ATTACHMENT F

TO REQUEST FOR *EX PARTE* REEXAMINATION OF U.S. PATENT NO. 7,868,912

Moving Object Detection and Event Recognition Algorithms for Smart Cameras

Thomas J. Olson
Frank Z. Brill

Texas Instruments
Research & Development
P.O. Box 655303, MS 8374, Dallas, TX 75265
E-mail: olson@csc.ti.com, brill@ti.com
http://www.ti.com/research/docs/iuba/index.html

Abstract

Smart video cameras analyze the video stream and translate it into a description of the scene in terms of objects, object motions, and events. This paper describes a set of algorithms for the core computations needed to build smart cameras. Together these algorithms make up the Autonomous Video Surveillance (AVS) system, a general-purpose framework for moving object detection and event recognition. Moving objects are detected using change detection, and are tracked using first-order prediction and nearest neighbor matching. Events are recognized by applying predicates to the graph formed by linking corresponding objects in successive frames.The AVS algorithms have been used to create several novel video surveillance applications. These include a video surveillance shell that allows a human to monitor the outputs of multiple cameras, a system that takes a single high-quality snapshot of every person who enters its field of view, and a system that learns the structure of the monitored environment by watching humans move around in the scene.

1 Introduction

Video cameras today produce images, which must be examined by humans in order to be useful. Future 'smart' video cameras will produce information, including descriptions of the environment they are monitoring and the events taking place in it. The information they produce may include images and video clips, but these will be carefully selected to maximize their useful information content. The symbolic information and images from smart cameras will be filtered by programs that extract data relevant to particular tasks. This filtering process will enable a single human to monitor hundreds or thousands of video streams.

In pursuit of our research objectives [Flinchbaugh, 1997], we are developing the technology needed to make smart cameras a reality. Two fundamental capabilities are needed. The first is the ability to describe scenes in terms of object motions and interactions. The second is the ability to recognize important events that occur in the scene, and to pick out those that are relevant to the current task. These capabilities make it possible to develop a variety of novel and useful video surveillance applications.

1.1 Video Surveillance and Monitoring Scenarios

Our work is motivated by a several types of video surveillance and monitoring scenarios.

Indoor Surveillance: Indoor surveillance provides information about areas such as building lobbies, hallways, and offices. Monitoring tasks in lobbies and hallways include detection of people depositing things (e.g., unattended luggage in an airport lounge), removing things (e.g., theft), or loitering. Office monitoring tasks typically require information about people's identities: in an office, for example, the office owner may do anything at any

The research described in this report was sponsored in part by the DARPA Image Understanding Program.

time, but other people should not open desk drawers or operate the computer unless the owner is present. Cleaning staff may come in at night to vacuum and empty trash cans, but should not handle objects on the desk.

Outdoor Surveillance: Outdoor surveillance includes tasks such as monitoring a site perimeter for intrusion or threats from vehicles (e.g., car bombs). In military applications, video surveillance can function as a sentry or forward observer, e.g. by notifying commanders when enemy soldiers emerge from a wooded area or cross a road.

In order for smart cameras to be practical for real-world tasks, the algorithms they use must be robust. Current commercial video surveillance systems have a high false alarm rate [Ringler and Hoover, 1995], which renders them useless for most applications. For this reason, our research stresses robustness and quantification of detection and false alarm rates. Smart camera algorithms must also run effectively on low-cost platforms, so that they can be implemented in small, low-power packages and can be used in large numbers. Studying algorithms that can run in near real time makes it practical to conduct extensive evaluation and testing of systems, and may enable worthwhile near-term applications as well as contributing to long-term research goals.

1.2 Approach

The first step in processing a video stream for surveillance purposes is to identify the important objects in the scene. In this paper it is assumed that the important objects are those that move independently. Camera parameters are assumed to be fixed. This allows the use of simple change detection to identify moving objects. Where use of moving cameras is necessary, stabilization hardware and stabilized moving object detection algorithms can be used (e.g. [Burt et al, 1989, Nelson, 1991]. The use of criteria other than motion (e.g., salience based on shape or color, or more general object recognition) is compatible with our approach, but these criteria are not used in our current applications.

Our event recognition algorithms are based on graph matching. Moving objects in the image are tracked over time. Observations of an object in successive video frames are linked to form a directed graph (the *motion graph*). Events are defined in terms of predicates on the motion graph. For instance, the beginning of a chain of successive observations of an object is defined to be an ENTER event. Event detection is described in more detail below.

Our approach to video surveillance stresses 2D, image-based algorithms and simple, low-level object representations that can be extracted reliably from the video sequence. This emphasis yields a high level of robustness and low computational cost. Object recognition and other detailed analyses are used only after the system has determined that the objects in question are interesting and merit further investigation.

1.3 Research Strategy

The primary technical goal of this research is to develop general-purpose algorithms for moving object detection and event recognition. These algorithms comprise the Autonomous Video Surveillance (AVS) system, a modular framework for building video surveillance applications. AVS is designed to be updated to incorporate better core algorithms or to tune the processing to specific domains as our research progresses.

In order to evaluate the AVS core algorithms and event recognition and tracking framework, we use them to develop applications motivated by the surveillance scenarios described above. The applications are small-scale implementations of future smart camera systems. They are designed for long-term operation, and are evaluated by allowing them to run for long periods (hours or days) and analyzing their output.

The remainder of this paper is organized as follows. The next section discusses related work. Section 3 presents the core moving object detection and event recognition algorithms, and the mechanism used to establish the 3D positions of objects. Section 4 presents applications that have been built using the AVS framework. The final section discusses the current state of the system and our future plans.

2 Related Work

Our overall approach to video surveillance has been influenced by interest in selective attention and task-oriented processing [Swain and Stricker, 1991, Rimey and Brown, 1993, Camus et al., 1993]. The fundamental problem with current video surveillance technology is that the useful information density of the images delivered to a human is very low; the vast majority of surveillance video frames contain no useful information at all. The fundamental role of the smart camera described above is to reduce the volume of data produced by the camera, and increase the value of that data. It does this by discarding irrelevant frames, and by expressing the information in the relevant frames primarily in symbolic form.

2.1 Moving Object Detection

Most algorithms for moving object detection using fixed cameras work by comparing incoming video frames to a reference image, and attributing significant differences either to motion or to noise. The algorithms differ in the form of the comparison operator they use, and in the way in which the reference image is maintained. Simple intensity differencing followed by thresholding is widely used [Jain et al., 1979, Yalamanchili et al., 1982, Kelly et al., 1995, Bobick and Davis, 1996, Courtney, 1997] because it is computationally inexpensive and works quite well in many indoor environments. Some algorithms provide a means of adapting the reference image over time, in order to track slow changes in lighting conditions and/or changes in the environment [Karmann and von Brandt, 1990, Makarov, 1996a]. Some also filter the image to reduce or remove low spatial frequency content, which again makes the detector less sensitive to lighting changes [Makarov et al., 1996b, Koller et al., 1994].

Recent work [Pentland, 1996, Kahn et al., 1996] has extended the basic change detection paradigm by replacing the reference image with a statistical model of the background. The comparison operator becomes a statistical test that estimates the probability that the observed pixel value belongs to the background.

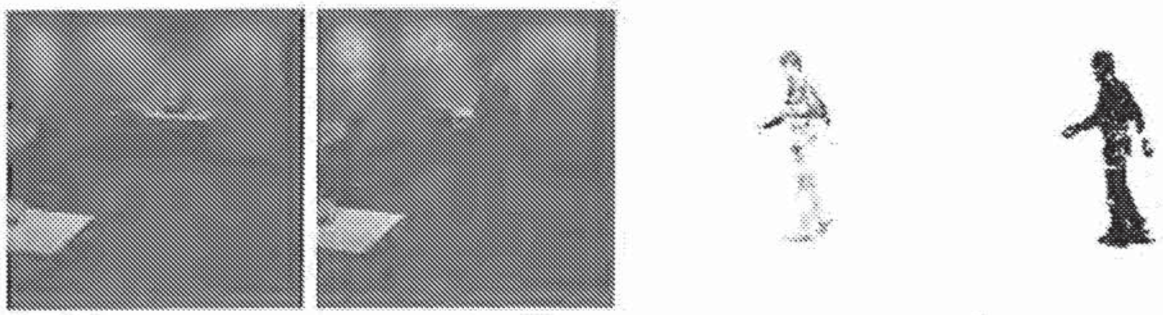
Our baseline change detection algorithm uses thresholded absolute differencing, since this works well for our indoor surveillance scenarios. For applications where lighting change is a problem, we use the adaptive reference frame algorithm of Karmann and von Brandt [1990]. We are also experimenting with a probabilistic change detector similar to Pfinder [Pentland, 1996.

Our work assumes fixed cameras. When the camera is not fixed, simple change detection cannot be used because of background motion. One approach to this problem is to treat the scene as a collection of independently moving objects, and to detect and ignore the visual motion due to camera motion [e.g. Burt et al., 1989] Other researchers have proposed ways of detecting features of the optical flow that are inconsistent with a hypothesis of self motion [Nelson, 1991].

In many of our applications moving object detection is a prelude to person detection. There has been significant recent progress in the development of algorithms to locate and track humans. Pfinder (cited above) uses a coarse statistical model of human body geometry and motion to estimate the likelihood that a given pixel is part of a human. Several researchers have described methods of tracking human body and limb movements [Gavrila and Davis, 1996, Kakadiaris and Metaxas, 1996] and locating faces in images [Sung and Poggio, 1994, Rowley et al., 1996]. Intille and Bobick [1995] describe methods of tracking humans through episodes of mutual occlusion in a highly structured environment. We do not currently make use of these techniques in live experiments because of their computational cost. However, we expect that this type of analysis will eventually be an important part of smart camera processing.

2.2 Event Recognition

Most work on event recognition has focussed on events that consist of a well-defined sequence of primitive motions. This class of events can be converted into spatiotemporal patterns and recognized using statistical pattern matching techniques. A number of researchers have demonstrated algorithms for recognizing gestures and sign language [e.g., Starner and Pentland, 1995]. Bobick and Davis [1996] describe a method of recognizing ste-



Reference Image Video Frame Difference Image Thresholded Image

Figure 1: Image processing steps for moving object detection.

reotypical motion patterns corresponding to actions such as sitting down, walking, or waving.

Our approach to event recognition is based on the video database indexing work of Courtney [1997], which introduced the use of predicates on the motion graph to represent events. Motion graphs are well suited to representing abstract, generic events such as 'depositing an object' or 'coming to rest', which are difficult to capture using the pattern-based approaches referred to above. On the other hand, pattern-based approaches can represent complex motions such as 'throwing an object' or 'waving', which would be difficult to express using motion graphs. It is likely that both pattern-based and abstract event recognition techniques will be needed to handle the full range of events that are of interest in surveillance applications.

3 AVS Tracking and Event Recognition Algorithms

This section describes the core technologies that provide the video surveillance and monitoring capabilities of the AVS system. There are three key technologies: moving object detection, visual tracking, and event recognition. The moving object detection routines determine when one or more objects enter a monitored scene, decide which pixels in a given video frame correspond to the moving objects versus which pixels correspond to the background, and form a simple representation of the object's image in the video frame. This representation is referred to as a *motion region*, and it exists in a single video frame, as distinguished from the *world objects* which exist in the world and give rise to the motion regions.

Visual tracking consists of determining correspondences between the motion regions over a sequence of video frames, and maintaining a single representation, or *track*, for the world object which gave rise to the sequence of motion regions in the sequence of frames. Finally, event recognition is a means of analyzing the collection of tracks in order to identify events of interest involving the world objects represented by the tracks.

3.1 Moving Object Detection

The moving object detection technology we employ is a 2D change detection technique similar to that described in Jain et al. [1979] and Yalamanchili et al [1982]. Prior to activation of the monitoring system, an image of the background, i.e., an image of the scene which contains no moving or otherwise interesting objects, is captured to serve as the *reference image*. When the system is in operation, the absolute difference of the current video frame from the reference image is computed to produce a *difference image*. The difference image is then thresholded at an appropriate value to obtain a binary image in which the "off" pixels represent background pixels, and the "on" pixels represent "moving object" pixels. The four-connected components of moving object pixels in the thresholded image are the motion regions (see Figure 1).

Simple application of the object detection procedure outlined above results in a number of errors, largely due to the limitations of thresholding. If the threshold used is too low, camera noise and shadows will produce spurious objects; whereas if the threshold is too high, some portions of the objects in the scene will fail to be separated from the back-

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.