice decision is based upon the output of many detectors, according to the decision rule in Eq. (11); (ii) at high contrasts, the forced-choice decision is based upon the output of a single detector, located at the fixation point. The switch in decision rules occurs at a contrast of about 1%.

In the case of spatial pooling, the following simplification was made for purposes of calculation. Only detectors centered on peaks and valleys of the 2.0 cpd signal grating were included in the decision process.[64]

The properties of the masking model were studied with a computer program simulation. For a given signal and masker, a forced-choice trial was simulated by the following steps: (i) A set of detectors was chosen by specifying their center positions x_i in the visual field; (ii) the linear filter response of each detector was calculated for the masker and signal-plus-masker from Eq. (6), and weighted by the retinal inhomogeneity factor; (iii) for each detector, the linear filter outputs were subjected to the nonlinear transformation of Eq. (8); (iv) a value drawn from a zero-mean, unit-variance Gaussian random process was added to the output of each nonlinear transformation, Eq. (9); (v) for each detector, the difference in signal-plus-masker and masker outputs was computed, and the forced-choice was determined by the decision rule given in either (10) or (11). For a given masker and signal, 250 trials were simulated, and the percent correct calculated. The staircase procedure used in the experiments of this paper estimated the signal contrast that yielded 79% correct. Accordingly, the computer simulation repeated its computations for signal contrasts until the corresponding values of percent correct ranged from 60% to 90%. From these simulated

"frequency-of-seeing" curves, the 79% signal contrast was obtained. Although the Gaussian noise component of the simulation means that the model's predictions have some variability, repetitions indicated that its threshold estimates are good to better than 5%.

The masking model has now been fully specified, except for evaluation of the constants a_1 and a_2 in Eq. (8). As noted earlier, at high contrasts, the nonlinear transducer reduces to $a_1|r|^{0.4}$ and is independent of a_2. The value of a_1 was chosen by simulating contrast discrimination at 51.2% contrast. A value of $a_1 = 45$ was chosen because it yielded a threshold of about 4.5% in agreement with observation. Similarly, $a_2 = 0.0075$ was chosen so that the model's predictions for contrast detection thresholds for wide fields would approximate closely the observed threshold of 0.3%.

The model may now be used to predict signal thresholds for contrast detection, discrimination, and masking. The model is characterized by a linear spatial frequency filter [Fig. 8(a)], a nonlinear transducer [Fig. 8(b)], separate decision rules for low and high contrast masking, and the constants a_1 and a_2.

PREDICTIONS OF THE MASKING MODEL

Contrast discrimination

Two theoretical curves accompany the contrast discrimination data in Fig. 6. The solid curve, labeled NO SPATIAL POOLING, was derived from the masking model by applying the single detector decision rule, Eq. (10), across the full range of contrasts. The broken line, labeled SPATIAL POOLING, is the masking model's prediction based upon the decision rule, Eq. (11), that incorporates spatial pooling over the 6° field. A third theoretical curve, not shown, incorporates effects of spatial pooling in the narrow field case. It lies very close to the NO SPATIAL POOLING curve at low contrasts, and drops well below it at high contrasts. The two theoretical curves in Fig. 6 indicate that the NO SPATIAL POOLING predictions fit the narrow field data at all contrasts, and the wide field data at high contrasts. On the other hand, spatial pooling leads to predictions that provide a reasonably good fit to the wide field data at low contrasts. Apparently, low contrast discrimination at 2.0 cpd involves some form of spatial pooling, but high contrast discrimination does not.

When the spatial pooling decision rule is used, the masking model predicts contrast detection thresholds of 0.31% and 0.55% for wide and narrow field gratings, respectively, compared with measured values (means across three observers) of 0.29% and 0.50%. The predicted psychometric functions for detection have the shape found experimentally by Foley and Legge.[4]

Masking tuning functions at high contrast

Figure 9 presents the masking model's tuning functions (solid curves) for masking contrasts of 51.2% and 25.6%. Panel (a) presents wide field results and panel (b) narrow field results. Data points are replotted from Figs. 3 and 5.

The model's predictions fit the wide field data quite well, reflecting the broad and asymmetric form of the tuning functions. The model's predictions for all masking contrasts in the range 3.2% to 51.2% are similar in shape, and give
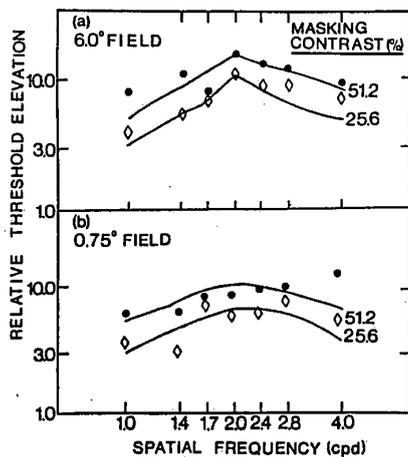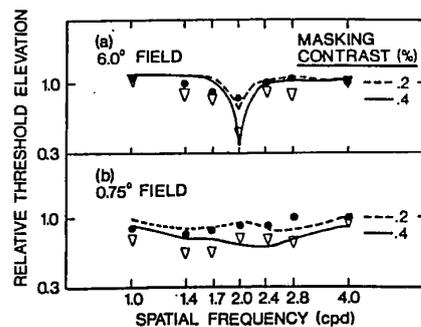
FIG. 9. Masking tuning functions at high contrast: data and model. Solid curves are predictions of the masking model for 2.0 cpd signal threshold elevation plotted as a function of the spatial frequency of masking gratings of 51.2% and 25.6% contrast. (a) Wide field: 6° by 6°, data replotted from Fig. 3. (b) Narrow field: 0.75° by 6°, data replotted from Fig. 5.

FIG. 10. Masking tuning functions at low contrast: data and model. Solid curves are predictions of the masking model for 2.0 cpd signal threshold reduction plotted as a function of the spatial frequency of masking gratings. (a) Wide field: 6° by 6°, data replotted from Fig. 3. (b) Narrow field: 0.75° by 6°, data replotted from Fig. 5.

comparably good approximations to the data. It is interesting to note that the channel sensitivity function, Fig. 8(a), is considerably narrower than the spread of masking predicted by the model. This difference can be traced to the form of the nonlinearity that follows the stage of linear filtering.[65] Clearly, the relation between the theoretical channel sensitivity and the empirical frequency selectivity of masking depends crucially on the model adopted to relate them.

The model's predicted tuning functions are broader for narrow field masking than for wide field masking, in conformity with the data. The broadening can be traced to the spectral broadening of grating stimuli that are truncated to 0.75°. The model gives a good account of the data at the masking contrast of 51.2%, except for the rightmost point. A possible source for the discrepancy between model and data is the estimate of the channel sensitivity function, Fig. 8(a). If the channel were to have a broad, high frequency tail, beyond 4.0 cpd, model predictions based on extrapolation of the straight lines in Fig. 8(a) would lead to low threshold estimates at high masking frequencies. Notice how peak masking shifts to frequencies above the signal frequency for high contrast, narrow field masking. The model predicts a slight peak shift of this sort, arising because the spectrum of the truncated 2.4 cpd masking stimulus actually exhibits slightly greater overlap with the channel sensitivity function than does the spectrum of the truncated 2.0 cpd stimulus grating.

Masking tuning functions at low contrast

Figure 10 presents the masking model's tuning functions (solid curves) for low masking contrasts. Panel (a) presents wide field predictions for masking contrasts of 0.2% and 0.4%. Panel (b) presents narrow field results for masking contrasts of 0.2% and 0.4%. Data points have been replotted from Figs. 3 and 5.

For the wide field, the model predicts the very narrow tuning of the facilitation effect. In part, the narrow tuning is a reflection of a form of phase sensitivity inherent in the

model. When signal and masker are both in cosine phase with the fixation mark at the center of the display, but have different spatial frequencies, their relative phases will differ everywhere except at points separated by distances equal to the period of the beat frequency. Detectors that are optimally stimulated by the signal grating are centered at the peaks and troughs of the signal grating. These detectors will be stimulated by maskers whose relative phase is often nonoptimal for stimulating the detector. When the masker and the signal have nearly the same spatial frequency, the masker acts to lower signal threshold. In the context of the model, this facilitation is a consequence of the positively accelerated form of the nonlinear transducer at low contrasts.

For narrow field masking at low contrasts, the model predicts very broad tuning of the facilitation effect. The broadening occurs because the phase selective narrowing due to spatial pooling has been removed and because stimulus spectra have been broadened. Note that both the data and the model's predictions are so broadly tuned that frequency sensitivity is hardly apparent. Comparison of the upper and lower panels of Fig. 10 shows the very large influence that field size has upon the frequency selectivity of the facilitation effect. For wide fields, upper panel, the frequency tuning is extremely narrow. For narrow fields, lower panel, the tuning is extremely broad. The masking model also possesses these properties.

A rather strong assumption of the masking model is that decisions concerning the presence or absence of the 2.0 cpd signal always result from decision rules that are based on the outputs of detectors that are optimally sensitive at 2.0 cpd. This is an assumption of parsimony. Instead, it may be that observers refer their decisions to different sets of detectors for different masking conditions. Limitations of the masking model in describing the data may result from a violation of this assumption.

Both the data and model exhibit different properties at low and high contrast. In terms of the model, low contrast processing is characterized by a linear filter, followed by an accelerating nonlinear transducer, followed by spatial pooling across detectors. Empirical consequences are that masking facilitates signal detection, is narrowly tuned in frequency, but sensitive to field size. High contrast processing is char-

rized by the same linear filter, followed by a compressive linearity, and the relative absence of spatial pooling. pirically, high contrast masking elevates signal threshold, ibits medium to broad tuning, and is relatively insensitive field size. The striking differences between masking at low high contrasts may reflect the properties of separate, derlying mechanisms. Accelerating and compressive tions of the nonlinear transducer may be descriptive of arate mechanisms. Foley and Legge[4] have suggested that accelerating portion of the transducer may be the result a threshold process followed by the addition of Gaussian ise. By comparison, the compressive portion of the nonlar transducer may reflect the operation of signal compression in the transmission of high amplitude signals along ual pathways.

Why is spatial pooling less prominent at high contrasts? he possible reason involves the properties of the Gaussian ise that is added to the transducer output of each detector. , as assumed in the masking model, the additive noise is correlated across detectors, the decision rule in (11) results spatial pooling. If, however, the Gaussian noise were perctly correlated across detectors, no spatial pooling would be resent. This account has some phenomenological plausiility. Observers sometimes report that for very low conrasts, grating detection involves "seeing something" at different places in the stimulus field on different trials, as if local esponse fluctuations were occurring. At high contrasts, however, observers do not seem to report seeing patches of a grating fluctuating in contrast. But why should the detector noise become correlated at high contrast? Although we have assumed constant-variance additive noise in the current model, further study may indicate that the noise is signal dependent, in which case some correlation might not be surprising.

An important feature of the masking model is the ordering of its elements. The linear filter *precedes* the nonlinear transducer. Burton[66] attributed visual "beats" to a frequency filter that followed a nonlinearity. The possibility exists that spatial frequency filtering occurs both before and after the nonlinearity. However, there are two compelling reasons for placing a stage of linear filtering *before* the nonlinearity. First, it would be difficult to account for the fact that the high contrast portions of the masking functions in Figs. 2 and 4 are approximately parallel without assuming that the scaling operation (linear filtering) occurs prior to the nonlinearity. Second, evidence that pattern detection thresholds can be predicted from a knowledge of pattern Fourier spectra[67-69] requires that a stage of linear filtering precedes the nonlinearity.

ACKNOWLEDGMENTS

[1] R. Fox, "Visual masking," in *Handbook of Sensory Physiology. VIII. Perception*, edited by R. Held, H. W. Leibowitz, and H.-L. Teuber (Springer-Verlag, Berlin, 1978).
[2] J. Nachmias and R. V. Sansbury, "Grating contrast: discrimination may be better than detection," Vision Res. 14, 1039–1042 (1974).
[3] C. F. Stromeyer, III and S. Klein, "Spatial frequency channels in human vision as asymmetric (edge) mechanisms," Vision Res. 14, 1409–1420 (1974).
[4] J. M. Foley and G. E. Legge, "Contrast detection and near-threshold discrimination in human vision," Vision Res. (in press).
[5] D. M. Green, *An Introduction to Hearing* (Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1976).
[6] U. Greis and R. Röhler, "Untersuchung der subjektiven Detailerkennbarkeit mit Hilfe der Ortsfrequenzfilterung," Opt. Acta 17, 515–526 (1970).
[7] C. F. Stromeyer, III and B. Julesz, "Spatial frequency masking in vision: critical bands and spread of masking," J. Opt. Soc. Am. 62, 1221–1232 (1972).
[8] A. Pantle, *Visual Information Processing of Complex Imagery* (Aerospace Medical Research Laboratory Report AMRL-TR-74-43, 1974).
[9] G. E. Legge, "Sustained and transient mechanisms in human vision: temporal and spatial properties," Vision Res. 18, 69–81 (1978).
[10] G. E. Legge, "Spatial frequency masking in human vision: binocular interactions," J. Opt. Soc. Am. 69, 838–847 (1979).
[11] A. Pantle and R. Sekuler, "Size detecting mechanisms in human vision," Science 162, 1146–1148 (1968).
[12] C. Blakemore and F. W. Campbell, "On the existence of neurones in the human visual system selectively sensitive to the orientation and size of retinal images," J. Physiol. (London) 203, 237–260 (1969).
[13] K. K. De Valois, "Spatial frequency adaptation can enhance contrast sensitivity," Vision Res. 17, 1057–1065 (1977).
[14] M. B. Sachs, J. Nachmias, and J. G. Robson, "Spatial frequency channels in human vision," J. Opt. Soc. Am. 61, 1176–1186 (1971).
[15] R. F. Quick, Jr. and T. Reichert, "Spatial frequency selectivity in contrast detection," Vision Res. 15, 637–643 (1975).
[16] C. F. Stromeyer, III and S. Klein, "Evidence against narrow-band spatial frequency channels in human vision: the detectability of frequency-modulated gratings," Vision Res. 15, 899–910 (1975).
[17] N. Graham, "Visual detection of aperiodic spatial stimuli by probability summation among narrow-band channels," Vision Res. 17, 637–652 (1977).
[18] R. F. Quick, Jr., W. W. Mullins, and T. A. Reichert, "Spatial summation effects on two-component grating thresholds," J. Opt. Soc. Am. 68, 116–121 (1978).
[19] D. J. Tolhurst, "Adaptation to square-wave gratings: inhibition between spatial frequency channels in the human visual system," J. Physiol. (London) 226, 231–248 (1972).
[20] R. S. Dealy and D. J. Tolhurst, "Is spatial frequency adaptation an aftereffect of prolonged inhibition?" J. Physiol. (London) 241, 261–270 (1974).
[21] C. F. Stromeyer, III, S. Klein, and C. E. Sternheim, "Is spatial adaptation caused by prolonged inhibition?" Vision Res. 17, 603–606 (1977).
[22] D. J. Tolhurst, "Separate channels for the analysis of the shape and the movement of a moving visual stimulus," J. Physiol. (London) 231, 385–402 (1973).
[23] F. W. Campbell and D. G. Green, "Optical and retinal factors affecting visual resolution," J. Physiol. (London) 181, 576–593 (1965).
[24] G. B. Wetherill and H. Levitt, "Sequential estimation of points on a psychometric function," Br. J. Math. Stat. Psychol. 18, 1–10 (1965).
[25] C. Blakemore and J. Nachmias, "The orientation specificity of two visual aftereffects," J. Physiol. (London) 213, 157–174 (1971).
[26] C. Blakemore, J. P. J. Muncey, and R. Ridley, "Stimulus specificity

Gordon E. Legge and John M. Foley   1469

HUAWEI EX. 1116 - 157/714

in the human visual system," Vision Res. 13, 1915–1931 (1973).

[27]F. W. Campbell and J. J. Kulikowski, "Orientational selectivity of the human visual system," J. Physiol. (London) 187, 437–445 (1966).

[28]I. Bodis-Wollner, C. D. Hendley, and J. J. Kulikowski, "Electrophysiological and psychophysical responses to modulation of contrast of a grating pattern," Perception 1, 341–349 (1972).

[29]D. J. Tolhurst and L. P. Barfield, "Interactions between spatial frequency channels," Vision Res. 18, 951–958 (1978).

[30]G. E. Legge, "In search of Weber's law for contrast discrimination," Suppl. Invest. Ophthalmol. Vis. Sci. 19, 43 (1980).

[31]I. Bodis-Wollner, C. D. Hendley, and M. Tajfel, "Contrast-modulation thresholds as a function of spatial frequency," J. Opt. Soc. Am. 63, 1297 (1973).

[32]J. J. Kulikowski, "Effective contrast constancy and linearity of contrast sensation," Vision Res. 16, 1419–1432 (1976).

[33]J. J. Kulikowski and A. Gorea, "Complete adaptation to pattern stimuli: a necessary and sufficient condition for Weber's law for contrast," Vision Res. 18, 1223–1227 (1978).

[34]J. Nachmias and E. C. Kocher, "Visual detection and discrimination of luminance increments," J. Opt. Soc. Am. 69, 382–389 (1970).

[35]T. E. Cohn, L. N. Thibos, and R. N. Kleinstein, "Detectability of a luminance increment," J. Opt. Soc. Am. 64, 1321–1327 (1974).

[36]G. E. Legge, "Space domain properties of a spatial frequency channel in human vision," Vision Res. 18, 959–969 (1978).

[37]J. J. Kulikowski and P. E. King-Smith, "Spatial arrangement of line, edge, and grating detectors revealed by subthreshold summation," Vision Res. 13, 1455–1478 (1973).

[38]J. Nachmias and A. Weber, "Discrimination of simple and complex gratings," Vision Res. 15, 217–223 (1975).

[39]R. V. Sansbury, Some Properties of Spatial Channels Revealed by Pulsed Simultaneous Masking, Ph.D. dissertation, Dept. of Psychology, University of Pennsylvania, Philadelphia, 1974 (unpublished).

[40]J. Hoekstra, D. P. J. van der Goot, G. van den Brink, and F. A. Bilsen, "The influence of the number of cycles upon the visual contrast threshold for spatial sine-wave patterns," Vision Res. 14, 365–368 (1974).

[41]R. L. Savoy and J. J. McCann, "Visibility of low-spatial-frequency sine-wave targets: dependence on number of cycles," J. Opt. Soc. Am. 65, 343–350 (1975).

[42]O. Estevez and C. R. Cavonius, "Low-frequency attenuation in the detection of gratings: sorting out the artifacts," Vision Res. 16, 497–500 (1976).

[43]J. J. McCann, R. L. Savoy, and J. A. Hall, Jr., "Visibility of low-frequency sine-wave targets: dependence on number of cycles and surround parameters," Vision Res. 18, 891–894 (1978).

[44]W. J. McGill and J. P. Goldberg, "A study of the near-miss involving Weber's law and pure tone intensity discrimination," Percept. Psychophys. 7, 105–109 (1968).

[45]J. H. Hall and M. M. Sondhi, "Detection threshold for a two-tone complex," J. Acoust. Soc. Am. 62, 636–640 (1977).

[46]J. P. Thomas, "Model for function of receptive fields in human vision," Psychol. Rev. 77, 121–134 (1970).

[47]I. D. G. Macleod and A. Rosenfeld, "The visibility of gratings: spatial frequency channels or bar detecting units?" Vision Res. 14, 909–915 (1974).

[48]H. R. Wilson and J. R. Bergen, "A four mechanism model for threshold spatial vision," Vision Res. 19, 19–32 (1979).

[49]The spatial frequency sensitivity function of Fig. 8(a) will be identified with the linear filter. Extrapolation of the fitted straight line toward 0 cpd (uniform field) suggests little sensitivity to mean luminance.

[50]In Eq. (4), let $x' = x - x_0$. Then:

$$r_0(f) = C \int_{-\infty}^{\infty} \cos 2\pi f(x' + x_0) S(x') dx'$$

$$= C \int_{-\infty}^{\infty} (\cos 2\pi f x' \cos 2\pi f x_0 - \sin 2\pi f x' \sin 2\pi f x_0) S(x') dx'$$

$$= C \cos 2\pi f x_0 \int_{-\infty}^{\infty} \cos 2\pi f x' S(x') dx'$$

$$- C \sin 2\pi f x_0 \int_{-\infty}^{\infty} \sin 2\pi f x' S(x') dx'.$$

Assuming S is even-symmetric and finite, the second term is 0, and we are left with Eq. (5).

[51]The only restriction upon Eq. (6) is that the spatial sensitivity function $S(x)$ be center-symmetric and finite. All "phase effects" due to the position of the detector are accounted for by the factor $\cos 2\pi f x_0$.

[52]R. Hilz and C. R. Cavonius, "Functional organization of the peripheral retina: sensitivity to periodic stimuli," Vision Res. 14, 1333–1337 (1974).

[53]J. J. Koenderink, M. A. Bouman, A. E. Bueno de Mesquita, and S. Slappendell, "Perimetry of contrast detection thresholds of moving spatial sinewave patterns. I. The near peripheral visual field," J. Opt. Soc. Am. 68, 845–849 (1978).

[54]H. R. Wilson and S. C. Giese, "Threshold visibility of frequency gradient patterns," Vision Res. 17, 1177–1190 (1977).

[55]Fechner (see Boring,[56] Chap. 14) derived a sensory magnitude function from discrimination data. If the effects of spatial pooling are ignored, Fechner's technique may be used to derive the transducer function F from the discrimination functions in Figs. 2 or 4. The discrimination function is proportional to the reciprocal of the derivative of F.

[56]E. G. Boring, A History of Experimental Psychology (Appleton-Century-Crofts, New York, 1957).

[57]If the constant variance assumption is relaxed, the transducer function cannot be inferred directly from forced-choice data. Information concerning the dependence of noise variance on contrast could be obtained from other psychophysical procedures, such as the "bootstrap" procedure of Nachmias and Kocher.[34] With this information in hand, a corrected form of the transducer function could be computed. Changes in variance with contrast would be expected to affect the shape of the channel sensitivity function, Fig. 8(a). There exists little evidence concerning the dependence of variance on stimulus contrast. Paired comparison data of Foley and Legge[4] indicate small changes in variance over a range of low contrasts. Further discussion of the effects of relaxing the constant variance assumption is given by Nachmias and Sansbury.[2]

[58]N. Graham and B. E. Rogowitz, "Spatial pooling properties deduced from the detectability of FM and quasi-AM gratings: a reanalysis," Vision Res. 16, 1021–1026 (1976).

[59]P. E. King-Smith and J. J. Kulikowski, "The detection of gratings by independent activation of line detectors," J. Physiol. (London) 247, 237–271 (1975).

[60]H. Mostafavi and D. J. Sakrison, "Structure and properties of a single channel in the human visual system," Vision Res. 16, 957–968 (1976).

[61]H. R. Wilson, "Quantitative prediction of line spread function measurements: implications for channel bandwidths," Vision Res. 18, 493–496 (1978).

[62]H. R. Wilson, "Quantitative characterization of two types of line-spread function near the fovea," Vision Res. 18, 971–981 (1978).

[63]C. F. Stromeyer, III has pointed out an interesting property of the contrast discrimination data of Fig. 6. If the narrow field data points are moved downward by a factor of 2 and to the left by a factor of 2, they are very nearly superimposed upon the wide field data. This result means that contrast sensitivity for the narrow field stimuli is a factor of 2 lower than contrast sensitivity for the wide field stimuli.

[64]If the sampling density is doubled, center-symmetric detectors will be included that are located at zero crossings of the signal. Since these detectors will be insensitive to the signal, they will add only noise, resulting in a small reduction in the improvement of sensitivity due to spatial pooling. The effect upon the masking model is to elevate slightly its predictions for wide field masking at low contrasts [Fig. 6, dashed curve, and Fig. 10(a)]. Refinement of the masking model to include signal-dependent noise and/or asymmetric receptive fields would reduce such sampling effects.

[65]If spatial pooling and spectral effects are ignored, a logarithmic transducer results in Weber's law for discrimination, and masking tuning functions that match the shape of the linear filter function. If the nonlinearity is a power function with exponent n, the properties of masking will depend on the value of n. (i) For 0 < n < 1, there is threshold elevation and the masking tuning functions are broader than the filter function. (ii) For n = 1, signal thresholds are unaffected by the masker. (iii) For 1 < n < 2, masking produces facilitation, but the tuning functions are broader than the filter function. (iv) For n > 2, masking produces facilitation and the tuning functions are narrower than the filter function.

[66]G. J. Burton, "Evidence for nonlinear response processes in the

human visual system from measurements of the thresholds of spatial beat frequencies," Vision Res. 13, 1211–1225 (1973).
[67] W. Campbell and J. G. Robson, "Applications of Fourier analysis to the visibility of gratings," J. Physiol. (London) 197, 551–566 (1968).

[68]F. W. Campbell, R. H. S. Carpenter, and J. Z. Levinson, "Visibility of aperiodic patterns compared with that of sinusoidal gratings," J. Physiol. (London) 204, 283–298 (1969).
[69]M. Hines, "Line spread function variation near the fovea," Vision Res. 16, 567–572 (1976).

Equations for chromatic discrimination models

Robert M. Boynton and John J. Wisowaty

Department of Psychology, University of California at San Diego, La Jolla, California 92093

Equations are derived to fit the cone sensitivity values of Smith and Pokorny, which are based mainly upon Judd's modified color matching functions and the spectral sensitivities of normal and dichromatic observers. The equations, which fit Smith and Pokorny's values to within 1 to 3%, are smooth and differentiable. Derivatives of cone sensitivities for an equal luminance spectrum, which are of particular importance for models of chromatic discrimination, have been calculated and are shown graphically.

INTRODUCTION

The initial nonoptical basis for seeing two colors as different lies in differential signals from the three classes of cone photoreceptors that populate the human retina. No matter what assumptions are made about the subsequent processing of visual signals, any reasonable model of chromatic discrimination must begin by considering these receptor differentials. To do so correctly requires precise information about the action spectra of the R, G, and B cones.

By the end of the nineteenth century, König and Dieterici[1] had already proposed a set of cone fundamentals consistent both with color mixture data and certain facts about red-green dichromacies considered as loss systems. Two similar sets, based on modern data, have appeared within the last decade. The first was published in 1971 by Vos and Walraven[2]; the second, in 1975, by Smith and Pokorny.[3] For reasons that will now be given, we have elected to fit equations to the second set.

The idea of "chromatic discrimination" implies equal luminance for the fields being compared. Experimentally, this requires that a method be developed for holding luminance constant as chromaticity is varied. Theoretically, this operation implies that only signals in the opponent-color channels be allowed to vary, while those in the achromatic visual channels are held constant. Widespread agreement exists that the B cones contribute little to luminance. Recent experimental data[4] strongly support the limiting version of this concept which states that B cones do not contribute to luminance at all. We have chosen the Smith-Pokorny functions because they are explicitly based upon this null assumption.[5]

A limitation of the Smith-Pokorny fundamentals is that they are specified only at 10-nm intervals, having been derived from Judd's modified CIE color mixture data, which were similarly given.[6] These then are not really functions, but rather tabled values, from which interpolations are required

to obtain sensitivities at intermediate wavelengths. Since wavelength differences of as little as 1 nm may be perceptible, the use of such tables is awkward for constructing models of chromatic discrimination.

Simple interpolations are not satisfactory. For example, if a curve is fit to a plot of the Smith-Pokorny values, using a cubic spline routine that constrains the fitted curve to pass exactly through the plotted points, the resulting curve looks smooth if seen at the usual scale. But this smoothness is deceptive: a detailed analysis shows a discontinuity at each plotted point and no smoothly differentiable function can pass exactly through these points. The same is true for the Vos-Walraven set, the 1931 CIE color mixture data, and Judd's modification of the CIE values.

The major purpose of this paper is to present mathematical expressions to represent the Smith-Pokorny fundamentals, which they called S_R, S_G, and S_B. We will use R, G, and B to stand for the calculated sensitivities of the three cone types whose action spectra are specified. Because discrimination models are concerned with differentials, we have added the requirement that the equations be differentiable. We know in advance that the equations for R, G, and B cannot meet this requirement and exactly fit the tabled values; therefore, we have realistically aspired only to come close.

The functions are presented as equations, as graphs, and in tabular form, the latter to allow exact comparison with the Smith-Pokorny values. A comparison is then made between the sum of the R and G functions and Judd's modification of the CIE spectral luminous efficiency function V_λ. Because chromatic discrimination requires that R + G be held constant, the calculated functions for $R/(R + G)$, $G/(R + G)$, and $B/(R + G)$ (to be called $\hat{R}$, $\hat{G}$, and $\hat{B}$) are shown graphically. Finally, the first derivatives of these functions ($\hat{R}'$, $\hat{G}'$, and $\hat{B}'$), which are of key importance for modeling chromatic discrimination, are presented. All plots are of energy-based sensitivity versus wavelength. An appendix elaborates upon the method used for obtaining the equations.

Experience with the new color facsimile standard

G.B. Beretta, E.F. Chan, K. Konstantinides, D.T. Lee, H.J. Lee, A.H. Mutz

Hewlett-Packard Laboratories, 1501 Page Mill Road, Palo Alto, CA 94304

Abstract

In 1994 the International Telecommunications Union – Telecommunications Sector (ITU-T, formerly known as CCITT) has approved annexes to the Group 3 and Group 4 facsimile standards that allow for the exchange of continuous-tone color and gray-scale images. After reviewing the highlights of the new capabilities, we report on our experience in implementing this standard and testing the protocol. We believe the quality of images transmitted with color facsimile is sufficient for many communication tasks of people working in the various color areas.

1 Overview of the color-facsimile standard

Research and development of color facsimile can be dated back to before World War II [4]. This effort has resulted in expensive and proprietary systems for special applications, such as the transmission of color photographs by news delivery services. Three breakthroughs have occurred that will soon make color facsimile as ubiquitous as plain paper facsimile is now.

The first break-through has occurred in electronic imaging. New lossy data compression algorithms that discard image data that cannot be perceived allow considerable reduction of the image size in bytes. This size reduction makes it viable to transmit the image over the public-switched telephone network (PSTN) using ordinary modems.

The second breakthrough is in the cost of hardware. High performance processors, color scanners, and color printers have become so cheap that they can be assembled to a color facsimile machine with a very accessible purchase price.

Last but not least, an international standard has been sanctioned [2, 3]. This means that users can universally exchange color images with remote users without bothering about the manufacturer or type of the facsimile stations.

1

1.1 Selection of a color space

In a study proposal introduced 1990 by the Nippon Telegraph and Telephone Company of Japan (NTT), business color images have been classified into four categories:

full color (color photographs)

multi-color (color charts and graphs)

bi-color (documents marked up with red ink)

mixed color (combinations of above documents, such as color pages of magazines)

Seventeen color spaces have been considered. They have included device spaces likes CMYK and YIQ, colorimetric spaces like the CIE color spaces, and color order systems such as the Munsell Renotation System.

These color space candidates have been evaluated using criteria relevant for the intended use of color facsimile mentioned above. Examples are quantization error, compatibility with compression algorithms, and color stability with white point change.

This evaluation has resulted in the selection of the CIE 1976 ($L^*a^*b^*$) color space (CIELAB) as the mandatory color space for color facsimile. The standard also specifies CIE Standard Illuminant D_{50} as the default and basic values for the white point/illuminant.

Special care has been necessary for the selection of an appropriate gamut range. Because the standard prescribes the allocation of 8 bits/pel/component (or optionally 12), a trade-off is necessary between the gamut size and the artifacts introduced by quantization into a limited data volume. It has been determined experimentally that with a gamut range of $a^* = [-80, 80]$, $b^* = [-80, 120]$, quantization artifacts under aggressive compression are relatively unobjectionable [6].

The standard prescribes a default gamut range of $a^* = [-85, 85]$, $b^* = [-75, 125]$. For those cases in which *a priori* knowledge about the gamut range is available and high color fidelity is necessary, the standard allows the specification of a custom gamut range. This custom gamut range can be changed on a *per* page basis.

1.2 Selection of a data compression method

For the image compression method, baseline JPEG (Joint Photographic Experts Group) [1] has been adopted. As shown in Fig. 1, this method consists of three stages. First, a sequential discrete cosine transform (DCT) is applied, which is an orthogonal and separable transform that allows near-optimum energy compaction and for which a number of fast algorithms with low computational complexity have been developed.
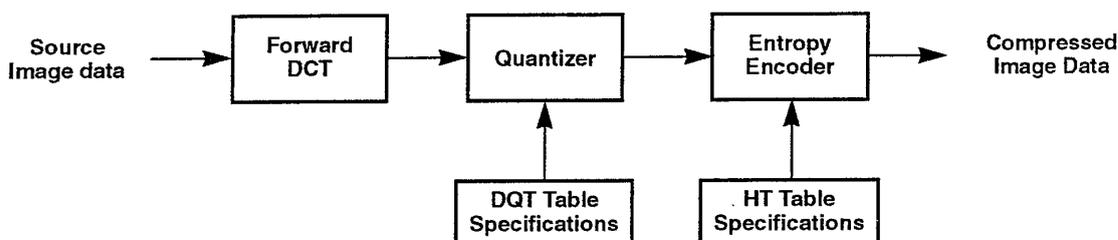
```
Source            Forward              Entropy          Compressed
Image data   →     DCT     →  Quantizer  →   Encoder   →   Image Data
                                 ↑                 ↑
                             DQT Table        HT Table
                           Specifications   Specifications
```

Figure 1. DCT-based encoder, simplified diagram. The lightness channels and the two chromatic channels are each run through the encoder. Typically, the tables are different for the various channels.(After reference 1, Fig. 4.)

In a second stage the data is quantized based on the discrete quantization table (DQT). This stage is lossy and implementors design the DQT so that no visible artifacts are introduced in the image. The compression ratio of a file can be increased by setting a so-called *q-factor* or *scaling factor*, which is essentially a uniform multiplicative parameter that is applied to the quantization tables. Even when the tables are carefully designed to be perceptually lossless, a large *q*-factor will introduce artifacts, such as blockiness in areas of constant color or ringing on text characters.

The final step is a lossless Huffman encoder, which eliminates the entropy in the image file. The Huffman table (HT) controls the effectiveness of the lossless compression. At the receiving end, the inverse transforms are applied in reverse order, creating a rendition of the original image.

It has been established [5] that the luminance acuity is 32 to 33 cycles/degree, the red–green chromatic acuity is 11 to 12 cycles/degree, and the blue–yellow chromatic acuity is around 11 cycles/degree. Therefore, by default the color facsimile standards prescribes subsampling the image by 4:1:1 in the CIELAB space. For critical applications, it is possible to override this spatial subsampling in the chrominance channels.

1.3 The color processing pipeline

A Group 3 color facsimile machine essentially consists of an RGB scanner, a JPEG coder, a module performing the facsimile protocol, a modem, and a CMYK printer. Fig. 2 illustrates the color pipeline.

The process begins with the translation of a color, 24-bits per pixel, RGB image into the CIELAB color space. At the baseline resolution of 200 dpi, an A size color image requires approximately 11.22M bytes of storage. Next, the chrominance components of the image are subsampled by a factor of two on both the horizontal and vertical coordinates. Subsampling decreases the size of the original image by a factor of two. Be-
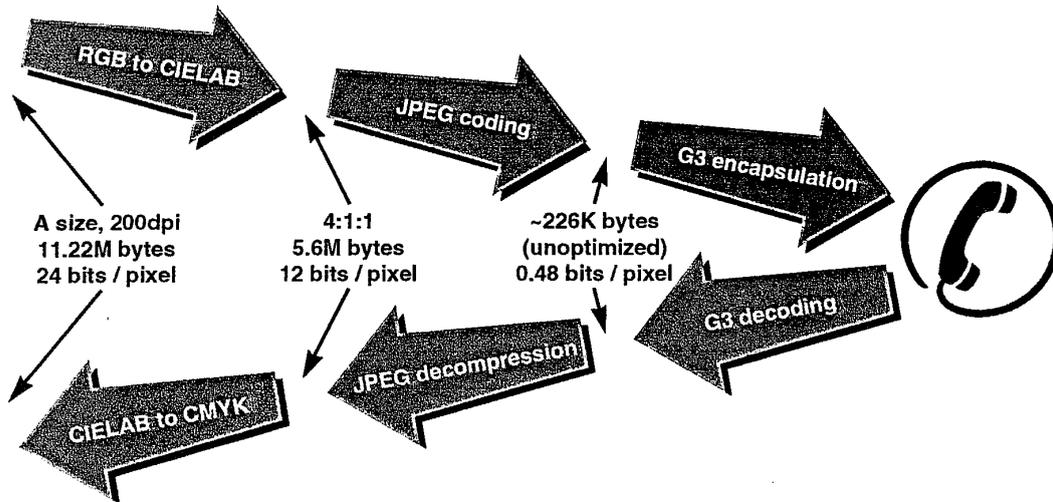
3

Figure 2. Color facsimile pipeline. The RGB bitmap from a scanner is converted to the CIELAB color space and the chrominance channels are spatially subsampled. The image is then compressed using the JPEG method and encapsulated for the conventional Group 3 facsimile protocol and transmitted over an ordinary phone line. At the receiving end, the inverse operations are applied.

fore transmission, the image is compressed using the baseline lossy JPEG algorithm. The compressed image is finally transmitted using the Group 3 facsimile protocol. On the receiver side, after Group 3 decoding, the image is decompressed and printed on a color or gray-scale printer.

The use of a device independent color space, a well defined gamut and the specification of the viewing conditions allow for a colorimetrically correct rendition of the image at the receiving color facsimile station. For artists, scientists, industrialists, educators, and the many others using color professionally, the characterization of the process is important also to know the limits of the system. This knowledge is important because the image is received remotely and by the very nature of the facsimile madium the sender can not compare visually the reproduction with the original.

2 Extensions to the facsimile protocol

To explore the issues for a practical realization we have assembled a system from a number existing components. This approach has allowed us to focus on the system issues and to ignore implementation details. After a short implementation time we were able to successfully verify the protocol with an external independent implementation.

4

The first step in augmenting an existing facsimile implementation for continuous-tone and gray-scale images is to provide for the new bits in the handshaking frames during the pre-message procedure. In this phase the receiving machine advertises its capabilities to the sending machine, which compares them with its own capabilities and selects the best common set.

The pre-message procedure allows users to communicate through a wide variety of facsimile machines. If the sender is a color facsimile system, the following receiver capabilities have to be evaluated:

1. Is the error correction mode (ECM) enabled?
2. Is the resolution at least 200 x 200 pels/25.4 mm?
3. Is JPEG coding, gray-scale mode supported?
4. Is JPEG coding, full color mode supported?

Any Group 3 facsimile machine supports the conventional modified Huffman coding that is used in all existing machines. If any of the above questions is negative, the sender falls back to this mode, ensuring full backwards compatibility with existing equipment. As the third test suggests, the new color facsimile protocol also supports a high quality black & white continuous tone mode.

If necessary, this negotiation can be repeated for each page, allowing for the transmission of mixed documents. For example, if a multi-page document contains only one color page and one gray-scale page, while the other pages are binary (e.g., black text) only the two continuous-tone pages will be JPEG encoded and only the color page will be sent in color mode.

The other components required to assemble a color facsimile system are readily available: scanner & printer drivers, and a JPEG coder. The JPEG coders available in the public domain usually require the image to be represented either in RGB or in YUV coordinates. Adding the conversion to and from the CIELAB color space is straight-forward.

3 Transmission experiments

For error free communication using a 9600 baud modem, experimental and simulation results showed that the effective facsimile transmission rate is approximately 64K bytes per minute. This includes negotiation start-up time and typical retransmission overhead. Using lossy JPEG and the default quantization tables, we have achieved perceptually lossless compression with compression ratios close to 20:1.

For such a compression ratio, and a transmission rate of 64K bytes per minute, color facsimile of an A size color image will require at least 8.6 minutes of communication time. This is certainly neither cost-efficient or practical. Therefore, higher compression ratios, at acceptable printing quality, are required for efficient color facsimile. For example, for a transmission time of less than three minutes, the minimum required compression ratio is 60:1.

In the JPEG standard, the compression ratio is controlled by two sets of tables: the quantization and the Huffman tables. The quantization tables affect the lossy-compression part. A common practice is to use the default JPEG quantization tables, but scaled with an appropriate q-factor as mentioned above. The Huffman tables control the lossless compression part. Most applications use the default tables. For high compression ratios and acceptable output, both of these tables will have to be optimized.

Prior work related to JPEG optimization addresses only the YUV or the RGB color spaces and focuses on perceptually lossless compression. Scanning or printing are seldom taken into consideration. By taking into consideration the complete color facsimile pipeline (scanning, color transformation, compression-decompression, and printing), we are able to improve significantly both the compressibility and the printing quality of color images. Using custom quantization tables we are able to improve the compression ratio by at least 20%, for similar visual quality.

Table 1 shows compression ratios for three color images using both the default (uniformly scaled) and custom quantization tables. All images were scanned at 200 dpi. The 4CP01 image is the color facsimile test image recommended by the ITU.

Image	Default table	Custom table
Real estate flier with photo	52:1	82:1
Book page with photos and text	53:1	63:1
4CP01 test chart	47:1	63:1

Table 1. **Typical compression ratios.** We compare the compression ratios obtained with the default tables used in most JPEG implementations and the custom tables we tweaked by hand for color facsimile. The latter allow for viable transmission times.

4 An open problem

The design of good quantization tables is more an art than a science. Authors that have studied the problem have done so in the optical domain, i.e., for images encoded in device coordinates. Because the color facsimile standard prescribes the CIELAB color space, the images are represented in the perceptual domain.

Much of the literature on color perception deals with aperture colors. There is only limited discussion of the spatial perception of color other than the study of visual acuity and contrast sensitivity. Knowledge about spatial color perception would allow the design of better quantization tables, which would shorten the transmission times for color facsimile communication.

5 Conclusions

We have shown that the new amendments to the Group 3 standard to provide for color and gray-scale facsimile are sound and do not present any particularly difficult to solve system issues. We have tested our implementation of the protocol with an independent realization and successfully exercised all aspects of the protocol.

With the default JPEG quantization and Huffman tables the transmission time are not practical for commercial applications. We were able to improve the transmission times with custom tables, but more work is necessary in this area.

References

1. ISO/IEC JTC 1/SC 29/WG 10 N52, Information technology — Digital compression and coding of continuous-tone images —, ISO/IEC IS 10918-1, 1992.

2. ITU (Q.5/8 Rapporteur), Amendments to ITU-T Rec. T.30 for enabling continuous-tone colour and grayscale mode for Group 3, COM 8-43-E, 1994.

3. ITU (Q.5/8 Rapporteur), Amendments to ITU-T Rec. T.4 to enable continuous-tone colour and grayscale mode for Group 3, COM 8-44-E, 1994.

4. M. Matsuki and D.T. Lee, Developments in Color-Facsimile Standards: An Overview, Proc. *SID 94 Digest*, 883-886, 1994.

5. Kathy T. Mullen: The Contrast Sensitivity of Human Colour Vision to Red-Green and Blue-Yellow Chromatic Gratings, in *J. Physiol.*, **359**, 381-400, 1985.

6. A.H. Mutz, Effect of Gamut Range Upon Color Quantization During JPEG Coding, Proc. *SID 94 Digest*, 891-893, 1994.

Luminance-Model-Based DCT Quantization for Color Image Compression

Albert J. Ahumada, Jr.
NASA Ames Research Center, Human Interface Research Branch
Moffett Field, California 94035
and
Heidi A. Peterson
IBM T. J. Watson Research Center
Yorktown Heights, NY

ABSTRACT

A model is developed to approximate visibility thresholds for discrete cosine transform (DCT) coefficient quantization error based on the peak-to-peak luminance of the error image. Experimentally measured visibility thresholds for R, G, and B DCT basis functions can be predicted by a simple luminance-based detection model. This model allows DCT coefficient quantization matrices to be designed for display conditions other than those of the experimental measurements: other display luminances, other veiling luminances, and other spatial frequencies (different pixel spacings, viewing distances, and aspect ratios).

1. INTRODUCTION

1.1 Discrete cosine transform-based image compression

The discrete cosine transform (DCT) has become a standard method of image compression[1,2]. Frequently the image is divided into 8×8-pixel blocks that are each transformed into 64 coefficients for reconstruction from the DCT basis functions. The DCT transform coefficients $I_{m,n}$, of an $N \times N$ block of image pixels $i_{j,k}$, are given by

$$I_{m,n} = \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} i_{j,k}\, c_{j,m}\, c_{k,n}\,, \quad m,n = 0, \ldots, N-1, \tag{1}$$

where

$$c_{j,m} = \sqrt{\frac{1}{N}}, \quad m = 0,$$

$$= \sqrt{\frac{2}{N}} \cos(\frac{\pi m}{2N}[2j+1]), \quad m > 0.$$

The block of image pixels is reconstructed by the inverse transform, which is the same as the forward transform for this normalization,

$$i_{j,k} = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} I_{m,n}\, c_{j,m}\, c_{k,n}\,, \quad j,k = 1, \ldots, N-1. \tag{2}$$

Quantization of the DCT coefficients $I_{m,n}$ results in image compression. Typically, compression standards allow the user to set the level of quantization for each coefficient. A quantization matrix $Q_{m,n}$ specifies the quantization of the DCT coefficients, where the m,n th entry in the matrix is the step size of the uniform quantizer for the m,n th DCT coefficient.

1.2 Threshold-based DCT coefficient quantization

Last year at this meeting Peterson, Peng, Morgan, and Pennebaker[3] presented experimental measurements of the detection thresholds for individual DCT basis functions in each of the R, G, and B color dimensions.

Visibility thresholds were measured for purely R, G, or B basis functions on a black background, masked by a pedestal containing the same color at the luminance that resulted from the mid-range digital value 128 (plus a small veiling luminance). From these measurements, Peterson et al. [3] derived DCT coefficient quantization matrices for separate compression of R, G, and B images. Their quantization matrices produced reasonable compression with barely visible artifacts. However, the viewing conditions in many applications may not be similar enough to the viewing conditions of their experiments to yield good compression without visible artifacts using their specific quantization matrices. The goal is to develop a formula to generate appropriate quantization matrices for different viewing conditions. In particular, we would like to interpolate and extrapolate the experimental results to other display luminances, other veiling luminances, and other spatial frequencies (different pixel spacings, viewing distances, and aspect ratios).

2. THEORY

The theoretical basis of the model developed here is the assumption that the detectability of distortion in the decoded RGB image can be predicted based on the luminance error caused by quantization of an individual DCT coefficient. Although quantization of DCT coefficients causes both luminance and chrominance errors, we assume that for all the coefficients other than the DC coefficient, the spatial frequency is high enough that the luminance error dominates. The model only takes into account the luminance of the error; it ignores the chrominance.

The display parameters are characterized as follows:

L_0 is the background or veiling luminance on the screen,
L_i is the luminance contributed by the image,
L is the total luminance, $L_0 + L_i$,
w_x is the horizontal width of a pixel in degrees of visual angle,
w_y is the vertical height of a pixel in degrees of visual angle.

Assuming ideal interpolation of a spatially sampled signal displayed on a linear monitor, the image of the error $E_{m,n}(x,y)$, due to quantization of the m,n th DCT basis function coefficient $I_{m,n}$ can be written

$$E_{m,n}(x,y) = L_e \cos(\frac{\pi m}{2N w_x}[2x+1]) \cos(\frac{\pi n}{2N w_y}[2y+1]), \quad 0 \le x < N w_x, \ 0 \le y < N w_y, \quad (3)$$

$$= 0, \quad \text{elsewhere,}$$

where L_e is the luminance amplitude of the error image. We assume that the luminance error due to quantization of a particular DCT coefficient is not visible if L_e is less than a threshold $T_{m,n}$, which is a function only of the coefficient indexes, m,n, and the above display parameters.

2.1 Single-Orientation Case

For either m or n equal to zero (but not both), and ignoring windowing (that is, the limited spatial extent of the test pattern in the experiment), the problem becomes one of predicting $T_{m,n}$ for a vertical grating with spatial frequency

$$f_{m,0} = \frac{m}{2N w_x} \quad (4a)$$

or a horizontal grating with spatial frequency

$$f_{0,n} = \frac{n}{2N w_y} . \quad (4b)$$

In our model we approximate the log of $T_{m,n}$ by a parabola in log spatial frequency,

$$\log T_{m,n} = \log (T_{\min}) + K (\log f_{j,k} - \log f_{\min})^2, \quad m=0 \text{ or } n=0, \quad (5)$$

where the parameters $T_{\min}$, K, and $f_{\min}$ are functions of the total luminance L. $T_{\min}$ is the luminance threshold at $f_{\min}$, the frequency where the threshold is smallest; and K determines the steepness of the parabola.

Figure 1 illustrates that contrast sensitivity (luminance divided by threshold luminance) for luminance-only sine wave gratings can be fit by parabolic functions of log spatial frequency over a range of luminances, when only the fit to high (> 1 cycle per degree) spatial frequencies is considered. The data are from the work of van Nes and Bouman[4], as reported by Olzak and Thomas[5]. The parabolas are least-squared error fits to the individual sets of luminance data. The parabola model is not valid for low spatial frequencies, since the decrease in sensitivity at low frequencies is slow for imagery that is not stabilized on the retina[6]. This model is not applicable for $T_{0,0}$. The results presented in Peterson et. al.[3] indicate a conservative estimate for $T_{0,0}$ is the smaller of $T_{1,0}$ and $T_{0,1}$.
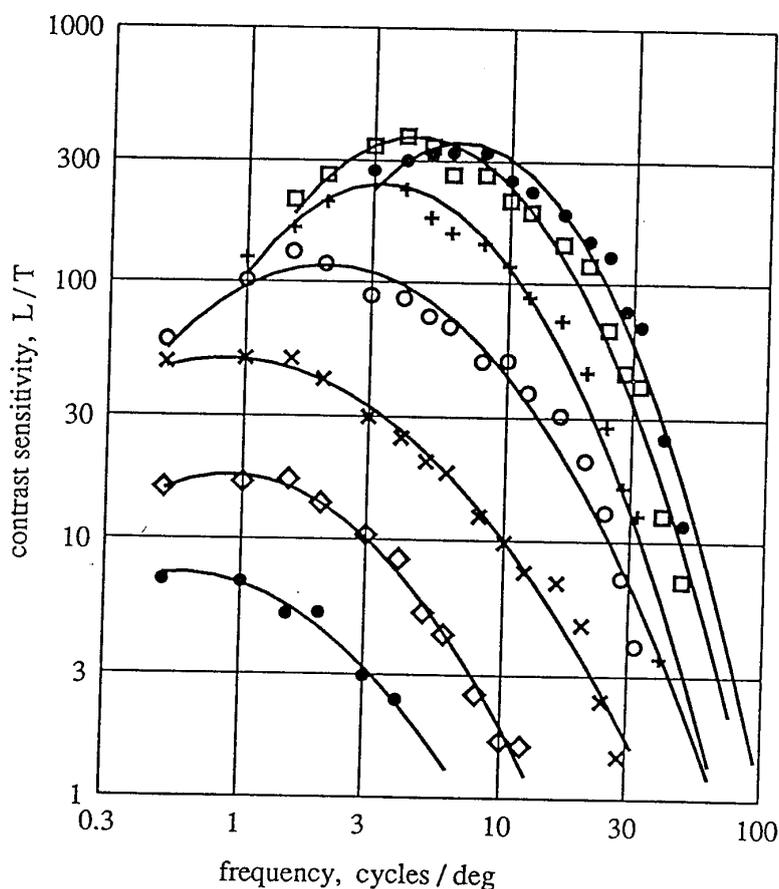
Figure 1. Contrast sensitivity data from van Nes and Bouman[4]. The data were measured at luminances spaced one decade apart from $2.9\times10^{-4}\,cd/m^2$ to $2.9\times10^{-2}\,cd/m^2$. Parabolas were fit by least squares to the log data with spatial frequencies of 1 cycle/degree or greater.

2.2 Two-Orientation Case

When neither m nor n are equal to zero, the trigonometric identity

$$2\cos(a)\cos(b) = \cos(a+b) + \cos(a-b) \tag{6}$$

shows that the DCT basis functions are the sum of two frequency components with the same spatial frequency

$$f = \sqrt{f_x^2 + f_y^2}, \tag{7}$$

but different orientations. The angle θ between the two components is

$$\theta = \arcsin \frac{2 f_x f_y}{f^2}. \tag{8}$$

There are two extreme cases. If either f_x or f_y is zero, the result is the single orientation case discussed above. If f_x equals f_y, the components are orthogonal, and detection can be predicted either by probability summation, or more simply, by summing the fourth powers of the amplitudes and taking the fourth root. The latter leads to the threshold being increased by the factor $2^{0.75} = 1.68$. This prediction for the effect of the summation of the two Fourier components follows the work of Phillips and Wilson[7] and that of Watson[8]. The effect of intermediate positions of the two Fourier components can be approximated as a multiplicative threshold factor

$$\frac{1}{r + (1-r)\cos^b \theta}, \tag{9}$$

where b represents the orientation tuning of the visual system in grating detection. We assume $b = 2$. In addition, when f_x and f_y are equal, the two Fourier components are in the oblique direction, which usually results in a further increase in threshold. The oblique effect can be included by decreasing the value of r. Incorporating the factor of Equation (9) in the model of Equation (5), the complete model for predicting the log visibility thresholds becomes

$$\log T_{m,n} = \log\left(\frac{T_{\min}}{r + (1-r)\cos^2\theta}\right) + K \, (\log f_{j,k} - \log f_{\min})^2, \quad m,n = 0, \ldots, N-1. \tag{10}$$

3. PARAMETER ESTIMATION AND GOODNESS OF FIT

A luminance-only model for DCT basis function visibility thresholds requires the model parameters $T_{\min}$, $f_{\min}$, and K each be expressed as a function of luminance. The van Nes and Bouman[4] data were collected over a wide range of luminances. The experiments of Peterson et al.[3] had a narrow luminance range, but were shown to lead to good quantization matrices. Therefore, our strategy for estimating the functions is first to use the van Nes and Bouman data to determine the form of these functions, and then to fit these form templates to the Peterson et al. data. Each function is assumed to be a power law (linear in log-log coordinates), clipped at some maximal value. The exponent of the power law is found by a least squares fit to the van Nes and Bouman parabola parameters in the log domain. The maximal value for each function is also based on that data. The resulting templates are the dashed lines in Figure 2. The van Nes and Bouman peak contrast sensitivities have

been shifted downwards by approximately 0.3 log units relative to those in Figure 1. This shift attempts to account for the difference in test pattern size for the two experiments, and corresponds to the log of the fourth root of the ratio of the two areas, the same summation rule discussed above [7,8]. (Peterson et al. used 50×50 pixel test patterns that occupied 1.47 degrees horizontally and 1.27 degrees vertically, while van Nes and Bouman used a 8.25 by 4.5 degree signal.)
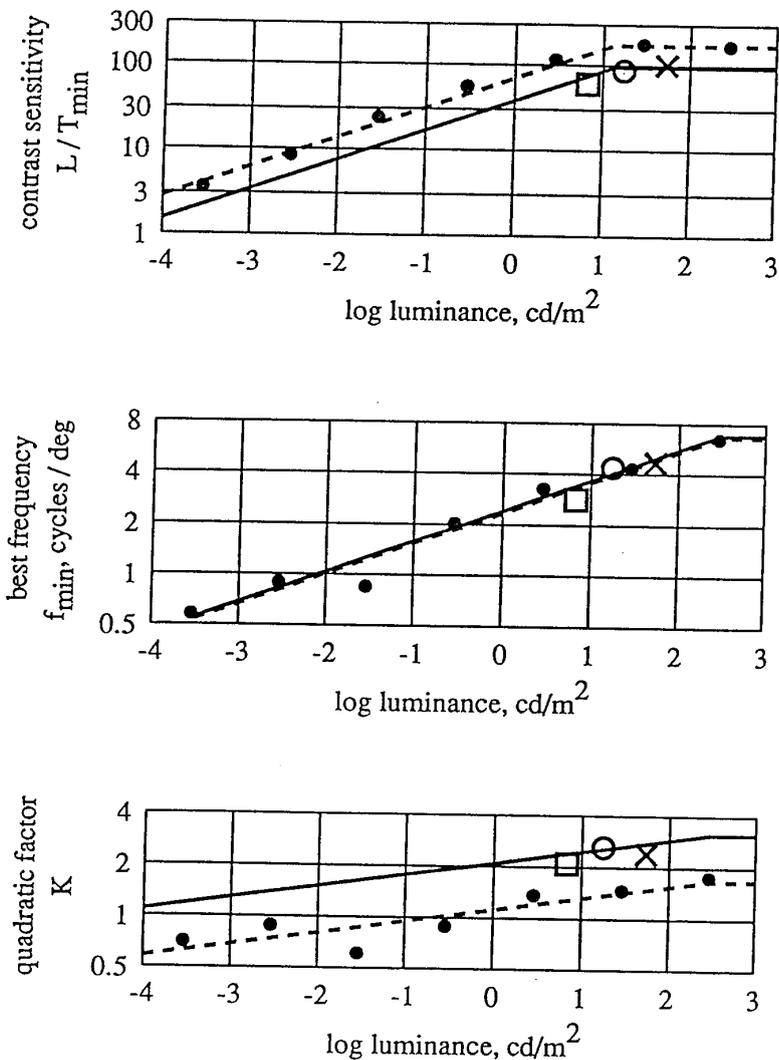

Figure 2. T_{min} (top), f_{min} (middle), and K are shown as a function of L. Open circles, X's, and squares represent parameter values of the predictions for the Peterson et al. data in Figure 2 for R, G, and B, respectively. Solid circles represent parameter values for the curves fit to the van Nes and Bouman data in Figure 1. The solid lines show Equations (12), (13), and (14), developed to extrapolate the measured DCT basis function thresholds.

These templates are then used to extrapolate the data of Peterson *et. al.* by finding the best vertical shift for fitting the templates to the Peterson *et. al.* data. The quality of the model fit to the data is evaluated by the chi-square measure

$$\chi^2 = \sum \frac{N \, (\log T_{obs} - \log T_{pred})^2}{\sigma^2}, \tag{11}$$

where N is the number of subjects (15), T_{pred} is the threshold predicted by the model (Equation (10)), and T_{obs} is the experimental threshold. The standard deviation was estimated from the between-subjects standard deviations, excluding thresholds whose value was limited by the range of available stimulus levels. The standard deviations of the log thresholds do not appear to vary with basis function, but a different standard deviation was used for the B ($\sigma = 0.31$) thresholds than for the R and G ($\sigma = 0.21$). The sum is over the 58 non-DC basis functions (18 R, 22 G, 18 B) included in the experiment.

3.1 Template fitting procedure

Figure 2 shows model parameters estimated directly from the Peterson *et al.* data along with the templates fit to that data. To assess quantitatively the fit of the templates a sequence of nested hypotheses was formed that successively added constraints to the model parameters. Then T_{min}, f_{min}, K, and r were estimated by finding the values minimizing the chi-square measure of fit, subject to their constraints. Initially, all four model parameters were allowed to vary arbitrarily with luminance (*i.e.* color). The results appear in the first column of Table 1. Next all parameters were re-estimated with r assumed constant. These results appear in the second column of Table 1 and are also the points in Figure 2 representing the estimates unconstrained by the templates. Then all parameters were re-estimated with r constant and K assumed to follow its template (with a vertical shift allowed). In this case the model has eight parameters: r, a vertical shift for the K template, three f_{min}'s (one for each luminance), and three T_{min}'s (one for each luminance). In turn, f_{min} and $S_{max} = L/T_{min}$ were additionally constrained, so that the rightmost column in Table 1 gives estimates of the model parameters consistent with all the templates. Table 1 also gives the minimum χ^2 fit of the model of Equation (10) for this series of nested constraints on the parameters. Differences between successive fits would be distributed approximately as χ^2 if the constraints were valid (and other assumptions held).

		next parameter modelled				
		none	r	K	f_{min}	S_{max}
fit χ^2		107.80	108.14	112.02	133.58	233.11
test χ^2		107.80	0.34	3.88	21.56	99.53
df		46	2	2	2	2
signif. level		<0.001	0.85	0.15	<0.001	<0.001
r	B	0.65	0.70	0.69	0.68	0.68
	R	0.65	0.70	0.69	0.68	0.68
	G	0.70	0.70	0.69	0.68	0.68
K	B	2.10	2.12	2.28	2.56	2.39
	R	2.62	2.67	2.44	2.64	2.56
	G	2.41	2.41	2.64	2.97	2.77
f_{min}	B	2.82	2.82	3.02	3.72	3.40
	R	4.36	4.36	4.17	4.43	4.04
	G	4.68	4.68	4.90	5.44	4.96
S_{max}	B	62.6	62.6	61.1	52.0	74.5
	R	91.3	91.3	89.2	93.4	94.7
	G	101.3	101.3	103.7	108.6	94.7

Table 1. Parameter estimates and goodness of fit.

These test χ^2 values indicate that parameters r and K are well fit by their templates, but f_{min} and T_{min} are not well fit statistically. The luminances for B, R and G are 6.8, 17.4, and 53.9 cd/m^2, respectively.

3.2 T_{min}, f_{min}, and K luminance functions

The relation between T_{min} and luminance with all model parameters constrained to their templates is

$$T_{min} = \frac{L^{a_T} L_T^{1-a_T}}{S_0}, \quad L \leq L_T,$$

$$= \frac{L}{S_0}, \qquad L > L_T,$$

(12)

where $L_T = 13.45\,cd/m^2$, $S_0 = 94.7$, and $a_T = 0.649$. Figure 2 shows this relation in contrast sensitivity $S_{max} = L/T_{min}$. Despite the adjustment for area, there is still a difference of about 0.25 log units between the two data sets.

The two data sets agree very closely on the relation of f_{min} to L. Figure 2 shows that both sets of data are fit well by the power function

$$f_{min} = f_0 L^{a_f} L_f^{-a_f}, \quad L \leq L_f,$$

$$= f_0, \qquad L > L_f,$$

(13)

where $f_0 = 6.78\,cycles/deg$, $a_f = 0.182$, and $L_f = 300\,cd/m^2$. This function is clipped at high luminances because van Nes and Bouman[4] found no difference between the contrast sensitivity functions for luminances of 290 and 1880 cd/m^2.

Although χ^2 tests could not reject the hypothesis that the same value of K could be used for R, G, and B, the van Nes and Bouman data suggest that K is also approximately a power function of L, as shown in Figure 2. Accordingly, we approximate K by the power function

$$K = K_0 L^{a_K} L_K^{-a_K}, \quad L \leq L_K,$$

$$= K_0, \qquad L > L_K,$$

(14)

where $K_0 = 3.125$, $a_K = 0.0706$, and $L_K = 300\,cd/m^2$. This function is clipped at high luminances for the same reason as the f_{min} function.

Equations (12), (13), and (14) give T_{min}, f_{min}, and K as functions of L. When they are substituted into Equation (10), the result is a luminance-only model for DCT basis function visibility thresholds that extrapolates the Peterson et al. data.

3.3 Orientation correction

Figure 3 shows threshold predictions using the templates described above in the model of Equation(10) (dashed curves from the rightmost column in Table 1). These curves appear to fit the experimental thresholds nearly as well as those estimated directly from the data (solid curves from the second column in Table 1). The m or n equal zero experimental data of Peterson et al. are indicated by small symbols in Figure 3. The two-orientation data, indicated by large symbols in Figure 3, are shown divided by the correction factor of Equation (9). The need for this correction factor is demonstrated in Figure 4. Figure 4 plots the ratio of the experimentally measured thresholds for two-orientation basis functions (m, $n \neq 0$) to the uncorrected model

predictions (Equation (5)) as a function of θ, the angle between the orientation components. These ratios show that the experimentally measured two-orientation thresholds are consistently higher than those predicted from just the spatial frequency of the basis function (that is, the model of Equation (5)). The solid line in Figure 4 shows Equation (9) with $r = 0.70$. This value was used to correct the two-orientation data points in Figure 3.
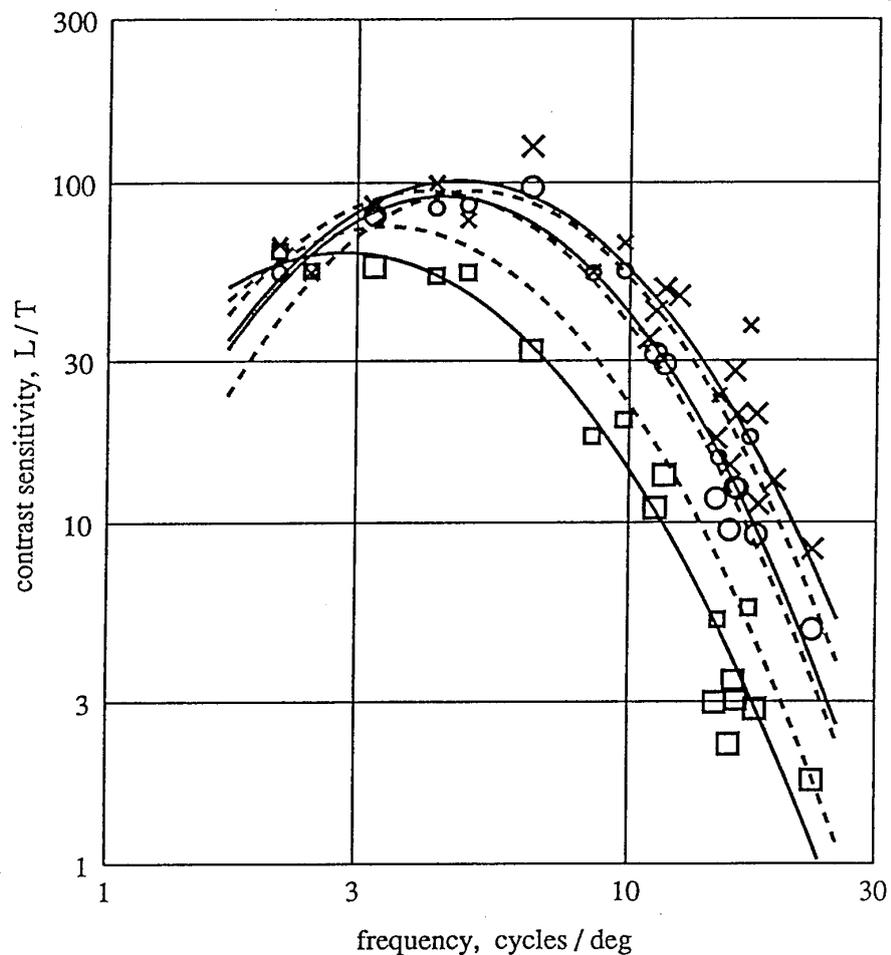

Figure 3. Visibility thresholds T for DCT basis functions measured by Peterson et al.[3] are represented as contrast sensitivities $S = L/T$, as a function of their spatial frequency f. Circles, X's, and squares correspond to R, G, and B experimental thresholds, respectively. Small symbols indicate single-orientation basis function thresholds. Large symbols indicate two-orientation basis function thresholds, after correction by Equation (9). Solid lines show the model predictions using the parameters T_{min}, f_{min}, and K estimated directly from the data by minimum χ^2 (the second column of Table 1). Dashed curves show model predictions using Equations (12), (13), and (14). The top, middle, and bottom (at the right edge of the graph) curves are fit to the G, R, and B, respectively.
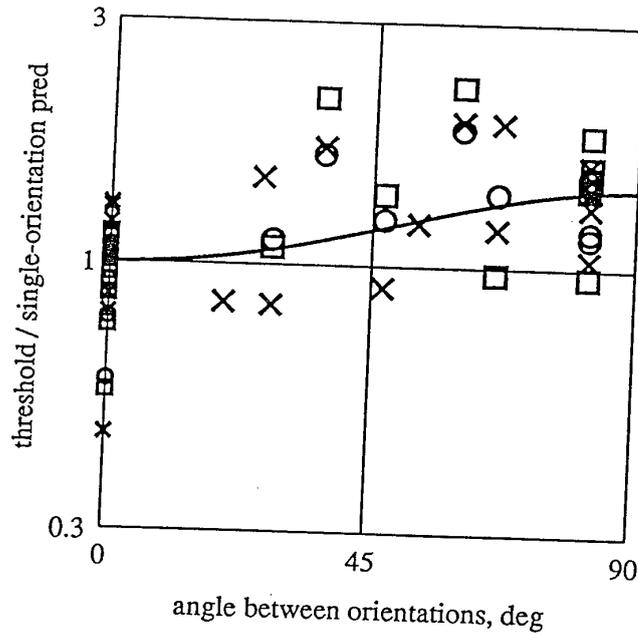
Figure 4. Ratios of uncorrected basis function visibility thresholds to their single-orientation model predictions are shown as a function of θ, the angle between the orientation components. The solid line shows the the graph of Equation (9) with $r = 0.70$. Points are labeled as in Figure 3.

4. GENERATION OF QUANTIZATION MATRICES

To convert visibility thresholds measured in luminance to quantization units, $T_{m,n}$ values are first multiplied by 2. This accounts for the maximum quantization error being one half of the quantizer step size. Two more factors are needed to complete the conversion. First we need ΔL_g, the change in luminance caused by a change of one quantization unit for gun g, at luminance L. For a linear display this is the difference between the minimum luminance $L_{g,\min}$ and maximum luminance $L_{g,\max}$ from gun g, divided by the total number of quantization units M,

$$\Delta L_g = \frac{(L_{g,\max} - L_{g,\min})}{M}.$$ (15)

The second factor accounts for the normalization constants of the DCT,

$$\alpha_m = \sqrt{\frac{1}{N}}, \quad m = 0,$$
$$= \sqrt{\frac{2}{N}}, \quad m > 0.$$ (16)

The quantization matrix elements for that gun are then

$$Q_{m,n} = \frac{2 T_{m,n}}{\alpha_m \alpha_n \Delta L_g}.$$ (17)

5. DISCUSSION

There remains the question of why the shape and the peak of the contrast sensitivity functions of this data are so different from the van Nes and Bouman[4] data. Two factors that would force the results in this direction are the spatial frequency transfer functions of the displays and of the observers. No attempt was made by Peterson et al. to check the optical correction of the observers. In Figure 3 the small symbols corresponding to horizontal and vertical gratings appear in pairs, with the horizontal grating frequency just to right of the vertical grating frequency. At high spatial frequencies the sensitivity for vertical gratings is much less than that for horizontal gratings, a result consistent in direction with cone spacing[9], but also with some low pass filtering by the display electronics. Other possible factors include the experience and selection of the observers. The above factors seem generally in the direction of making the results more practical, but if one wants to insure that artifacts are not visible, one could use the templates as fit to the van Nes and Bouman data, at the cost of less compression. One major difference between the two studies was that the small area signals of Peterson et al. were on a dark surround instead of an adapting field of the same mean luminance. It is well known that adaptation near the masking level is optimal for brightness discrimination[10]. This raises the interesting question of how the luminance term of the model should be computed from the image luminance distribution on the display to optimize compression.

A luminance-based model appears to adequately account for the variations between the R, G, and B sensitivities found by Peterson et al.[3]. There seems to be no need to introduce chromatic mechanisms to account for their data. An advantage of separate compression for R, G, and B images is that only the luminance of the output guns is needed to generate the quantization matrices. Transformation to luminance and two iso-luminant color coordinates will give better compression, but then it may be critical that the reconstruction be done so that the iso-luminant color planes are accurately reconstructed at the display. Whatever color transformation is used, at least one dimension will be luminance dominated at high spatial frequencies. Our model provides an experimentally based method for generating quantization matrices for such dimensions for a range of display parameters and viewing conditions.

6. REFERENCES

1. G. Wallace, "The JPEG still picture compression standard", *Communications of the ACM*, vol. 34, no. 4, pp. 30-44, 1991.
2. D. LeGall, "MPEG: A video compression standard for multimedia applications", *Communications of the ACM*, vol. 34, no. 4, pp. 46-58, 1991.
3. H. A. Peterson, H. Peng, J. H. Morgan, W. B. Pennebaker, "Quantization of color image components in the DCT domain", in B. E. Rogowitz, M. H. Brill, J. P. Allebach, eds., *Human Vision, Visual Processing, and Digital Display II*, Proc. SPIE, vol. 1453, pp. 210-222, 1991.
4. F. L. van Nes, M. A. Bouman, "Spatial modulation transfer in the human eye", *Journal of the Optical Society of America*, vol. 57, pp. 401-406, 1967.
5. L. A. Olzak, J. P. Thomas, "Seeing spatial patterns", in K. R. Boff, L. Kaufman, J. P. Thomas, eds., *Handbook of Perception and Human Performance*, pp. 7.1-7.56, Wiley, New York, 1986.
6. D. H. Kelly, "Visual processing of moving stimuli", *Journal of the Optical Society of America A*, vol. 2, pp. 216-225, 1985.
7. G. C. Phillips, H. R. Wilson, "Orientation bandwidths of spatial mechanisms measured by masking", *Journal of the Optical Society of America A*, vol. 1, pp. 226-232, 1984.
8. A. B. Watson, "Detection and recognition of simple spatial forms", in O. J. Braddick, A. C. Sleigh, eds., *Physical and Biological Processing of Images*, Springer-Verlag, Berlin, 1983.
9. D. R. Williams, "Topography of the foveal cone mosaic in the living human eye". *Vision Research*, vol. 28, pp. 433-454, 1988.
10. K. J. W. Craik, "The effect of adaptation on differential brightness discrimination", *Journal of Physiology*, vol. 92, pp. 406-421, 1938.

Spatial-frequency characteristics of letter identification

Kenneth R. Alexander, Wei Xie, and Deborah J. Derlacki

Department of Ophthalmology and Visual Sciences, University of Illinois at Chicago College of Medicine, Chicago, Illinois 60612

To investigate the spatial-frequency components that govern letter identification we compared contrast thresholds for three types of visual stimulus (1) standard Sloan letters, (2) Sloan letters that were spatially bandpass filtered by cosine log filters, and (3) D6 patterns (sixth spatial derivatives of Gaussians). Stimuli were presented on a gray-scale display screen of a Macintosh computer-based testing system at temporal frequencies primarily of 2 and 16 Hz. Contrast thresholds were measured in two subjects with normal visual acuity with use of forced-choice staircases. Contrast sensitivity functions for standard Sloan letters and D6 patterns were comparable at a temporal frequency of 16 Hz but differed systematically at a temporal frequency of 2 Hz. The measurement of contrast sensitivity for cosine log filtered letters presented at a temporal frequency of 2 Hz indicated that the object spatial frequency of maximum sensitivity shifted to lower frequencies as letter size decreased, whereas the retinal spatial frequency of maximum sensitivity remained relatively constant. When letters were spatially bandpass filtered at a peak object spatial frequency of 2.5 cycles/letter, then contrast sensitivity functions for letter identification were equivalent to those for D6 patterns at both temporal frequencies. These results suggest that spatially filtered letters may provide a more appropriate test of visual function than do standard letter optotypes.

1. INTRODUCTION

The loss of visual acuity in visual disorders has traditionally been assessed with the use of letter optotypes. Letters have also been suggested as an alternative or supplement to the use of sinusoidal grating stimuli in the measurement of contrast sensitivity.[1,2] As discussed previously,[2,3] letters have certain advantages over sinusoidal grating patterns in that they contain components with multiple orientations, are more familiar to patients, and are used in everyday life. Moreover, given the importance of letter identification to the performance of common visual tasks, it is relevant to determine the ability of patients to identify letters of low contrast and small size. However, letters contain a broad range of spatial frequencies,[4] which potentially complicates the theoretical interpretation of test results in terms of underlying visual mechanisms.

Nevertheless, evidence[5] suggests that, despite their broad spatial-frequency content, letters can provide a reasonable test of visual function, because only a limited range of object spatial frequencies[6] appears to be important for letter identification. However, there is some uncertainty as to the specific object spatial frequencies that are involved in letter identification. Some reports have suggested that relatively low object spatial frequencies [approximately 1–2 cycles per letter width (cycles/letter)] are the most important for letter identification and reading.[4,5,7,8] However, these studies used either low-pass filters[5,7,8] or filters with relatively broad bandpass characteristics,[4] so that the exact range of object spatial frequencies that is most relevant for letter identification is uncertain. Other investigators have suggested that these lower object spatial frequencies do not contribute significantly to letter identification and on this basis have designed visual acuity charts that use letter optotypes containing only high object spatial frequencies.[9,10]

The goal of the present study was to clarify the spatial-frequency components that govern letter identification. The first issue examined was the degree of similarity between contrast sensitivity functions obtained with standard letter stimuli and those obtained with stimuli that are more delimited in spatial-frequency content and orientation. Previous studies[11–13] have suggested that both types of stimulus produce similar contrast sensitivity functions, but explicit comparisons between these two types of test stimulus have not been made under equivalent testing conditions. Therefore we compared contrast sensitivity functions for Sloan letter optotypes[14] with those for D6 patterns (sixth spatial derivatives of Gaussians), a type of stimulus that is both spatially localized and delimited in spatial-frequency content, with a bandwidth at half-height of 1 octave.[15] D6 patterns are similar in spatial-frequency content to a Gaussian-windowed sinusoidal grating or Gabor patch.[16] We compared contrast sensitivity functions for letters and D6 patterns at two different temporal frequencies. Evidence indicates that the spatial characteristics of visual mechanisms can vary with temporal frequency such that spatial bandwidths become broader at higher temporal frequencies.[17] Therefore a comparison of contrast sensitivity at two disparate temporal frequencies provides a more stringent test of the similarity between letters and D6 patterns than does testing at a single temporal frequency.

The second issue addressed in this study was whether the same object-spatial-frequency components govern letter identification across letter sizes. This has been an implicit assumption in the use of letters as test stimuli but has been questioned recently in a study of the effect of refractive error on letter identification.[18] To test this assumption we measured contrast thresholds for the identification of spatially bandpass filtered letters at a range of letter sizes. Letters were filtered with a set of cosine log filters[19] that have the following properties: (1) the bandwidth at half-height is 1 octave, (2) the peaks of the
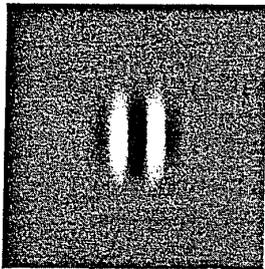
Fig. 1. Illustration of a one-dimensional D6 (sixth spatial derivative of a Gaussian) test stimulus.

filters differ by 1 octave, and (3) the filters are symmetrical on a log spatial-frequency axis. The two-dimensional cosine log filters were radially symmetric (torus shaped) in the frequency domain. Contrast thresholds for filtered letters were measured as a function of filter peak object frequency at a range of letter sizes.

The third issue addressed was whether restricting the object-spatial-frequency content of letters increases the degree of similarity between contrast sensitivity functions for letters and D6 patterns. Specifically, we compared contrast sensitivity functions for letters that were bandpass filtered at a fixed object spatial frequency with contrast sensitivity functions for D6 patterns. Comparisons were made at two different temporal frequencies, as in the first part of the study.

2.   SUBJECTS AND METHODS

Subjects
Two of the authors, ages 30 and 46, with normal visual acuity and practice in psychophysical experiments, served as subjects.

Test Stimuli
Contrast sensitivity was measured for three types of target: D6 patterns, standard Sloan letters, and Sloan letters that were spatially bandpass filtered with two-dimensional cosine log filters. D6 patterns were defined by a sixth spatial derivative of a Gaussian in one direction and by a Gaussian in the orthogonal direction. The space constant of the Gaussian was 0.8 times the peak frequency of the D6 pattern, so that the patterns were approximately circular.[20] An illustration of a D6 pattern is presented in Fig. 1. A set of ten Sloan letters (C, D, H, K, N, O, R, S, V, Z) was constructed according to published guidelines.[14] These Sloan letters were then spatially bandpass filtered with a set of five cosine log filters[19] with peak object spatial frequencies of 0.63, 1.25, 2.5, 5, and 10 cycles/letter. Illustrations of an unfiltered Sloan letter and the cosine log filtered versions are presented in Fig. 2. Object spatial frequencies lower than 0.63 cycle/letter were not used because, as described by Parish and Sperling[4] and confirmed by our pilot testing, these low object spatial frequencies contained little useful information about letter identity. Object spatial frequencies higher than 10 cycles/letter were not used, because the letter spectra contained little amplitude at these object spatial frequencies.

Test stimuli were generated by an Apple Macintosh IIfx microcomputer and were displayed on an Apple high-resolution gray-scale monitor that had a P4 phosphor and a vertical scan rate of 66.67 Hz. Stimuli were presented in the center of a rectangular adapting field that was displayed continuously throughout the session. The stimulus display was viewed monocularly through a phorometer with a best refractive correction, and a 2-mm artificial pupil was used to control the retinal illuminance of the stimuli. The display monitor, which was the only source of illumination in the test area, was placed behind the subjects, with stray light shielded by black cloth, and the letters were viewed in a front-surface mirror. Except where indicated, the viewing distance was 7.2 m, at which distance the adapting field subtended 1.7° horizontally and 1.3° vertically. Stimulus luminances were controlled by an ISR video attenuator and VideoToolbox software, as described by Pelli and Zhang.[21] Linearized color lookup tables that were loaded during the video retrace periods defined the pixel luminances for each video frame.

The peak retinal spatial frequency (ω_p) of the D6 patterns was defined as in previous studies[15]:

$$\omega_p = \sqrt{3}/\pi\sigma_s, \tag{1}$$

where σ_s is the space constant of the D6 pattern. As in previous studies,[15] the contrast of the D6 patterns was defined by a Weber-type formulation:

$$C_{D6} = (L_{peak} - L_{mean})/L_{mean}, \tag{2}$$

where L_{peak} refers to the maximum luminance (or, for patterns of negative contrast, minimum luminance) and L_{mean} refers to the space-average luminance, which was 1.9 log troland (Td) as measured with a Spectra Spotmeter and an EG&G model 550 photometer. Sloan letter size was specified in log minimum angle of resolution (MAR) units, and letter contrast was defined according to the Weber definition:
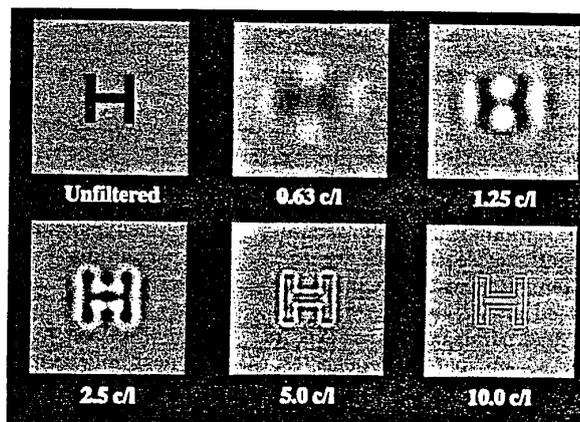
$$C_L = (L_T - L_B)/L_B, \tag{3}$$

Fig. 2. Illustration of an unfiltered Sloan letter H and spatially bandpass filtered versions, with use of cosine log filters with peak object spatial frequencies of 0.63, 1.25, 2.5, 5.0, and 10.0 cycles/letter.
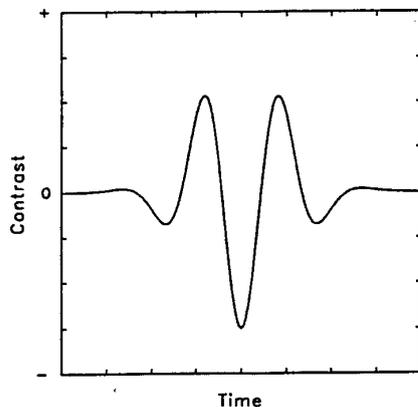
Fig. 3. Temporal waveform used to modulate the stimulus contrast. When plotted as luminance versus spatial position, this function also represents the horizontal spatial luminance profile of the D6 pattern shown in Fig. 1.

where L_T and L_B were the luminances of the target and background, respectively, and $L_B = 1.9$ log Td.

The spatial properties of the filtered letters were defined in terms of object spatial frequencies, which were also converted to retinal spatial frequencies on the basis of viewing distance. Because the contrast of complex images, such as bandpass-filtered letters, is difficult to define satisfactorily,[19,22] we defined the contrast of the filtered images relative to the contrast of the unfiltered letters from which they were derived. That is, when the contrast of the original unfiltered letter was 1.0, each of the filtered images was assigned a relative contrast of 1.0, regardless of the actual spatial distribution of luminance values. For example, each of the filtered images in Fig. 2 was considered to have a relative contrast of 1.0, since the original letter had a contrast of 1.0. When the contrast of the unfiltered letter was reduced by some proportion, then the relative contrast of the filtered images was considered to have been reduced by an equal proportion.

To restrict the temporal-frequency content of the test stimuli and to limit stimulus duration, we modulated the stimulus contrast temporally by a D6 waveform, as illustrated in Fig. 3. The peak temporal frequency was defined as in Eq. (1), with σ_t (time constant) substituted for σ_s. At the point of maximum contrast during stimulus presentation, all stimuli had negative contrast (i.e., $L_{peak} < L_{mean}$ and $L_T < L_B$), as illustrated in Figs. 1 and 2. The temporal characteristics of the stimuli were confirmed by a photocell and an oscilloscope.

Procedure

Before each of the testing sessions the subjects were given a series of practice trials in which the test stimuli were presented at a variety of sizes and contrasts. During the experimental sessions thresholds were measured with a forced-choice staircase procedure. The starting contrast (or size, when appropriate) was chosen by the experimenter, typically at a presumed suprathreshold level. A brief warning tone preceded each stimulus presentation. For D6 patterns, subjects judged whether the target was vertical or horizontal on each trial (i.e., two-alternative

forced-choice), with the orientation determined by a Fellows sequence.[23] The initial staircase reversal point was approached with a one-down, one-up decision rule. Subsequently a three-down, one-up staircase decision rule was used,[24] according to which contrast (or size, as appropriate) was reduced by one step following three consecutive correct responses and was increased by one step following an incorrect response. This rule provides an estimate of the 79% correct point on a psychometric function.[25] For letter stimuli (both unfiltered and filtered versions), one of the ten Sloan letters was randomly presented on each trial, and subjects were required to identify the letter verbally (i.e., ten-alternative forced-choice). The initial staircase reversal point was approached with a one-down, one-up decision rule. Then, a two-down, one-up staircase decision rule was used, which provides a measure of the 71% correct point on a psychometric function.[25] Different decision rules were used for D6 patterns and letters, because the number of alternatives was different; in each case the decision rule provided a point on the psychometric function that was slightly greater than halfway between chance and 100% performance. The interstimulus interval was typically 2 to 3 s, during which time the subject's response was entered into the computer by the examiner and the next stimulus was generated. No feedback was given. Each condition was tested twice, and contrast thresholds were defined as the means of 12 staircase reversals, with sensitivity defined as the reciprocal of threshold.

In the first experiment, contrast thresholds were measured for D6 patterns and Sloan letters at temporal frequencies of 2 and 16 Hz for a range of retinal spatial frequencies (D6 patterns) or log MAR values (letters). For the two largest stimulus sizes the viewing distance was 2.7 m; for all other sizes the viewing distance was 7.2 m. Pilot studies confirmed that varying the test distance over this range (and consequently changing the adapting field subtense) had no effect on contrast thresholds for a target of a fixed angular subtense. We also measured size thresholds for letters and D6 patterns, presented at maximum contrast, at temporal frequencies of 0.5, 1, 2, 4, 8, and 16 Hz at a test distance of 7.2 m. We used a forced-choice staircase procedure in which log retinal spatial frequency (D6) or log MAR (letters) was varied in 0.1-log-unit steps.

In the second experiment, we measured contrast thresholds for spatially filtered Sloan letters presented at a temporal frequency of 2 Hz. First, an intermediate size of Sloan letter was filtered with the set of five cosine log filters. Then the resulting set of filtered letters was presented at each of three test distances: 13.2, 7.2, and 2.7 m. At these test distances the unfiltered Sloan letter from which the filtered letters were derived had log MAR values of 0.0, 0.3, and 0.7 (Snellen equivalents of 20/20, 20/40, and 20/100, respectively).

In the third experiment we measured contrast thresholds for cosine log filtered letters that had a peak object spatial frequency of 2.5 cycles/letter. As in the first experiment, the test distance was 2.7 m for the two largest targets and 7.2 m for all other sizes. The unfiltered letters from which the filtered letters were derived ranged in size from 0.1 to 1.3 log MAR (20/25–20/400 Snellen
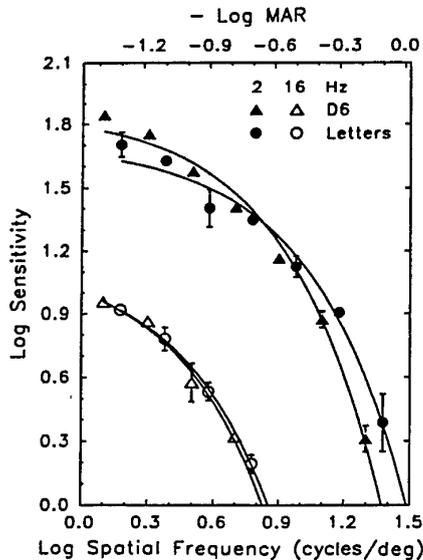
- Log MAR

Fig. 4.   Mean contrast sensitivity functions for D6 patterns and for unfiltered Sloan letters presented at temporal frequencies of 2 and 16 Hz.   Data for the D6 patterns are plotted relative to the lower x axis; data for letters are plotted relative to the upper x axis, which represents the log of the reciprocal of MAR.   The two x axes are equated such that a spatial frequency of 30 cycles/deg corresponds to a log MAR value of 0.   Data points represent the means for two subjects; the error bars represent ±1 SEM.   The curves represent the least-squares best fits of Eq. (4) to each data set.

equivalent).   The corresponding filtered letters had peak retinal spatial frequencies ranging from 1.4 to 0.2 log cycles/deg.

3.   RESULTS

Contrast Sensitivity Functions for D6 Patterns and Sloan Letters

Contrast sensitivity functions for Sloan letters and D6 patterns presented at temporal frequencies of 2 and 16 Hz are shown in Fig. 4.   The data points in this and subsequent figures represent the mean results for the two subjects, and the error bars indicate standard errors of the means (SEM's).   In this figure both data sets have been plotted in the format of contrast sensitivity functions.   Contrast sensitivity values for the D6 patterns have been plotted in terms of log retinal spatial frequency.   To permit direct comparison with these results, we have plotted the contrast sensitivity values for Sloan letters in terms of the log of the reciprocal of MAR (−log MAR).   The data sets for the two types of test stimulus have been equated such that a log MAR value of 0 (20/20 Snellen equivalent) corresponded to a retinal spatial frequency of 30 cycles/deg, as per convention.[5]   The curves in Fig. 4 represent least-squares best fits of the log form of an equation that has been used previously to describe contrast sensitivity functions[26]:

$$s = a \exp(-pf), \qquad (4)$$

where s represents sensitivity, f represents either the peak retinal spatial frequency (D6 patterns) or the reciprocal of MAR (letters), and a and p are scaling parameters representing vertical and horizontal positions,

respectively, on log–log coordinates.   The data sets for both letters and D6 patterns are reasonably well fitted by this equation.

There was a considerable degree of similarity between the contrast sensitivity functions for D6 patterns and for letters, particularly at a temporal frequency of 16 Hz.   At that temporal frequency the parameter estimates (and standard errors of the estimates) for log a were 1.18 (0.05) and 1.16 (0.01) and for (linear) p were 0.40 (0.04) and 0.38 (0.01), for D6 patterns and for letters, respectively (the value of p for letters was corrected for the scaling difference between the horizontal axes).   The statistically similar values of the parameter estimates for D6 patterns and letters indicate that the data were comparable at a temporal frequency of 16 Hz.

However, at a temporal frequency of 2 Hz there were systematic differences between the results for D6 patterns and for letters.   Contrast sensitivity was lower for letters than for D6 patterns at low spatial frequencies (large log MAR values) and higher for letters than for D6 patterns at high spatial frequencies (small log MAR values).   In addition, the curve for letters extended to smaller stimulus sizes.   The parameter estimates (and standard errors of the estimates) for log a were 1.87 (0.04) and 1.71 (0.04) and for p were 0.18 (0.01) and 0.15 (0.01), for D6 patterns and for letters, respectively (the value of p for letters was corrected for the scaling difference between the horizontal axes).   The differences between the parameter estimates in units of the common standard error were 4.00 and 3.00 for log a and p, respectively, which indicates that data for D6 patterns and letters were significantly different at a temporal frequency of 2 Hz.

To confirm this differential effect of temporal frequency on contrast sensitivity for the two types of test stimulus, we measured size thresholds for D6 patterns and letters at a range of temporal frequencies with test stimuli at maximum contrast.   This procedure is equivalent to measuring visual acuity.   The results are shown in Fig. 5.   As in Fig. 4, the data sets for D6 patterns and letters were equated such that a log MAR value of 0 corresponded to a retinal spatial frequency of 30 cycles/deg.   At temporal frequencies of 2 Hz and below there was a difference of approximately 0.2 log unit between the two data sets, consistent with the data shown in Fig. 4.   The difference between the data sets decreased at higher temporal frequencies.   These results confirm that temporal frequency had a systematic effect on the relationship between the contrast sensitivity values for D6 patterns and for letters.

Contrast Sensitivity for Cosine Log Filtered Letters

The effect of restricting the object-spatial-frequency content of letters on contrast sensitivity for letter identification is illustrated in Fig. 6.   We have normalized the data in Fig. 6 by plotting contrast sensitivities for the cosine log filtered letters relative to the contrast sensitivities of the unfiltered letters from which the filtered images were derived.   We adopted this procedure to permit comparisons among the shapes of the functions for the different object spatial frequencies without regard to the actual contrast sensitivity values for the unfiltered letters.   In fact, as shown in Fig. 4, the contrast sensitivities for the unfiltered letters changed by approximately 1 log unit over this range of letter sizes (0.7–0.0 log MAR).
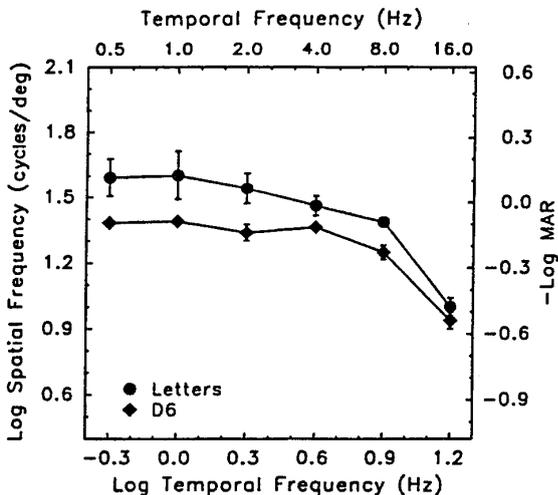
Fig. 5. Mean values of log peak spatial frequency (D6 patterns) and log reciprocal of MAR (letters) as a function of temporal frequency. Data for D6 patterns are plotted relative to the left-hand *y* axis; data for letters are plotted relative to the right-hand *y* axis. The two *y* axes are equated such that a spatial frequency of 30 cycles/deg corresponds to a log MAR value of 0. Data points represent the mean for two subjects; error bars represent ±1 SEM.
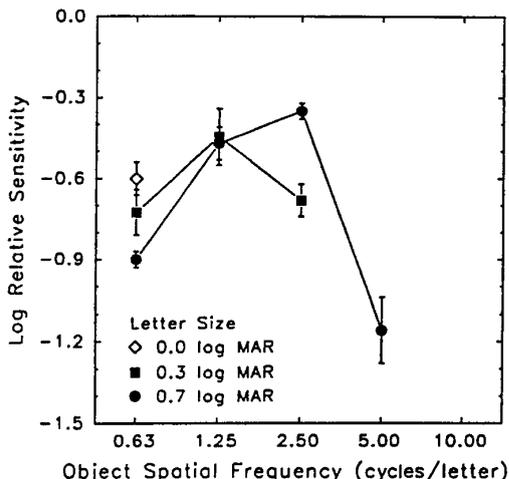
Fig. 6. Mean log relative contrast sensitivities for cosine log filtered letters with five peak object spatial frequencies, presented at a temporal frequency of 2 Hz. Data are plotted on an object-spatial-frequency axis, and contrast sensitivities for filtered letters are plotted relative to the contrast sensitivities for the unfiltered letters from which the filtered letters were derived. Error bars represent ± SEM.

The data shown in Fig. 6 represent the means for the two subjects; the individual subjects showed identical patterns of results. The object spatial frequency of greatest sensitivity differed depending on the log MAR value of the unfiltered letter. For the largest letter size (0.7 log MAR; 20/100 equivalent), the maximum contrast sensitivity occurred at a peak object spatial frequency of 2.5 cycles/letter. For a letter size of 0.3 log MAR (20/40 equivalent), the maximum contrast sensitivity occurred at a peak object spatial frequency of 1.25 cycles/letter.

However, for the smallest letter (0.0 log MAR; 20/20 equivalent), threshold measurements could be made only at an object spatial frequency of 0.63 cycle/letter. At higher object spatial frequencies the contrast threshold exceeded the maximum available contrast.

Figure 7 shows the same contrast sensitivities plotted in terms of retinal rather than object spatial frequencies. The maximum contrast sensitivity occurred at nearly the same retinal spatial frequency (approximately 7 cycles/deg) regardless of letter size. Consequently, even though letter log MAR changed by a factor of 5 (0.7 log unit), the retinal spatial frequency of maximum sensitivity remained relatively constant over this range.

Contrast Sensitivity Functions for D6 Patterns and Filtered Letters

The effect of restricting the object-spatial-frequency content of letters is shown in Fig. 8. This figure presents a comparison between contrast sensitivity functions obtained with D6 patterns and with letters that were filtered at a peak object spatial frequency of 2.5 cycles/letter, which is the object spatial frequency of maximal sensitivity for large letters (see Fig. 6). Measurements were made at temporal frequencies of 2 and 16 Hz. The data points for D6 patterns and the solid curves fitted to these data have been replotted from Fig. 4. The symbols fitted by the dashed curves represent the measured contrast sensitivities for the cosine log filtered letters. The parameter estimates (and standard errors of the estimates) for log a and p for the dashed curves were 1.51 (0.02) and 0.22 (0.01) at 2 Hz and 0.78 (0.02) and 0.48 (0.02) at 16 Hz, respectively.

To facilitate a comparison with the data for D6 patterns, the data points for filtered letters were also all shifted vertically by 0.4 log unit (squares falling along the solid curves), which represents the mean difference between the parameter estimates for log a for D6 patterns and filtered letters. When normalized in this way, the data for the spatially filtered letters corresponded
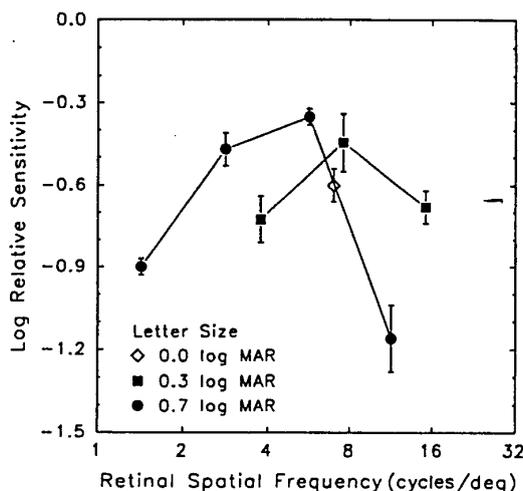

Fig. 7. Mean log relative contrast sensitivities for cosine log filtered letters, replotted from Fig. 6 on a retinal-spatial-frequency axis. Data points represent the means for two subjects; error bars represent ±1 SEM.
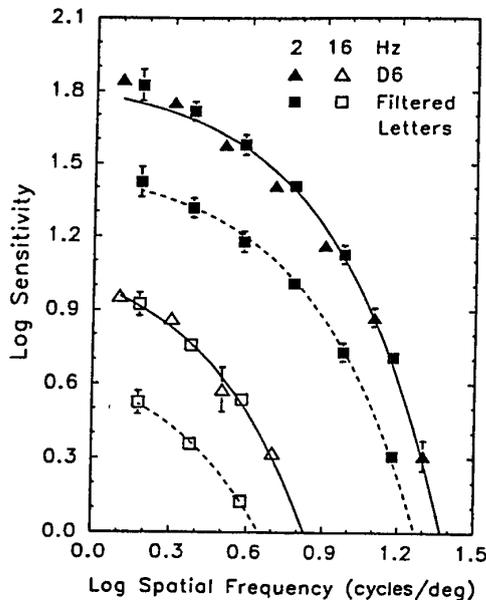
Fig. 8. Mean contrast sensitivity functions for D6 patterns and letters filtered with a cosine log filter that had a peak object spatial frequency of 2.5 cycles/letter. Test stimuli were presented at temporal frequencies of 2 and 16 Hz. The curves represent the least-squares best fits of Eq. (4) to the data sets. The solid curves were fitted to the contrast sensitivities for D6 patterns, replotted from Fig. 4. The dashed curves were fitted to the contrast sensitivities for filtered letters. The squares along the solid curves represent the data for filtered letters displaced vertically by 0.4 log unit, which was the mean difference between the values of log a for the filtered letters and for D6 patterns. Data points represent the means for two subjects; error bars represent ±1 SEM.

well to those for the D6 patterns at both temporal frequencies. This result is in contrast to the findings for unfiltered letters (Fig. 4), where the degree of correspondence between the results for Sloan letters and the D6 patterns differed systematically with temporal frequency. Consequently, restricting the spatial-frequency content of letters by spatial-bandpass filtering results in contrast sensitivity functions that are quite similar in shape to those for D6 patterns at both temporal frequencies.

4. DISCUSSION

Stimuli that are restricted in spatial-frequency content, such as D6 patterns and Gabor patches, are advantageous for studying the properties of spatiotemporal visual mechanisms in visually normal individuals. However, such stimuli may not be sensitive to visual pathology in persons with disorders of the visual system. For example, because of spurious resolution, the use of periodic test stimuli such as gratings may underestimate the extent of uncorrected refractive error.[12,18] Moreover, because of spatial aliasing, the use of periodic test stimuli may misrepresent the extent of visual abnormality in diseases in which there is considerable spatial undersampling.[12,27] Since letters are nonperiodic stimuli, they are less subject to these factors and thus appear to have an advantage in probing the extent of visual loss in disorders of the visual system.[12,18] For example, uncorrected refrac-

tive error has a greater effect on letter identification than on grating resolution; this phenomenon is most likely due to the greater importance of spatial-phase information to letter identification.[12,18,28] Nevertheless, as discussed in Section 1, letters contain a broad range of object spatial frequencies, which can potentially complicate the theoretical interpretation of test results when such stimuli are used.

Our first experiment showed that, despite the broad spatial-frequency content of letters, contrast sensitivity functions for letter identification were quite similar in shape to those for stimuli that were more restricted in spatial-frequency content. In fact, contrast sensitivities for Sloan letters and for D6 patterns were essentially equivalent at a temporal frequency of 16 Hz (Fig. 4). Nevertheless, at lower temporal frequencies there were small but systematic differences between the contrast sensitivity values for the two types of stimulus. At a temporal frequency of 2 Hz, contrast sensitivity was higher for D6 patterns than for letters at low spatial frequencies (large log MAR values), whereas contrast sensitivity was higher for letters than for D6 patterns at high spatial frequencies (small log MAR values).

For D6 patterns the effect of temporal frequency on contrast sensitivity (Fig. 4) was as expected. At low spatial frequencies the difference between the log contrast sensitivity values for D6 patterns at the two temporal frequencies was approximately equal to the log of the ratio of the temporal frequencies (0.9 log unit), a relationship that represents temporal summation. As the spatial frequency of the D6 patterns increased, the difference between the contrast sensitivity functions at the two temporal frequencies increased, indicating a selective sensitivity loss at high spatial frequencies, consistent with previous studies.[29] For Sloan letters the difference in log sensitivity values at large letter sizes was not so great as for D6 patterns, and there was a proportionally greater loss of sensitivity with increasing temporal frequency at small letter sizes than there was for D6 patterns.

The fact that the differences between the contrast sensitivity functions for letters and D6 patterns varied systematically with temporal frequency indicates that these differences were not due to methodological factors, such as the use of different staircase decision rules or different definitions of stimulus contrast. Instead, our second experiment suggested that the most likely explanation for the systematic differences between the contrast sensitivity functions at low temporal frequencies is that different object-spatial-frequency ranges were involved in letter identification as log MAR varied. At intermediate and large letter sizes the object spatial frequency of greatest sensitivity was approximately 2.5 cycles/letter (Fig. 6), which is in general agreement with the results of previous studies.[4,5,7,8] If this were the object spatial frequency that is most relevant for letter identification at all letter sizes, then a letter size of 0.0 log MAR (20/20 Snellen equivalent) would correspond to a retinal spatial frequency of 30 cycles/deg, which is the conventional method for specifying the relationship between letter log MAR and retinal spatial frequency[5] and was the basis for equating the log MAR value of letters with the log spatial frequency of D6 patterns in Figs. 3 and 4.

However, our results showed that the object spatial

frequency of maximum sensitivity was not constant but varied systematically with letter size. For example, at a letter size of 0.0 log MAR the object spatial frequency of greatest sensitivity was displaced to 0.63 cycle/letter (Fig. 6), as might be expected because of the attenuation of the higher object spatial frequencies by the optical and neural properties of the visual system.[30] This result is in agreement with the hypothesis of Thorn and Schwartz[18] but is in contrast to the findings of Parish and Sperling,[4] who reported that the highest efficiency in discriminating letters in noise occurred at a constant object spatial frequency despite changes in viewing distance (letter size). The most likely explanation for this apparent difference in findings is that their range of retinal spatial frequencies was not so high as ours, extending only to 2.3 cycles/deg, whereas our spatially filtered letters had peak retinal spatial frequencies extending to approximately 15 cycles/deg (Figs. 7 and 8). Other possible reasons for the differences between the findings are that their filter width was generally 2 octaves and was high pass for the highest object spatial frequency, whereas our spatial filters were 1 octave wide at half-height; and their images were normalized, so that their filtered images at high object frequencies were effectively contrast enhanced, whereas our images were not normalized.

As a result of the shift in peak object spatial frequency with letter size (Fig. 6), the peak retinal spatial frequency remained relatively constant at approximately 7 cycles/deg over the range of letter sizes that were tested (20/100–20/20; Fig. 7). This finding implies that the retinal spatial frequency that is tested by a conventional visual acuity chart also remains fairly constant as letter size decreases. As a result, the reduction in performance that typically occurs with decreasing size of Sloan letters must be due to factors other than a change in retinal spatial frequency, such as the fact that lower object spatial frequencies contain less information about letter identity[4] (cf. Fig. 2). Consequently, although log MAR is a reasonable metric for specifying letter size, it is somewhat misleading for one to equate letter size measured in units of log MAR with spatial frequency measured in cycles per degree of visual angle in order to derive contrast sensitivity functions based on letter identification. This difference between log MAR and retinal spatial frequency is likely to account for the fact that the data points for letters and D6 patterns in Fig. 4 do not lie along the same curve at high spatial frequencies (small log MAR values).

When letters were bandpass filtered at a peak object spatial frequency of 2.5 cycles/letter, then contrast sensitivity functions were quite similar in shape for filtered letters and D6 patterns at both temporal frequencies of presentation (Fig. 8). This finding is consistent with the hypothesis that letter identification can be predicted on the basis of the spatial-frequency components of letters in combination with the contrast response properties of the visual system.[31] The fact that the data for filtered letters in Fig. 8 were displaced downward from the data for the D6 patterns implies that more than this single 1-octave band of object spatial frequencies is involved in the identification of standard Sloan letters. That is, since we defined the contrast of the filtered letters relative to the contrast of the unfiltered letters, then the contrast sensitivity values for the filtered letters in Fig. 8 would

have corresponded to the contrast sensitivity values for unfiltered letters shown in Fig. 4 if letter identification were based solely on a single object-frequency band centered at 2.5 cycles/letter. The overall lower contrast sensitivity for filtered letters than for unfiltered letters (and for D6 patterns) indicates that more than this restricted band of object spatial frequencies is involved in the identification of unfiltered letters. A similar conclusion can be drawn from the data in Figs. 6 and 7, in which the log relative sensitivities for filtered letters were all less than 0.0. This means that a higher contrast was required for the identification of the filtered letters than was present at that object-spatial-frequency band in the original unfiltered letters. Therefore, identification of unfiltered letters could not be based on that object-spatial-frequency band alone.

Although contrast sensitivity functions for spatially filtered letters are similar to those for D6 patterns when letters are filtered at an object spatial frequency of 2.5 cycles/letter, it is not apparent whether this is the optimal filter peak frequency for all letters. For example, our examination of the object-spatial-frequency spectrum of Sloan letters indicated that the Sloan letter S has higher object-spatial-frequency components than the Sloan letter Z. A cosine log filter centered on 2.5 cycles/letter might pass a higher proportion of the spectrum of the letter S than of the letter Z, whereas a different relationship would obtain at other object-spatial-frequency bands. This could account for the observation that letter confusions can change systematically as a function of letter size.[31] In addition, font characteristics have an important influence on letter confusions,[32] and it is likely that results obtained with Sloan letters may not be representative of letter identification with other letter sets.

In conclusion, restricting the spatial-frequency content of letters increases the similarity between contrast sensitivity functions for letters and for patterns that are inherently more delimited in spatial-frequency content, such as D6 patterns. This result suggests that it is necessary to limit the object-spatial-frequency content of letters for adequate testing of the contrast sensitivity of specific visual mechanisms. Whether spatially filtered letters will prove useful in testing disorders of the visual system remains to be determined.

ACKNOWLEDGMENTS

REFERENCES AND NOTES

1. D. Regan and D. Neima, "Low-contrast letter charts as a test of visual function," Ophthalmology **90**, 1192–1200 (1983).
2. D. G. Pelli, J. G. Robson, and A. J. Wilkins, "The design of a new letter chart for measuring contrast sensitivity," Clin. Vis. Sci. **2**, 187–199 (1988).

3. D. Regan, "Do letter charts measure contrast sensitivity?" Clin. Vis. Sci. **6**, 401–408 (1991).
4. D. H. Parish and G. Sperling, "Object spatial frequencies, retinal spatial frequencies, noise, and the efficiency of letter discrimination," Vision Res. **31**, 1399–1415 (1991).
5. D. Regan, J. Raymond, A. Ginsburg, and T. J. Murray, "Contrast sensitivity, visual acuity and the discrimination of Snellen letters in multiple sclerosis," Brain **104**, 333–350 (1981).
6. As discussed by Parish and Sperling,[4] object spatial frequency refers to the spatial properties of a stimulus without regard to viewing distance and is usually specified in cycles per stimulus dimension, such as letter width (cycles/letter). In comparison, retinal spatial frequency is governed by viewing distance and is specified in units such as cycles per degree (cycles/deg) of visual angle.
7. A. P. Ginsburg, "Visual information processing based upon spatial filters constrained by biological data." Ph.D. dissertation (Cambridge University, Cambridge, UK, 1978; reprinted as AFAMRL Tech. Rep. 78-129, Library of Congress 79-600156).
8. G. E. Legge, D. G. Pelli, G. S. Rubin, and M. M. Schleske, "Psychophysics of reading—I. Normal vision," Vision Res. **25**, 239–252 (1985).
9. B. Howland, A. Ginsburg, and F. Campbell, "High-pass spatial frequency letters as clinical optotypes," Vision Res. **18**, 1063–1066 (1978).
10. A. Medina and B. Howland, "A novel high-frequency visual acuity chart," Ophthalmol. Physiol. Opt. **8**, 14–18 (1988).
11. G. E. Legge, G. S. Rubin, and A. Luebker, "Psychophysics of reading—V. The role of contrast in normal vision," Vision Res. **27**, 1165–1177 (1987).
12. P. R. Herse and H. E. Bedell, "Contrast sensitivity for letter and grating targets under various stimulus conditions," Optom. Vis. Sci. **66**, 774–781 (1989).
13. K. R. Alexander, D. J. Derlacki, and G. A. Fishman, "Contrast thresholds for letter identification in retinitis pigmentosa," Invest. Ophthalmol. Vis. Sci. **33**, 1846–1852 (1992).
14. National Academy of Sciences–National Research Council, "Recommended standard procedures for the clinical measurement and specification of visual acuity: report of working group 39," Adv. Ophthalmol. **41**, 103–148 (1980).
15. W. H. Swanson, H. R. Wilson, and S. C. Giese, "Contrast matching data predicted from contrast increment thresholds," Vision Res. **24**, 63–75 (1984).
16. The Fourier transform and spatial frequency properties of D6 patterns are given in H. R. Wilson, D. K. McFarlane, and G. C. Phillips, "Spatial frequency tuning of orientation selective units estimated by oblique masking," Vision Res. **23**, 873–882 (1983).
17. A. B. Watson and J. G. Robson, "Discrimination at threshold: labelled detectors in human vision," Vision Res. **21**, 1115–1122 (1981).
18. F. Thorn and F. Schwartz, "Effects of dioptric blur on Snellen and grating acuity," Optom. Vis. Sci. **67**, 3–7 (1990).
19. E. Peli, "Contrast in complex images," J. Opt. Soc. Am. A **7**, 2032–2040 (1990).
20. W. H. Swanson and E. E. Birch, "Infant spatiotemporal vision: dependence of spatial contrast sensitivity on temporal frequency," Vision Res. **30**, 1033–1048 (1990).
21. D. G. Pelli and L. Zhang, "Accurate control of contrast on microcomputer displays," Vision Res. **31**, 1337–1350 (1991).
22. A problem with the use of standard contrast definitions such as the Weber, Michelson, or rms formulation for spatially filtered letters is that the individual letters within a set have quite different luminance profiles and therefore have markedly different contrast values by these definitions. Equating the filtered letters in terms of these standard contrast definitions then results in stimuli with very different degrees of visibility. Since a major goal of the present study was to compare contrast sensitivity for spatially filtered letters with contrast sensitivity for unfiltered letters, we adopted the procedure for specifying relative contrast that is described in the text.
23. B. J. Fellows, "Chance stimulus sequences for discrimination tasks," Psychol. Bull. **67**, 87–92 (1967).
24. R. S. Schlauch and R. M. Rose, "Two-, three-, and four-interval forced-choice staircase procedures: estimator bias and efficiency," J. Acoust. Soc. Am. **88**, 732–740 (1990).
25. H. Levitt, "Transformed up–down methods in psychoacoustics," J. Acoust. Soc. Am. **49**, 467–477 (1970).
26. H. R. Wilson and S. C. Giese, "Threshold visibility of frequency gradient patterns," Vision Res. **17**, 1177–1190 (1977).
27. A. M. Geller, P. A. Sieving, and D. G. Green, "Effect on grating identification of sampling with degenerate arrays," J. Opt. Soc. Am. A **9**, 472–477 (1992).
28. R. S. Anderson, Y.-Z. Wang, and L. N. Thibos, "Factors affecting letter discrimination in the fovea and periphery," Invest. Ophthalmol. Vis. Sci. Suppl. **33**, 1344 (1992).
29. R. F. Hess and R. J. Snowden, "Temporal properties of human visual filters: number, shapes, and spatial covariation," Vision Res. **32**, 47–59 (1992).
30. M. S. Banks, W. S. Geisler, and P. J. Bennett, "The physical limits of grating visibility," Vision Res. **27**, 1915–1924 (1987).
31. M. J. Gervais, L. O. Harvey, and J. O. Roberts, "Identification confusions among letters of the alphabet," J. Exp. Psychol. **10**, 655–666 (1984).
32. H. Bouma, "Visual recognition of isolated lower-case letters," Vision Res. **11**, 459–474 (1971).

Adaptive Coding of Monochrome and Color Images

WEN-HSIUNG CHEN AND C. HARRISON SMITH, MEMBER, IEEE

Abstract—An efficient adaptive encoding technique using a new implementation of the Fast Discrete Cosine Transform (FDCT) for bandwidth compression of monochrome and color images is described. Practical system application is attained by maintaining a balance between complexity of implementation and performance. FDCT sub-blocks are sorted into four classes according to level of image activity, measured by the total ac energy within each sub-block. Adaptivity is provided by distributing bits between classes, favoring higher levels of activity over lower levels. Excellent performance is demonstrated in terms of mean square error and direct comparison of original and reconstructed images. Results are presented for both noiseless and noisy transmission at a total rate of 1 bit and 0.5 bit per pixel for a monochrome image and for a total rate of 2 bits and 1 bit per pixel for a color image. In every case the total bit rate includes all overhead required for image reconstruction and bit protection.

I. INTRODUCTION

NUMEROUS bandwidth compression techniques have been proposed in response to the continually increasing requirements for transmission and storage of high resolution imagery. Adaptive transform encoding based upon the statistics of imagery data has proved to be very effective but introduces system complexity. The key to a practical system implementation lies in selecting an efficient coding scheme which achieves a successful compromise between complexity and performance. This paper describes an efficient adaptive coding technique using the fast discrete cosine transform. Transform blocks are sorted into classes by the level of image activity present. Within each activity (ac energy) level, coding bits are allocated to individual transform elements according to the variance matrix of the transformed data. Bits are then distributed between "busy" and "quiet" image areas to provide the desired adaptivity—more bits being assigned to the areas of high image activity and fewer bits to those of low activity. Use of the fast discrete cosine transform (FDCT) provides high energy compaction and helps to simplify implementation. Transform elements are nonuniformly quantized and coded as prescribed by the bit allocation matrices of the various energy levels. Overhead is held to a minimum consistent with performance. Mean square error between original and reconstructed images is used as the primary performance criterion although subjective performance is exercisable by visual inspection of the monochrome and color image coding examples presented.

The discrete cosine transform representation and its properties are briefly described in Section II. In Section III the

statistical properties of cosine transform samples are reviewed. Section IV contains a detailed description of the FDCT adaptive image coding system for monochrome images. Section V describes the extension of the adaptive scheme to color images. Simulated examples of monochrome and color images are presented and discussed in Section VI.

II. COSINE TRANSFORM

The one dimensional cosine transform of a discrete function $f(j), j = 0, 1, \cdots, N - 1$, is defined as [1]

$$F(u) = \frac{2c(u)}{N} \sum_{j=0}^{N-1} f(j) \cos \left[\frac{(2j + 1)u\pi}{2N} \right] ;$$

$$u = 0, 1, \cdots, N - 1 \qquad (1)$$

where

$$c(u) = \frac{1}{\sqrt{2}}, \qquad \text{for } u = 0$$

$$= 1, \qquad \text{for } u = 1, 2, \cdots, N - 1$$

$$= 0, \qquad \text{elsewhere}$$

and the inverse transform is

$$f(j) = \sum_{u=0}^{N-1} c(u)F(u) \cos \left[\frac{(2j + 1)u\pi}{2N} \right] ;$$

$$j = 0, 1, \cdots, N - 1. \qquad (2)$$

In the class of transforms with a known fast computational algorithm the cosine transform has a superior energy compaction property.

Figure 1 contains a plot of the mean square error of an image with a Markov process covariance as a function of block size for various transformations [2]. In this plot the 25% of the coefficients containing the largest variances were selected and the remainder discarded. It is seen from the figure that the cosine transform results in virtually the same energy compaction performance as the Karhunen-Loeve transform, known to be the best in the mean square error sense. The transform possesses a circular convolution-multiplication relationship which can readily be used in linear system theory [3]. The transform can also be calculated using a computational algorithm which is different from the conventional approach. The conventional approach of implementing the discrete cosine transform involves computation of a double size fast Fourier transform (FFT) of $2N$ coefficients employing complex arithmetic [1]. An alternative algorithm, known as a Fast Discrete Cosine Transform (FDCT), requiring only real opera-
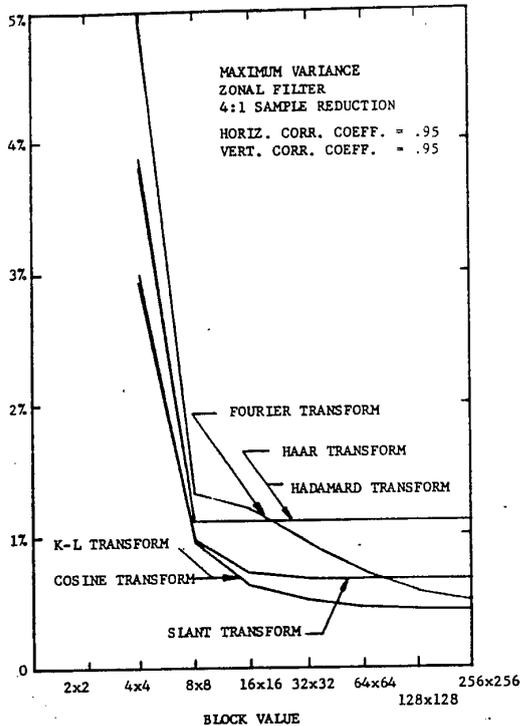
Figure 1.   Mean square error performance of image transform as a function of block size.

Figure 2.   Fast cosine transform flowchart, $N = 16$, $Ci = \cos(\pi i/32)$, $Si = \sin(\pi i/32)$.

tions on a set of N points is also possible. A general method for derivation of the FDCT for any desired value of $N = 2^m$, $m \geqslant 2$ has recently been found [4]. The method requires $(3N/2)(\log_2 N - 1) + 2$ real additions and $N \log_2 N - (3N/2) + 4$ real multiplications and is six times faster than the conventional approach. An example of this faster algorithm for $N = 16$ is shown in flowchart form in Figure 2.

III. STATISTICAL PROPERTIES OF COSINE TRANSFORM SAMPLES

Statistical image information plays an important role in developing an efficient coder. This section presents some of the statistical properties which are useful in devising the adaptive coder.

A. Mean and Variance

The one dimensional cosine transform discussed in the previous section can easily be extended to two dimensions. Suppose that the image pixel $f(j, k)$ is a sample of a random process with mean m. Then $E\{f(j, k)\}$ can be replaced by m and the expected value of the cosine transform samples can be expressed as

$$E\{F(u, v)\} = \frac{4mc(u)c(v)}{N^2} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} \cos \frac{(2j + 1)u\pi}{2N}$$
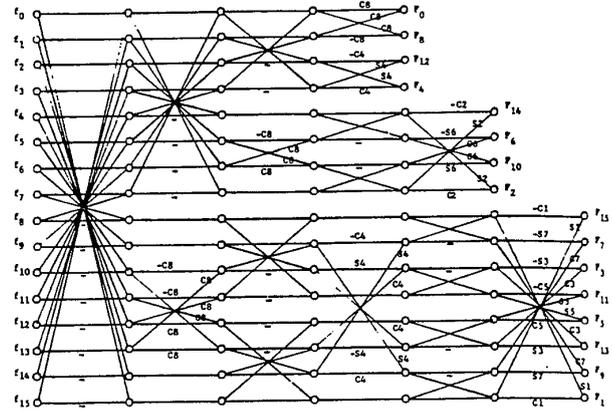$$\cdot \cos \frac{(2k + 1)v\pi}{2N} \qquad (3)$$

or

$$E\{F(0, 0)\} = 2m \qquad (4a)$$

and

$$E\{F(u, v)\} = 0; \qquad (u, v) \neq 0. \qquad (4b)$$

With the mean denoted as above the variance of the transform samples can be computed from

$$\sigma^2(0, 0) = E\{[F(0, 0)]^2\} - 4m^2 \qquad (5a)$$

and

$$\sigma^2(u, v) = E\{[F(u, v)]^2\}; \qquad (u, v) \neq 0. \qquad (5b)$$

B. Probability Model

Since each cosine transform domain sample is formed from the cosine weighted sum of all the pixels in the original image, the central limit theorem [5] tells us that the probability density functions can be modeled approximately by Gaussian densities, i.e.,

$$p_{(0,0)}(x) = \frac{1}{\sqrt{2\pi}\sigma(0, 0)} \exp\left\{-\frac{(x - 2m)^2}{2\sigma^2(0, 0)}\right\} \qquad (6a)$$

and

$$p_{(u,v)}(x) = \frac{1}{\sqrt{2\pi}\sigma(u, v)} \exp\left\{-\frac{x^2}{2\sigma^2(u, v)}\right\};$$
$$(u, v) \neq 0. \qquad (6b)$$

IV. COSINE TRANSFORM ADAPTIVE CODING OF MONOCHROME IMAGES

The activity levels of images are proportional to the transform domain ac energy within the images. AC energy can be associated with image detail—each dc sample in the transform domain is excluded since it determines only the brightness level. All transform sub-blocks can then be classifed into various groups according to the ac energy inside the sub-blocks.
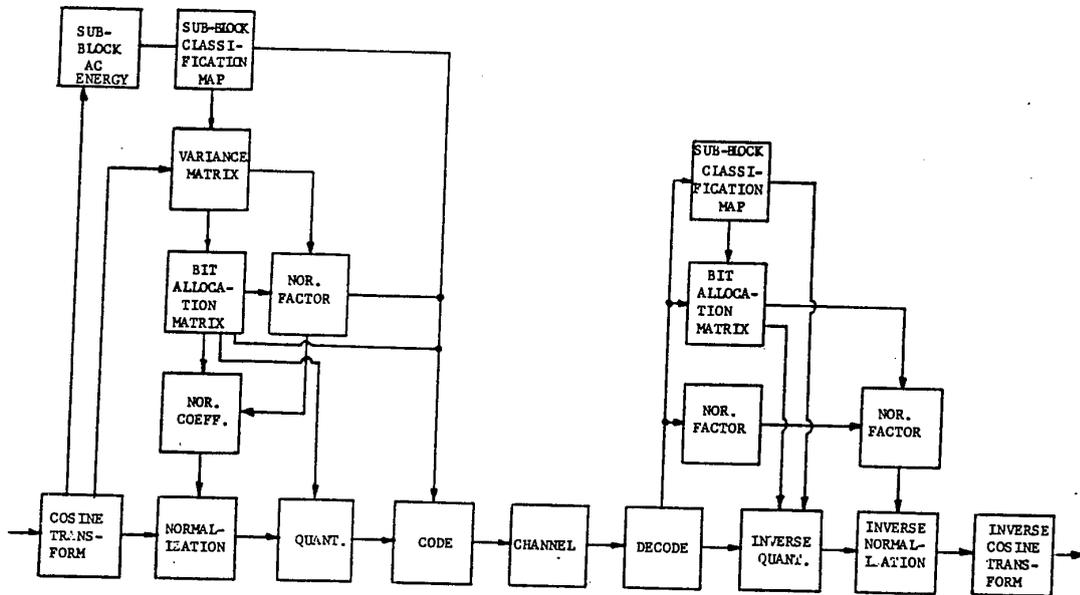
Figure 3.   Cosine transform adaptive coding system.

Adaptivity is provided by assigning more bits to the higher energy levels and fewer bits to the lower energy levels. This not only provides good quality reconstruction for the high activity regions but also achieves an efficient coding for the low activity regions. For adaptivity to be successful the span of adaptivity must be sufficiently large to draw statistically significant parameters from the image data. Coding in small sub-blocks meets this requirement and is also computationally efficient. In the system to be described, the entire digitized image is divided into a number of 16 × 16 pixel sub-blocks and a two dimensional FDCT performed on each sub-block.

Figure 3 contains a block diagram of the adaptive coding system for monochrome images. The sum of the ac energy of the transform samples within each sub-block is first calculated and accepted as the measure of the level of activity of that sub-block. Next, a histogram is constructed and the transform blocks are classified into four groups according to this measure of activity. The normalized transform samples within each class are then nonuniformly quantized and adaptively coded. The quantization method used here is Max's optimal quantization-scheme [6] with probability density of the transform sample modeled as equation (6).

For real time applications it is necessary to make two passes: the first pass to generate the sub-block classification maps, set up the bit assignment matrices, and calculate the normalization factor; the second pass to multiply by the normalization factor, quantize the transform coefficients, code the data, add the overhead information and transmit the resulting data over a channel. This is accomplished by a one-frame delay in which the first pass data are processed and stored while the previous second pass data are coded and transmitted. This is practicable if the frame size is restricted to a value compatible with storage capability and first pass processing time. At the receiver, the received data are decoded and an inverse cosine transform is performed to reconstruct the image.
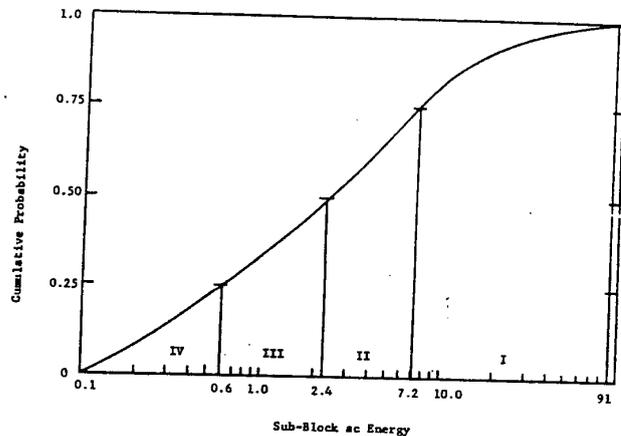

Figure 4.   Probability distribution of sub-block ac energy.

A. Transform Sub-block Classification

After the FDCT is performed the transform sub-blocks are classified by level of activity (ac energy).

The ac energy in the (m, l)th sub-block can be defined as

$$E_{m,l} = \sum_{u=0}^{15} \sum_{v=0}^{15} [F_{m,l}(u, v)]^2 - [F_{m,l}(0, 0)]^2 ;$$

$$m, l = 1, 2, \cdots, M/16* \qquad (7)$$

Figure 4 shows a typical cumulative probability distribution of ac energy within sub-blocks. The energy has been classified into four levels, nonuniformly separated by the .25, .50, and .75 of the cumulative probability distribution, such that each class contains $\frac{1}{4}(M/16)^2$ sub-blocks. The classification of the

*Unless otherwise specified, the image is represented by $M \times M$ pixels.

transform sub-blocks into equally populated levels provides a simple way to ensure that the average coding rate over the entire image is maintained. The number of classification levels is generally dependent upon the image size, the relative population of the activity levels, the degree of change of activity within an image and the average bits required to code the compressed data. While not necessarily optimum, four equally populated levels of classification are sufficient to code images with an average code rate of 0.5 bits per pixel or more.

The classification maps serve as an index or reference to the proper bit allocation matrices and become a part of the overhead information needed by the receiver to decode the received data. Figure 5 illustrates a typical classification map for the central 20 X 20 transform sub-blocks for the monochrome image shown in Fig. 8a. The classification of sub-blocks by the levels of activity is clearly demonstrated.

B. Adaptive Coding

The ensemble average of the variance of each of the transform samples within each class is next computed. From equation (5) we have

$$\sigma_k{}^2(0,0) = \frac{256K}{M^2} \sum_{m=1}^{16\sqrt{K}} \sum_{l=1}^{16\sqrt{K}} [F_{m,l}(0,0)]^2 - 4m^2 \quad (8a)$$

and

$$\sigma_k{}^2(u,v) = \frac{256K}{M^2} \sum_{m=1}^{16\sqrt{K}} \sum_{l=1}^{16\sqrt{K}} [F_{m,l}(u,v)]^2,$$

$$(u,v) \neq 0 \quad (8b)$$

where $k = 1, 2, \cdots, K$ and K is the number of sub-block classifications. Once the variances for each class are computed, the bit allocation matrix for that class then can be determined. The bit assignment, $N_{B_k}(u,v)$, for each bit allocation matrix is based upon a relationship from rate distortion theory. [7, 8] The number of bits is given by

$$N_{B_k}(u,v) = \frac{1}{2} \log_2 [\sigma_k{}^2(u,v)] - \log_2 [D] ;$$

$$(u,v) \neq (0,0) \quad (9)$$

where $\sigma_k{}^2(u,v)$ is the variance of a transform sample as defined in equation (8) and D is a parameter. Bits are assigned by initializing D and iteratively calculating the summation of $N_{B_k}(u,v)$ in equation (9) until the desired total number of bits is achieved. If the exact total number of bits desired is not reached after a fixed number of iterations, bits are arbitrarily adjusted. Figure 6 illustrates typical bit allocation matrices for the monochrome image with an average code bit of 1 bit/pixel.

Referring to the block diagram of Figure 3, the incoming transform samples are now normalized by a normalization coefficient $\sigma_k{}'(u,v)$:

$$\sigma_k{}'(u,v) = c \cdot 2^{N_{B_k}(u,v)-1}, \quad (u,v) \neq (0,0) \quad (10)$$
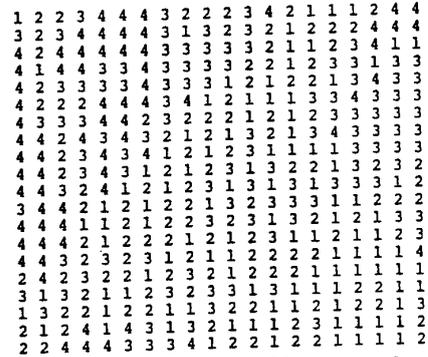

Figure 5. Classification of the central 20 X 20 transform sub-blocks for the monochrome image. 1 specifies the highest activity sub-blocks; 4 specifies the lowest activity sub-blocks.
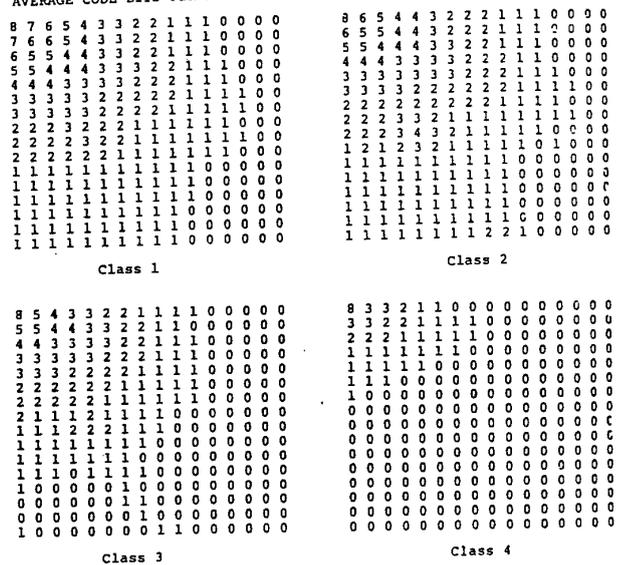

Figure 6. Bit allocation matrices for the monochrome image.

where $N_{B_k}(u,v)$ is as defined in equation (9) (see bit tables in Figure 6) and c is the normalization factor. In equation (10) if $N_{B_k}(u,v)$ is assigned to one bit the normalization factor c will equal $\sigma_k{}'(u,v)$. Since $\sigma_k{}'(u,v)$ is simply the estimation of $\sigma_k(u,v)$, c can be determined during the bit assignment process by setting it equal to the maximum standard deviation of those elements in the variance matrix which were assigned to one bit. The maximum value is selected rather than the average to avoid excessive clipping. As for the dc samples, eight bits are assigned to avoid brightness differences at sub-block boundaries.

For an image of N_D bits dynamic range the maximum possible value for the dc coefficients for a $2D$ cosine transform can be determined from equation (11) as $2(2^{N_D} - 1)$. Therefore, if the dc coefficient is assigned at eight bits the dc normalization factor is $2(2^{N_D} - 1)$ divided by the highest eight bit decision level with unit variance. This guarantees that none of the dc samples are clipped. The dc normalization factor need not be transmitted as long as the dc bit assignment

is fixed. A normalization factor assigned to each transform sample might yield slightly increased performance but would lead to excessive overhead. Next, the normalized samples are optimally quantized with the number of quantization levels (bits) set according to the bit allocation matrices. The Max's quantization scheme [6] is used here since it gives the optimal result for a given probability density function. The probability density model used in finding decision and reconstruction levels in the Max's scheme is defined in equation (6). The optimally quantized samples are then coded into a string of binary data. These data incorporated with their overhead information are then transmitted over a channel.

C. Overhead

Overhead is the information necessary to decode the compressed image data at the receiving end of the channel. In the adaptive coding system just described, the total overhead information consists of one classification map, one normalization factor, and four bit tables. This can be written as

$$B = B_c + B_n + B_t \tag{11}$$

where B_c, B_n, and B_t represent the average overhead code bits required to code the classification map, the normalization factor and the bit allocation tables, respectively. They are represented by the following set of equations:

$$B_c = \frac{2\left(\frac{M}{16}\right)^2}{M^2} = \frac{1}{128} \text{ bits} \tag{12}$$

$$B_n = \frac{c'}{M^2} \text{ bits} \tag{13}$$

$$B_t = \frac{h(16)^2 \cdot 4}{M^2} \text{ bits.} \tag{14}$$

where c' is the number of bits used to represent the normalization factor c. The factor h is a function of the compression ratio. Utilizing an efficient coder such as the Huffman code, h is equal to 1.62 for the 1 bit case and 1.41 for the 0.5 bit case, based upon the fact that the majority of bit assignments equal zero for large compression ratios.

It is known that the inherent "error averaging" property of transform coding provides image coding superior to conventional spatial coding. However, any adaptive transform coding technique such as the one used here tends to propagate channel errors and some form of bit protection is desirable. Since little degradation in the image itself results at channel bit error rates of 10^{-4} or less, only the overhead part of the data requires error correction to avoid catastrophic errors and produce reliable data transmission. In each of the subsequent examples the allowance for overhead has been doubled for BER = 10^{-4} and tripled for BER = 10^{-2} for bit protection.

It can be computed that for the errorless case less than 0.034 bits of overhead information is needed to code a one bit monochrome image of 256 × 256 pixels. For a one bit monochrome image of 1024 × 1024 pixels, the overhead information s less than 0.01 bit.
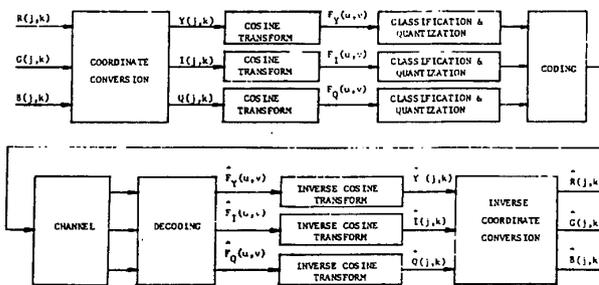

Figure 7.   Cosine transform color image coding system.

V. COSINE TRANSFORM ADAPTIVE CODING OF COLOR IMAGES

Figure 7 contains a block diagram of a cosine transform color image coding system. In the system, the color image represented by three source tristimulus signals $R(j, k)$, $G(j, k)$, $B(j, k)$ at coordinate (j, k) are first converted to a new three dimensional space, $Y(j, k), I(j, k), Q(j, k)$ according to

$$
\begin{bmatrix} Y(j, k) \\ I(j, k) \\ Q(j, k) \end{bmatrix} =
\begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.253 & 0.312 \end{bmatrix}
\begin{bmatrix} R(j, k) \\ G(j, k) \\ B(j, k) \end{bmatrix}. \tag{15}
$$

The conversion compacts most of the color energy into the Y plane such that a more efficient design of the adaptive coder can be accomplished. The converted YIQ signals then individually undergo a two dimensional cosine transform in 16 × 16 pixel sub-blocks to form three transform planes $F_Y(u, v)$, $F_I(u, v)$, and $F_Q(u, v)$. The adaptive coding process as described in coding the monochrome images is then applied to each color plane and the resulting code bits are transmitted serially over the channel. At the receiver, the receiver code bits are decoded, and inverse cosine transform and inverse coordinate conversions are performed to reconstruct the source tristimulus signals. The inverse coordinate conversion is described by

$$
\begin{bmatrix} R(j, k) \\ G(j, k) \\ B(j, k) \end{bmatrix} =
\begin{bmatrix} 1.000 & 0.956 & 0.621 \\ 1.000 & -0.272 & -0.647 \\ 1.000 & -1.106 & 1.703 \end{bmatrix}
\begin{bmatrix} Y(j, k) \\ I(j, k) \\ Q(j, k) \end{bmatrix}. \tag{16}
$$

VI. EXPERIMENTAL RESULTS

A series of computer simulations has been conducted to evaluate the performance of the cosine transform adaptive coder. All images were generated with the DICOMED, Model D47, Image Recorder manufactured by Dicomed Corp., Minneapolis, MN. It is worthwhile to mention that there is no trial and error process involved in simulating the coder. Every result to be presented here is based on a "one shot" experiment.

Each set of simulation results have been grouped into a single figure consisting of four quadrants, a, b, c, and d, for ease of comparison. The original image in all cases is at the

CHEN

Figure 8. Simulation results for compression of monochrome image at one bit per pixel (see text). (a) Original. (b) Reconstructed image with BER = 0. (c) BER = 10^{-4}. (d) BER = 10^{-2}.
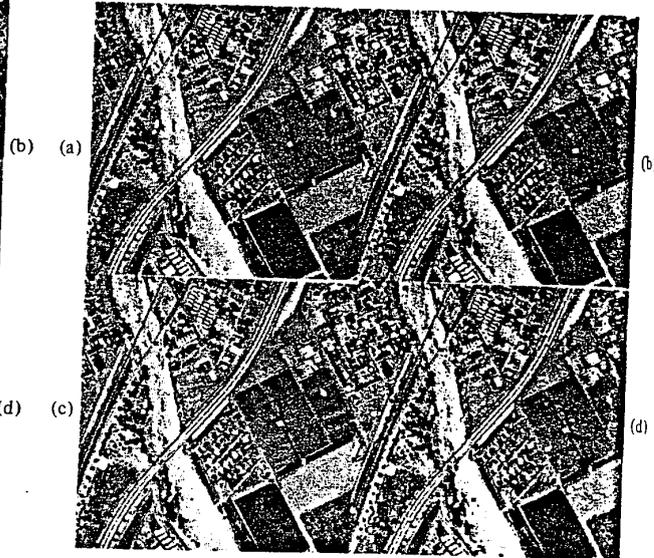

Figure 9. Simulation results for compression of monochrome image at 0.5 bit per pixel (see text). (a) Original. (b) Reconstructed image with BER = 0. (c) BER = 10^{-4}. (d) BER = 10^{-2}.

Figure 10. Simulation results for compression of color image at 2 bits per pixel (see text). (a) Original. (b) Reconstructed image with BER = 0. (c) BER = 10^{-4}. (d) BER = 10^{-2}.

upper left in quadrant *a*. The reconstructed image without channel error is at upper right in quadrant *b*. The reconstructed images for channel errors of 10^{-4} and 10^{-2} are at lower left, quadrant *c*, and lower right, quadrant *d*, respectively.

The original monochrome image shown in Figure 8(a) contains 1024 × 1024 pixels with each intensity value uniformly quantized to 256 levels (8 bits per pixel). Figure 8 is

the set of simulation results for illustrating the performance for compression at one bit per pixel. Figure 9 shows performance at one-half bit per pixel.

The original color image shown in Figure 10(a) contained 256 × 256 pixels with each red, green, and blue tristimulus value uniformly quantized to 256 levels (24 bits per pixel). Figure 10 is the set of simulation results for adaptive compres-

sion at
reconst
corresp
In both
of chan
For
is classi
two le\
ponent:
compor
Figu
versus
square ε
square
input v
images.
is due t
image.

The
minimal
The cla:
vides ef
Adaptivi
rather tl
more bit
than to
protecti:
by bett:
devised

Figure 11. Simulation results for compression of color image at 1 bit per pixel (see text). (a) Original. (b) Reconstructed image with BER = 0. (c) BER = 10^{-4}. (d) BER = 10^{-2}.

sion at 2 bits per pixel. Figure 10(b) shows the high quality reconstruction without channel error. Figure 11 is the corresponding set of results for compression at 1 bit per pixel. In both sets of images, quadrants (c) and (d) show the effect of channel BER of 10^{-4} and 10^{-2}, respectively.

For simplicity in simulation each F_Y, F_I and F_Q component is classified into four classes, even though the classifications of two levels may have been enough for the F_I and F_Q components. The bit distributions among the F_Y, F_I, and F_Q components are determined empirically from reference 8.

Figure 12 shows a plot of normalized mean square error versus average code bits for these images. Excellent mean square error performance of the scheme is demonstrated. Mean square error has been normalized to the maximum reference input value of 256 levels for both monochrome and color images. The apparent superior performance of the color image is due to the much greater level of detail in the monochrome image.

VII. SUMMARY

The adaptive coding technique described herein requires minimal overhead information yet performs extremely well. The classification of image sub-blocks by activity level provides efficient coding for the transform domain elements. Adaptivity in coding provides code bits to individual samples rather than to specific zones. As can be seen from Figure 6, more bits are sometimes assigned to higher frequency samples than to nearby lower frequency samples. This feature helps in protecting high frequency information and in reducing errors by better reconstruction of large transform samples. The devised scheme is based upon analytical optimization of the

$$MSE = \frac{\frac{1}{N^2} \sum_j \sum_k [f(j,k) - \hat{f}(j,k)]^2}{255^2}$$
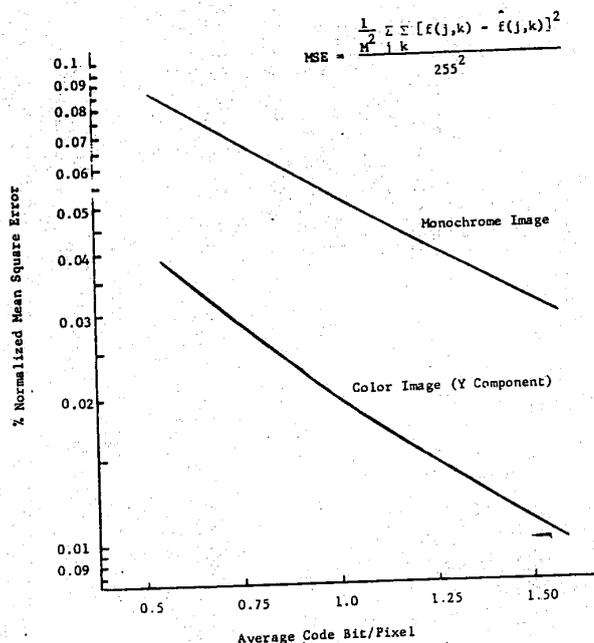
Figure 12. Mean square error performance of adaptive cosine transform coder.

mean square quantization error where the deleted samples are treated as part of the quantization noise. Further optimization may be possible by iteration of parameters such as the normalization factor c but might not be acceptable in a practical application. The low mean square error (MSE) achieved, e.g.,

MSE = 0.049% for the 1 bit monochrome image and MSE = 0.019% for the 1 bit color image (Y component) and the excellent subjective performance would imply that further optimization may not be rewarding.

REFERENCES

1. N. Ahmed, T. Natarjan, and K. R. Rao, "Discrete Cosine Transform," *IEEE Transactions on Computers*, January 1974, pp. 90-93.
2. W. K. Pratt, W. Chen, and L. R. Welch, "Slant Transform Image Coding," *IEEE Transactions on Communications*, Vol. COM-22, No. 8, August 1974, pp. 1075-1093.
3. W. Chen and S. Fralick, "Image Enhancement Using Cosine Transform Filtering," Proceedings of the Symposium on Current Mathematical Problems in Image Science, Monterey, CA, November 1976, pp. 186-192.
4. W. Chen, C. H. Smith, and S. Fralick, "A Fast Computational Algorithm for the Discrete Cosine Transform," *IEEE Transactions on Communications*, September 1977.
5. A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, New York: McGraw-Hill, 1965.
6. J. Max, "Quantization for Minimum Distortion," *IRE Transactions on Information Theory*, March 1960, pp. 7-12.
7. J. Pearl, H. C. Andrews, and W. K. Pratt, "Performance Measure for Transform Data Coding," *IEEE Transactions on Communications*, Vol. COM-20, No. 3, June 1972, pp. 411-415.
8. L. D. Davisson, "Rate-Distortion Theory and Application," *Proceedings of IEEE*, Vol. 60, July 1972, pp. 800-828.

★

Wen-hsiung Chen received the B.S. degree from the National Taiwan University, Taipei, Taiwan, in 1962, the M.S. degree from the Kansas State University, Manhattan, in 1966, and the Ph.D. degree from the University of Southern California, Los Angeles, in 1973, all in electrical engineering.

He joined the Electrical Engineering Department of the National Taiwan University as a Teaching Assistant in 1963. From 1966 to 1968 he was employed with the Allis-Chalmers Company, Chicago, IL, as an Electrical Engineer. He was a Teaching and Research Assistant at the Electrical Engineering Department of the University of Southern California from 1968 to 1972, and was a Research Associate at the Image Processing Institute of USC in 1973. From October 1973 to June 1977 he was a Senior Engineering Specialist with the Ford Aerospace & Communications Corporation, Palo Alto, CA. Since July 1977, he has been employed with Compression Labs, Inc., Campbell, CA as Chief Scientist in charge of R & D work in image processing.

★

C. Harrison Smith (M'40) received the BSEE degree from George Washington University, Washington, DC, in 1939.

From 1940 to 1952 he was employed at the Naval Research Laboratory in Bellevue, DC. Following this he was Chief Electronics Engineer at Chicago Midway Laboratories until January 1955 when he became head of the Guidance & Control Department, Missile Systems Division, Lockheed Aircraft Systems at Van Nuys, CA. In Dec. 1955, he left to help form a new company, Systems Research Corp. in Van Nuys, CA. In July 1956, this company was purchased by Ford Motor Company and became Aeronutronic Systems, Inc., finally locating in Newport Beach, CA. There, Mr. Smith's interests were primarily in telemetering and tactical weapons. In late 1968, he was transferred to his present location with Ford Aerospace & Communications Corp., Western Development Laboratories Division at Palo Alto, CA. His current interests include digital systems simulation, signal processing, and image processing.

Effects of Round-Off Noise on Hadamard Transformed Imagery

JOHN J. KNAB, MEMBER, IEEE

Abstract—The effects of hardware round-off error when using the two-dimensional Hadamard transform to code 8 by 8 pixel subpicture blocks of imagery are presented. The simulations show that at least 8 bits of precision are desirable.

INTRODUCTION

IN RECENT years, the two-dimensional Hadamard transform has been quite successfully employed in reducing the amount of digital data needed to adequately represent a two-dimensional image [1]. Simulations [1] as well as hardware implementation [2] have demonstrated that data compression for TV imagery is practical using real-time Hadamard transform coding techniques.

In this paper we observe the detrimental effects of hardware round-off error when calculating the Hadamard transform coefficients. An image comprised of 512 by 512 pixels, quantized linearly to 6 bits (0 to 63), will be considered. The image is divided into 8 by 8 pixel sub-blocks before the two-dimensional Hadamard transform is applied. Round-off errors are then introduced during the computation of the Hadamard transform. When the inverse Hadamard transform is computed to obtain the original image, no round-off error is introduced. The effects of round-off errors are also examined when a non-

INTERNATIONAL TELECOMMUNICATION UNION

CCITT

THE INTERNATIONAL
TELEGRAPH AND TELEPHONE
CONSULTATIVE COMMITTEE

T.81

(09/92)

TERMINAL EQUIPMENT AND PROTOCOLS
FOR TELEMATIC SERVICES

INFORMATION TECHNOLOGY –
DIGITAL COMPRESSION AND CODING
OF CONTINUOUS-TONE STILL IMAGES –
REQUIREMENTS AND GUIDELINES

Recommendation T.81

Foreword

ITU (International Telecommunication Union) is the United Nations Specialized Agency in the field of telecommunications. The CCITT (the International Telegraph and Telephone Consultative Committee) is a permanent organ of the ITU. Some 166 member countries, 68 telecom operating entities, 163 scientific and industrial organizations and 39 international organizations participate in CCITT which is the body which sets world telecommunications standards (Recommendations).

The approval of Recommendations by the members of CCITT is covered by the procedure laid down in CCITT Resolution No. 2 (Melbourne, 1988). In addition, the Plenary Assembly of CCITT, which meets every four years, approves Recommendations submitted to it and establishes the study programme for the following period.

In some areas of information technology, which fall within CCITT's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC. The text of CCITT Recommendation T.81 was approved on 18th September 1992. The identical text is also published as ISO/IEC International Standard 10918-1.

CCITT NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized private operating agency.

© ITU 1993

Contents

Introduction

This CCITT Recommendation I ISO/IEC International Standard was prepared by CCITT Study Group VIII and the Joint Photographic Experts Group (JPEG) of ISO/IEC JTC 1/SC 29/WG 10. This Experts Group was formed in 1986 to establish a standard for the sequential progressive encoding of continuous tone grayscale and colour images.

Digital Compression and Coding of Continuous-tone Still images, is published in two parts:

- Requirements and guidelines;
- Compliance testing.

This part, Part 1, sets out requirements and implementation guidelines for continuous-tone still image encoding and decoding processes, and for the coded representation of compressed image data for interchange between applications. These processes and representations are intended to be generic, that is, to be applicable to a broad range of applications for colour and grayscale still images within communications and computer systems. Part 2, sets out tests for determining whether implementations comply with the requirments for the various encoding and decoding processes specified in Part 1.

The user's attention is called to the possibility that – for some of the coding processes specified herein – compliance with this Recommendation I International Standard may require use of an invention covered by patent rights. See Annex L for further information.

The requirements which these processes must satisfy to be useful for specific image communications applications such as facsimile, Videotex and audiographic conferencing are defined in CCITT Recommendation T.80. The intent is that the generic processes of Recommendation T.80 will be incorporated into the various CCITT Recommendations for terminal equipment for these applications.

In addition to the applications addressed by the CCITT and ISO/IEC, the JPEG committee has developped a compression standard to meet the needs of other applications as well, including desktop publishing, graphic arts, medical imaging and scientific imaging.

Annexes A, B, C, D, E, F, G, H and J are normative, and thus form an integral part of this Specification. Annexes K, L and M are informative and thus do not form an integral part of this Specification.

This Specification aims to follow the guidelines of CCITT and ISO/IEC JTC 1 on *Rules for presentation of CCITT I ISO/IEC common text.*

INTERNATIONAL STANDARD

CCITT RECOMMENDATION

INFORMATION TECHNOLOGY – DIGITAL COMPRESSION AND CODING OF CONTINUOUS-TONE STILL IMAGES – REQUIREMENTS AND GUIDELINES

1   Scope

This CCITT Recommendation I International Standard is applicable to continuous-tone – grayscale or colour – digital still image data. It is applicable to a wide range of applications which require use of compressed images. It is not applicable to bi-level image data.

This Specification

- specifies processes for converting source image data to compressed image data;

- specifies processes for converting compressed image data to reconstructed image data;

- gives guidance on how to implement these processes in practice;

- specifies coded representations for compressed image data.

   NOTE – This Specification does not specify a complete coded image representation. Such representations may include certain parameters, such as aspect ratio, component sample registration, and colour space designation, which are application-dependent.

2   Normative references

The following CCITT Recommendations and International Standards contain provisions which, through reference in this text, constitute provisions of this CCITT Recommendation I International Standard. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this CCITT Recommendation I International Standard are encouraged to investigate the possibility of applying the most recent edition of the Recommendations and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The CCITT Secretariat maintains a list of currently valid CCITT Recommendations.

- *CCITT Recommendation T.80 (1992), Common components for image compression and communication – Basic principles.*

3   Definitions, abbreviations and symbols

3.1   Definitions and abbreviations

For the purposes of this Specification, the following definitions apply.

3.1.1   abbreviated format: A representation of compressed image data which is missing some or all of the table specifications required for decoding, or a representation of table-specification data without frame headers, scan headers, and entropy-coded segments.

3.1.2   AC coefficient: Any DCT coefficient for which the frequency is not zero in at least one dimension.

3.1.3   (adaptive) (binary) arithmetic decoding: An entropy decoding procedure which recovers the sequence of symbols from the sequence of bits produced by the arithmetic encoder.

3.1.4   (adaptive) (binary) arithmetic encoding: An entropy encoding procedure which codes by means of a recursive subdivision of the probability of the sequence of symbols coded up to that point.

3.1.5   application environment: The standards for data representation, communication, or storage which have been established for a particular application.

3.1.6   **arithmetic decoder:** An embodiment of arithmetic decoding procedure.

3.1.7   **arithmetic encoder:** An embodiment of arithmetic encoding procedure.

3.1.8   **baseline (sequential):** A particular sequential DCT-based encoding and decoding process specified in this Specification, and which is required for all DCT-based decoding processes.

3.1.9   **binary decision:** Choice between two alternatives.

3.1.10   **bit stream:** Partially encoded or decoded sequence of bits comprising an entropy-coded segment.

3.1.11   **block:** An 8 × 8 array of samples or an 8 × 8 array of DCT coefficient values of one component.

3.1.12   **block-row:** A sequence of eight contiguous component lines which are partitioned into 8 × 8 blocks.

3.1.13   **byte:** A group of 8 bits.

3.1.14   **byte stuffing:** A procedure in which either the Huffman coder or the arithmetic coder inserts a zero byte into the entropy-coded segment following the generation of an encoded hexadecimal X'FF' byte.

3.1.15   **carry bit:** A bit in the arithmetic encoder code register which is set if a carry-over in the code register overflows the eight bits reserved for the output byte.

3.1.16   **ceiling function:** The mathematical procedure in which the greatest integer value of a real number is obtained by selecting the smallest integer value which is greater than or equal to the real number.

3.1.17   **class (of coding process):** Lossy or lossless coding processes.

3.1.18   **code register:** The arithmetic encoder register containing the least significant bits of the partially completed entropy-coded segment. Alternatively, the arithmetic decoder register containing the most significant bits of a partially decoded entropy-coded segment.

3.1.19   **coder:** An embodiment of a coding process.

3.1.20   **coding:** Encoding or decoding.

3.1.21   **coding model:** A procedure used to convert input data into symbols to be coded.

3.1.22   **(coding) process:** A general term for referring to an encoding process, a decoding process, or both.

3.1.23   **colour image:** A continuous-tone image that has more than one component.

3.1.24   **columns:** Samples per line in a component.

3.1.25   **component:** One of the two-dimensional arrays which comprise an image.

3.1.26   **compressed data:** Either compressed image data or table specification data or both.

3.1.27   **compressed image data:** A coded representation of an image, as specified in this Specification.

3.1.28   **compression:** Reduction in the number of bits used to represent source image data.

3.1.29   **conditional exchange:** The interchange of MPS and LPS probability intervals whenever the size of the LPS interval is greater than the size of the MPS interval (in arithmetic coding).

3.1.30   **(conditional) probability estimate:** The probability value assigned to the LPS by the probability estimation state machine (in arithmetic coding).

3.1.31   **conditioning table:** The set of parameters which select one of the defined relationships between prior coding decisions and the conditional probability estimates used in arithmetic coding.

3.1.32   **context:** The set of previously coded binary decisions which is used to create the index to the probability estimation state machine (in arithmetic coding).

3.1.33   **continuous-tone image:** An image whose components have more than one bit per sample.

3.1.34   **data unit:** An 8 × 8 block of samples of one component in DCT-based processes; a sample in lossless processes.

3.1.35 **DC coefficient:** The DCT coefficient for which the frequency is zero in both dimensions.

3.1.36 **DC prediction:** The procedure used by DCT-based encoders whereby the quantized DC coefficient from the previously encoded 8 × 8 block of the same component is subtracted from the current quantized DC coefficient.

3.1.37 **(DCT) coefficient:** The amplitude of a specific cosine basis function – may refer to an original DCT coefficient, to a quantized DCT coefficient, or to a dequantized DCT coefficient.

3.1.38 **decoder:** An embodiment of a decoding process.

3.1.39 **decoding process:** A process which takes as its input compressed image data and outputs a continuous-tone image.

3.1.40 **default conditioning:** The values defined for the arithmetic coding conditioning tables at the beginning of coding of an image.

3.1.41 **dequantization:** The inverse procedure to quantization by which the decoder recovers a representation of the DCT coefficients.

3.1.42 **differential component:** The difference between an input component derived from the source image and the corresponding reference component derived from the preceding frame for that component (in hierarchical mode coding).

3.1.43 **differential frame:** A frame in a hierarchical process in which differential components are either encoded or decoded.

3.1.44 **(digital) reconstructed image (data):** A continuous-tone image which is the output of any decoder defined in this Specification.

3.1.45 **(digital) source image (data):** A continuous-tone image used as input to any encoder defined in this Specification.

3.1.46 **(digital) (still) image:** A set of two-dimensional arrays of integer data.

3.1.47 **discrete cosine transform; DCT:** Either the forward discrete cosine transform or the inverse discrete cosine transform.

3.1.48 **downsampling (filter):** A procedure by which the spatial resolution of an image is reduced (in hierarchical mode coding).

3.1.49 **encoder:** An embodiment of an encoding process.

3.1.50 **encoding process:** A process which takes as its input a continuous-tone image and outputs compressed image data.

3.1.51 **entropy-coded (data) segment:** An independently decodable sequence of entropy encoded bytes of compressed image data.

3.1.52 **(entropy-coded segment) pointer:** The variable which points to the most recently placed (or fetched) byte in the entropy encoded segment.

3.1.53 **entropy decoder:** An embodiment of an entropy decoding procedure.

3.1.54 **entropy decoding:** A lossless procedure which recovers the sequence of symbols from the sequence of bits produced by the entropy encoder.

3.1.55 **entropy encoder:** An embodiment of an entropy encoding procedure.

3.1.56 **entropy encoding:** A lossless procedure which converts a sequence of input symbols into a sequence of bits such that the average number of bits per symbol approaches the entropy of the input symbols.

3.1.57 **extended (DCT-based) process:** A descriptive term for DCT-based encoding and decoding processes in which additional capabilities are added to the baseline sequential process.

3.1.58 **forward discrete cosine transform; FDCT:** A mathematical transformation using cosine basis functions which converts a block of samples into a corresponding block of original DCT coefficients.

3.1.59   frame: A group of one or more scans (all using the same DCT-based or lossless process) through the data of one or more of the components in an image.

3.1.60   frame header: A marker segment that contains a start-of-frame marker and associated frame parameters that are coded at the beginning of a frame.

3.1.61   frequency: A two-dimensional index into the two-dimensional array of DCT coefficients.

3.1.62   (frequency) band: A contiguous group of coefficients from the zig-zag sequence (in progressive mode coding).

3.1.63   full progression: A process which uses both spectral selection and successive approximation (in progressive mode coding).

3.1.64   grayscale image: A continuous-tone image that has only one component.

3.1.65   hierarchical: A mode of operation for coding an image in which the first frame for a given component is followed by frames which code the differences between the source data and the reconstructed data from the previous frame for that component. Resolution changes are allowed between frames.

3.1.66   hierarchical decoder: A sequence of decoder processes in which the first frame for each component is followed by frames which decode an array of differences for each component and adds it to the reconstructed data from the preceding frame for that component.

3.1.67   hierarchical encoder: The mode of operation in which the first frame for each component is followed by frames which encode the array of differences between the source data and the reconstructed data from the preceding frame for that component.

3.1.68   horizontal sampling factor: The relative number of horizontal data units of a particular component with respect to the number of horizontal data units in the other components.

3.1.69   Huffman decoder: An embodiment of a Huffman decoding procedure.

3.1.70   Huffman decoding: An entropy decoding procedure which recovers the symbol from each variable length code produced by the Huffman encoder.

3.1.71   Huffman encoder: An embodiment of a Huffman encoding procedure.

3.1.72   Huffman encoding: An entropy encoding procedure which assigns a variable length code to each input symbol.

3.1.73   Huffman table: The set of variable length codes required in a Huffman encoder and Huffman decoder.

3.1.74   image data: Either source image data or reconstructed image data.

3.1.75   interchange format: The representation of compressed image data for exchange between application environments.

3.1.76   interleaved: The descriptive term applied to the repetitive multiplexing of small groups of data units from each component in a scan in a specific order.

3.1.77   inverse discrete cosine transform; IDCT: A mathematical transformation using cosine basis functions which converts a block of dequantized DCT coefficients into a corresponding block of samples.

3.1.78   Joint Photographic Experts Group; JPEG: The informal name of the committee which created this Specification. The "joint" comes from the CCITT and ISO/IEC collaboration.

3.1.79   latent output: Output of the arithmetic encoder which is held, pending resolution of carry-over (in arithmetic coding).

3.1.80   less probable symbol; LPS: For a binary decision, the decision value which has the smaller probability.

3.1.81   level shift: A procedure used by DCT-based encoders and decoders whereby each input sample is either converted from an unsigned representation to a two's complement representation or from a two's complement representation to an unsigned representation.

ISO/IEC 10918-1 : 1993(E)

3.1.82   lossless: A descriptive term for encoding and decoding processes and procedures in which the output of the decoding procedure(s) is identical to the input to the encoding procedure(s).

3.1.83   lossless coding: The mode of operation which refers to any one of the coding processes defined in this Specification in which all of the procedures are lossless (see Annex H).

3.1.84   lossy: A descriptive term for encoding and decoding processes which are not lossless.

3.1.85   marker: A two-byte code in which the first byte is hexadecimal FF (X'FF') and the second byte is a value between 1 and hexadecimal FE (X'FE').

3.1.86   marker segment: A marker and associated set of parameters.

3.1.87   MCU-row: The smallest sequence of MCU which contains at least one line of samples or one block-row from every component in the scan.

3.1.88   minimum coded unit; MCU: The smallest group of data units that is coded.

3.1.89   modes (of operation): The four main categories of image coding processes defined in this Specification.

3.1.90   more probable symbol; MPS: For a binary decision, the decision value which has the larger probability.

3.1.91   non-differential frame: The first frame for any components in a hierarchical encoder or decoder. The components are encoded or decoded without subtraction from reference components. The term refers also to any frame in modes other than the hierarchical mode.

3.1.92   non-interleaved: The descriptive term applied to the data unit processing sequence when the scan has only one component.

3.1.93   parameters: Fixed length integers 4, 8 or 16 bits in length, used in the compressed data formats.

3.1.94   point transform: Scaling of a sample or DCT coefficient.

3.1.95   precision: Number of bits allocated to a particular sample or DCT coefficient.

3.1.96   predictor: A linear combination of previously reconstructed values (in lossless mode coding).

3.1.97   probability estimation state machine: An interlinked table of probability values and indices which is used to estimate the probability of the LPS (in arithmetic coding).

3.1.98   probability interval: The probability of a particular sequence of binary decisions within the ordered set of all possible sequences (in arithmetic coding).

3.1.99   (probability) sub-interval: A portion of a probability interval allocated to either of the two possible binary decision values (in arithmetic coding).

3.1.100   procedure: A set of steps which accomplishes one of the tasks which comprise an encoding or decoding process.

3.1.101   process: See coding process.

3.1.102   progressive (coding): One of the DCT-based processes defined in this Specification in which each scan typically improves the quality of the reconstructed image.

3.1.103   progressive DCT-based: The mode of operation which refers to any one of the processes defined in Annex G.

3.1.104   quantization table: The set of 64 quantization values used to quantize the DCT coefficients.

3.1.105   quantization value: An integer value used in the quantization procedure.

3.1.106   quantize: The act of performing the quantization procedure for a DCT coefficient.

3.1.107   reference (reconstructed) component: Reconstructed component data which is used in a subsequent frame of a hierarchical encoder or decoder process (in hierarchical mode coding).

3.1.108 renormalization: The doubling of the probability interval and the code register value until the probability interval exceeds a fixed minimum value (in arithmetic coding).

3.1.109 restart interval: The integer number of MCUs processed as an independent sequence within a scan.

3.1.110 restart marker: The marker that separates two restart intervals in a scan.

3.1.111 run (length): Number of consecutive symbols of the same value.

3.1.112 sample: One element in the two-dimensional array which comprises a component.

3.1.113 sample-interleaved: The descriptive term applied to the repetitive multiplexing of small groups of samples from each component in a scan in a specific order.

3.1.114 scan: A single pass through the data for one or more of the components in an image.

3.1.115 scan header: A marker segment that contains a start-of-scan marker and associated scan parameters that are coded at the beginning of a scan.

3.1.116 sequential (coding): One of the lossless or DCT-based coding processes defined in this Specification in which each component of the image is encoded within a single scan.

3.1.117 sequential DCT-based: The mode of operation which refers to any one of the processes defined in Annex F.

3.1.118 spectral selection: A progressive coding process in which the zig-zag sequence is divided into bands of one or more contiguous coefficients, and each band is coded in one scan.

3.1.119 stack counter: The count of X'FF' bytes which are held, pending resolution of carry-over in the arithmetic encoder.

3.1.120 statistical conditioning: The selection, based on prior coding decisions, of one estimate out of a set of conditional probability estimates (in arithmetic coding).

3.1.121 statistical model: The assignment of a particular conditional probability estimate to each of the binary arithmetic coding decisions.

3.1.122 statistics area: The array of statistics bins required for a coding process which uses arithmetic coding.

3.1.123 statistics bin: The storage location where an index is stored which identifies the value of the conditional probability estimate used for a particular arithmetic coding binary decision.

3.1.124 successive approximation: A progressive coding process in which the coefficients are coded with reduced precision in the first scan, and precision is increased by one bit with each succeeding scan.

3.1.125 table specification data: The coded representation from which the tables used in the encoder and decoder are generated and their destinations specified.

3.1.126 transcoder: A procedure for converting compressed image data of one encoder process to compressed image data of another encoder process.

3.1.127 (uniform) quantization: The procedure by which DCT coefficients are linearly scaled in order to achieve compression.

3.1.128 upsampling (filter): A procedure by which the spatial resolution of an image is increased (in hierarchical mode coding).

3.1.129 vertical sampling factor: The relative number of vertical data units of a particular component with respect to the number of vertical data units in the other components in the frame.

3.1.130 zero byte: The X'00' byte.

3.1.131 zig-zag sequence: A specific sequential ordering of the DCT coefficients from (approximately) lowest spatial frequency to highest.

3.1.132 3-sample predictor: A linear combination of the three nearest neighbor reconstructed samples to the left and above (in lossless mode coding).

3.2 Symbols

The symbols used in this Specification are listed below.

A	probability interval
AC	AC DCT coefficient
AC_{ji}	AC coefficient predicted from DC values
Ah	successive approximation bit position, high
Al	successive approximation bit position, low
Ap_i	ith 8-bit parameter in APP_n segment
APP_n	marker reserved for application segments
B	current byte in compressed data
B2	next byte in compressed data when B = X'FF'
BE	counter for buffered correction bits for Huffman coding in the successive approximation process
BITS	16-byte list containing number of Huffman codes of each length
BP	pointer to compressed data
BPST	pointer to byte before start of entropy-coded segment
BR	counter for buffered correction bits for Huffman coding in the successive approximation process
Bx	byte modified by a carry-over
C	value of bit stream in code register
C_i	component identifier for frame
C_u	horizontal frequency dependent scaling factor in DCT
C_v	vertical frequency dependent scaling factor in DCT
CE	conditional exchange
C-low	low order 16 bits of the arithmetic decoder code register
Cm_i	ith 8-bit parameter in COM segment
CNT	bit counter in NEXTBYTE procedure
CODE	Huffman code value
CODESIZE(V)	code size for symbol V
COM	comment marker
Cs	conditioning table value
Cs_i	component identifier for scan
CT	renormalization shift counter
Cx	high order 16 bits of arithmetic decoder code register
CX	conditional exchange
d_{ji}	data unit from horizontal position i, vertical position j
d_{ji}^{k}	d_{ji} for component k
D	decision decoded

Da	in DC coding, the DC difference coded for the previous block from the same component; in lossless coding, the difference coded for the sample immediately to the left
DAC	define-arithmetic-coding-conditioning marker
Db	the difference coded for the sample immediately above
DC	DC DCT coefficient
DC_i	DC coefficient for ith block in component
DC_k	kth DC value used in prediction of AC coefficients
DHP	define hierarchical progression marker
DHT	define-Huffman-tables marker
DIFF	difference between quantized DC and prediction
DNL	define-number-of-lines marker
DQT	define-quantization-tables marker
DRI	define restart interval marker
E	exponent in magnitude category upper bound
EC	event counter
ECS	entropy-coded segment
ECS_i	ith entropy-coded segment
Eh	horizontal expansion parameter in EXP segment
EHUFCO	Huffman code table for encoder
EHUFSI	encoder table of Huffman code sizes
EOB	end-of-block for sequential; end-of-band for progressive
EOBn	run length category for EOB runs
EOBx	position of EOB in previous successive approximation scan
EOB0, EOB1, ..., EOB14	run length categories for EOB runs
EOI	end-of-image marker
Ev	vertical expansion parameter in EXP segment
EXP	expand reference components marker
FREQ(V)	frequency of occurrence of symbol V
H_i	horizontal sampling factor for ith component
H_{max}	largest horizontal sampling factor
HUFFCODE	list of Huffman codes corresponding to lengths in HUFFSIZE
HUFFSIZE	list of code lengths
HUFFVAL	list of values assigned to each Huffman code
i	subscript index
I	integer variable
Index(S)	index to probability estimation state machine table for context index S
j	subscript index
J	integer variable

JPG	marker reserved for JPEG extensions
JPG_n	marker reserved for JPEG extensions
k	subscript index
K	integer variable
Kmin	index of 1st AC coefficient in band (1 for sequential DCT)
Kx	conditioning parameter for AC arithmetic coding model
L	DC and lossless coding conditioning lower bound parameter
L_i	element in BITS list in DHT segment
$L_i(t)$	element in BITS list in the DHT segment for Huffman table t
La	length of parameters in APP_n segment
LASTK	largest value of K
Lc	length of parameters in COM segment
Ld	length of parameters in DNL segment
Le	length of parameters in EXP segment
Lf	length of frame header parameters
Lh	length of parameters in DHT segment
Lp	length of parameters in DAC segment
LPS	less probable symbol (in arithmetic coding)
Lq	length of parameters in DQT segment
Lr	length of parameters in DRI segment
Ls	length of scan header parameters
LSB	least significant bit
m	modulo 8 counter for RST_m marker
m_t	number of $V_{i,j}$ parameters for Huffman table t
M	bit mask used in coding magnitude of V
Mn	nth statistics bin for coding magnitude bit pattern category
MAXCODE	table with maximum value of Huffman code for each code length
MCU	minimum coded unit
MCU_i	ith MCU
MCUR	number of MCU required to make up one MCU-row
MINCODE	table with minimum value of Huffman code for each code length
MPS	more probable symbol (in arithmetic coding)
MPS(S)	more probable symbol for context-index S
MSB	most significant bit
M2, M3, M4, ... , M15	designation of context-indices for coding of magnitude bits in the arithmetic coding models
n	integer variable
N	data unit counter for MCU coding
N/A	not applicable

Nb	number of data units in MCU
Next_Index_LPS	new value of Index(S) after a LPS renormalization
Next_Index_MPS	new value of Index(S) after a MPS renormalization
Nf	number of components in frame
NL	number of lines defined in DNL segment
Ns	number of components in scan
OTHERS(V)	index to next symbol in chain
P	sample precision
Pq	quantizer precision parameter in DQT segment
Pq(t)	quantizer precision parameter in DQT segment for quantization table t
PRED	quantized DC coefficient from the most recently coded block of the component
Pt	point transform parameter
Px	calculated value of sample
Q_{ji}	quantizer value for coefficient AC_{ji}
Q_{vu}	quantization value for DCT coefficient S_{vu}
Q_{00}	quantizer value for DC coefficient
QAC_{ji}	quantized AC coefficient predicted from DC values
QDC_k	kth quantized DC value used in prediction of AC coefficients
Qe	LPS probability estimate
Qe(S)	LPS probability estimate for context index S
Qk	kth element of 64 quantization elements in DQT segment
r_{vu}	reconstructed image sample
R	length of run of zero amplitude AC coefficients
R_{vu}	dequantized DCT coefficient
Ra	reconstructed sample value
Rb	reconstructed sample value
Rc	reconstructed sample value
Rd	rounding in prediction calculation
RES	reserved markers
Ri	restart interval in DRI segment
RRRR	4-bit value of run length of zero AC coefficients
RS	composite value used in Huffman coding of AC coefficients
RST_m	restart marker number m
s_{yx}	reconstructed value from IDCT
S	context index
S_{vu}	DCT coefficient at horizontal frequency u, vertical frequency v

SC	context-index for coding of correction bit in successive approximation coding
Se	end of spectral selection band in zig-zag sequence
SE	context-index for coding of end-of-block or end-of-band
SI	Huffman code size
SIGN	1 if decoded sense of sign is negative and 0 if decoded sense of sign is positive
SIZE	length of a Huffman code
SLL	shift left logical operation
SLL α β	logical shift left of α by β bits
SN	context-index for coding of first magnitude category when V is negative
SOF_0	baseline DCT process frame marker
SOF_1	extended sequential DCT frame marker, Huffman coding
SOF_2	progressive DCT frame marker, Huffman coding
SOF_3	lossless process frame marker, Huffman coding
SOF_5	differential sequential DCT frame marker, Huffman coding
SOF_6	differential progressive DCT frame marker, Huffman coding
SOF_7	differential lossless process frame marker, Huffman coding
SOF_9	sequential DCT frame marker, arithmetic coding
SOF_{10}	progressive DCT frame marker, arithmetic coding
SOF_{11}	lossless process frame marker, arithmetic coding
SOF_{13}	differential sequential DCT frame marker, arithmetic coding
SOF_{14}	differential progressive DCT frame marker, arithmetic coding
SOF_{15}	differential lossless process frame marker, arithmetic coding
SOI	start-of-image marker
SOS	start-of-scan marker
SP	context-index for coding of first magnitude category when V is positive
Sq_{vu}	quantized DCT coefficient
SRL	shift right logical operation
SRL α β	logical shift right of α by β bits
Ss	start of spectral selection band in zig-zag sequence
SS	context-index for coding of sign decision
SSSS	4-bit size category of DC difference or AC coefficient amplitude
ST	stack counter
Switch_MPS	parameter controlling inversion of sense of MPS
Sz	parameter used in coding magnitude of V
S0	context-index for coding of $V = 0$ decision
t	summation index for parameter limits computation
T	temporary variable

Ta$_j$	AC entropy table destination selector for jth component in scan
Tb	arithmetic conditioning table destination identifier
Tc	Huffman coding or arithmetic coding table class
Td$_j$	DC entropy table destination selector for jth component in scan
TEM	temporary marker
Th	Huffman table destination identifier in DHT segment
Tq	quantization table destination identifier in DQT segment
Tq$_i$	quantization table destination selector for ith component in frame
U	DC and lossless coding conditioning upper bound parameter
V	symbol or value being either encoded or decoded
V$_i$	vertical sampling factor for ith component
V$_{i,j}$	jth value for length i in HUFFVAL
V$_{max}$	largest vertical sampling factor
V$_t$	temporary variable
VALPTR	list of indices for first value in HUFFVAL for each code length
V1	symbol value
V2	symbol value
x$_i$	number of columns in ith component
X	number of samples per line in component with largest horizontal dimension
X$_i$	ith statistics bin for coding magnitude category decision
X1, X2, X3, ... , X15	designation of context-indices for coding of magnitude categories in the arithmetic coding models
XHUFCO	extended Huffman code table
XHUFSI	table of sizes of extended Huffman codes
X'values'	values within the quotes are hexadecimal
y$_i$	number of lines in ith component
Y	number of lines in component with largest vertical dimension
ZRL	value in HUFFVAL assigned to run of 16 zero coefficients
ZZ(K)	Kth element in zig-zag sequence of quantized DCT coefficients
ZZ(0)	quantized DC coefficient in zig-zag sequence order

4   General

The purpose of this clause is to give an informative overview of the elements specified in this Specification. Another purpose is to introduce many of the terms which are defined in clause 3. These terms are printed in *italics* upon first usage in this clause.

4.1   Elements specified in this Specification

There are three elements specified in this Specification:

a)   An *encoder* is an embodiment of an *encoding process*. As shown in Figure 1, an encoder takes as input *digital source image data* and *table specifications*, and by means of a specified set of *procedures* generates as output *compressed image data*.

b)   A *decoder* is an embodiment of a *decoding process*. As shown in Figure 2, a decoder takes as input compressed image data and table specifications, and by means of a specified set of procedures generates as output *digital reconstructed image data*.

c)   The *interchange format*, shown in Figure 3, is a compressed image data representation which includes all table specifications used in the encoding process. The interchange format is for exchange between *application environments*.
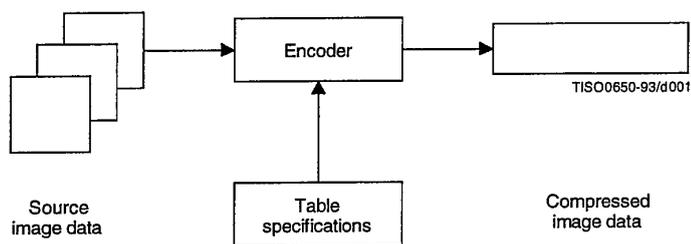
Figure 1 – Encoder
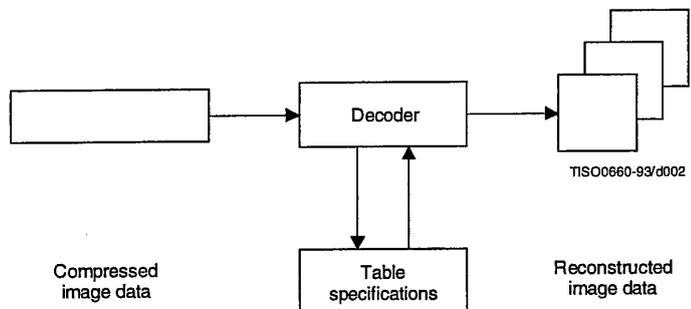
Figure 2 – Decoder

Figures 1 and 2 illustrate the general case for which the *continuous-tone* source and reconstructed image data consist of multiple *components*. (A *colour* image consists of multiple components; a *grayscale* image consists only of a single component.) A significant portion of this Specification is concerned with how to handle multiple-component images in a flexible, application-independent way.
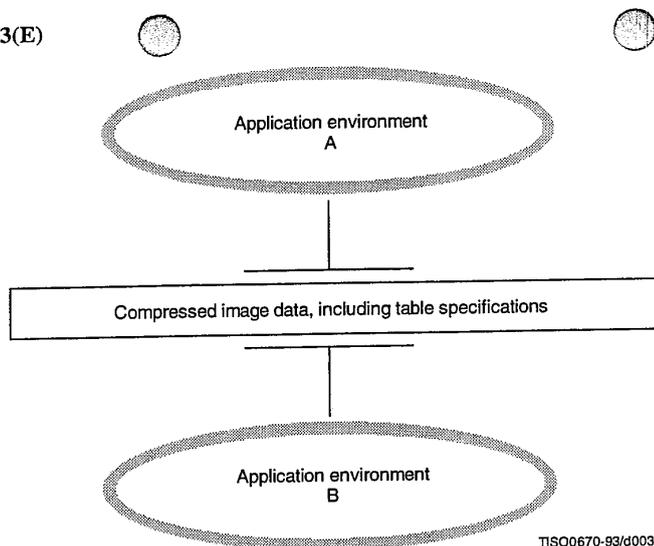
Figure 3 – Interchange format for compressed image data

These figures are also meant to show that the same tables specified for an encoder to use to compress a particular image must be provided to a decoder to reconstruct that image. However, this Specification does not specify how applications should associate tables with compressed image data, nor how they should represent source image data generally within their specific environments.

Consequently, this Specification also specifies the interchange format shown in Figure 3, in which table specifications are included within compressed image data. An image compressed with a specified encoding process within one application environment, A, is passed to a different environment, B, by means of the interchange format. The interchange format does not specify a complete coded image representation. Application-dependent information, e.g. colour space, is outside the scope of this Specification.

4.2 Lossy and lossless compression

This Specification specifies two *classes* of encoding and decoding processes, *lossy* and *lossless* processes. Those based on the *discrete cosine transform* (DCT) are lossy, thereby allowing substantial *compression* to be achieved while producing a reconstructed image with high visual fidelity to the encoder's source image.

The simplest DCT-based *coding process* is referred to as the *baseline sequential* process. It provides a capability which is sufficient for many applications. There are additional DCT-based processes which extend the baseline sequential process to a broader range of applications. In any decoder using *extended DCT-based decoding processes*, the baseline decoding process is required to be present in order to provide a default decoding capability.

The second class of coding processes is not based upon the DCT and is provided to meet the needs of applications requiring lossless compression. These lossless encoding and decoding processes are used independently of any of the DCT-based processes.

A table summarizing the relationship among these lossy and lossless coding processes is included in 4.11.

The amount of compression provided by any of the various processes is dependent on the characteristics of the particular image being compressed, as well as on the picture quality desired by the application and the desired speed of compression and decompression.

4.3   DCT-based coding

Figure 4 shows the main procedures for all encoding processes based on the DCT. It illustrates the special case of a single-component image; this is an appropriate simplification for overview purposes, because all processes specified in this Specification operate on each image component independently.
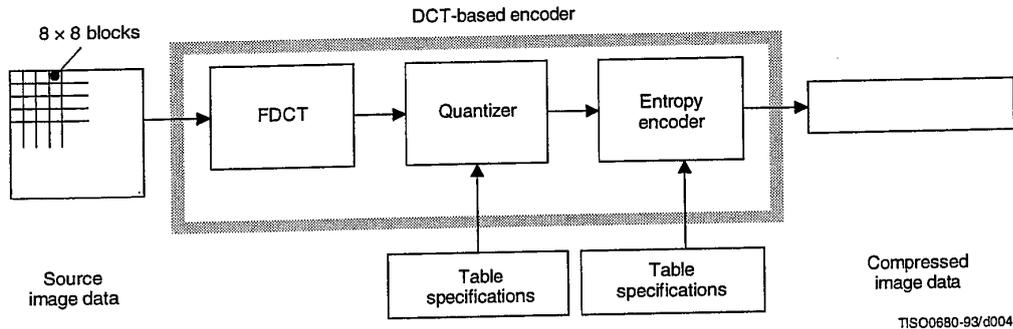
Figure 4 – DCT-based encoder simplified diagram

In the encoding process the input component's *samples* are grouped into 8×8 *blocks*, and each block is transformed by the *forward DCT* (FDCT) into a set of 64 values referred to as *DCT coefficients*. One of these values is referred to as the *DC coefficient* and the other 63 as the *AC coefficients*.

Each of the 64 coefficients is then *quantized* using one of 64 corresponding values from a *quantization table* (determined by one of the table specifications shown in Figure 4). No default values for quantization tables are specified in this Specification; applications may specify values which customize picture quality for their particular image characteristics, display devices, and viewing conditions.

After quantization, the DC coefficient and the 63 AC coefficients are prepared for *entropy encoding*, as shown in Figure 5. The previous quantized DC coefficient is used to predict the current quantized DC coefficient, and the difference is encoded. The 63 quantized AC coefficients undergo no such differential encoding, but are converted into a one-dimensional *zig-zag sequence*, as shown in Figure 5.

The quantized coefficients are then passed to an entropy encoding procedure which compresses the data further. One of two entropy coding procedures can be used, as described in 4.6. If *Huffman encoding* is used, *Huffman table* specifications must be provided to the encoder. If *arithmetic encoding* is used, arithmetic coding *conditioning table* specifications may be provided, otherwise the default conditioning table specifications shall be used.

Figure 6 shows the main procedures for all DCT-based decoding processes. Each step shown performs essentially the inverse of its corresponding main procedure within the encoder. The entropy decoder decodes the zig-zag sequence of quantized DCT coefficients. After *dequantization* the DCT coefficients are transformed to an 8×8 block of samples by the *inverse DCT* (IDCT).

4.4   Lossless coding

Figure 7 shows the main procedures for the lossless encoding processes. A *predictor* combines the reconstructed values of up to three neighbourhood samples at positions a, b, and c to form a prediction of the sample at position x as shown in Figure 8. This prediction is then subtracted from the actual value of the sample at position x, and the difference is losslessly entropy-coded by either Huffman or arithmetic coding.

Differential DC encoding

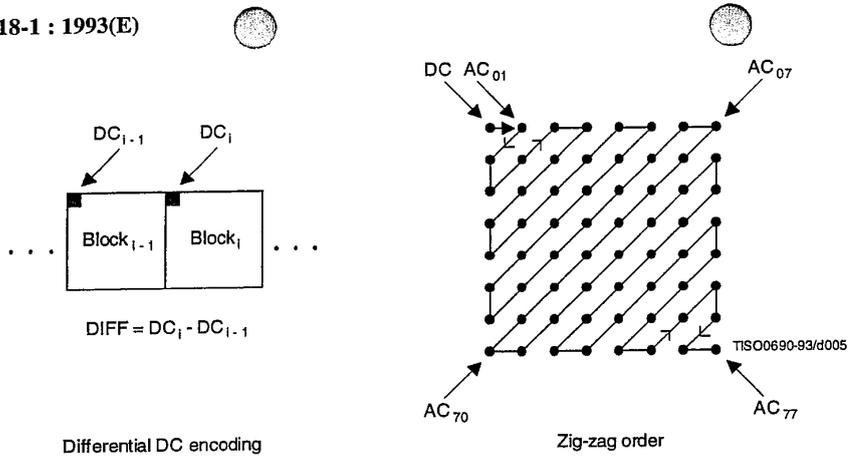Zig-zag order

DIFF = DC$_i$ - DC$_{i-1}$

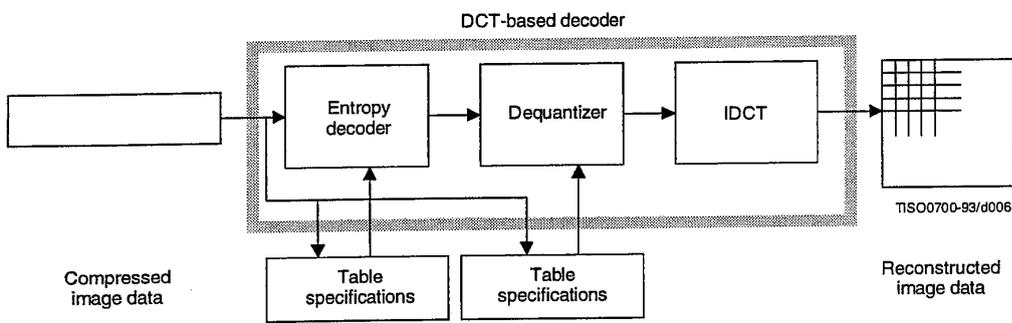Figure 5 – Preparation of quantized coefficients for entropy encoding

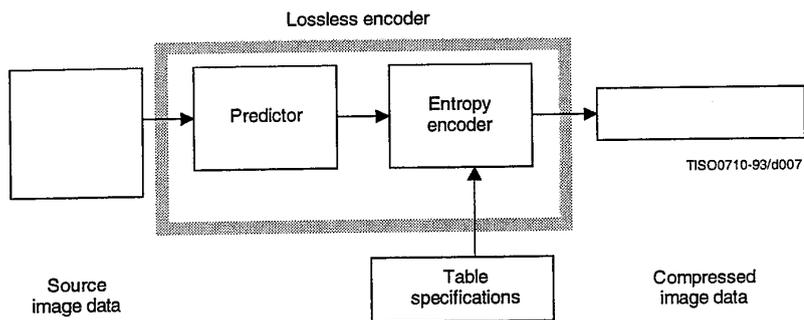Figure 6 – DCT-based decoder simplified diagram

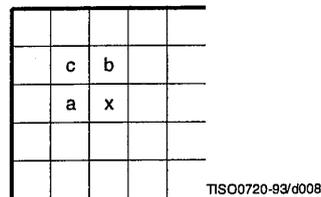Figure 7 – Lossless encoder simplified diagram

Figure 8 — 3-sample prediction neighbourhood

This encoding process may also be used in a slightly modified way, whereby the *precision* of the input samples is reduced by one or more bits prior to the lossless coding. This achieves higher compression than the lossless process (but lower compression than the DCT-based processes for equivalent visual fidelity), and limits the reconstructed image's worst-case sample error to the amount of input precision reduction.

4.5   Modes of operation

There are four distinct *modes of operation* under which the various coding processes are defined: *sequential DCT-based, progressive DCT-based*, lossless, and *hierarchical*. (Implementations are not required to provide all of these.) The lossless mode of operation was described in 4.4. The other modes of operation are compared as follows.

For the sequential DCT-based mode, 8 × 8 sample blocks are typically input block by block from left to right, and block-row by block-row from top to bottom. After a block has been transformed by the forward DCT, quantized and prepared for entropy encoding, all 64 of its quantized DCT coefficients can be immediately entropy encoded and output as part of the compressed image data (as was described in 4.3), thereby minimizing coefficient storage requirements.

For the progressive DCT-based mode, 8 × 8 blocks are also typically encoded in the same order, but in multiple *scans* through the image. This is accomplished by adding an image-sized coefficient memory buffer (not shown in Figure 4) between the quantizer and the entropy encoder. As each block is transformed by the forward DCT and quantized, its coefficients are stored in the buffer. The DCT coefficients in the buffer are then partially encoded in each of multiple scans. The typical sequence of image presentation at the output of the decoder for sequential versus progressive modes of operation is shown in Figure 9.

There are two procedures by which the quantized coefficients in the buffer may be partially encoded within a scan. First, only a specified *band* of coefficients from the zig-zag sequence need be encoded. This procedure is called *spectral selection*, because each band typically contains coefficients which occupy a lower or higher part of the *frequency* spectrum for that 8 × 8 block. Secondly, the coefficients within the current band need not be encoded to their full (quantized) accuracy within each scan. Upon a coefficient's first encoding, a specified number of most significant bits is encoded first. In subsequent scans, the less significant bits are then encoded. This procedure is called *successive approximation*. Either procedure may be used separately, or they may be mixed in flexible combinations.

In hierarchical mode, an image is encoded as a sequence of *frames*. These frames provide *reference reconstructed components* which are usually needed for prediction in subsequent frames. Except for the first frame for a given component, *differential frames* encode the difference between source components and reference reconstructed components. The coding of the differences may be done using only DCT-based processes, only lossless processes, or DCT-based processes with a final lossless process for each component. *Downsampling* and *upsampling filters* may be used to provide a pyramid of spatial resolutions as shown in Figure 10. Alternatively, the hierarchical mode can be used to improve the quality of the reconstructed components at a given spatial resolution.

Hierarchical mode offers a progressive presentation similar to the progressive DCT-based mode but is useful in environments which have multi-resolution requirements. Hierarchical mode also offers the capability of progressive coding to a final lossless stage.
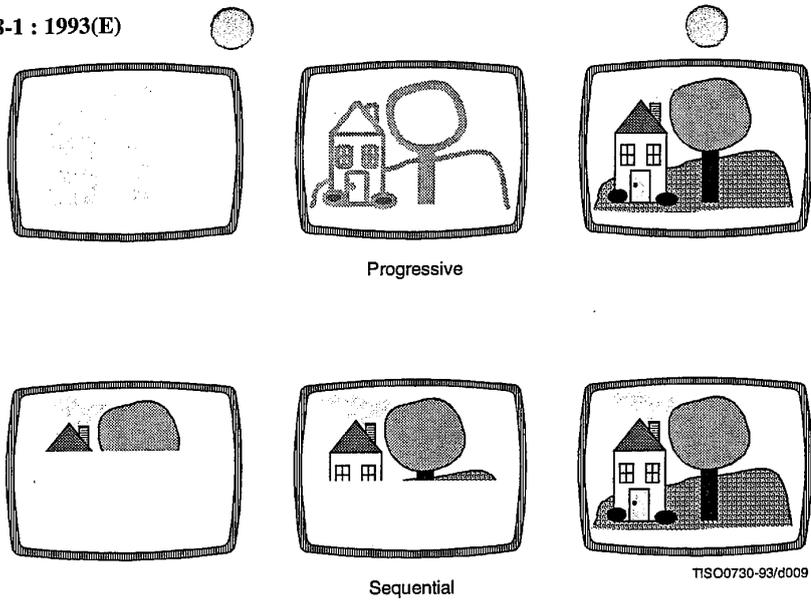
Progressive

Sequential

TISO0730-93/d009

Figure 9 – Progressive versus sequential presentation
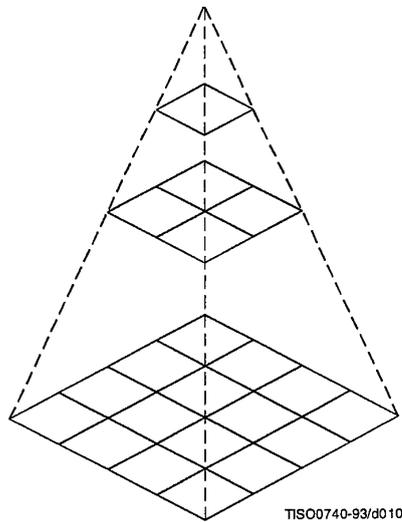

TISO0740-93/d010

Figure 10 – Hierarchical multi-resolution encoding

4.6 Entropy coding alternatives

Two alternative entropy coding procedures are specified: Huffman coding and arithmetic coding. Huffman coding procedures use Huffman tables, determined by one of the table specifications shown in Figures 1 and 2. Arithmetic coding procedures use arithmetic coding conditioning tables, which may also be determined by a table specification. No default values for Huffman tables are specified, so that applications may choose tables appropriate for their own environments. Default tables are defined for the arithmetic coding conditioning.

The baseline sequential process uses Huffman coding, while the extended DCT-based and lossless processes may use either Huffman or arithmetic coding.

4.7   Sample precision

For DCT-based processes, two alternative sample precisions are specified: either 8 bits or 12 bits per sample. Applications which use samples with other precisions can use either 8-bit or 12-bit precision by shifting their source image samples appropriately. The baseline process uses only 8-bit precision. DCT-based implementations which handle 12-bit source image samples are likely to need greater computational resources than those which handle only 8-bit source images. Consequently in this Specification separate normative requirements are defined for 8-bit and 12-bit DCT-based processes.

For lossless processes the sample precision is specified to be from 2 to 16 bits.

4.8   Multiple-component control

Subclauses 4.3 and 4.4 give an overview of one major part of the encoding and decoding processes – those which operate on the sample values in order to achieve compression. There is another major part as well – the procedures which control the order in which the image data from multiple components are processed to create the compressed data, and which ensure that the proper set of table data is applied to the proper *data units* in the image. (A data unit is a sample for lossless processes and an 8 × 8 block of samples for DCT-based processes.)

4.8.1   Interleaving multiple components

Figure 11 shows an example of how an encoding process selects between multiple source image components as well as multiple sets of table data, when performing its encoding procedures. The source image in this example consists of the three components A, B and C, and there are two sets of table specifications. (This simplified view does not distinguish between the quantization tables and entropy coding tables.)
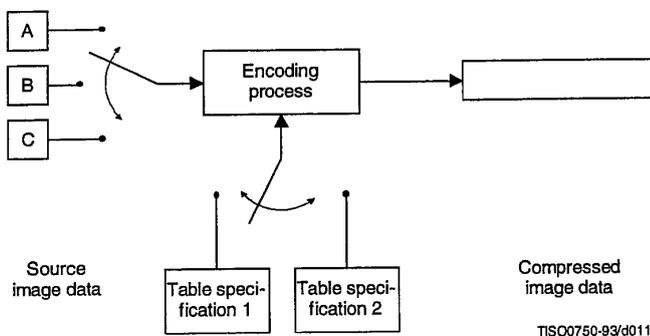

Figure 11 – Component-interleave and table-switching control

In sequential mode, encoding is *non-interleaved* if the encoder compresses all image data units in component A before beginning component B, and then in turn all of B before C. Encoding is *interleaved* if the encoder compresses a data unit from A, a data unit from B, a data unit from C, then back to A, etc. These alternatives are illustrated in Figure 12, which shows a case in which all three image components have identical dimensions: X *columns* by Y lines, for a total of n data units each.

$A_1, A_2, ...A_n,$ $B_1, B_2, ...B_n,$ $C_1, C_2, ...C_n$

Scan 1   Scan 2   Scan 3

Data unit encoding order, non-interleaved

$A_1, B_1, C_1, A_2, B_2, C_2, ...A_n, B_n, C_n$

Scan 1

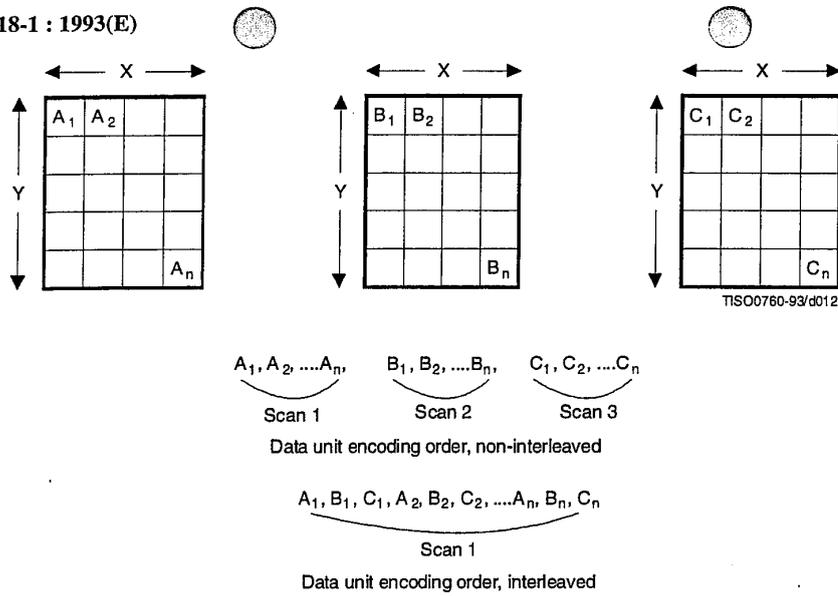Data unit encoding order, interleaved

Figure 12 – Interleaved versus non-interleaved encoding order

These control procedures are also able to handle cases in which the source image components have different dimensions. Figure 13 shows a case in which two of the components, B and C, have half the number of horizontal samples relative to component A. In this case, two data units from A are interleaved with one each from B and C. Cases in which components of an image have more complex relationships, such as different horizontal and vertical dimensions, can be handled as well. (See Annex A.)
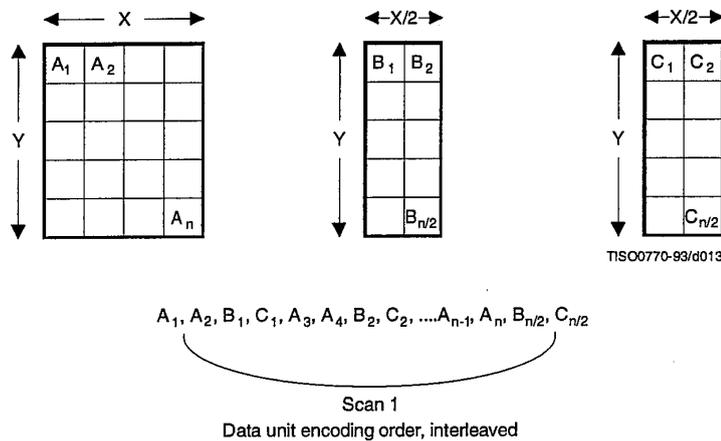

$A_1, A_2, B_1, C_1, A_3, A_4, B_2, C_2, ...A_{n-1}, A_n, B_{n/2}, C_{n/2}$

Scan 1

Data unit encoding order, interleaved

Figure 13 – Interleaved order for components with different dimensions

4.8.2 Minimum coded unit

Related to the concepts of multiple-component interleave is the *minimum coded unit* (MCU). If the compressed image data is non-interleaved, the MCU is defined to be one data unit. For example, in Figure 12 the MCU for the non-interleaved case is a single data unit. If the compressed data is interleaved, the MCU contains one or more data units from each component. For the interleaved case in Figure 12, the (first) MCU consists of the three interleaved data units A_1, B_1, C_1. In the example of Figure 13, the (first) MCU consists of the four data units A_1, A_2, B_1, C_1.

4.9 Structure of compressed data

Figures 1, 2, and 3 all illustrate slightly different views of compressed image data. Figure 1 shows this data as the output of an encoding process, Figure 2 shows it as the input to a decoding process, and Figure 3 shows compressed image data in the interchange format, at the interface between applications.

Compressed image data are described by a uniform structure and set of *parameters* for both classes of encoding processes (lossy or lossless), and for all modes of operation (sequential, progressive, lossless, and hierarchical). The various parts of the compressed image data are identified by special two-byte codes called *markers*. Some markers are followed by particular sequences of parameters, as in the case of table specifications, *frame header*, or *scan header*. Others are used without parameters for functions such as marking the start-of-image and end-of-image. When a marker is associated with a particular sequence of parameters, the marker and its parameters comprise a *marker segment*.

The data created by the entropy encoder are also segmented, and one particular marker – *the restart marker* – is used to isolate *entropy-coded data segments*. The encoder outputs the restart markers, intermixed with the entropy-coded data, at regular *restart intervals* of the source image data. Restart markers can be identified without having to decode the compressed data to find them. Because they can be independently decoded, they have application-specific uses, such as parallel encoding or decoding, isolation of data corruptions, and semi-random access of entropy-coded segments.

There are three compressed data formats:

a) the interchange format;

b) the *abbreviated format* for compressed image data;

c) the abbreviated format for table-specification data.

4.9.1 Interchange format

In addition to certain required marker segments and the entropy-coded segments, the interchange format shall include the marker segments for all quantization and entropy-coding table specifications needed by the decoding process. This guarantees that a compressed image can cross the boundary between application environments, regardless of how each environment internally associates tables with compressed image data.

4.9.2 Abbreviated format for compressed image data

The abbreviated format for compressed image data is identical to the interchange format, except that it does not include all tables required for decoding. (It may include some of them.) This format is intended for use within applications where alternative mechanisms are available for supplying some or all of the table-specification data needed for decoding.

4.9.3 Abbreviated format for table-specification data

This format contains only table-specification data. It is a means by which the application may install in the decoder the tables required to subsequently reconstruct one or more images.

4.10 Image, frame, and scan

Compressed image data consists of only one image. An image contains only one frame in the cases of sequential and progressive coding processes; an image contains multiple frames for the hierarchical mode.

A frame contains one or more scans. For sequential processes, a scan contains a complete encoding of one or more image components. In Figures 12 and 13, the frame consists of three scans when non-interleaved, and one scan if all three components are interleaved together. The frame could also consist of two scans: one with a non-interleaved component, the other with two components interleaved.

For progressive processes, a scan contains a partial encoding of all data units from one or more image components. Components shall not be interleaved in progressive mode, except for the DC coefficients in the first scan for each component of a progressive frame.

4.11   Summary of coding processes

Table 1 provides a summary of the essential characteristics of the various coding processes specified in this Specification. The full specification of these processes is contained in Annexes F, G, H, and J.

Table 1 – Summary: Essential characteristics of coding processes

Baseline process (required for all DCT-based decoders)
• DCT-based process • Source image: 8-bit samples within each component • Sequential • Huffman coding: 2 AC and 2 DC tables • Decoders shall process scans with 1, 2, 3, and 4 components • Interleaved and non-interleaved scans

Extended DCT-based processes
• DCT-based process • Source image: 8-bit or 12-bit samples • Sequential or progressive • Huffman or arithmetic coding: 4 AC and 4 DC tables • Decoders shall process scans with 1, 2, 3, and 4 components • Interleaved and non-interleaved scans

Lossless processes
• Predictive process (not DCT-based) • Source image: P-bit samples ($2 \leq P \leq 16$) • Sequential • Huffman or arithmetic coding: 4 DC tables • Decoders shall process scans with 1, 2, 3, and 4 components • Interleaved and non-interleaved scans

Hierarchical processes
• Multiple frames (non-differential and differential) • Uses extended DCT-based or lossless processes • Decoders shall process scans with 1, 2, 3, and 4 components • Interleaved and non-interleaved scans

5   Interchange format requirements

The interchange format is the coded representation of compressed image data for exchange between application environments.

The interchange format requirements are that any compressed image data represented in interchange format shall comply with the syntax and code assignments appropriate for the decoding process selected, as specified in Annex B.

Tests for whether compressed image data comply with these requirements are specified in Part 2 of this Specification.

6   Encoder requirements

An encoding process converts source image data to compressed image data. Each of Annexes F, G, H, and J specifies a number of distinct encoding processes for its particular mode of operation.

An encoder is an embodiment of one (or more) of the encoding processes specified in Annexes F, G, H, or J. In order to comply with this Specification, an encoder shall satisfy at least one of the following two requirements.

An encoder shall

a)   with appropriate accuracy, convert source image data to compressed image data which comply with the interchange format syntax specified in Annex B for the encoding process(es) embodied by the encoder;

b)   with appropriate accuracy, convert source image data to compressed image data which comply with the abbreviated format for compressed image data syntax specified in Annex B for the encoding process(es) embodied by the encoder.

For each of the encoding processes specified in Annexes F, G, H, and J, the compliance tests for the above requirements are specified in Part 2 of this Specification.

NOTE – There is no requirement in this Specification that any encoder which embodies one of the encoding processes specified in Annexes F, G, H, or J shall be able to operate for all ranges of the parameters which are allowed for that process. An encoder is only required to meet the compliance tests specified in Part 2, and to generate the compressed data format according to Annex B for those parameter values which it does use.

7   Decoder requirements

A decoding process converts compressed image data to reconstructed image data. Each of Annexes F, G, H, and J specifies a number of distinct decoding processes for its particular mode of operation.

A decoder is an embodiment of one (or more) of the decoding processes specified in Annexes F, G, H, or J. In order to comply with this Specification, a decoder shall satisfy all three of the following requirements.

A decoder shall

a)   with appropriate accuracy, convert to reconstructed image data any compressed image data with parameters within the range supported by the application, and which comply with the interchange format syntax specified in Annex B for the decoding process(es) embodied by the decoder;

b)   accept and properly store any table-specification data which comply with the abbreviated format for table-specification data syntax specified in Annex B for the decoding process(es) embodied by the decoder;

c)   with appropriate accuracy, convert to reconstructed image data any compressed image data which comply with the abbreviated format for compressed image data syntax specified in Annex B for the decoding process(es) embodied by the decoder, provided that the table-specification data required for decoding the compressed image data has previously been installed into the decoder.

Additionally, any DCT-based decoder, if it embodies any DCT-based decoding process other than baseline sequential, shall also embody the baseline sequential decoding process.

For each of the decoding processes specified in Annexes F, G, H, and J, the compliance tests for the above requirements are specified in Part 2 of this Specification.

Annex A

Mathematical definitions

(This annex forms an integral part of this Recommendation I International Standard)

A.1 Source image

Source images to which the encoding processes specified in this Specification can be applied are defined in this annex.

A.1.1 Dimensions and sampling factors

As shown in Figure A.1, a source image is defined to consist of Nf components. Each component, with unique identifier C_i, is defined to consist of a rectangular array of samples of x_i columns by y_i lines. The component dimensions are derived from two parameters, X and Y, where X is the maximum of the x_i values and Y is the maximum of the y_i values for all components in the frame. For each component, sampling factors H_i and V_i are defined relating component dimensions x_i and y_i to maximum dimensions X and Y, according to the following expressions:

$$x_i = \left\lceil X \times \frac{H_i}{H_{max}} \right\rceil \text{ and } y_i \left\lceil Y \times \frac{V_i}{V_{max}} \right\rceil,$$

where H_{max} and V_{max} are the maximum sampling factors for all components in the frame, and $\lceil \ \rceil$ is the ceiling function.

As an example, consider an image having 3 components with maximum dimensions of 512 lines and 512 samples per line, and with the following sampling factors:

Component 0 $H_0 = 4,\ V_0 = 1$
Component 1 $H_1 = 2,\ V_1 = 2$
Component 2 $H_2 = 1,\ V_2 = 1$

Then $X = 512$, $Y = 512$, $H_{max} = 4$, $V_{max} = 2$, and x_i and y_i for each component are

Component 0 $x_0 = 512,\ y_0 = 256$
Component 1 $x_1 = 256,\ y_1 = 512$
Component 2 $x_2 = 128,\ y_2 = 256$

NOTE – The X, Y, H_i, and V_i parameters are contained in the frame header of the compressed image data (see B.2.2), whereas the individual component dimensions x_i and y_i are derived by the decoder. Source images with x_i and y_i dimensions which do not satisfy the expressions above cannot be properly reconstructed.

A.1.2 Sample precision

A sample is an integer with precision P bits, with any value in the range 0 through 2^{P-1}. All samples of all components within an image shall have the same precision P. Restrictions on the value of P depend on the mode of operation, as specified in B.2 to B.7.

A.1.3 Data unit

A data unit is a sample in lossless processes and an 8×8 block of contiguous samples in DCT-based processes. The left-most 8 samples of each of the top-most 8 rows in the component shall always be the top-left-most block. With this top-left-most block as the reference, the component is partitioned into contiguous data units to the right and to the bottom (as shown in Figure A.4).

A.1.4 Orientation

Figure A.1 indicates the orientation of an image component by the terms top, bottom, left, and right. The order by which the data units of an image component are input to the compression encoding procedures is defined to be left-to-right and top-to-bottom within the component. (This ordering is precisely defined in A.2.) Applications determine which edges of a source image are defined as top, bottom, left, and right.
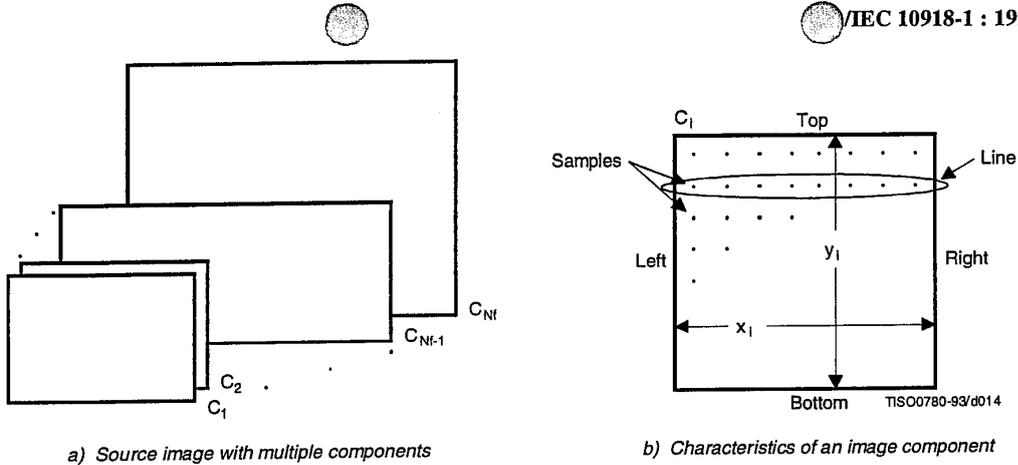
a) Source image with multiple components

b) Characteristics of an image component

Figure A.1 – Source image characteristics

A.2 Order of source image data encoding

The scan header (see B.2.3) specifies the order by which source image data units shall be encoded and placed within the compressed image data. For a given scan, if the scan header parameter Ns = 1, then data from only one source component – the component specified by parameter Cs_1 – shall be present within the scan. This data is non-interleaved by definition. If Ns > 1, then data from the Ns components Cs_1 through Cs_{Ns} shall be present within the scan. This data shall always be interleaved. The order of components in a scan shall be according to the order specified in the frame header.

The ordering of data units and the construction of minimum coded units (MCU) is defined as follows.

A.2.1 Minimum coded unit (MCU)

For non-interleaved data the MCU is one data unit. For interleaved data the MCU is the sequence of data units defined by the sampling factors of the components in the scan.

A.2.2 Non-interleaved order (Ns = 1)

When Ns = 1 (where Ns is the number of components in a scan), the order of data units within a scan shall be left-to-right and top-to-bottom, as shown in Figure A.2. This ordering applies whenever Ns = 1, regardless of the values of H_1 and V_1.
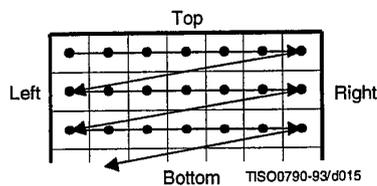

Figure A.2 – Non-interleaved data ordering

A.2.3   Interleaved order (Ns > 1)

When Ns > 1, each scan component Cs_i is partitioned into small rectangular arrays of H_k horizontal data units by V_k vertical data units. The subscripts k indicate that H_k and V_k are from the position in the frame header component-specification for which $C_k = Cs_i$. Within each H_k by V_k array, data units are ordered from left-to-right and top-to-bottom. The arrays in turn are ordered from left-to-right and top-to-bottom within each component.

As shown in the example of Figure A.3, Ns = 4, and MCU_1 consists of data units taken first from the top-left-most region of Cs_1, followed by data units from the corresponding region of Cs_2, then from Cs_3 and then from Cs_4. MCU_2 follows the same ordering for data taken from the next region to the right for the four components.
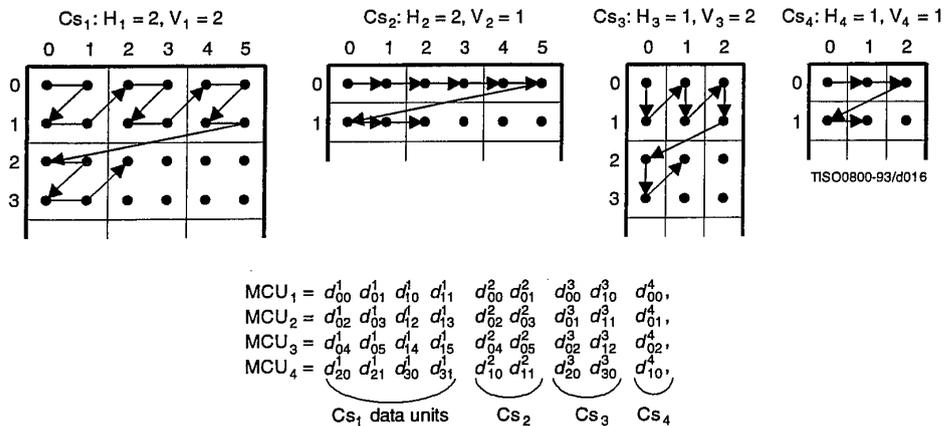

Figure A.3 – Interleaved data ordering example

A.2.4   Completion of partial MCU

For DCT-based processes the data unit is a block. If x_i is not a multiple of 8, the encoding process shall extend the number of columns to complete the right-most sample blocks. If the component is to be interleaved, the encoding process shall also extend the number of samples by one or more additional blocks, if necessary, so that the number of blocks is an integer multiple of H_i. Similarly, if y_i is not a multiple of 8, the encoding process shall extend the number of lines to complete the bottom-most block-row. If the component is to be interleaved, the encoding process shall also extend the number of lines by one or more additional block-rows, if necessary, so that the number of block-rows is an integer multiple of V_i.

NOTE – It is recommended that any incomplete MCUs be completed by replication of the right-most column and the bottom line of each component.

For lossless processes the data unit is a sample. If the component is to be interleaved, the encoding process shall extend the number of samples, if necessary, so that the number is a multiple of H_i. Similarly, the encoding process shall extend the number of lines, if necessary, so that the number of lines is a multiple of V_i.

Any sample added by an encoding process to complete partial MCUs shall be removed by the decoding process.

A.3   DCT compression

A.3.1   Level shift

Before a non-differential frame encoding process computes the FDCT for a block of source image samples, the samples shall be level shifted to a signed representation by subtracting 2^{P-1}, where P is the precision parameter specified in B.2.2. Thus, when P = 8, the level shift is by 128; when P = 12, the level shift is by 2048.

After a non-differential frame decoding process computes the IDCT and produces a block of reconstructed image samples, an inverse level shift shall restore the samples to the unsigned representation by adding 2^{P-1} and clamping the results to the range 0 to $2^{P}-1$.

A.3.2   Orientation of samples for FDCT computation

Figure A.4 shows an image component which has been partitioned into 8×8 blocks for the FDCT computations. Figure A.4 also defines the orientation of the samples within a block by showing the indices used in the FDCT equation of A.3.3.

The definitions of block partitioning and sample orientation also apply to any DCT decoding process and the output reconstructed image. Any sample added by an encoding process to complete partial MCUs shall be removed by the decoding process.
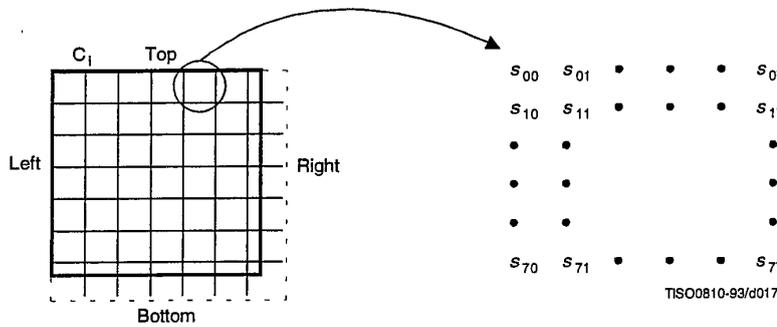

Figure A.4 – Partition and orientation of 8 x 8 sample blocks

A.3.3   FDCT and IDCT (informative)

The following equations specify the ideal functional definition of the FDCT and the IDCT.

NOTE – These equations contain terms which cannot be represented with perfect accuracy by any real implementation. The accuracy requirements for the combined FDCT and quantization procedures are specified in Part 2 of this Specification. The accuracy requirements for the combined dequantization and IDCT procedures are also specified in Part 2 of this Specification.

$$\text{FDCT:} \qquad S_{vu} = \frac{1}{4} C_u C_v \sum_{x=0}^{7} \sum_{y=0}^{7} s_{yx} \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16}$$

$$\text{IDCT:} \qquad s_{yx} = \frac{1}{4} \sum_{u=0}^{7} \sum_{v=0}^{7} C_u C_v S_{vu} \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16}$$

where

$$C_u, C_v = 1/\sqrt{2} \text{ for } u, v = 0$$

$$C_u, C_v = 1 \text{ otherwise}$$

otherwise.

A.3.4   DCT coefficient quantization (informative) and dequantization (normative)

After the FDCT is computed for a block, each of the 64 resulting DCT coefficients is quantized by a uniform quantizer. The quantizer step size for each coefficient S_{vu} is the value of the corresponding element Q_{vu} from the quantization table specified by the frame parameter Tq_i (see B.2.2).

The uniform quantizer is defined by the following equation. Rounding is to the nearest integer:

$$Sq_{vu} = round \left(\frac{S_{vu}}{Q_{vu}} \right)$$

Sq_{vu} is the quantized DCT coefficient, normalized by the quantizer step size.

NOTE – This equation contains a term which may not be represented with perfect accuracy by any real implementation. The accuracy requirements for the combined FDCT and quantization procedures are specified in Part 2 of this Specification.

At the decoder, this normalization is removed by the following equation, which defines dequantization:

$$R_{vu} = Sq_{vu} \times Q_{vu}$$

NOTE – Depending on the rounding used in quantization, it is possible that the dequantized coefficient may be outside the expected range.

The relationship among samples, DCT coefficients, and quantization is illustrated in Figure A.5.

A.3.5   Differential DC encoding

After quantization, and in preparation for entropy encoding, the quantized DC coefficient Sq_{00} is treated separately from the 63 quantized AC coefficients. The value that shall be encoded is the difference (DIFF) between the quantized DC coefficient of the current block (DC_i which is also designated as Sq_{00}) and that of the previous block of the same component (PRED):

$$DIFF = DC_i - PRED$$

A.3.6   Zig-zag sequence

After quantization, and in preparation for entropy encoding, the quantized AC coefficients are converted to the zig-zag sequence. The quantized DC coefficient (coefficient zero in the array) is treated separately, as defined in A.3.5. The zig-zag sequence is specified in Figure A.6.

A.4   Point transform

For various procedures data may be optionally divided by a power of 2 by a point transform prior to coding. There are three processes which require a point transform: lossless coding, lossless differential frame coding in the hierarchical mode, and successive approximation coding in the progressive DCT mode.

In the lossless mode of operation the point transform is applied to the input samples. In the difference coding of the hierarchical mode of operation the point transform is applied to the difference between the input component samples and the reference component samples. In both cases the point transform is an integer divide by 2^{Pt}, where Pt is the value of the point transform parameter (see B.2.3).

In successive approximation coding the point transform for the AC coefficients is an integer divide by 2^{Al}, where Al is the successive approximation bit position, low (see B.2.3). The point transform for the DC coefficients is an arithmetic-shift-right by Al bits. This is equivalent to dividing by 2^{Pt} before the level shift (see A.3.1).

The output of the decoder is rescaled by multiplying by 2^{Pt}. An example of the point transform is given in K.10.
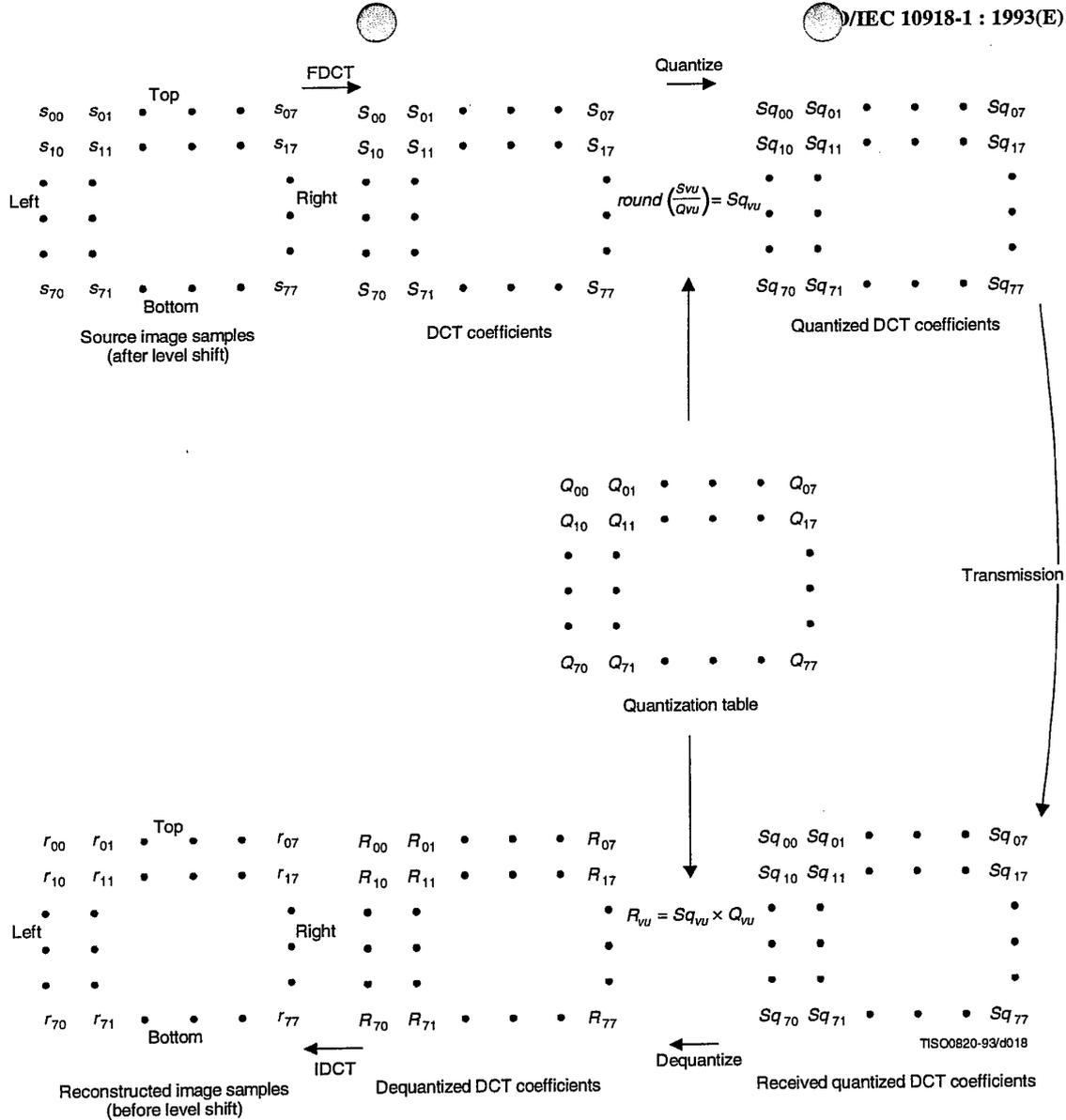
$$round\left(\frac{S_{vu}}{Q_{vu}}\right)= Sq_{vu}$$

$$R_{vu} = Sq_{vu} \times Q_{vu}$$

Figure A.5 – Relationship between 8 × 8-block samples and DCT coefficients

0	1	5	6	14	15	27	28
2	4	7	13	16	26	29	42
3	8	12	17	25	30	41	43
9	11	18	24	31	40	44	53
10	19	23	32	39	45	52	54
20	22	33	38	46	51	55	60
21	34	37	47	50	56	59	61
35	36	48	49	57	58	62	63

Figure A.6 – Zig-zag sequence of quantized DCT coefficients

A.5   Arithmetic procedures in lossless and hierarchical modes of operation

In the lossless mode of operation predictions are calculated with full precision and without clamping of either overflow or underflow beyond the range of values allowed by the precision of the input. However, the division by two which is part of some of the prediction calculations shall be approximated by an arithmetic-shift-right by one bit.

The two's complement differences which are coded in either the lossless mode of operation or the differential frame coding in the hierarchical mode of operation are calculated modulo 65 536, thereby restricting the precision of these differences to a maximum of 16 bits. The modulo values are calculated by performing the logical AND operation of the two's complement difference with X'FFFF'. For purposes of coding, the result is still interpreted as a 16 bit two's complement difference. Modulo 65 536 arithmetic is also used in the decoder in calculating the output from the sum of the prediction and this two's complement difference.

Annex B

Compressed data formats

(This annex forms an integral part of this Recommendation I International Standard)

This annex specifies three compressed data formats:

    a)   the interchange format, specified in B.2 and B.3;

    b)   the abbreviated format for compressed image data, specified in B.4;

    c)   the abbreviated format for table-specification data, specified in B.5.

B.1 describes the constituent parts of these formats. B.1.3 and B.1.4 give the conventions for symbols and figures used in the format specifications.

B.1   General aspects of the compressed data format specifications

Structurally, the compressed data formats consist of an ordered collection of parameters, markers, and entropy-coded data segments. Parameters and markers in turn are often organized into marker segments. Because all of these constituent parts are represented with byte-aligned codes, each compressed data format consists of an ordered sequence of 8-bit bytes. For each byte, a most significant bit (MSB) and a least significant bit (LSB) are defined.

B.1.1   Constituent parts

This subclause gives a general description of each of the constituent parts of the compressed data format.

B.1.1.1   Parameters

Parameters are integers, with values specific to the encoding process, source image characteristics, and other features selectable by the application. Parameters are assigned either 4-bit, 1-byte, or 2-byte codes. Except for certain optional groups of parameters, parameters encode critical information without which the decoding process cannot properly reconstruct the image.

The code assignment for a parameter shall be an unsigned integer of the specified length in bits with the particular value of the parameter.

For parameters which are 2 bytes (16 bits) in length, the most significant byte shall come first in the compressed data's ordered sequence of bytes. Parameters which are 4 bits in length always come in pairs, and the pair shall always be encoded in a single byte. The first 4-bit parameter of the pair shall occupy the most significant 4 bits of the byte. Within any 16-, 8-, or 4-bit parameter, the MSB shall come first and LSB shall come last.

B.1.1.2   Markers

Markers serve to identify the various structural parts of the compressed data formats. Most markers start marker segments containing a related group of parameters; some markers stand alone. All markers are assigned two-byte codes: an X'FF' byte followed by a byte which is not equal to 0 or X'FF' (see Table B.1). Any marker may optionally be preceded by any number of fill bytes, which are bytes assigned code X'FF'.

       NOTE – Because of this special code-assignment structure, markers make it possible for a decoder to parse the compressed data and locate its various parts without having to decode other segments of image data.

B.1.1.3   Marker assignments

All markers shall be assigned two-byte codes: a X'FF' byte followed by a second byte which is not equal to 0 or X'FF'. The second byte is specified in Table B.1 for each defined marker. An asterisk (*) indicates a marker which stands alone, that is, which is not the start of a marker segment.

Table B.1 – Marker code assignments

Code Assignment	Symbol	Description
Start Of Frame markers, non-differential, Huffman coding		
X'FFC0' X'FFC1' X'FFC2' X'FFC3'	SOF_0 SOF_1 SOF_2 SOF_3	Baseline DCT Extended sequential DCT Progressive DCT Lossless (sequential)
Start Of Frame markers, differential, Huffman coding		
X'FFC5' X'FFC6' X'FFC7'	SOF_5 SOF_6 SOF_7	Differential sequential DCT Differential progressive DCT Differential lossless (sequential)
Start Of Frame markers, non-differential, arithmetic coding		
X'FFC8' X'FFC9' X'FFCA' X'FFCB'	JPG SOF_9 SOF_{10} SOF_{11}	Reserved for JPEG extensions Extended sequential DCT Progressive DCT Lossless (sequential)
Start Of Frame markers, differential, arithmetic coding		
X'FFCD' X'FFCE' X'FFCF'	SOF_{13} SOF_{14} SOF_{15}	Differential sequential DCT Differential progressive DCT Differential lossless (sequential)
Huffman table specification		
X'FFC4'	DHT	Define Huffman table(s)
Arithmetic coding conditioning specification		
X'FFCC'	DAC	Define arithmetic coding conditioning(s)
Restart interval termination		
X'FFD0' through X'FFD7'	RST_m*	Restart with modulo 8 count "m"
Other markers		
X'FFD8' X'FFD9' X'FFDA' X'FFDB' X'FFDC' X'FFDD' X'FFDE' X'FFDF' X'FFE0' through X'FFEF' X'FFF0' through X'FFFD' X'FFFE'	SOI* EOI* SOS DQT DNL DRI DHP EXP APP_n JPG_n COM	Start of image End of image Start of scan Define quantization table(s) Define number of lines Define restart interval Define hierarchical progression Expand reference component(s) Reserved for application segments Reserved for JPEG extensions Comment
Reserved markers		
X'FF01' X'FF02' through X'FFBF'	TEM* RES	For temporary private use in arithmetic coding Reserved

B.1.1.4 Marker segments

A marker segment consists of a marker followed by a sequence of related parameters. The first parameter in a marker segment is the two-byte length parameter. This length parameter encodes the number of bytes in the marker segment, including the length parameter and excluding the two-byte marker. The marker segments identified by the SOF and SOS marker codes are referred to as headers: the frame header and the scan header respectively.

B.1.1.5 Entropy-coded data segments

An entropy-coded data segment contains the output of an entropy-coding procedure. It consists of an integer number of bytes, whether the entropy-coding procedure used is Huffman or arithmetic.

NOTES

1   Making entropy-coded segments an integer number of bytes is performed as follows: for Huffman coding, 1-bits are used, if necessary, to pad the end of the compressed data to complete the final byte of a segment. For arithmetic coding, byte alignment is performed in the procedure which terminates the entropy-coded segment (see D.1.8).

2   In order to ensure that a marker does not occur within an entropy-coded segment, any X'FF' byte generated by either a Huffman or arithmetic encoder, or an X'FF' byte that was generated by the padding of 1-bits described in NOTE 1 above, is followed by a "stuffed" zero byte (see D.1.6 and F.1.2.3).

B.1.2   Syntax

In B.2 and B.3 the interchange format syntax is specified. For the purposes of this Specification, the syntax specification consists of:

    — the required ordering of markers, parameters, and entropy-coded segments;
    — identification of optional or conditional constituent parts;
    — the name, symbol, and definition of each marker and parameter;
    — the allowed values of each parameter;
    — any restrictions on the above which are specific to the various coding processes.

The ordering of constituent parts and the identification of which are optional or conditional is specified by the syntax figures in B.2 and B.3. Names, symbols, definitions, allowed values, conditions, and restrictions are specified immediately below each syntax figure.

B.1.3   Conventions for syntax figures

The syntax figures in B.2 and B.3 are a part of the interchange format specification. The following conventions, illustrated in Figure B.1, apply to these figures:

    — **parameter/marker indicator:** A thin-lined box encloses either a marker or a single parameter;
    — **segment indicator:** A thick-lined box encloses either a marker segment, an entropy-coded data segment, or combinations of these;
    — **parameter length indicator:** The width of a thin-lined box is proportional to the parameter length (4, 8, or 16 bits, shown as E, B, and D respectively in Figure B.1) of the marker or parameter it encloses; the width of thick-lined boxes is not meaningful;
    — **optional/conditional indicator:** Square brackets indicate that a marker or marker segment is only optionally or conditionally present in the compressed image data;
    — **ordering:** In the interchange format a parameter or marker shown in a figure precedes all of those shown to its right, and follows all of those shown to its left;
    — **entropy-coded data indicator:** Angled brackets indicate that the entity enclosed has been entropy encoded.

| Segment | [Optional segment] | [B] | D | E | F |
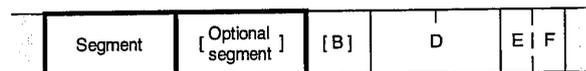
TISO0830-93/d019

Figure B.1 – Syntax notation conventions

B.1.4   Conventions for symbols, code lengths, and values

Following each syntax figure in B.2 and B.3, the symbol, name, and definition for each marker and parameter shown in the figure are specified. For each parameter, the length and allowed values are also specified in tabular form.

The following conventions apply to symbols for markers and parameters:

- all marker symbols have three upper-case letters, and some also have a subscript. Examples: SOI, SOF_n;

- all parameter symbols have one upper-case letter; some also have one lower-case letter and some have subscripts. Examples: Y, Nf, H_i, Tq_i.

B.2   General sequential and progressive syntax

This clause specifies the interchange format syntax which applies to all coding processes for sequential DCT-based, progressive DCT-based, and lossless modes of operation.

B.2.1   High-level syntax

Figure B.2 specifies the order of the high-level constituent parts of the interchange format for all non-hierarchical encoding processes specified in this Specification.
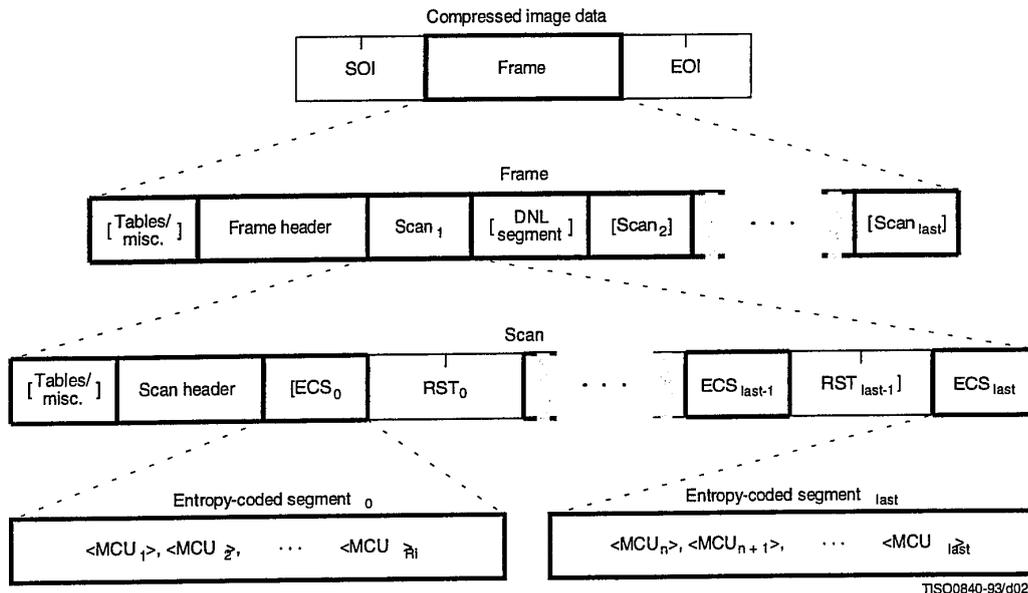

Figure B.2 – Syntax for sequential DCT-based, progressive DCT-based,
and lossless modes of operation

The three markers shown in Figure B.2 are defined as follows:

SOI: Start of image marker – Marks the start of a compressed image represented in the interchange format or abbreviated format.

EOI: End of image marker – Marks the end of a compressed image represented in the interchange format or abbreviated format.

RST_m: Restart marker – A conditional marker which is placed between entropy-coded segments only if restart is enabled. There are 8 unique restart markers (m = 0 - 7) which repeat in sequence from 0 to 7, starting with zero for each scan, to provide a modulo 8 restart interval count.

The top level of Figure B.2 specifies that the non-hierarchical interchange format shall begin with an SOI marker, shall contain one frame, and shall end with an EOI marker.

The second level of Figure B.2 specifies that a frame shall begin with a frame header and shall contain one or more scans. A frame header may be preceded by one or more table-specification or miscellaneous marker segments as specified in B.2.4. If a DNL segment (see B.2.5) is present, it shall immediately follow the first scan.

For sequential DCT-based and lossless processes each scan shall contain from one to four image components. If two to four components are contained within a scan, they shall be interleaved within the scan. For progressive DCT-based processes each image component is only partially contained within any one scan. Only the first scan(s) for the components (which contain only DC coefficient data) may be interleaved.

The third level of Figure B.2 specifies that a scan shall begin with a scan header and shall contain one or more entropy-coded data segments. Each scan header may be preceded by one or more table-specification or miscellaneous marker segments. If restart is not enabled, there shall be only one entropy-coded segment (the one labeled "last"), and no restart markers shall be present. If restart is enabled, the number of entropy-coded segments is defined by the size of the image and the defined restart interval. In this case, a restart marker shall follow each entropy-coded segment except the last one.

The fourth level of Figure B.2 specifies that each entropy-coded segment is comprised of a sequence of entropy-coded MCUs. If restart is enabled and the restart interval is defined to be Ri, each entropy-coded segment except the last one shall contain Ri MCUs. The last one shall contain whatever number of MCUs completes the scan.

Figure B.2 specifies the locations where table-specification segments may be present. However, this Specification hereby specifies that the interchange format shall contain all table-specification data necessary for decoding the compressed image. Consequently, the required table-specification data shall be present at one or more of the allowed locations.

B.2.2   Frame header syntax

Figure B.3 specifies the frame header which shall be present at the start of a frame. This header specifies the source image characteristics (see A.1), the components in the frame, and the sampling factors for each component, and specifies the destinations from which the quantized tables to be used with each component are retrieved.
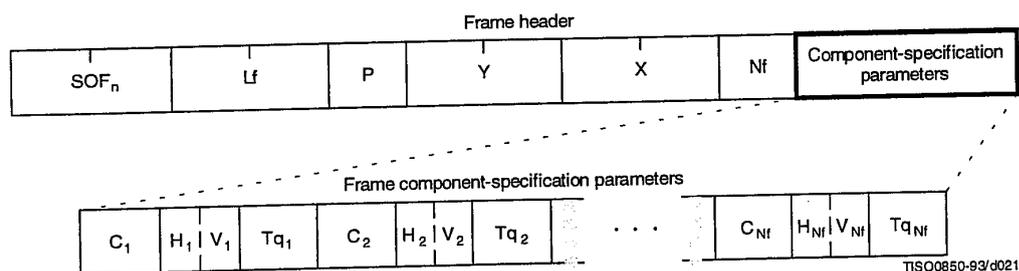

Figure B.3 – Frame header syntax

The markers and parameters shown in Figure B.3 are defined below. The size and allowed values of each parameter are given in Table B.2. In Table B.2 (and similar tables which follow), value choices are separated by commas (e.g. 8, 12) and inclusive bounds are separated by dashes (e.g. 0 - 3).

SOF$_n$: Start of frame marker – Marks the beginning of the frame parameters. The subscript n identifies whether the encoding process is baseline sequential, extended sequential, progressive, or lossless, as well as which entropy encoding procedure is used.

SOF$_0$:   Baseline DCT

SOF$_1$:   Extended sequential DCT, Huffman coding

SOF$_2$:   Progressive DCT, Huffman coding

SOF₃: Lossless (sequential), Huffman coding

SOF₉: Extended sequential DCT, arithmetic coding

SOF₁₀: Progressive DCT, arithmetic coding

SOF₁₁: Lossless (sequential), arithmetic coding

Lf: Frame header length – Specifies the length of the frame header shown in Figure B.3 (see B.1.1.4).

P: Sample precision – Specifies the precision in bits for the samples of the components in the frame.

Y: Number of lines – Specifies the maximum number of lines in the source image. This shall be equal to the number of lines in the component with the maximum number of vertical samples (see A.1.1). Value 0 indicates that the number of lines shall be defined by the DNL marker and parameters at the end of the first scan (see B.2.5).

X: Number of samples per line – Specifies the maximum number of samples per line in the source image. This shall be equal to the number of samples per line in the component with the maximum number of horizontal samples (see A.1.1).

Nf: Number of image components in frame – Specifies the number of source image components in the frame. The value of Nf shall be equal to the number of sets of frame component specification parameters (C_i, H_i, V_i, and Tq_i) present in the frame header.

C_i: Component identifier – Assigns a unique label to the ith component in the sequence of frame component specification parameters. These values shall be used in the scan headers to identify the components in the scan. The value of C_i shall be different from the values of C_1 through C_{i-1}.

H_i: Horizontal sampling factor – Specifies the relationship between the component horizontal dimension and maximum image dimension X (see A.1.1); also specifies the number of horizontal data units of component C_i in each MCU, when more than one component is encoded in a scan.

V_i: Vertical sampling factor – Specifies the relationship between the component vertical dimension and maximum image dimension Y (see A.1.1); also specifies the number of vertical data units of component C_i in each MCU, when more than one component is encoded in a scan.

Tq_i: Quantization table destination selector – Specifies one of four possible quantization table destinations from which the quantization table to use for dequantization of DCT coefficients of component C_i is retrieved. If the decoding process uses the dequantization procedure, this table shall have been installed in this destination by the time the decoder is ready to decode the scan(s) containing component C_i. The destination shall not be re-specified, or its contents changed, until all scans containing C_i have been completed.

Table B.2 – Frame header parameter sizes and values

Parameter	Size (bits)	Values			
		Sequential DCT		Progressive DCT	Lossless
		Baseline	Extended		
Lf	16	$8 + 3 \times Nf$			
P	8	8	8, 12	8, 12	2-16
Y	16	0-65 535			
X	16	1-65 535			
Nf	8	1-255	1-255	1-4	1-255
C_i	8	0-255			
H_i	4	1-4			
V_i	4	1-4			
Tq_i	8	0-3	0-3	0-3	0

B.2.3   Scan header syntax

Figure B.4 specifies the scan header which shall be present at the start of a scan. This header specifies which component(s) are contained in the scan, specifies the destinations from which the entropy tables to be used with each component are retrieved, and (for the progressive DCT) which part of the DCT quantized coefficient data is contained in the scan. For lossless processes the scan parameters specify the predictor and the point transform.
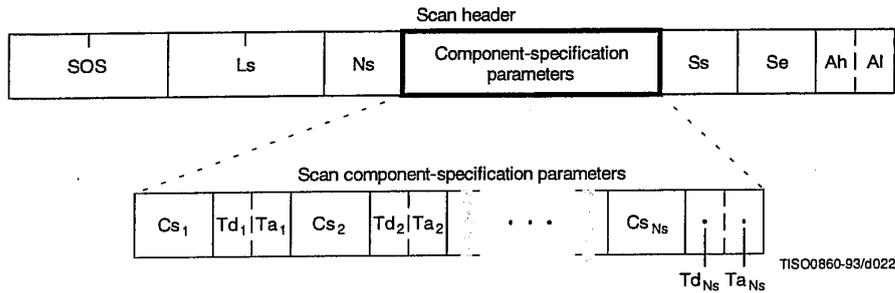
Figure B.4 – Scan header syntax

The marker and parameters shown in Figure B.4 are defined below. The size and allowed values of each parameter are given in Table B.3.

SOS: Start of scan marker – Marks the beginning of the scan parameters.

Ls: Scan header length – Specifies the length of the scan header shown in Figure B.4 (see B.1.1.4).

Ns: Number of image components in scan – Specifies the number of source image components in the scan. The value of Ns shall be equal to the number of sets of scan component specification parameters (Cs$_j$, Td$_j$, and Ta$_j$) present in the scan header.

Cs$_j$: Scan component selector – Selects which of the Nf image components specified in the frame parameters shall be the *j*th component in the scan. Each Cs$_j$ shall match one of the C$_i$ values specified in the frame header, and the ordering in the scan header shall follow the ordering in the frame header. If Ns > 1, the order of interleaved components in the MCU is Cs$_1$ first, Cs$_2$ second, etc. If Ns > 1, the following restriction shall be placed on the image components contained in the scan:

$$\sum_{j=1}^{N_s} H_j \times V_j \leq 10,$$

where H$_j$ and V$_j$ are the horizontal and vertical sampling factors for scan component j. These sampling factors are specified in the frame header for component i, where i is the frame component specification index for which frame component identifier C$_i$ matches scan component selector Cs$_j$.

As an example, consider an image having 3 components with maximum dimensions of 512 lines and 512 samples per line, and with the following sampling factors:

Component 0 $H_0 = 4,$ $V_0 = 1$
Component 1 $H_1 = 1,$ $V_1 = 2$
Component 2 $H_2 = 2$ $V_2 = 2$

Then the summation of H$_j$ × V$_j$ is (4 × 1) + (1 × 2) + (2 × 2) = 10.

The value of Cs$_j$ shall be different from the values of Cs$_1$ to Cs$_{j-1}$.

Td$_j$: DC entropy coding table destination selector – Specifies one of four possible DC entropy coding table destinations from which the entropy table needed for decoding of the DC coefficients of component Cs$_j$ is retrieved. The DC entropy table shall have been installed in this destination (see B.2.4.2 and B.2.4.3) by the time the decoder is ready to decode the current scan. This parameter specifies the entropy coding table destination for the lossless processes.

Ta$_j$: AC entropy coding table destination selector – Specifies one of four possible AC entropy coding table destinations from which the entropy table needed for decoding of the AC coefficients of component Cs$_j$ is retrieved. The AC entropy table selected shall have been installed in this destination (see B.2.4.2 and B.2.4.3) by the time the decoder is ready to decode the current scan. This parameter is zero for the lossless processes.

Ss: Start of spectral or predictor selection – In the DCT modes of operation, this parameter specifies the first DCT coefficient in each block in zig-zag order which shall be coded in the scan. This parameter shall be set to zero for the sequential DCT processes. In the lossless mode of operations this parameter is used to select the predictor.

Se: End of spectral selection – Specifies the last DCT coefficient in each block in zig-zag order which shall be coded in the scan. This parameter shall be set to 63 for the sequential DCT processes. In the lossless mode of operations this parameter has no meaning. It shall be set to zero.

Ah: Successive approximation bit position high – This parameter specifies the point transform used in the preceding scan (i.e. successive approximation bit position low in the preceding scan) for the band of coefficients specified by Ss and Se. This parameter shall be set to zero for the first scan of each band of coefficients. In the lossless mode of operations this parameter has no meaning. It shall be set to zero.

Al: Successive approximation bit position low or point transform – In the DCT modes of operation this parameter specifies the point transform, i.e. bit position low, used before coding the band of coefficients specified by Ss and Se. This parameter shall be set to zero for the sequential DCT processes. In the lossless mode of operations, this parameter specifies the point transform, Pt.

The entropy coding table destination selectors, Td$_j$ and Ta$_j$, specify either Huffman tables (in frames using Huffman coding) or arithmetic coding tables (in frames using arithmetic coding). In the latter case the entropy coding table destination selector specifies both an arithmetic coding conditioning table destination and an associated statistics area.

Table B.3 – Scan header parameter size and values

Parameter	Size (bits)	Values			
		Sequential DCT		Progressive DCT	Lossless
		Baseline	Extended		
Ls	16	$6 + 2 \times Ns$			
Ns	8	1-4			
Cs$_j$	8	0-255[a]			
Td$_j$	4	0-1	0-3	0-3	0-3
Ta$_j$	4	0-1	0-3	0-3	0
Ss	8	0	0	0-63	1-7[b]
Se	8	63	63	Ss-63[c]	0
Ah	4	0	0	0-13	0
Al	4	0	0	0-13	0-15

a) Cs$_j$ shall be a member of the set of C$_i$ specified in the frame header.

b) 0   for lossless differential frames in the hierarchical mode (see B.3).

c) 0   if Ss equals zero.

B.2.4   Table-specification and miscellaneous marker segment syntax

Figure B.5 specifies that, at the places indicated in Figure B.2, any of the table-specification segments or miscellaneous marker segments specified in B.2.4.1 through B.2.4.6 may be present in any order and with no limit on the number of segments.

If any table specification for a particular destination occurs in the compressed image data, it shall replace any previous table specified for this destination, and shall be used whenever this destination is specified in the remaining scans in the frame or subsequent images represented in the abbreviated format for compressed image data. If a table specification for a given destination occurs more than once in the compressed image data, each specification shall replace the previous specification. The quantization table specification shall not be altered between progressive DCT scans of a given component.
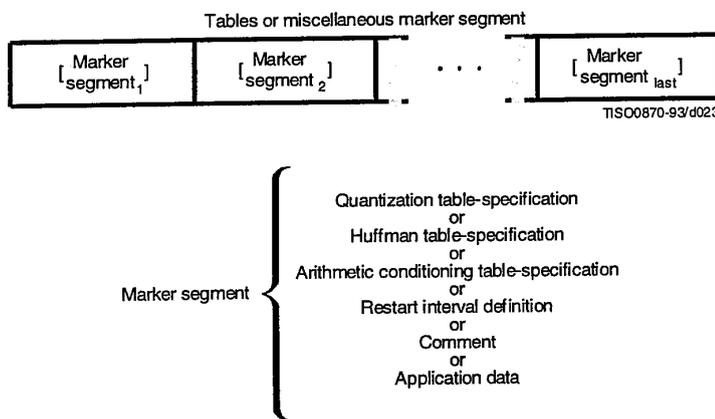

Figure B.5 – Tables/miscellaneous marker segment syntax

B.2.4.1   Quantization table-specification syntax

Figure B.6 specifies the marker segment which defines one or more quantization tables.
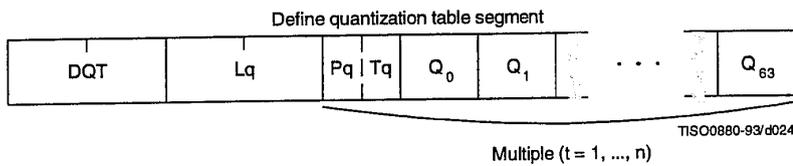

Figure B.6 – Quantization table syntax

The marker and parameters shown in Figure B.6 are defined below. The size and allowed values of each parameter are given in Table B.4.

DQT: Define quantization table marker – Marks the beginning of quantization table-specification parameters.

Lq: Quantization table definition length – Specifies the length of all quantization table parameters shown in Figure B.6 (see B.1.1.4).

Pq: Quantization table element precision – Specifies the precision of the Q_k values. Value 0 indicates 8-bit Q_k values; value 1 indicates 16-bit Q_k values. Pq shall be zero for 8 bit sample precision P (see B.2.2).

Tq: Quantization table destination identifier – Specifies one of four possible destinations at the decoder into which the quantization table shall be installed.

Q$_k$: Quantization table element – Specifies the kth element out of 64 elements, where k is the index in the zig-zag ordering of the DCT coefficients. The quantization elements shall be specified in zig-zag scan order.

Table B.4 – Quantization table-specification parameter sizes and values

Parameter	Size (bits)	Values			
		Sequential DCT		Progressive DCT	Lossless
		Baseline	Extended		
Lq	16	$2 + \sum\limits_{t=1}^{n} \left(65 + 64 \times Pq(t)\right)$			Undefined
Pq	4	0	0, 1	0, 1	Undefined
Tq	4	0-3			Undefined
Q$_k$	8, 16	1-255, 1-65 535			Undefined

The value n in Table B.4 is the number of quantization tables specified in the DQT marker segment.

Once a quantization table has been defined for a particular destination, it replaces the previous tables stored in that destination and shall be used, when referenced, in the remaining scans of the current image and in subsequent images represented in the abbreviated format for compressed image data. If a table has never been defined for a particular destination, then when this destination is specified in a frame header, the results are unpredictable.

An 8-bit DCT-based process shall not use a 16-bit precision quantization table.

B.2.4.2 Huffman table-specification syntax

Figure B.7 specifies the marker segment which defines one or more Huffman table specifications.
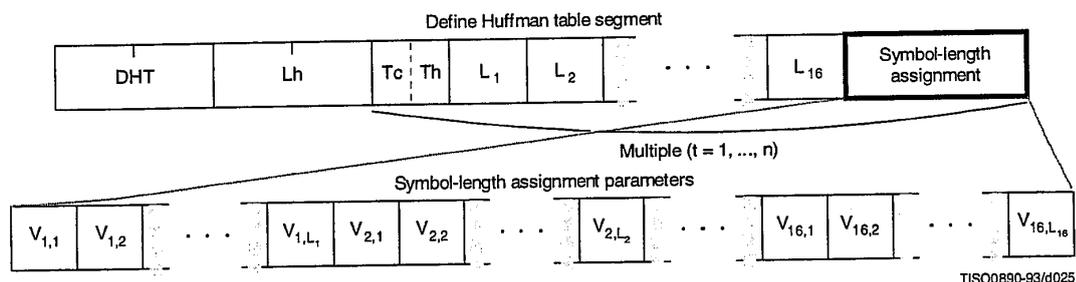

Figure B.7 – Huffman table syntax

The marker and parameters shown in Figure B.7 are defined below. The size and allowed values of each parameter are given in Table B.5.

DHT: Define Huffman table marker – Marks the beginning of Huffman table definition parameters.

Lh: Huffman table definition length – Specifies the length of all Huffman table parameters shown in Figure B.7 (see B.1.1.4).

Tc: Table class – 0 = DC table or lossless table, 1 = AC table.

Th: Huffman table destination identifier – Specifies one of four possible destinations at the decoder into which the Huffman table shall be installed.

L_i: Number of Huffman codes of length i – Specifies the number of Huffman codes for each of the 16 possible lengths allowed by this Specification. L_i's are the elements of the list BITS.

$V_{i,j}$: Value associated with each Huffman code – Specifies, for each i, the value associated with each Huffman code of length i. The meaning of each value is determined by the Huffman coding model. The $V_{i,j}$'s are the elements of the list HUFFVAL.

Table B.5 – Huffman table specification parameter sizes and values

Parameter	Size (bits)	Values			
		Sequential DCT		Progressive DCT	Lossless
		Baseline	Extended		
Lh	16	$2 + \sum_{t=1}^{n} (17 + m_t)$			
Tc	4	0, 1			0
Th	4	0, 1	0-3		
L_i	8	0-255			
$V_{i,j}$	8	0-255			

The value n in Table B.5 is the number of Huffman tables specified in the DHT marker segment. The value m_t is the number of parameters which follow the 16 $L_i(t)$ parameters for Huffman table t, and is given by:

$$m_t = \sum_{i=1}^{16} L_i$$

In general, m_t is different for each table.

Once a Huffman table has been defined for a particular destination, it replaces the previous tables stored in that destination and shall be used when referenced, in the remaining scans of the current image and in subsequent images represented in the abbreviated format for compressed image data. If a table has never been defined for a particular destination, then when this destination is specified in a scan header, the results are unpredictable.

B.2.4.3   Arithmetic conditioning table-specification syntax

Figure B.8 specifies the marker segment which defines one or more arithmetic coding conditioning table specifications. These replace the default arithmetic coding conditioning tables established by the SOI marker for arithmetic coding processes. (See F.1.4.4.1.4 and F.1.4.4.2.1.)
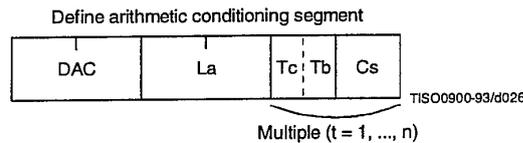
Define arithmetic conditioning segment

TISO0900-93/d026

Multiple (t = 1, ..., n)

Figure B.8 – Arithmetic conditioning table-specification syntax

The marker and parameters shown in Figure B.8 are defined below. The size and allowed values of each parameter are given in Table B.6.

DAC: Define arithmetic coding conditioning marker – Marks the beginning of the definition of arithmetic coding conditioning parameters.

La: Arithmetic coding conditioning definition length – Specifies the length of all arithmetic coding conditioning parameters shown in Figure B.8 (see B.1.1.4).

Tc: Table class – $0 =$ DC table or lossless table, $1 =$ AC table.

Tb: Arithmetic coding conditioning table destination identifier – Specifies one of four possible destinations at the decoder into which the arithmetic coding conditioning table shall be installed.

Cs: Conditioning table value – Value in either the AC or the DC (and lossless) conditioning table. A single value of Cs shall follow each value of Tb. For AC conditioning tables Tc shall be one and Cs shall contain a value of Kx in the range $1 \le Kx \le 63$. For DC (and lossless) conditioning tables Tc shall be zero and Cs shall contain two 4-bit parameters, U and L. U and L shall be in the range $0 \le L \le U \le 15$ and the value of Cs shall be $L + 16 \times U$.

The value n in Table B.6 is the number of arithmetic coding conditioning tables specified in the DAC marker segment. The parameters L and U are the lower and upper conditioning bounds used in the arithmetic coding procedures defined for DC coefficient coding and lossless coding. The separate value range 1-63 listed for DCT coding is the Kx conditioning used in AC coefficient coding.

Table B.6 – Arithmetic coding conditioning table-specification parameter sizes and values

Parameter	Size (bits)	Values			
		Sequential DCT		Progressive DCT	Lossless
		Baseline	Extended		
La	16	Undefined	$2 + 2 \times n$		
Tc	4	Undefined	0, 1		0
Tb	4	Undefined	0-3		
Cs	8	Undefined	0-255 (Tc = 0), 1-63 (Tc = 1)		0-255

B.2.4.4   Restart interval definition syntax

Figure B.9 specifies the marker segment which defines the restart interval.

Define restart interval segment

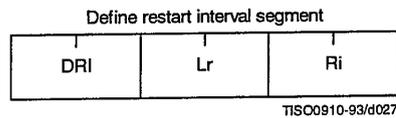| DRI	Lr	Ri

TISO0910-93/d027

Figure B.9 – Restart interval definition syntax

The marker and parameters shown in Figure B.9 are defined below. The size and allowed values of each parameter are given in Table B.7.

DRI: Define restart interval marker – Marks the beginning of the parameters which define the restart interval.

Lr: Define restart interval segment length – Specifies the length of the parameters in the DRI segment shown in Figure B.9 (see B.1.1.4).

Ri: Restart interval – Specifies the number of MCU in the restart interval.

In Table B.7 the value n is the number of rows of MCU in the restart interval. The value MCUR is the number of MCU required to make up one line of samples of each component in the scan. The SOI marker disables the restart intervals. A DRI marker segment with Ri nonzero shall be present to enable restart interval processing for the following scans. A DRI marker segment with Ri equal to zero shall disable restart intervals for the following scans.

Table B.7 – Define restart interval segment parameter sizes and values

Parameter	Size (bits)	Values			
		Sequential DCT		Progressive DCT	Lossless
		Baseline	Extended		
Lr	16	4			
Ri	16	0-65 535			n × MCUR

B.2.4.5   Comment syntax

Figure B.10 specifies the marker segment structure for a comment segment.

Comment segment

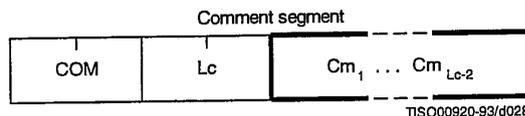| COM	Lc	Cm$_1$ ... Cm$_{Lc-2}$

TISO000920-93/d028

Figure B.10 – Comment segment syntax

The marker and parameters shown in Figure B.10 are defined below. The size and allowed values of each parameter are given in Table B.8.

COM: Comment marker – Marks the beginning of a comment.

Lc: Comment segment length – Specifies the length of the comment segment shown in Figure B.10 (see B.1.1.4).

Cm$_i$: Comment byte – The interpretation is left to the application.

Table B.8 – Comment segment parameter sizes and values

Parameter	Size (bits)	Values			
		Sequential DCT		Progressive DCT	Lossless
		Baseline	Extended		
Lc	16	2-65 535			
Cm$_i$	8	0-255			

B.2.4.6 Application data syntax

Figure B.11 specifies the marker segment structure for an application data segment.
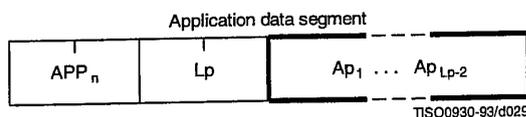

Figure B.11 – Application data syntax

The marker and parameters shown in Figure B.11 are defined below. The size and allowed values of each parameter are given in Table B.9.

APP$_n$: Application data marker – Marks the beginning of an application data segment.

Lp: Application data segment length – Specifies the length of the application data segment shown in Figure B.11 (see B.1.1.4).

Ap$_i$: Application data byte – The interpretation is left to the application.

The APP$_n$ (Application) segments are reserved for application use. Since these segments may be defined differently for different applications, they should be removed when the data are exchanged between application environments.

Table B.9 – Application data segment parameter sizes and values

Parameter	Size (bits)	Values			
		Sequential DCT		Progressive DCT	Lossless
		Baseline	Extended		
Lp	16	2-65 535			
Ap$_i$	8	0-255			

B.2.5   Define number of lines syntax

Figure B.12 specifies the marker segment for defining the number of lines. The DNL (Define Number of Lines) segment provides a mechanism for defining or redefining the number of lines in the frame (the Y parameter in the frame header) at the end of the first scan. The value specified shall be consistent with the number of MCU-rows encoded in the first scan. This segment, if used, shall only occur at the end of the first scan, and only after coding of an integer number of MCU-rows. This marker segment is mandatory if the number of lines (Y) specified in the frame header has the value zero.
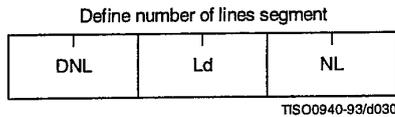
Define number of lines segment

DNL	Ld	NL

TISO0940-93/d030

Figure B.12 – Define number of lines syntax

The marker and parameters shown in Figure B.12 are defined below. The size and allowed values of each parameter are given in Table B.10.

DNL: Define number of lines marker – Marks the beginning of the define number of lines segment.

Ld: Define number of lines segment length – Specifies the length of the define number of lines segment shown in Figure B.12 (see B.1.1.4).

NL: Number of lines – Specifies the number of lines in the frame (see definition of Y in B.2.2).

Table B.10 – Define number of lines segment parameter sizes and values

Parameter	Size (bits)	Values			
		Sequential DCT		Progressive DCT	Lossless
		Baseline	Extended		
Ld	16	4			
NL	16	1-65 535[a]			

a)   The value specified shall be consistent with the number of lines coded at the point where the DNL segment terminates the compressed data segment.

B.3   Hierarchical syntax

B.3.1   High level hierarchical mode syntax

Figure B.13 specifies the order of the high level constituent parts of the interchange format for hierarchical encoding processes.
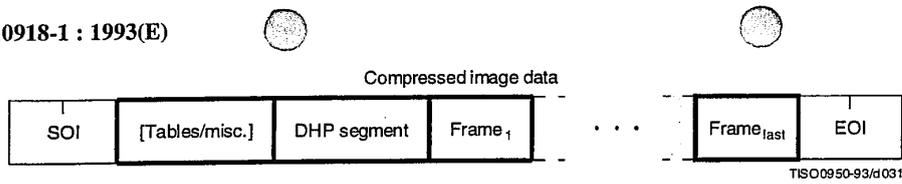
Compressed image data

| SOI | [Tables/misc.] | DHP segment | Frame$_1$ | $\cdots$ | Frame$_{last}$ | EOI |

TISO0950-93/d031

Figure B.13 – Syntax for the hierarchical mode of operation

Hierarchical mode syntax requires a DHP marker segment that appears before the non-differential frame or frames. The hierarchical mode compressed image data may include EXP marker segments and differential frames which shall follow the initial non-differential frame. The frame structure in hierarchical mode is identical to the frame structure in non-hierarchical mode.

The non-differential frames in the hierarchical sequence shall use one of the coding processes specified for SOF$_n$ markers: SOF$_0$, SOF$_1$, SOF$_2$, SOF$_3$, SOF$_9$, SOF$_{10}$ and SOF$_{11}$. The differential frames shall use one of the processes specified for SOF$_5$, SOF$_6$, SOF$_7$, SOF$_{13}$, SOF$_{14}$ and SOF$_{15}$. The allowed combinations of SOF markers within one hierarchical sequence are specified in Annex J.

The sample precision (P) shall be constant for all frames and have the identical value as that coded in the DHP marker segment. The number of samples per line (X) for all frames shall not exceed the value coded in the DHP marker segment. If the number of lines (Y) is non-zero in the DHP marker segment, then the number of lines for all frames shall not exceed the value in the DHP marker segment.

B.3.2   DHP segment syntax

The DHP segment defines the image components, size, and sampling factors for the completed hierarchical sequence of frames. The DHP segment shall precede the first frame; a single DHP segment shall occur in the compressed image data.

The DHP segment structure is identical to the frame header syntax, except that the DHP marker is used instead of the SOF$_n$ marker. The figures and description of B.2.2 then apply, except that the quantization table destination selector parameter shall be set to zero in the DHP segment.

B.3.3   EXP segment syntax

Figure B.14 specifies the marker segment structure for the EXP segment. The EXP segment shall be present if (and only if) expansion of the reference components is required either horizontally or vertically. The EXP segment parameters apply only to the next frame (which shall be a differential frame) in the image. If required, the EXP segment shall be one of the table-specification segments or miscellaneous marker segments preceding the frame header; the EXP segment shall not be one of the table-specification segments or miscellaneous marker segments preceding a scan header or a DHP marker segment.
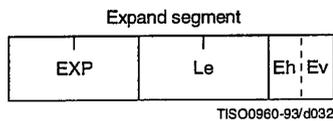
Expand segment

| EXP | Le | Eh | Ev |

TISO0960-93/d032

Figure B.14 – Syntax of the expand segment

The marker and parameters shown in Figure B.14 are defined below. The size and allowed values of each parameter are given in Table B.11.

EXP: Expand reference components marker – Marks the beginning of the expand reference components segment.

Le: Expand reference components segment length – Specifies the length of the expand reference components segment (see B.1.1.4).

Eh: Expand horizontally – If one, the reference components shall be expanded horizontally by a factor of two. If horizontal expansion is not required, the value shall be zero.

Ev: Expand vertically – If one, the reference components shall be expanded vertically by a factor of two. If vertical expansion is not required, the value shall be zero.

Both Eh and Ev shall be one if expansion is required both horizontally and vertically.

Table B.11 – Expand segment parameter sizes and values

Parameter	Size (bits)	Values			
		Sequential DCT		Progressive DCT	Lossless
		Baseline	Extended		
Le	16	3			
Eh	4	0, 1			
Ev	4	0, 1			

B.4 Abbreviated format for compressed image data

Figure B.2 shows the high-level constituent parts of the interchange format. This format includes all table specifications required for decoding. If an application environment provides methods for table specification other than by means of the compressed image data, some or all of the table specifications may be omitted. Compressed image data which is missing any table specification data required for decoding has the abbreviated format.

B.5 Abbreviated format for table-specification data

Figure B.2 shows the high-level constituent parts of the interchange format. If no frames are present in the compressed image data, the only purpose of the compressed image data is to convey table specifications or miscellaneous marker segments defined in B.2.4.1, B.2.4.2, B.2.4.5, and B.2.4.6. In this case the compressed image data has the abbreviated format for table specification data (see Figure B.15).
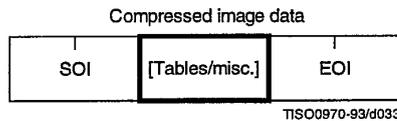
Compressed image data

SOI	[Tables/misc.]	EOI

TISO0970-93/d033

Figure B.15 – Abbreviated format for table-specification data syntax

B.6 Summary

The order of the constituent parts of interchange format and all marker segment structures is summarized in Figures B.16 and B.17. Note that in Figure B.16 double-lined boxes enclose marker segments. In Figures B.16 and B.17 thick-lined boxes enclose only markers.

The EXP segment can be mixed with the other tables/miscellaneous marker segments preceding the frame header but not with the tables/miscellaneous marker segments preceding the DHP segment or the scan header.
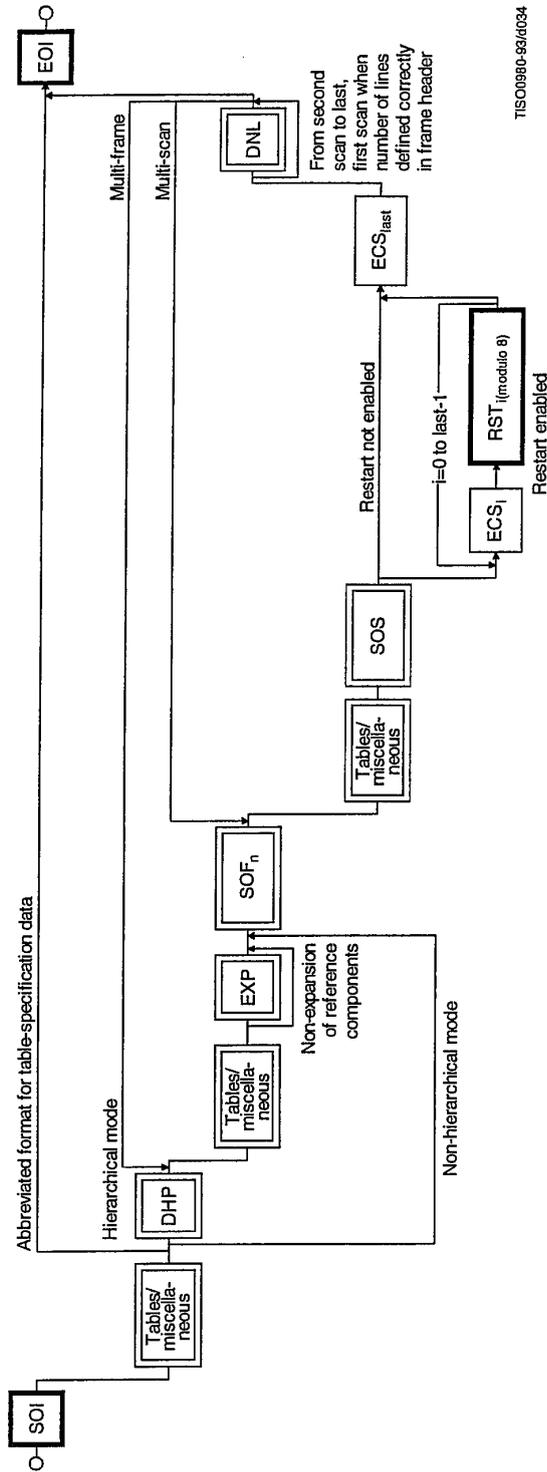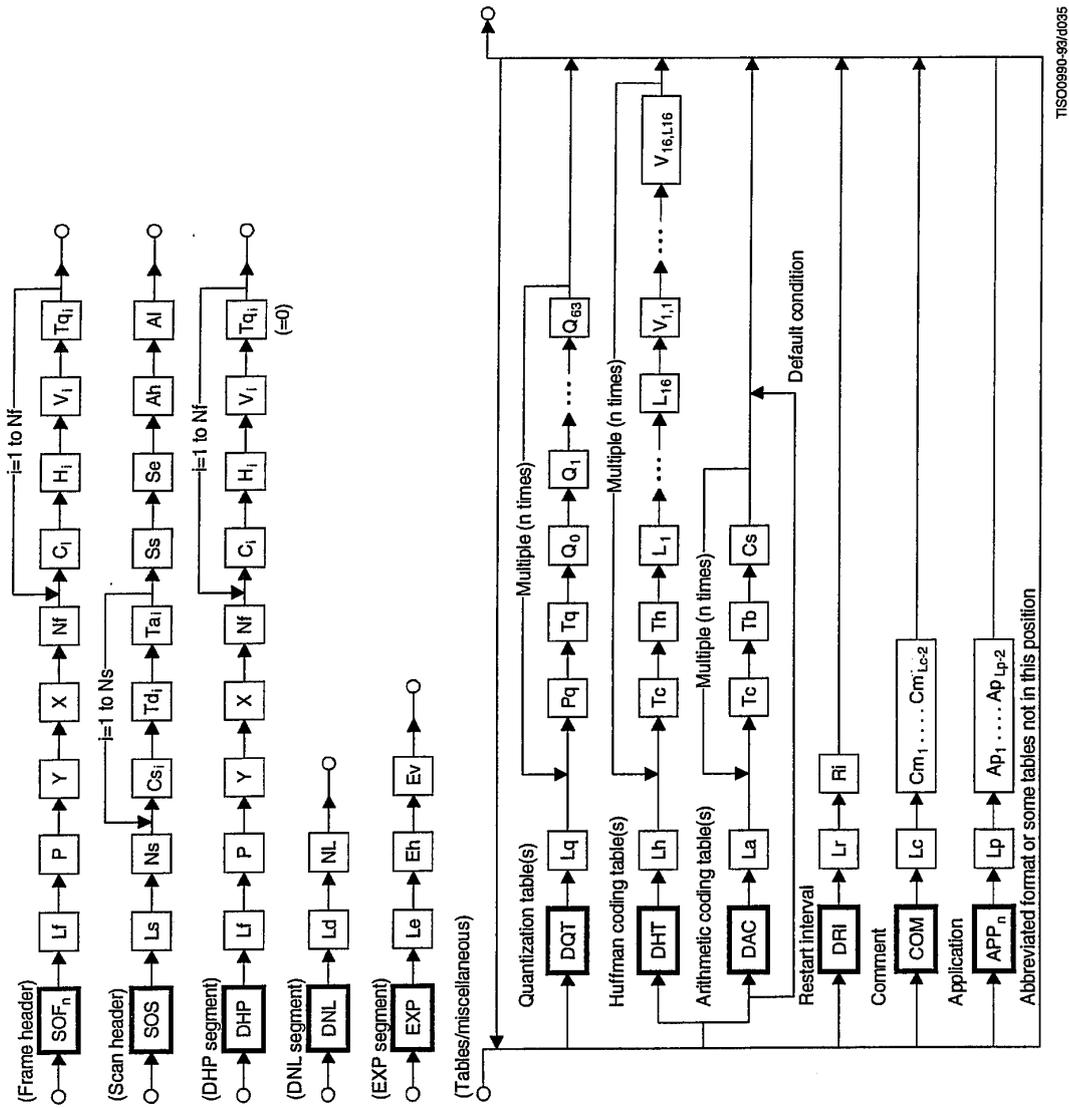
Figure B.16 – Flow of compressed data syntax

Figure B.17 – Flow of marker segment

Annex C

Huffman table specification

(This annex forms an integral part of this Recommendation I International Standard)

A Huffman coding procedure may be used for entropy coding in any of the coding processes. Coding models for Huffman encoding are defined in Annexes F, G, and H. In this Annex, the Huffman table specification is defined.

Huffman tables are specified in terms of a 16-byte list (BITS) giving the number of codes for each code length from 1 to 16. This is followed by a list of the 8-bit symbol values (HUFFVAL), each of which is assigned a Huffman code. The symbol values are placed in the list in order of increasing code length. Code lengths greater than 16 bits are not allowed. In addition, the codes shall be generated such that the all-1-bits code word of any length is reserved as a prefix for longer code words.

NOTE – The order of the symbol values within HUFFVAL is determined only by code length. Within a given code length the ordering of the symbol values is arbitrary.

This annex specifies the procedure by which the Huffman tables (of Huffman code words and their corresponding 8-bit symbol values) are derived from the two lists (BITS and HUFFVAL) in the interchange format. However, the way in which these lists are generated is not specified. The lists should be generated in a manner which is consistent with the rules for Huffman coding, and it shall observe the constraints discussed in the previous paragraph. Annex K contains an example of a procedure for generating lists of Huffman code lengths and values which are in accord with these rules.

NOTE – There is **no requirement** in this Specification that any encoder or decoder shall implement the procedures in precisely the manner specified by the flow charts in this annex. It is necessary only that an encoder or decoder implement the **function** specified in this annex. The sole criterion for an encoder or decoder to be considered in compliance with this Specification is that it satisfy the requirements given in clause 6 (for encoders) or clause 7 (for decoders), as determined by the compliance tests specified in Part 2.

C.1   Marker segments for Huffman table specification

The DHT marker identifies the start of Huffman table definitions within the compressed image data. B.2.4.2 specifies the syntax for Huffman table specification.

C.2   Conversion of Huffman table specifications to tables of codes and code lengths

Conversion of Huffman table specifications to tables of codes and code lengths uses three procedures. The first procedure (Figure C.1) generates a table of Huffman code sizes. The second procedure (Figure C.2) generates the Huffman codes from the table built in Figure C.1. The third procedure (Figure C.3) generates the Huffman codes in symbol value order.

Given a list BITS (1 to 16) containing the number of codes of each size, and a list HUFFVAL containing the symbol values to be associated with those codes as described above, two tables are generated. The HUFFSIZE table contains a list of code lengths; the HUFFCODE table contains the Huffman codes corresponding to those lengths.

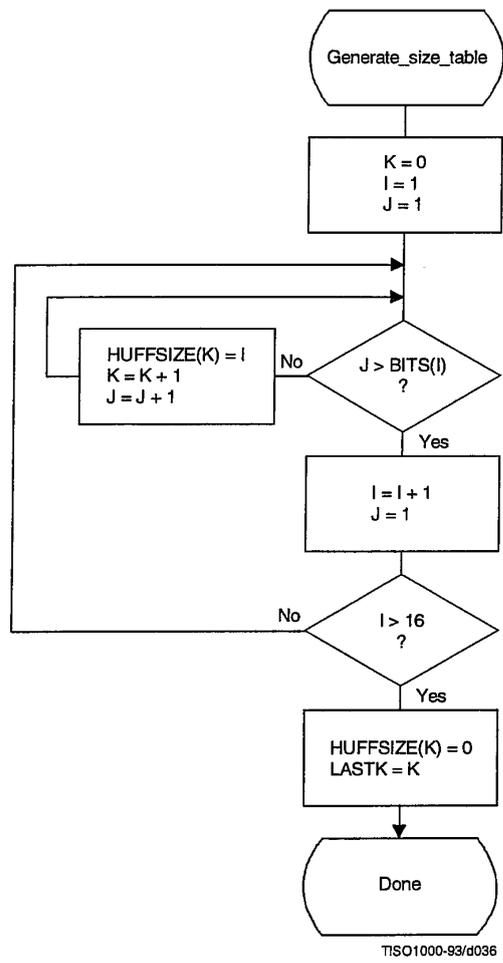Note that the variable LASTK is set to the index of the last entry in the table.

TISO1000-93/d036

Figure C.1 – Generation of table of Huffman code sizes

A Huffman code table, HUFFCODE, containing a code for each size in HUFFSIZE is generated by the procedure in Figure C.2. The notation "SLL CODE 1" in Figure C.2 indicates a shift-left-logical of CODE by one bit position.
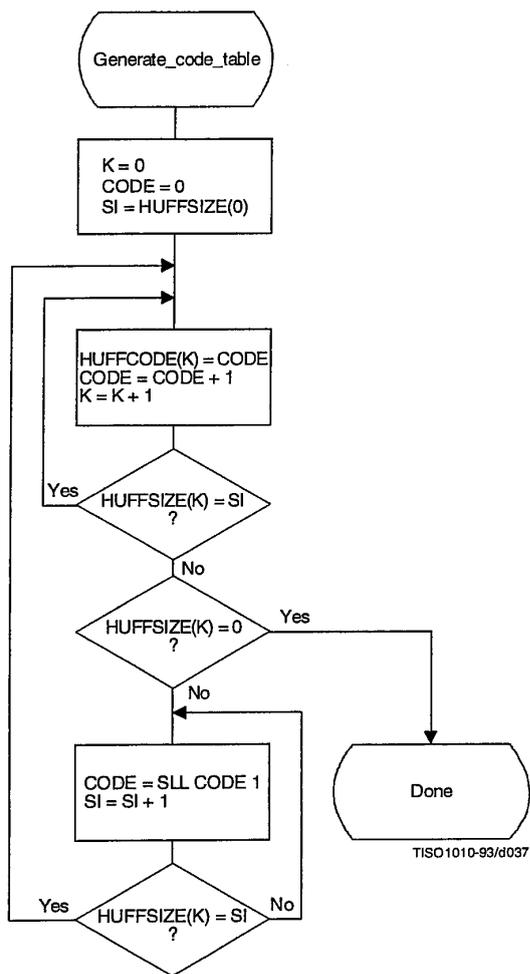

Figure C.2 – Generation of table of Huffman codes

Two tables, HUFFCODE and HUFFSIZE, have now been generated. The entries in the tables are ordered according to increasing Huffman code numeric value and length.

The encoding procedure code tables, EHUFCO and EHUFSI, are created by reordering the codes specified by HUFFCODE and HUFFSIZE according to the symbol values assigned to each code in HUFFVAL.
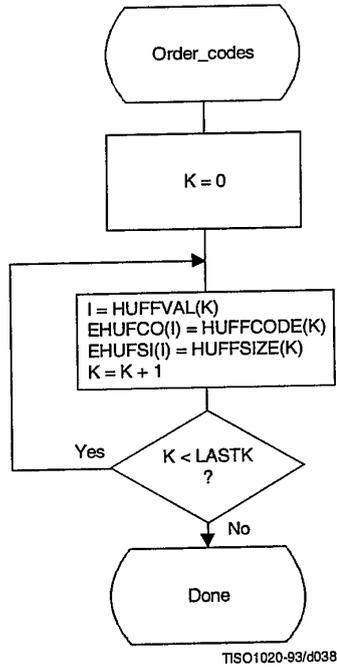
Figure C.3 illustrates this ordering procedure.

TISO1020-93/d038

Figure C.3 – Ordering procedure for encoding procedure code tables

C.3 Bit ordering within bytes

The root of a Huffman code is placed toward the MSB (most-significant-bit) of the byte, and successive bits are placed in the direction MSB to LSB (least-significant-bit) of the byte. Remaining bits, if any, go into the next byte following the same rules.

Integers associated with Huffman codes are appended with the MSB adjacent to the LSB of the preceding Huffman code.

Annex D

Arithmetic coding

(This annex forms an integral part of this Recommendation I International Standard)

An adaptive binary arithmetic coding procedure may be used for entropy coding in any of the coding processes except the baseline sequential process. Coding models for adaptive binary arithmetic coding are defined in Annexes F, G, and H. In this annex the arithmetic encoding and decoding procedures used in those models are defined.

In K.4 a simple test example is given which should be helpful in determining if a given implementation is correct.

NOTE – There is **no requirement** in this Specification that any encoder or decoder shall implement the procedures in precisely the manner specified by the flow charts in this annex. It is necessary only that an encoder or decoder implement the **function** specified in this annex. The sole criterion for an encoder or decoder to be considered in compliance with this Specification is that it satisfy the requirements given in clause 6 (for encoders) or clause 7 (for decoders), as determined by the compliance tests specified in Part 2.

D.1 Arithmetic encoding procedures

Four arithmetic encoding procedures are required in a system with arithmetic coding (see Table D.1).

Table D.1 – Procedures for binary arithmetic encoding

Procedure	Purpose
Code_0(S)	Code a "0" binary decision with context-index S
Code_1(S)	Code a "1" binary decision with context-index S
Initenc	Initialize the encoder
Flush	Terminate entropy-coded segment

The "Code_0(S)"and "Code_1(S)" procedures code the 0-decision and 1-decision respectively; S is a context-index which identifies a particular conditional probability estimate used in coding the binary decision. The "Initenc" procedure initializes the arithmetic coding entropy encoder. The "Flush" procedure terminates the entropy-coded segment in preparation for the marker which follows.

D.1.1 Binary arithmetic encoding principles

The arithmetic coder encodes a series of binary symbols, zeros and ones, each symbol representing one possible result of a binary decision.

Each "binary decision" provides a choice between two alternatives. The binary decision might be between positive and negative signs, a magnitude being zero or nonzero, or a particular bit in a sequence of binary digits being zero or one.

The output bit stream (entropy-coded data segment) represents a binary fraction which increases in precision as bytes are appended by the encoding process.

D.1.1.1 Recursive interval subdivision

Recursive probability interval subdivision is the basis for the binary arithmetic encoding procedures. With each binary decision the current probability interval is subdivided into two sub-intervals, and the bit stream is modified (if necessary) so that it points to the base (the lower bound) of the probability sub-interval assigned to the symbol which occurred.

In the partitioning of the current probability interval into two sub-intervals, the sub-interval for the less probable symbol (LPS) and the sub-interval for the more probable symbol (MPS) are ordered such that usually the MPS sub-interval is closer to zero. Therefore, when the LPS is coded, the MPS sub-interval size is added to the bit stream. This coding convention requires that symbols be recognized as either MPS or LPS rather than 0 or 1. Consequently, the size of the LPS sub-interval and the sense of the MPS for each decision must be known in order to encode that decision.