Computer Vision and Image Understanding 113 (2009) 291–307

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Detecting motion patterns via direction maps with application to surveillance

Jacob M. Gryn *, Richard P. Wildes, John K. Tsotsos

Department of Computer Science & Engineering, Centre for Vision Research, York University, 4700 Keele St., Toronto, Ont., Canada M3J 1P3

ARTICLE INFO

Article history:
Received 13 July 2005
Accepted 23 October 2008
Available online 13 November 2008

Keywords:
Surveillance
Direction maps
Dominant direction
Event detection
Spatiotemporal analysis
Motion analysis

ABSTRACT

Detection of motion patterns in video data can be significantly simplified by abstracting away from pixel intensity values towards representations that explicitly and compactly capture movement across space and time. A novel representation that captures the spatiotemporal distributions of motion across regions of interest, called the "Direction Map," abstracts video data by assigning a two-dimensional vector, representative of local direction of motion, to quantized regions in space-time. Methods are presented for recovering direction maps from video, constructing direction map templates (defining target motion patterns of interest) and comparing templates to newly acquired video (for pattern detection and localization). These methods have been successfully implemented and tested (with real-time considerations) on over 6300 frames across seven surveillance/traffic videos, detecting potential targets of interest as they traverse the scene in specific ways. Results show an overall recognition rate of approximately 91% hits vs 8% false positives.

1. Introduction

1.1. Motivation

There is a need for systems to automatically detect motion patterns of interest (e.g., potential 'threats') in surveillance and other video to reduce the load on personnel, who simultaneously watch numerous monitors. As an example: Interviews with surveillance personnel at York University in Toronto have indicated that surveillance requires a high degree of knowledge of the areas being surveyed (both in the physical and image domain) as well as constant awareness of what is happening at the current time in these areas [1]. This knowledge is then applied to select which cameras to attend to, using the limited number of video monitors available. However, even with the most strategic 'game plan' of selecting which cameras to survey, detecting every potentially threatening incident as it happens is practically impossible due to the high ratio of cameras to surveillance staff. As a result, a need arises for intelligent surveillance systems to be developed.

Potential events of interest may include a group of people suddenly converging to or diverging from a point in a surveillance video, someone entering through an exit, or, in the case of traffic monitoring, a car making an illegal left turn; all of these encompass *global motion information* (i.e., globally visible motion patterns corresponding to objects moving across different large-scale regions of a surveyed area). As a result, it is advantageous to develop an algorithm to detect such global motion patterns. An intelligent system must be able to run on multiple cameras simultaneously and detect potentially threatening activity. Security staff should be notified of such an incident (with video footage) within a reasonable amount of time; the system should also log the incident for later retrieval by authorities, if necessary.

Consider the aforementioned car making a left turn. As humans, we recognize the car as making a left turn, simply by watching the car as it moves forward, towards the center of the intersection, and then left into the lane where the car is turning. For a computer, it is hypothesized that even a coarse representation of motion is sufficient to detect such an event; one such representation is the proposed *direction map* used for capturing *global motion* patterns.

Global motion patterns can be mapped onto spatiotemporal *direction maps* containing a coarse representation of motion described in terms of local dominant directions of motion (i.e., the direction that best accounts for the aggregate movement in a localized spatiotemporal region; this terminology is used as each point in space-time potentially has a direction of motion; therefore, each region has a dominant direction). For example, a car making a left turn at an intersection can be mapped to a series of regions in space-time with localized directions pointing towards the center of the intersection, and as time progresses, towards the lane into which the car is turning (see Fig. 1). Algorithms using direction maps eliminate the need for explicit tracking and/or segmentation and therefore eliminate an unnecessary layer of complexity that can potentially result in errors. Such a tracking and segmentation-free algorithm to detect user-defined motion patterns is proposed in the current work. It is hypothesized that video containing motion can be translated into *direction maps* by the

* Corresponding author. Fax: +1 416 736 5857.
E-mail addresses: jgryn@cse.yorku.ca (J.M. Gryn), wildes@cse.yorku.ca (R.P. Wildes), tsotsos@cse.yorku.ca (J.K. Tsotsos).

1077-3142/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2008.10.006

Fig. 1. A direction map representing a car making a left turn. At first, motion (signified by small arrows) is directed forward from the left-turn lane, as time progresses (represented by additional images in the figure), the arrows point towards the destination. The time course is shown from left to right, top to bottom.

use of spatiotemporal analysis to be compared with hand-made templates of motion patterns for detection.

1.2. Related research

Extensive research has been performed in image motion analysis [2,3]; however, research considered in this discussion will be limited to previous work in detection of motion patterns most closely related to the proposed approach. Such research can generally be classified into two main categories: (a) tracking and segmenting of objects/people and (b) detection of behavior-based events.

With respect to tracking and segmentation, the majority of algorithms make use of background subtraction [4–9]; others include the use of optical flow [10–12]; other various methods are proposed [13–16], including, for example, a multi-model system that fits the best model for any given frame [14].

Traditional work in background subtraction relied on manual initialization and had little, if any, ability to adapt to irrelevant changes, e.g., [17]. One straightforward way to increase robustness is to reconstruct the background by using a temporal median filter; however, it was found that performance was low in scenes with lighting intensity changes [4,9]. A slightly more complex method of background modeling was employed in the W4 system [5,18]. This system keeps track of various parameters extracted from intensity values that are then used to create a threshold needed to segment the foreground pixels. The system makes use of morphological techniques (erosion and dilation) in an attempt to compensate for changes in lighting. This work has an advantage that it makes use of basic shape analysis in order to continue tracking objects during and after occlusion. Other useful additions to this tracking system include a heuristic model to process splitting and merging of tracked regions, as well as a basic ability to track the main body parts. Extensions to this system have been made to track multiple people "before, during and after occlusions" [6], and to determine whether people are carrying objects - that can be separately tracked [7]. A discussion of the combination of these works is in [8]. Further extensions to this line of research in background modeling include shadow suppression by exploiting separable RGB channels during segmentation [13]. Another approach that has shown success in modeling complex backgrounds is to make use of adaptive mixture models [19].

A number of techniques have been developed that rely on optical flow recovery. Vehicular traffic has been tracked in two dimensions by using optical flow and predictions of displacement rate [10]. An extension to this approach has made use of three-dimensional models by considering parameters of the camera, vehicles, lanes, the three-dimensional scene, illumination and motion [11]. Unfortunately, in order for a vehicle to be tracked, it must be completely un-occluded for a period of time. In a rather different approach, optical flow is computed between a video mosaic reference image and a current image of interest to track multiple objects on a moving background as acquired from an airborne platform [12]. Optical flow-based techniques also have been applied to the tracking and interpretation of moving people. For example, individuals have been distinguished by their gait on the basis of extracting dense optical flow, followed by conversion to scale independent scalar features via moment weighting. Thereafter, the spatial distribution and analysis of periodic structure results from consideration of the time series of scalar descriptors [20].

Other approaches to tracking and segmentation include a system for automatic tracking of abandoned objects [16] as well as carried objects [7,13]. Still other research has been particularly concerned with detection of people, e.g., through training of a support vector machine on multiscale, spatial-based wavelet features recovered from several adjacent video frames [21]. Alternatively, pedestrian detection has been accomplished through the use of AdaBoost [22] defined templates operating with spatial and temporal support across consecutive video frames [23]. Finally, various approaches have been applied to modeling and extracting individual motion components from complex imagery. The probability that a pixel has changed due to motion has been modeled in terms of the rate of change in video RGB values [15]. A motion tracking

and segmentation system has been developed that dynamically fits various motion segmentation models to input video data by constantly calculating which of the models has the least error for current data [14]. Also considered has been the use of statistical analysis of the spatiotemporal gradient tensor to cluster background motion in video into a set of flow fields that capture dominant patterns of motion [24].

Review of literature discussing tracking and segmentation of objects shows a wide range of approaches, including some that are implemented for very specific tasks. Although an essential component in surveillance systems, tracking and segmentation alone are not sufficient for real-world security systems. To complement these systems, behavior and event detection also are necessary. A brief review of behavior and event detection literature follows.

Behavior and event recognition systems can be categorized further into those that incorporate a state-based system of individual motion or interactions between objects [12,25–31], those that are based on statistical approaches [32–44] and other approaches that employ appearance models and various projection techniques (e.g., Eigenspace) [45–50].

State-based behavior and event recognition systems include those that describe behavior in terms of a system that detects a series of states of interactions between objects being tracked [29]. A language is created based on events such as 'moves close to,' 'moves away from,' 'standing,' 'running,' etc. The authors describe a few behaviors of interest; however, the paper does not clearly describe how each event is detected. The tracking system is described in somewhat more detail in a related paper [30]. An extension dealing with implementing behavior recognition across multiple cameras as well as detection of groups also has been described [25]. Similar behavior recognition is discussed in an approach that computes object properties such as height, width, speed, motion direction and the distance to a reference object as well as motion trajectories to be potentially used in a multi-state system to detect events [12]. Finally, a class of 'atomic' activities for human body part movement has been defined as movements that are structurally similar over the range of performers and can be mapped onto a finite temporal window (i.e., non-periodic) [31]. The system, which recovers information based on principal component analysis, assumes initial segmentation of the body into parts that are tracked via the use of parameterized optical flow.

Statistical approaches have been key to a variety of behavior and event recognition systems. Interaction states between tracked objects (similar to those considered elsewhere, e.g., [29]) have been detected using a Bayesian network [38]. Anomalies in behavior of people and traffic have been detected and predicted based on hidden Markov models (HMMs) of optical flow fields [33,34]. A system has been developed for classifying traffic by generating probability mass functions from tracking data that is classified into motion patterns through the use of binary trees [43]. Atypical movements have been recognized by applying probability density functions of motion trajectories to a neural network in order to obtain a model of expected traffic motion patterns [39]. A method of analyzing traffic behavior based on 'qualitative reasoning and statistical analysis of video input' has been proposed [37]. In that work, a system is described that defines dynamic 'equi-temporal' regions based on temporal paths and spatial regions; this temporal path and region data is used to detect events such as cars passing each other as well as statistical anomalies to previous training data. Finally, preliminary results on a system for detection of local patterns of behavior have been presented that build on previous basic work in spatiotemporal analysis [51] as well as behavior recognition [45]. This system attempts to detect motion-based behaviors within a fixed region of the screen [35]. Two approaches were tested, one that is histogram-based and another based on HMMs; their histogram-based approach resulted in a very low detection rate while the HMM-based approach did recognize most of the events presented with a relatively low number of misclassifications. It appears that there was an assumption that each video sequence contained a known event, and that no control data was used in evaluation.

A variety of additional methods have been documented for behavior and event recognition, including the following. A system has been developed to provide information from traffic scenes such as stalled vehicles, vehicle counts and some lane changes; this was based on an estimated shape model of cars used for segmentation that was then tracked [46]. A system has been developed for classifying and counting vehicles as well as estimating their speed based on appearance models of the vehicles [50]. Neural networks have been used to model spatiotemporal behavior patterns of objects to predict future behavior [48]. A universal eigenspace, made up of images "of every possible kind of human behavior" to detect behaviors (really body positions), has been developed [47]; this work leverages previous work on building eigenspace recognition models from examples [52]. Human movement detection has been cast in terms of templates based on motion-energy images and motion-history images to detect certain local/individual human motions; unfortunately, it heavily relies on the assumptions that there is only one object segregated as moving, there is no occlusion and that the background is relatively static [45]. Finally, some approaches are distinguished by making very limited use of a priori knowledge of the imaged scene and event. For example, a distance measure has been proposed for comparing scenes by looking at temporal textures at multiple levels of a Gaussian pyramid that is blurred and subsampled in time only [49]. The authors note that while it is advantageous to do this type of event comparison/detection, the accuracy is significantly lower than event detection with a priori knowledge.

Similar to the tracking literature, the event recognition literature provides a wide variety of approaches for a broad range of applications. While some papers discuss the detection of a few very specific events with predefined context information, others are very general and, in theory, are capable of comparing events without a priori knowledge (yet yielded a lower accuracy rate). Behavior and event detection systems discussed here can be classified as those that search for statistical anomalies, those that are trained to classify patterns of motion, and those looking for specific predefined threats. The current work falls into the latter category.

Overall, research on motion analysis for surveillance systems appears to lack work on detection of specific events of interest (e.g., potential 'threats') incorporating global motion information. Systems that track individual objects or interactions discard contextual information that can be used to identify relevant events. In addition, research focusing on the detection of specific events uses tracking and/or segmentation, introducing an unnecessary layer of complexity, potentially resulting in errors.

In light of previous research, contributions of the current effort are as follows. (i) Direction maps, based on spatiotemporal distributions of local dominant directions, are proposed for capturing motion patterns. This representation makes it possible to capture both local and global patterns of interest without the need for explicit segmentation or tracking. Methods for (ii) recovering direction maps from video, (iii) constructing templates for target patterns of interest and (iv) comparing templates to video have been algorithmically defined, implemented and tested. (v) In empirical evaluation, involving over 6300 frames from seven video clips that depicted 141 events of interest as well as control events, recognition was observed at approximately 91% hits with 8% false positives. A description of an earlier version of this work appears in [53].

1.3. Paper outline

This paper consists of four main sections. Section 1 motivates the research and briefly outlines previous related work. Section 2 provides a technical explanation of how the developed system works. Section 3 details the experimental evaluation of the system. Section 4 provides a brief summary of and conclusions drawn from the work. Finally, a pair of appendices provide additional details regarding motion analysis and template examples.

2. Technical approach

2.1. Overview

The *first central hypothesis* of this work is that interesting patterns of motion (e.g., with respect to video surveillance applications) can be defined over vector fields that capture image displacement as a function of time. Such vector fields capture not only local motion (through the locally defined vectors), but also global motion patterns (through the spatiotemporal distribution of vectors) that serve to provide context. In this regard, we define global motion as motion in the two-dimensional image plane that captures the displacement of large-scale patches of interest across regions of a spatial grid as a function of time. The displacement can be characterized by vector fields that represent interactions between multiple motion paths or between motion paths and the scene structure.

Although useful information can be obtained from a continuously defined field of velocities, small deviations from the ideal due to low signal-to-noise ratio (e.g., arising from the interaction of small magnitude target motions with distractor motions, estimation errors, sensor noise, etc.) can inappropriately penalize comparisons between newly acquired data and predefined patterns of interest. In response to these observations, the *second central hypothesis* of this work is that usable, compact representations of motion patterns can result from coarse quantization of the video, in space-time, into discrete regions. Within each region, local dominant direction of motion is calculated and stored. The resulting representation, recovered from the projection of three-dimensional scene motion onto the image plane, is referred to as a *Direction Map* (an example is shown in Fig. 1). This leads to the *third central hypothesis* which is that *Template Direction Maps* representing events of interest can be created and matched with direction maps recovered from video data (Fig. 1 is actually a Template Direction Map).

In the remainder of this section, details of the approach are described: First, recovery of directional energy is presented as a way to estimate local dominant direction of motion. Second, direction maps are formally defined in terms of locally dominant directions, including methods for recovery from video, hand construction of target templates and comparison of videos containing events of interest to predefined templates. Finally, the main ideas are recapitulated from a system point of view.

2.2. Directional energies and image motion

Fundamental to the proposed calculation of local dominant direction is the notion that motion, as visible in the image domain (i.e., on a monitor), can be viewed as orientation in visual space–time (x–y–t space) [54,51,55]. For example, if each two-dimensional image is treated as a sheet that is stacked up into a three-dimensional spatiotemporal cube, a slice of this cube orthogonal to the x–y plane (for example, an x–t plane, with t downward), would show leftward motion as a diagonal line starting at the top right and moving towards the bottom left (see Fig. 2). The orientation of the line in this x–t slice would correspond to the image velocity. Consideration of motion as orientation in space-time is the basis of the dominant direction detection used in the current work. For the duration of this work, reference to motion, and direction of motion, shall refer to motion, as visible in the image domain.

Given the concern with directions of motion, spatiotemporal energy is recovered by convolving oriented, three-dimensional second derivative Gaussian filters, $G_2^{(\alpha,\beta,\gamma)}$, and their Hilbert transforms, $H_2^{(\alpha,\beta,\gamma)}$, with the video [56,57], followed by rectification via pointwise squaring, summation and square root (to convert the output of oriented filtering to an energy representation where larger values are indicative of greater orientation strength). In particular, directional energy, $E_{(\alpha,\beta,\gamma)}(x,y,t)$, with respect to three-dimensional direction cosines, (α,β,γ), is given as:

$$E_{(\alpha,\beta,\gamma)}(x,y,t)=\sqrt{(G_2^{(\alpha,\beta,\gamma)}(x,y,t)*I(x,y,t))^2+(H_2^{(\alpha,\beta,\gamma)}(x,y,t)*I(x,y,t))^2}. \tag{1}$$

For each filtered orientation, (α,β,γ), the output is further low-pass filtered with a three-dimensional Gaussian filter, $G(x,y,t;\sigma)$, with σ as the standard deviation of the Gaussian. This final filtering removes unwanted interference effects that appear as high frequency noise in the output. The low-pass filtered energy is denoted as $\tilde{E}_{(\alpha,\beta,\gamma)}(x,y,t)=G*E_{(\alpha,\beta,\gamma)}(x,y,t)$ in the following presentation.

Since second-order derivative filters are employed, filtering is performed along six different directions, as six directions spans the space of three-dimensional orientation for second-order Gaussian derivatives [56,57]. To equally space the basis directions in three-space, the direction vectors, (α,β,γ), are taken as the normals to the faces of a dodecahedron [58], with antipodal directions identified. The dodecahedron is oriented in three-space so that (α,β,γ), take the following values.

$$\begin{aligned}
\hat{\mathbf{n}}_1&=c(a,0,b)^{\mathrm{T}},\\
\hat{\mathbf{n}}_2&=c(-a,0,b)^{\mathrm{T}},\\
\hat{\mathbf{n}}_3&=c(b,a,0)^{\mathrm{T}},\\
\hat{\mathbf{n}}_4&=c(b,-a,b)^{\mathrm{T}},\\
\hat{\mathbf{n}}_5&=c(0,b,a)^{\mathrm{T}},\\
\hat{\mathbf{n}}_6&=c(0,b,-a)^{\mathrm{T}},
\end{aligned} \tag{2}$$

with

$$\begin{aligned}
a&=2,\\
b&=\left(1+\sqrt{5}\right),\\
c&=\left(10+2\sqrt{5}\right)^{-\frac{1}{2}}.
\end{aligned} \tag{3}$$

Given oriented energies computed along the specified directions, (2), it is a straightforward matter to recover the locally dominant three-dimensional orientation for a region of interest in visual space–time, (x,y,t) [59]. Further, for a temporal period of interest, the corresponding image velocity, $\mathbf{v}=(u,v)^{\mathrm{T}}$ can be recovered by projecting the orientation vector onto the image plane. While applicable methods are well known, details of a particular technique that is used in conjunction with the present work [59] are, for the sake of keeping the current presentation self-contained, outlined in Appendix A. In essence: The recovered energies along the spanning set of directions are used to construct the local covariance matrix; the eigenvalue/eigenvector structure of this matrix reveals the locally dominant orientation in three-dimensions; projection of the three-dimensional dominant direction into the image plane yields image velocity, $\mathbf{v}$.

2.3. Direction maps

2.3.1. Basic definitions

A direction map is defined as a three-dimensional array, $\mathscr{D}(x,y,t)$, with x, y, t corresponding to quantized horizontal, vertical

Fig. 2. Motion as orientation in space-time. Four frames of a video with a man walking towards the left (top). An expanded x–t slice from the video (bottom).

and temporal dimensions, respectively. Quantization of visual space–time adds robustness, as the representation, recovery and matching of templates for events of interest becomes less sensitive to precise spatiotemporal locations. This is significant, as in real-world surveillance scenarios, anticipated low signal-to-noise ratios (e.g., due to interaction of small target motions with distractors, small support targets, poor illumination, etc.) make it beneficial to abstract from precise estimates of location to a coarser representation. Each element (cell) of the direction map contains a direction of motion, $\hat{\mathbf{v}} = \mathbf{v}/\|\mathbf{v}\|$ or a symbol for *no motion*, (0,0), for the case where the locally recovered magnitude of motion, $\|\mathbf{v}\|$, does not exceed a motion detection threshold, τ. The estimate of image motion, $\mathbf{v}$, for each cell's direction is recovered as specified in Section 2.2, with the input to the estimation the set of oriented energies, $\tilde{E}_{(\alpha,\beta,\gamma)}(x,y,t)$, with each energy summed over the cell's spatiotemporal video support.

In general, cell quantization (Q_X, Q_Y and Q_T for horizontal, vertical and temporal dimensions, respectively) is scenario dependent. Spatial units of quantization are chosen (by the user) so that resulting cells are not significantly larger than the expected image size of monitored objects. For example, in traffic scenes, if the expected car size is approximately 20×20 pixels, Q_X and Q_Y should not be set much larger than 20×20 (i.e., the majority of a region should contain motion for a single object). The temporal unit of quantiza-

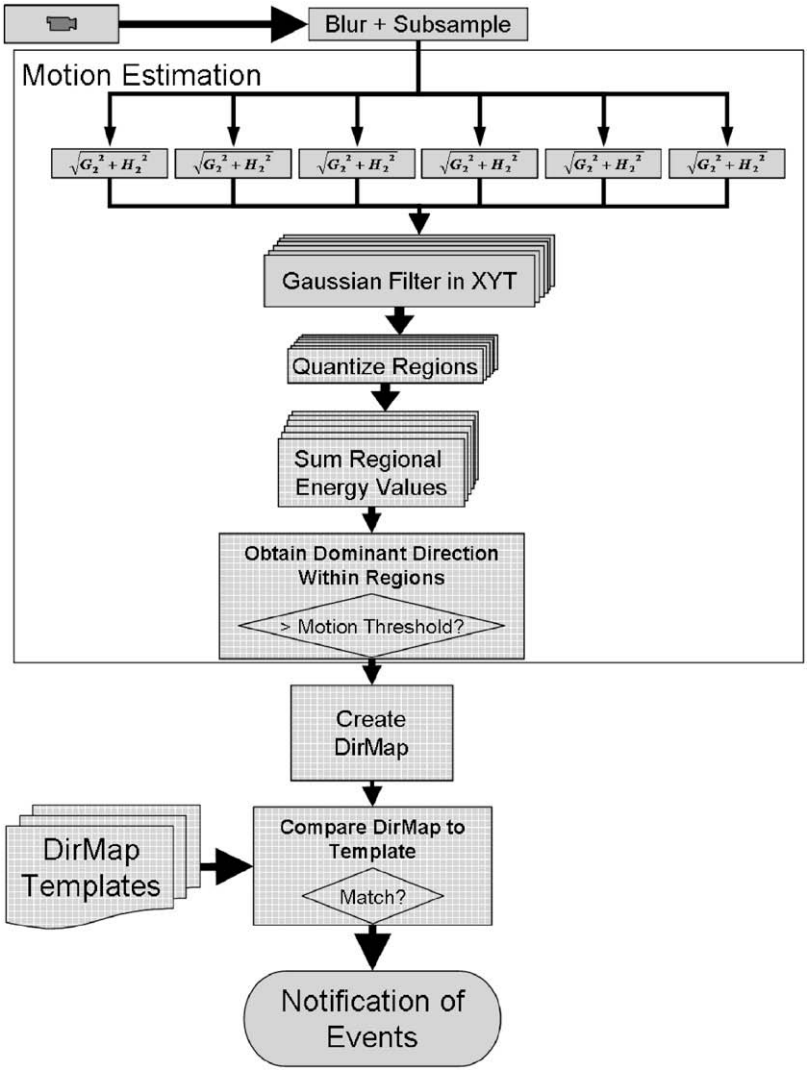

Fig. 3. An outline of the technical approach.

tion, Q_T, is determined by the maximum expected speed of objects of interest as well as the input frame rate. For example, if one expects a car to make a left turn in no less than 24 frames of video, and each individual component of the turn (i.e., the move into the intersection, or the beginning of the turn, etc.) would take approximately six frames, one would set Q_T to be 6. In order to take into account motion that may have begun or ended slightly outside the boundaries of a particular element in the direction map, the elements are defined to have slight overlap with their spatiotemporal neighbors. Specific spatiotemporal quantization rates used in evaluation of the described approach are specified in Section 3.

The importance of template representation in a manner that is robust to small template shifts relative to the image is widely recognized. In the present work, such robustness is achieved through the use of coarse quantization in space and time as well as the use of overlapping support of adjacent template cells. This approach is somewhat reminiscent of the use of histograms of local image intensity gradients accumulated about some central keypoint followed by interpolation across histogram bins used in SIFT representation [60]. In the present case, we found through initial empirical investigation that better results were had by directly computing a single dominant direction of motion over a quantized cell (as described in Appendix A) rather than via combination of multiple purely local estimates within a cell.

2.3.2. Direction map templates

Similar to the recovery of direction maps from input video, template direction maps can be specified to define target patterns of interest. In general, 'interesting' target patterns of motion, whether for surveillance or otherwise, are application specific (i.e., an activity considered threatening in one scenario might not be in another). Currently, templates are specified by a human operator through a user interface that supports manual specification of spatial and temporal pattern extent (i.e., spatial coverage and temporal duration) as well as quantization rates. Resulting spatiotemporal cells are then populated with relevant directions plus *no motion*, as consistent with the definition of direction maps. Also allowed is the specification of a *don't care* flag (represented by the null set, $\emptyset$) for spatiotemporal regions that are of no interest. Template direction maps are created via a GUI tool called 'DirMapEdit' (see Appendix C in [1]). DirMapEdit allows the creation of any of the template direction maps previously discussed and is also capable of viewing output direction maps created from real video. A match threshold, μ, in units of *directional distance* (discussed below), is then assigned for each template. An example direction map template, shown as a vector plot overlaid on a frame of an applicable video, is shown in Fig. 1. Additional examples of template direction maps are presented in Appendix B.

To help achieve a high hit rate, it is important, when creating templates, to capture only the portion of the motion pattern that is intrinsic to the event itself; however, keeping some preceding motion (i.e., moving forward before making a left turn) and succeeding motion may be required to prevent false positives. In some cases, the creation of an optimal template direction map can be a tedious and repetitive task; in others, such a template can be achieved in a matter of seconds. There are a few factors that determine how easy it will be to create an accurate direction map template, they include: uniqueness of the motion pattern to be detected, the degree of freedom of motion enjoyed by moving objects that are surveyed (i.e., whether they are constrained to lanes or narrow pathways), camera angle and placement, variations in size and speed of objects as well as the input frame rate. The sensitivity of the templates are determined by the size of moving objects relative to the spatial quantization (Q_X and Q_Y) and a user-defined threshold. Development of more automated methods for template generation (e.g., based on learning techniques) is a subject for future research. The present research is focused on establishing the basic efficacy of the proposed approach to representing and matching video surveillance events.

2.3.3. Comparing template and recovered direction maps

To detect a pattern of interest (as defined by a direction map template) in a video (as abstracted to a recovered direction map), it is necessary to define a measure of the distance (i.e., *directional distance*) between a template at a specific spatiotemporal position (i.e., offset) with respect to a recovered direction map. Let $\mathscr{T}(x,y,t)$ and $\mathscr{R}(x,y,t)$ be template and recovered direction maps, respectively. Let u, v, w be offsets within $\mathscr{R}$ at which a comparison is performed. Distance between two corresponding cells in the template and recovered maps (x,y,t) and $(x+u, y+v, t+w)$ is defined to capture the following considerations. First, the distance between two directions is defined as their angular difference, with a maximum of 180, for opposite directions. Second, the distance between *no motion* and any motion is defined as 90 (half of the maximum dis-

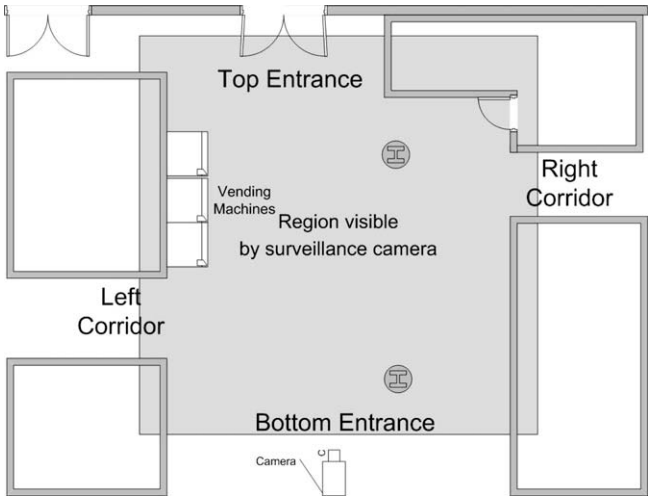

Fig. 5. Floor plan of scene in Video 1. Gray region corresponds to the approximate region visible by the surveillance camera.

Fig. 4. Three frames from Video 1.

tance); the distance between *no motion* and *no motion* is 0. Third, the distance between *don't care*, $\emptyset$, and anything is 0. Formally, let $\hat{\mathscr{R}} = \mathscr{R}(x+u, y+v, t+w)$ and $\hat{\mathscr{T}} = \mathscr{T}(x,y,t)$, then distance is given as

$$d(\hat{\mathscr{R}}, \hat{\mathscr{T}}) = \begin{cases} 0, & [\hat{\mathscr{T}} = \emptyset] \vee [\hat{\mathscr{R}} = \hat{\mathscr{T}} = (0,0)], \\ \| \arccos(\hat{\mathscr{R}} \cdot \hat{\mathscr{T}}) \|, & \text{otherwise}. \end{cases} \quad (4)$$

Notice that since *no motion* is represented as (0,0), the desired result of 90 is achieved when its distance is calculated with respect to any other direction. The distance for an entire template, $\mathscr{T}$, at a specific offset, (u,v,w), with respect to $\mathscr{R}$ is taken as the sum of the distances between individual corresponding cells, i.e.,

$$Distance(\mathscr{R}, \mathscr{T}) = \sum_{(x,y,t)\in\mathscr{T}} d(\hat{\mathscr{R}}, \hat{\mathscr{T}}). \quad (5)$$

A map containing distance measures for all possible offsets u,v and w can be created by calculating the distance given in Eq. (5) for all values of u, v and for existing values of w in time (in the case where the x, y dimensions of the template and the recovered direction map are the same, the variables u, v are ignored). A lower distance measure corresponds to greater similarity between the two direction maps. A match between $\mathscr{T}$ and $\mathscr{R}$ at position (u,v,w) is defined whenever $Distance(\mathscr{R}, \mathscr{T})$ is less than the user-defined match threshold, μ. Thresholds are scaled on a template-by-template basis to account for differences in spatiotemporal support.

Additional flexibility for matching events is obtained by allowing for templates to stretch in space and time, corresponding to variations in size and speed of interesting events. For spatial size invariance, the template direction map is scaled in the x and y dimensions by an additional parameter, κ; for speed invariance, the template direction map is scaled in the t dimension by the parameter λ. Formally, a multi-dimensional map containing distance measures for the offsets u, v, w and scales κ, λ can be created by calculating the following equation for all possible offsets and scales:

$$Distance(\mathscr{R}, \mathscr{T}) = \sum_{(x,y,t)\in\mathscr{T}} d(\widetilde{\mathscr{R}}, \hat{\mathscr{T}}), \quad (6)$$

Fig. 6. Three frames from Video 2.

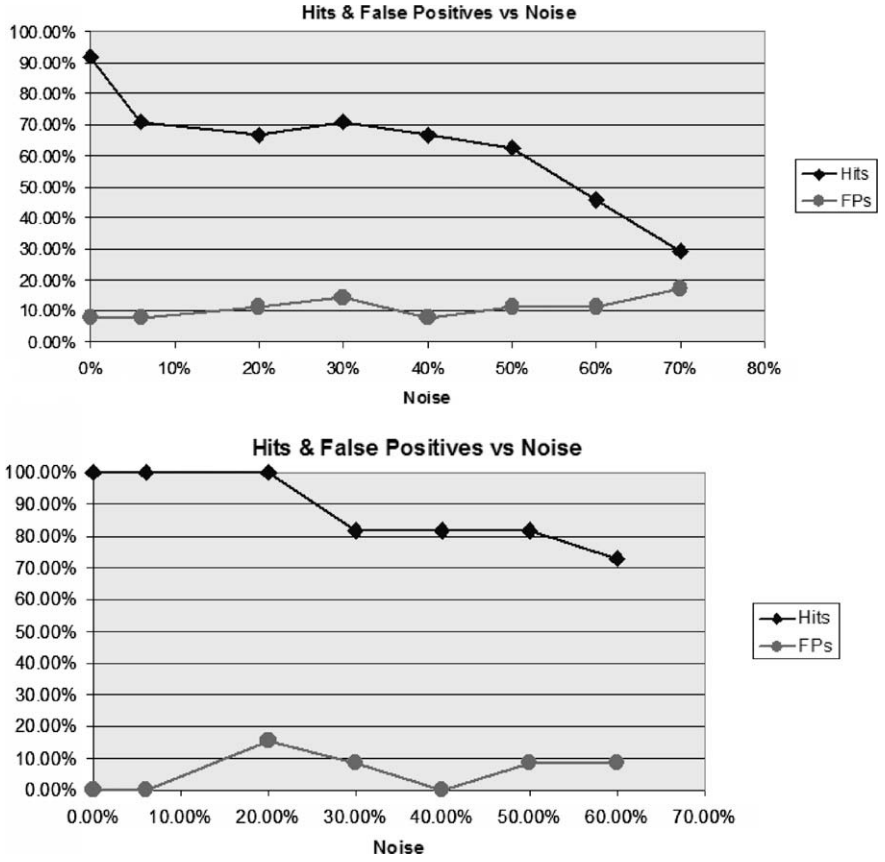

Fig. 7. Hits and false positives varying levels of noise. Video 1 (top), Video 2 (bottom).

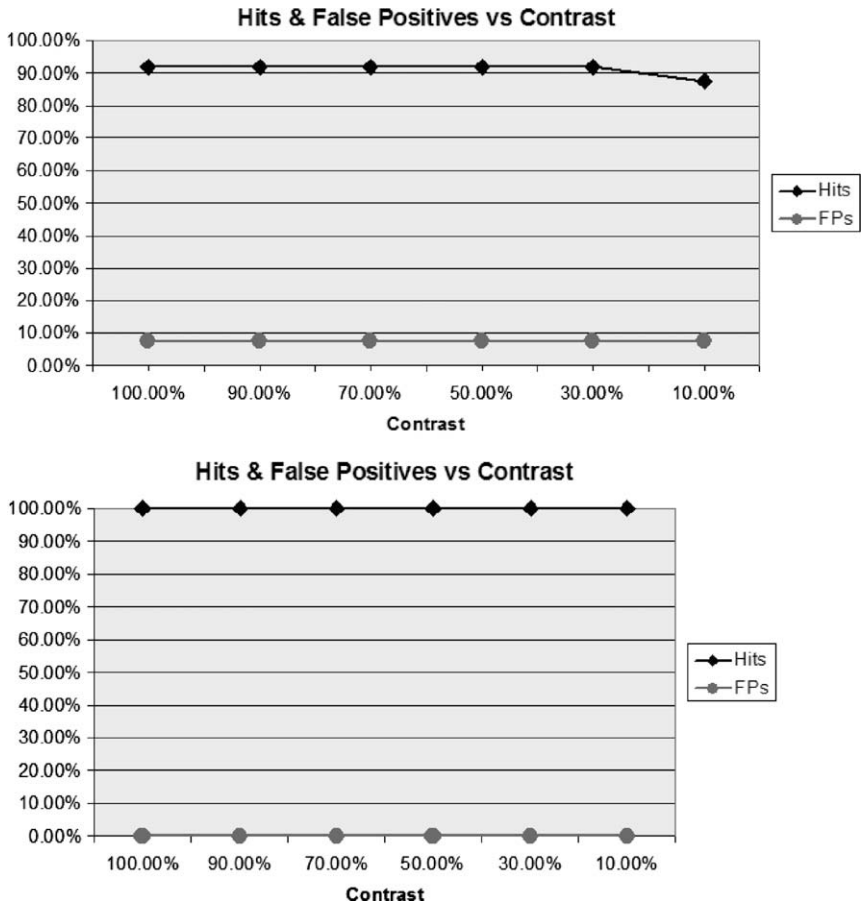

Fig. 8. Hits and false positives varying levels of contrast. Video 1 (top), Video 2 (bottom).

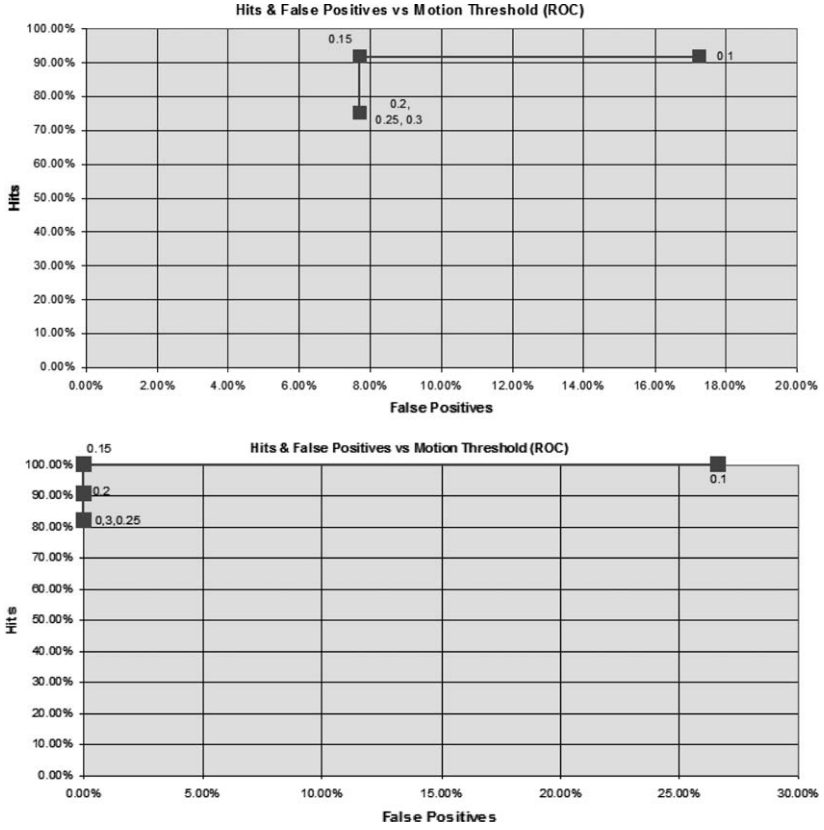

Fig. 9. Hits vs false positives varying motion thresholds (ROC) — Video 1 (top), Video 2 (bottom).

where $\widetilde{\mathscr{R}} = \mathscr{R}(\kappa x + u, \kappa y + v, \lambda t + w)$.

Correspondingly, locations u, v, w and scales κ, λ, where $Distance(\mathscr{R}, \mathscr{T})$ is below the match threshold, μ, are taken to be indicative of patterns of interest.

2.4. Recapitulation

A summary of the processing described in this section is provided by Fig. 3. Input video data is first preprocessed via Gaussian blur and subsampling to reduce the sheer quantity of data to be considered. It is then filtered with three-dimensional Gaussian, second derivative filters and their Hilbert transforms along a basis set of directions, (2). The resulting filter outputs are combined in quadrature to yield a set of directional energies, (1). This output is then passed through a Gaussian filter to remove unwanted high frequency noise. Direction of image motion (or *no motion*) is then calculated for each region of the spatiotemporal data (A.3); the result is a Direction Map. Template Direction Maps (as discussed in 2.3.2) are then compared against the recovered direction map using the algorithm in Section 2.3.3. If a match is found, presence of an 'interesting' event is noted.

Due to the potentially non-trivial length of a video, special consideration has been given to implementing processing so that operation can proceed with bounded memory and time. This is achieved by buffering input video with a sliding window. The buffer can constantly accept and process new data as it becomes available and not worry about memory use continually increasing. Temporal quantization, Q_T, the filter kernel width and the temporal subsample factor are taken into consideration when determining the buffer size. Significantly, all of the actual image processing involved in direction map recovery from video and comparison to stored event templates involves only straightforward, local operations (e.g., separable filtering, linear algebraic calculations and pointwise non-linearities).

3. Empirical evaluation

3.1. Experimental design

Seven video image sequences have been captured from various scenarios corresponding to scenes that include motion patterns of interest as well as control (uninteresting) motion. These videos

Table 1
Results for Video 3 (left) and Video 4 (right).

Hits	18	Hits	7
Misses	3	Misses	1
False positives	4	False positives	3

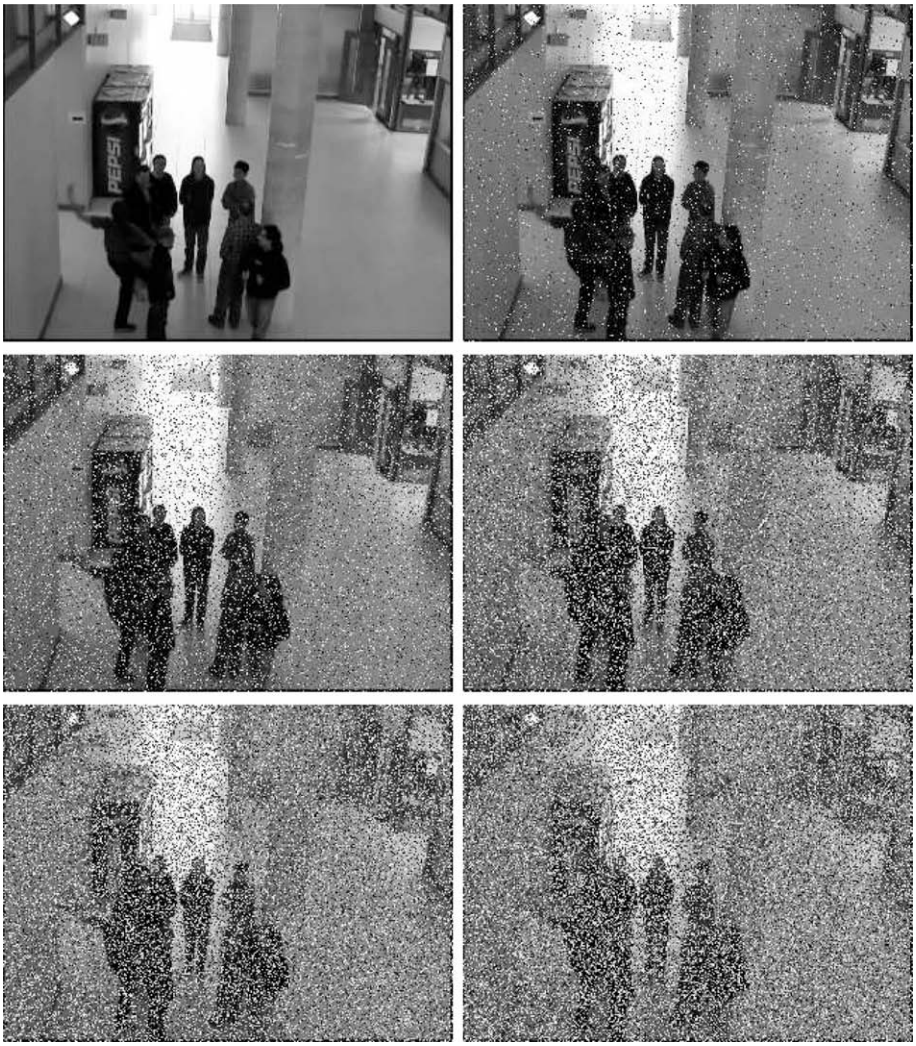

Fig. 10. One frame of Video 1 with 0%, 6%, 20%, 40%, 50% and 60% added shot noise (left to right, top to bottom).

Fig. 11. Three frames from Video 5.

Table 2
Results for Video 5.

Hits	11
Misses	0
False positives	0

have been digitized at a spatial sampling of 320×240 (and 368×240 in the case of Video 2) at 8 bits-per-pixel grayscale. For each video, direction map templates were manually created for each event of interest, as described in Section 2.3.2. Spatiotemporal coordinates were recorded for each actual occurrence of an event of interest to serve as ground truth detection data. Each video is then sent through the system for matching with a current average unoptimized execution rate of 2 fps in Linux on a 3.6 GHz Xeon processor for videos with a resolution of 320×240.

For two videos, a detailed analysis is presented including the detection accuracy with varying levels of shot noise, contrast and motion threshold, τ, (defined in Section 2.3.1). Other variables (e.g., quantization, blur kernel width, multiple frame rates and overlap buffer size) could have been systematically manipulated to further evaluate the system performance; however, noise, contrast and motion threshold were considered the most significant. For the remaining videos, overall hit and false positive rates were recorded.

The decision to use shot noise as opposed to, e.g., additive Gaussian noise, is due to the fact that shot noise is more representative of noise anticipated in real-world surveillance video. (As surveillance cameras are either wireless or have lengthy cables, they are particularly susceptible to this kind of noise from spurious electromagnetic pulses due to motors and appliances near the transmission path [61]). The corruption algorithm used is as follows: For n% noise, in each frame of video, a random n% of the pixels have their values replaced by noise from a uniform distribution over all legal gray-level values. Contrast (considered here as the dynamic range between the maximum and minimum gray-level values) is manipulated by multiplying each pixel by the new contrast value.

In all examples, unless otherwise specified, a default motion threshold of $\tau = 0.15$ and quantizations of $Q_T = 6$ (frames) and $Q_X = Q_Y = 20$ (pixels) were used; all numerical values are with respect to the input video resolution. Almost all objects to be detected in our experiments had image dimensions of at least 25×25 pixels. These parameter values were selected based on preliminary experimentation.

3.2. Detailed examples

Two videos are analyzed here in detail; Video 1 was taken, by the first author, from the second floor of a building overlooking a hallway on the first floor; it contains 24 events of interest in approximately 550 frames, Fig. 4 displays three frames of this video and Fig. 5 shows a map of the area. There are four main entrances to this hallway, one at each side of the screen. Twelve templates were created corresponding to entrances and exits from

Table 3
Results for Video 6 (left) and Video 7 (right).

Hits	36	Hits	23
Misses	2	Misses	5
False positives	1	False positives	2

Fig. 12. Three frames from Video 6 (top) and Video 7 (bottom).

each side of the screen as well as an additional template corresponding to a group of people converging at one point (*convergent crowding*). A subset of some of the template direction maps can be found in Appendix B, Figs. 17–20.

Video 2, as shown in Fig. 6, was taken from a window of a building, overlooking a four-way intersection in Cambridge, MA, and was taken by Brand and Kettnaker (used in [34]). The video contains 11 events of interest in approximately 1520 frames, Fig. 6 displays three frames of this video. Templates were created for four motion patterns, corresponding to a right turn going eastbound (it is assumed that north is at the top of the video), a right turn going westbound, a right turn going northbound and a left turn going westbound (as depicted in Fig. 1).

Overview charts of the results for Videos 1 and 2 are provided in Figs. 7–9.

From the initial analysis of the unmodified Video 1, containing 24 events of interest and much control data (e.g., various pedestrian motions not corresponding to any constructed template), 22 events were correctly detected plus two false positives; this gives a 91.67% hit rate, and 7.69% false positive rate.

Analysis of Video 2, unmodified, with 11 events of interest gave a 100% hit rate and a 0% false positive rate. It is believed that the

Fig. 13. Subject enters from bottom at an angle. This example demonstrates the need for multiple templates in such scenarios where one event can occur with different motion patterns.

Fig. 14. Example false positive. Subject walks towards right corridor and then walks away as if she is leaving the corridor.

constraint on motion given by the traffic lanes, resulted in the high accuracy.

As noise was progressively added to Video 1, the number of hits saw an initial decrement at 6% noise, thereafter, the number of hits did not significantly decrease until shot noise surpassed 50% (significantly higher than typical real-world noise levels; see, e.g., Fig. 10). Further, the number of false positives did not significantly increase. See Fig. 7 for the hits & false positives vs noise chart.

Addition of noise to Video 2 had no effect on the number of hits until 30% noise was reached, while the number of false positives saw an increment at 20% noise, which subsequently reduced and leveled off at higher noise levels (see Fig. 7).

Contrast was systematically reduced to 10% in both videos and detection results show that there was almost no consequence of contrast reduction in either video (see Fig. 8).

Fig. 15. Example false positive. Subjects walk into scene with intense background lighting casting shadows appearing as an *Exit at Bottom* event.

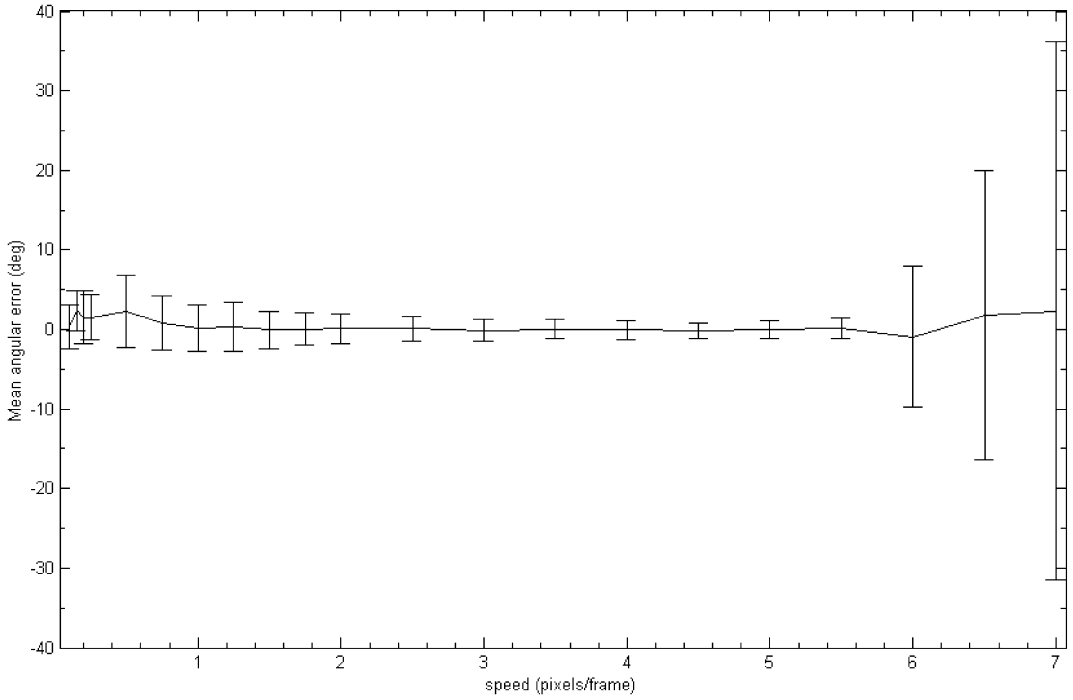

Fig. 16. Error in estimated image motion direction as a function of speed. The abscissa shows speed from 0 to 7 pixels/frame, with explicit sampling at 0.1, 0.15, 0.2, 0.25, 0.5, 1.0 and half-unit sampling thereafter. The ordinate shows mean error of recovered direction in degrees. Bars demark one standard deviation from the mean. The input test imagery corresponded to a translating white noise pattern, $(x, y, t) = (128, 128, 128)$, over which motion estimates were recovered in regions of support used in the main body of the paper.

For both videos, the motion threshold, τ, was varied from 0.1 to 0.3 (in increments of 0.05). It was found, as expected, that in both videos, a lower value of τ corresponded to a higher hit rate and a higher false positive rate; while a higher value of τ corresponded to a lower hit rate and lower false positive rate (see Fig. 9). In both videos a threshold of 0.15 yielded best performance from a receiver operating characteristic (ROC) perspective.

3.3. Additional videos

Videos 3 and 4 were taken from the same scene as Video 1 (see Fig. 4). Video 3 contained 21 events of interest and Video 4 contained eight. Results are shown in Table 1 and are comparable to the results for Video 1.

Video 5 depicts a relatively narrow hallway with a secure door on the left side; the events of interest contained entrances

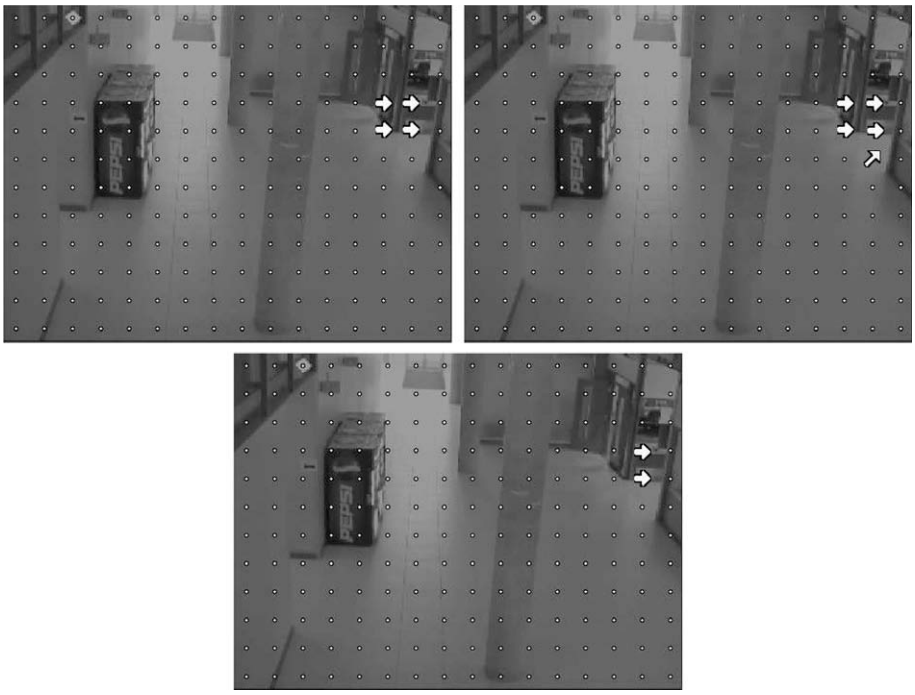

Fig. 17. A direction map template representing a person exiting at right corridor (for Videos 1, 3 and 4). In the first frame (left-to-right, top-to-bottom), motion is directed towards the right corridor; the next frame is a continuation with an additional arrow pointing up-right towards the corridor allowing for a subject to enter later in the template; finally, in the third frame, the entrance to the corridor is completed.

Fig. 18. A direction map template representing a person entering scene from the top (for Videos 1, 3 and 4). Each frame (left-to-right, top-to-bottom) progressively shows motion as people enter the scene from the top doors or corridor.

and exits through the secure door to and from the top and bottom of the screen as well as entrances and exits into the hallway from the top and the bottom. Three frames of this video are displayed in Fig. 11 and results are shown in Table 2. In this video, there were 11 possible events to detect; it should be noted that all subjects in this particular video have been classified as of interest, as a result, there was no control data (uninteresting motion) for this scene.

Videos 6 and 7 have been taken from a real traffic surveillance camera overlooking a pair of intersections (courtesy of the Department of Traffic Management in Bellevue, Washington). In Video 6, taken at around sunset, at the beginning it was light and approaches dusk by the end; in Video 7, taken at dusk, at the beginning it was somewhat dark and it became night by the end. Three frames of each video are shown in Fig. 12 and results are shown in Table 3. Video 6 contained 38 possible events of interest and Video 7 contained 28 events of interest; these events correspond to right and left turns at the intersections nearest to the camera. A direction map template for a car making a right turn going southbound can be found in Fig. 21.

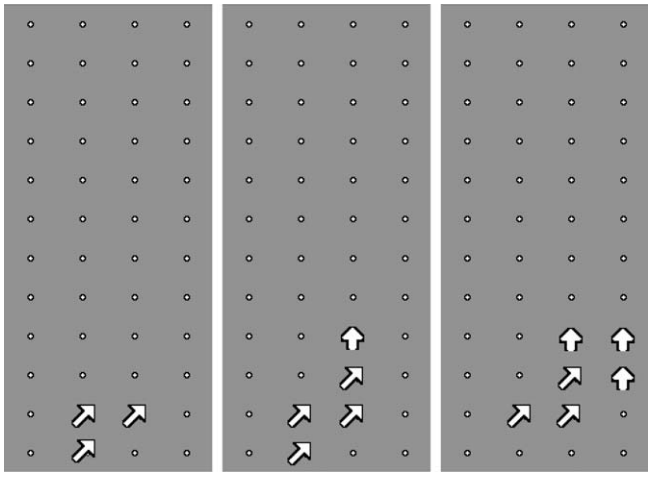

Fig. 19. A direction map template representing a person entering scene from the bottom at a rightward angle (for Videos 1, 3 and 4). This template is intended to be compared systematically across all imaged horizontal positions. The three frames (left-to-right) progressively show motion as a person might enter the scene from the bottom at a rightward angle.

3.4. Discussion

Over all experiments conducted, without artificial corruption, 141 possible events were tested for matches; there were 128 matches and 12 false positives, giving an overall hit rate of 90.78% and a false positive rate of 7.8%. It should be noted that in all but one of the experiments, there was a high amount of 'uninteresting' control data to potentially increase the rate of false positives; however, this remained low. Further, artificial manipulation of noise and contrast showed that the system is interestingly robust to such corruption. Significantly, those results were attained with a single motion threshold ($\tau = 0.15$); systematic manipulation of this variable in two detailed examples illustrated its effect on performance.

In some cases, two templates were needed to correctly detect an event of interest, this was motivated by the difficulty in classifying certain motions by a single global motion pattern. For example, a possible *Enter from Bottom* template in Videos 1, 3 and 4 could be defined as motion that proceeds upwards into the visible scene; this would not account for those entering from below at a diagonal. See Fig. 13 for example imagery, and Fig. 19 for an example of one of the two *Enter from Bottom* templates used. Two potential solutions to this problem are: (i) create multiple templates for such scenarios (as chosen for Videos 1, 3 and 4), (ii) constrain the environment with ribbon fences (e.g., as used for lineups at a bank) to restrict such 'unexpected' motion (note that accuracy was significantly high-

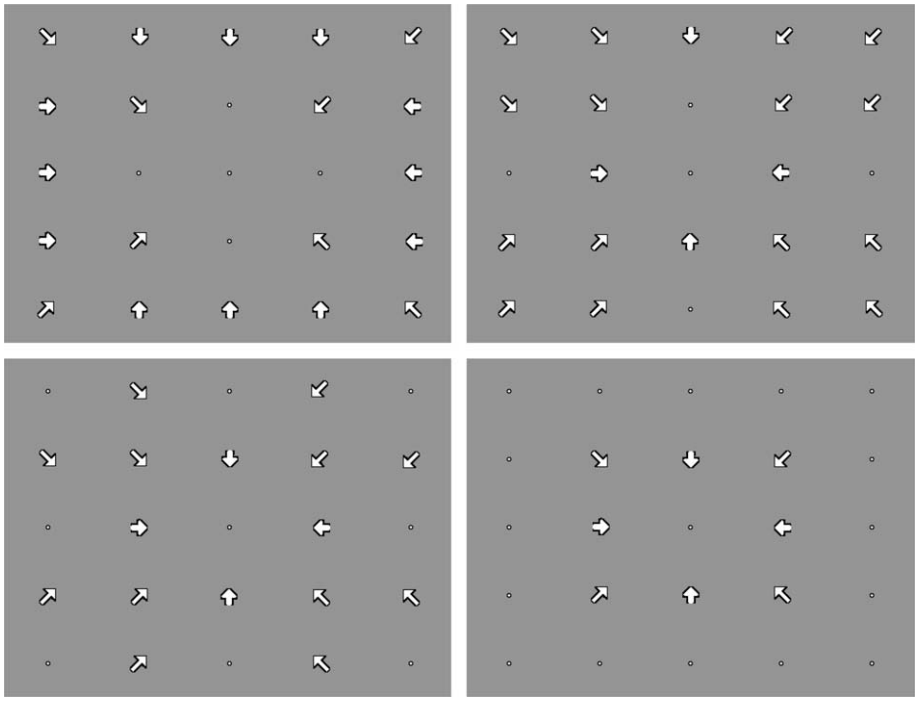

Fig. 20. A direction map template representing crowding (for Videos 1, 3 and 4). This template is intended to be compared systematically across all imaged horizontal and vertical positions as well as at multiple scales. The four frames (left-to-right, top-to-bottom) progressively show motion as it converges at a point in the scene.

Fig. 21. A direction map template (for Videos 6 and 7) Representing a car making a right turn going southbound. The four frames (left-to-right, top-to-bottom) progressively show motion as a car enters the vertical road (from the west), and turns right, moving southbound.

er in videos where moving objects were more constrained, i.e., Videos 2, 5, 6 and 7). Creating a single template with a broader scope and a higher match threshold is generally not recommended as this will result in a higher number of false positives in most scenarios.

In Videos 6 and 7, it is clear that since the camera was placed on an angle monitoring two separate intersections, cars further to the north appeared significantly smaller than those closer to the camera resulting in a lower hit rate.

Consideration of false positives detected suggests that the majority of false positives are due to motions that appear to move according to the template, but are generally unexpected for the scene. Examples include a subject walking near the right corridor (in Video 1) before turning around as if she is walking out of it (see Fig. 14), or inordinately long shadows of people walking against strong backlighting that appear as if the subjects are walking out of the region (see Fig. 15). Other false positives were caused by shadows or car headlights. Suggestions to reduce the number of false positives include (i) adjustment to the camera placement and angle to decrease the imaging of superfluous motion, (ii) environmental constraints (e.g., ribbon fences), (iii) additional lighting to attempt to minimize the contrast of shadows and the projections of headlights or (iv) modification to templates to include *no motion* regions following the expected motion.

Spatial quantization of the direction maps are believed to be the primary cause for the robustness to unusually high amounts of noise. Additional contributing factors to the stability to noise include, for example, in Video 1, the relatively small region size ($Q_X \times Q_Y$), relative to the size of people in the scene, as well as, for example, the scene structure in Video 2, where cars have a tendency to stay within their lanes. The high degree of robustness to contrast (e.g., synthetic contrast manipulation) and lighting variation (e.g., daytime to nighttime in Videos 6 and 7) is due primarily to the method for computing direction of motion. In particular, reliance on the relative magnitudes of the eigenvalues (and their eigenvectors) to compute local motion is highly robust to contrast variation that does not corrupt local orientation structure in the spatiotemporal domain.

On the whole, large variations in contrast as well as lighting (i.e., day to night) proved to be of little significance to the system as the method for calculating dominant direction had taken account of these factors. Further, the system showed a resilience to high levels of noise. It also has been found that a motion threshold, τ, of approximately 0.15, leads to a uniformly strong performance from an ROC perspective [62], regardless of tested scene structure or lighting conditions. Results of all the experiments are promising and suggest that a direction map based system can be used to detect global motion events of interest with a high degree of robustness.

4. Summary and conclusions

A novel approach to the simultaneous detection and classification of motion patterns, as depicted in video, has been presented. Key to the approach are direction maps, which capture the spatiotemporal distribution of local dominant directions of motion across a video. The recovery of direction maps from input video involves oriented bandpass spatiotemporal filtering and coarse quantization of space and time. The resulting direction maps emphasize local and global motion information across both space and time without the need for explicit segmentation or tracking. Templates, based on this direction map representation, can also be manually defined by the user to capture application-specific patterns of interest. These templates can then be compared to direction maps recovered from processed video using a simple distance calculation, which is used to quantify the similarity of motion patterns and thereby detect target motion patterns in a video. The approach has been evaluated in application to the detection of user-defined patterns of interest in surveillance and traffic videos. Seven videos, encompassing a wide range of variability (scene structure, indoor/outdoor settings, illumination, targets of interest, noise and contrast), have been considered. The results suggest

the applicability of the approach to real-world scenarios of interest.

Acknowledgments

This work was funded in part by an Ontario Graduate Scholarship to J. Gryn, an IRIS grant to J. Tsotsos and an NSERC grant to R. Wildes. J. Tsotsos holds the Canada Research Chair in Computational Vision. K. Derpanis provided valuable comments on the technical approach.

Appendix A. Recovery of image velocity from oriented energy measurements

Given a set of oriented energy measurements at a spanning set of directions for a given dimension, it is possible to recover an estimate of the locally dominant orientation. For the case of three-dimensional space–time imagery (x–y–t) it is further possible to interpret recovered orientation as image velocity. Detailed derivation and discussion of the method is available [59]; here, for the sake of making the present paper self-contained, the approach is described briefly. Other examples of research that has exploited spatiotemporal filtering to recover image velocity include a non-linear regression technique operating over three-dimensional Gabor filter outputs across a wider range of orientations and scales [63] and an approach that concentrates on zero-crossing analysis of three-dimensional Laplacian filtering with an emphasis on depth recovery [64]; additional examples can be found in reviews of image motion analysis (see, e.g., [2]).

Let the energy measurements arising from filtering along direction $\hat{\mathbf{n}}_k$, as specified in the spanning set (2), be q_k. For a spatiotemporal region of support over which three-dimensional dominant orientation is to be computed, construct a 3×3 matrix of the following form

$$\mathsf{T} = \sum_{k=1}^{6} q_k \mathsf{M}_k, \tag{A.1}$$

where

$$\mathsf{M}_k = \alpha \hat{\mathbf{n}}_k \hat{\mathbf{n}}_k^{\mathrm{T}} - \beta \mathsf{I}, \tag{A.2}$$

with the dyadic product $\hat{\mathbf{n}}_k \hat{\mathbf{n}}_k^{\mathrm{T}}$ establishing the frame implied by orientation $\hat{\mathbf{n}}_k$, I the 3×3 identity matrix and $\alpha = \frac{5}{4}$, $\beta = \frac{1}{4}$ numerical constants. In essence, T, captures the covariance structure of the support region.

The dominant orientation over the spatiotemporal region of support is specified by the eigenvector, $\hat{\mathbf{e}}_s$, corresponding to the smallest eigenvalue of T, provided the region contains adequate structure. To interpret $\hat{\mathbf{e}}_s$, in terms of image velocity, $\mathbf{v}$, the eigenvector must be projected onto the image plane: Let $\hat{\xi}_x$ and $\hat{\xi}_y$ be unit vectors defining the image plane, while $\hat{\mathbf{t}}$ is the unit vector along the temporal direction. Image velocity is then recovered as

$$\mathbf{v} = \left(e_x \hat{\xi}_x + e_y \hat{\xi}_y \right) \Big/ e_t, \tag{A.3}$$

where e_x, e_y and e_t are the projections of $\hat{\mathbf{e}}_s$ on $\hat{\xi}_x, \hat{\xi}_y$ and $\hat{\mathbf{t}}$, respectively, i.e.,

$$\begin{aligned} e_x &= \hat{\mathbf{e}}_s \cdot \hat{\xi}_x, \\ e_y &= \hat{\mathbf{e}}_s \cdot \hat{\xi}_y, \\ e_t &= \hat{\mathbf{e}}_s \cdot \hat{\mathbf{t}}. \end{aligned} \tag{A.4}$$

Empirical evaluation of the described algorithm for estimating three-dimensional orientation (and subsequently image velocity, for the case of spatiotemporal imagery) shows that it is able to provide accurate and precise estimates, even in the presence of challenging signal-to-noise ratios [65,59]. With respect to the particular implementation used in the present work, Fig. 16 shows the accuracy of estimated image motion direction as a function of speed for the case of a translating white noise pattern; it is seen that negligible bias and small variance is had across a range of speeds, 0.1–7 pixels/frame.

Appendix B. Direction map template examples

In the empirical evaluation of the described approach, a total of 35 direction map templates are employed. These can be coarsely categorized as (but not necessarily limited to) capturing turns of various directions, motion along particular directions, entrances, exits and converging/diverging motion for a pair or larger groups of individuals. Fig. 1 already introduced an example of a left-turn template. While space precludes an exhaustive presentation of all templates employed, additional example direction map templates are shown in Figs. 17–21. Notice that some templates are tied to specific spatial locations (e.g., a turn at a specific traffic intersection); these templates are depicted overlaid on the corresponding imagery. Other templates are allowed to be centered at arbitrary spatial positions (e.g., to detect a converging crowd anywhere in an image); those templates are depicted overlaid on a uniform background.

References

[1] J. Gryn, Automated surveillance using local dominant direction templates, Master's thesis, York University, May 2004.
[2] S. Beauchemin, J. Barron, The computation of optical-flow, ACM Computing Surveys 27 (3) (1995) 433–467.
[3] H. Nagel, Image sequence evaluation: 30 years and still going strong, in: Proceedings of the International Conference on Pattern Recognition, vol. I, 2000, pp. 149–158.
[4] A. Baumberg, Learning deformable models for tracking human motion, Ph.D. thesis, University of Leeds, 1995.
[5] I. Haritaoglu, D. Harwood, L. Davis, W4: Who? When? Where? What? A real time system for detecting and tracking people, in: Proceedings of the International Conference on Automatic Face and Gesture Recognition, 1998, pp. 222–227.
[6] I. Haritaoglu, D. Harwood, L. Davis, Hydra: Multiple people detection and tracking using silhouettes, in: Proceedings of the International Conference on Image Analysis and Processing, 1999, pp. 280–285.
[7] I. Haritaoglu, D. Harwood, L. Davis, Backpack: detection of people carrying objects using silhouettes, in: Proceedings of the IEEE International Conference on Computer Vision, vol. 1, 1999, pp. 102–107.
[8] I. Haritaoglu, D. Harwood, L. Davis, W4: Real-time surveillance of people and their activities, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 809–830.
[9] N. Siebel, S. Maybank, Real-time tracking of pedestrians and vehicles, in: Proceedings of the IEEE Workshop on Performance Evaluation of Tracking and Surveillance, 2001.
[10] M. Haag, H. Nagel, Tracking of complex driving maneuvers in traffic image sequences, Image and Vision Computing 16 (8) (1998) 517–527.
[11] M. Haag, H. Nagel, Combination of edge element and optical flow estimates for 3D-model-based vehicle tracking in traffic image sequences, International Journal of Computer Vision 35 (3) (1999) 295–319.
[12] G. Medioni, I. Cohen, F. Bremond, S. Hongeng, R. Nevatia, Event detection and analysis from video streams, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (8) (2001) 873–889.
[13] L. Davis, Real time computer surveillance for crime detection, Tech. Rep. 192734, US Department of Justice, February 2002.
[14] C. Lu, H. Liu, N. Ferrier, Multidimensional motion segmentation and identification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. II, 2000, pp. 629–636.
[15] J. Ng, S. Gong, Learning pixel-wise signal energy for understanding semantics, Image and Vision Computing 21 (12–13) (2003) 1183–1189.
[16] M. Spengler, B. Schiele, Automatic detection and tracking of abandoned objects, in: Proceedings of the Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2003.
[17] N. Hoose, L.G. Willumsen, Automatically extracting traffic data from videotape using CLIP4 parallel image processor, Pattern Recognition Letters 6 (3) (1987) 199–213.
[18] I. Haritaoglu, D. Harwood, L. Davis, A fast background scene modeling and maintenance for outdoor surveillance, in: Proceedings of the International Conference on Pattern Recognition, vol. 4, 2000, pp. 179–183.

[19] C. Stauffer, E. Grimson, Adaptive background mixture models for real-time tracking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1999, pp. 246–252.
[20] J. Little, J. Boyd, Recognizing people by their gait: The shape of motion, Videre 1 (2) (1998) 2–32.
[21] C. Papageorgiou, T. Poggio, A pattern classification approach to dynamical object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 1999, pp. 1223–1228.
[22] Y. Freund, R. Schapire, A decision theoretic generalization of on-line learning and an application to boosting, in: Proceedings of the European Conference on Computational Learning Theory, 1995, pp. 23–37.
[23] P. Viola, M. Jones, D. Snow, Detecting pedestrians using patterns of motion and appearance, International Journal of Computer Vision 63 (2) (2005) 153–161.
[24] R. Pless, J. Wright, Analysis of persistent motion patterns using the 3D structure tensor, in: Proceedings of the IEEE Workshop on Visual Motion, 2005, pp. 14–19.
[25] F. Cupillard, F. Bremond, M. Thonnat, Group behavior recognition with multiple cameras, in: Proceedings of the IEEE Workshop on Applications of Computer Vision, 2002, pp. 177–183.
[26] A. Efros, A. Berg, G. Mori, J. Malik, Recognizing action at a distance, in: Proceedings of the IEEE International Conference on Computer Vision, 2003, pp. 726–733.
[27] R. Howarth, H. Buxton, Conceptual descriptions from monitoring and watching image sequences, Image and Vision Computing 18 (2) (2000) 105–135.
[28] R. Polana, R. Nelson, Recognizing activities, in: Proceedings of the International Conference on Pattern Recognition, vol. A, 1994, pp. 815–818.
[29] N. Rota, M. Thonnat, Video sequence interpretation for visual surveillance, in: Proceedings of the IEEE International Workshop on Visual Surveillance, 2000, pp. 59–68.
[30] N. Rota, R. Stahr, R. Thonnat, Tracking for visual surveillance in VSIS, in: Proceedings of the IEEE Workshop on Performance Evaluation of Tracking and Surveillance, 2000.
[31] Y. Yacoob, M. Black, Parameterized modeling and recognition of activities, Computer Vision and Image Understanding 73 (2) (1999) 232–247.
[32] J. Alon, S. Sclaroff, G. Kollios, V. Pavlovic, Discovering clusters in motion time-series data, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. I, 2003, pp. 375–381.
[33] V. Kettnaker, M. Brand, Minimum-entropy models of scene activity, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. I, 1999, pp. 281–286.
[34] M. Brand, V. Kettnaker, Discovery and segmentation of activities in video, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 844–851.
[35] O. Chomat, J. Martin, J. Crowley, A probabilistic sensor for the perception and the recognition of activities, in: Proceedings of the European Conference of Computer Vision, vol. I, 2000, pp. 487–503.
[36] R. Fablet, P. Bouthemy, Motion recognition using nonparametric image motion models estimated from temporal and multiscale co-occurrence statistics, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (12) (2003) 1619–1624.
[37] J. Fernyhough, A. Cohn, D. Hogg, Constructing qualitative event models automatically from video input, Image and Vision Computing 18 (2) (2000) 81–103.
[38] S. Hongeng, F. Bremond, R. Nevatia, Bayesian framework for video surveillance application, in: Proceedings of the International Conference on Pattern Recognition, vol. 1, 2000.
[39] N. Johnson, D. Hogg, Learning the distribution of object trajectories for event recognition, Image and Vision Computing 14 (8) (1996) 609–615.
[40] I. Laptev, T. Lindeberg, Velocity adaptation of spatio-temporal receptive fields for direct recognition of activities: An experimental study, Image and Vision Computing 22 (2) (2004) 105–116.
[41] B. North, A. Blake, M. Isard, J. Rittscher, Learning and classification of complex dynamics, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (9) (2000) 1016–1034.
[42] N. Peyrard, P. Bouthemy, Detection of meaningful events in videos based on a supervised classification approach, in: Proceedings of the IEEE International Conference on Image Processing, vol. III, 2003, pp. 621–624.
[43] C. Stauffer, W. Grimson, Learning patterns of activity using real-time tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 747–757.
[44] H. Zhong, J. Shi, M. Visontai, Detecting unusual activity in video, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. II, 2004, pp. 819–826.
[45] A. Bobick, J. Davis, The recognition of human movement using temporal templates, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (3) (2001) 257–267.
[46] D. Koller, J. Weber, T. Huang, J. Malik, G. Ogasawara, B. Rao, S. Russell, Towards robust automatic traffic scene analysis in real-time, in: Proceedings of the International Conference on Pattern Recognition, vol. A, 1994, pp. 126–131.
[47] M. Rahman, K. Nakamura, S. Ishikawa, Recognizing human behavior using universal eigenspace, in: Proceedings of the International Conference on Pattern Recognition, vol. 1, 2002, pp. 295–298.
[48] N. Sumpter, A. Bulpitt, Learning spatio-temporal patterns for predicting object behaviour, Image and Vision Computing 18 (9) (2000) 697–704.
[49] L. Zelnik-Manor, M. Irani, Event-based analysis of video, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. II, 2001, pp. 123–130.
[50] Z. Zhu, B. Yang, G. Xu, D. Shi, A real-time vision system for automatic traffic monitoring based on 2D spatio-temporal images, in: Proceedings of the IEEE Workshop on Applications of Computer Vision, 1996, pp. 162–167.
[51] E. Adelson, J. Bergen, Spatiotemporal energy models for the perception of motion, Journal of the Optical Society of America A 2 (2) (1985) 284–299.
[52] H. Murase, S. Nayar, Visual learning and recognition of 3-D objects from appearance, International Journal of Computer Vision 14 (1) (1995) 5–24.
[53] J. Gryn, R. Wildes, J. Tsotsos, Detecting motion patterns via direction maps with application to surveillance, in: Proceedings of the IEEE Workshop on Applications of Computer Vision, 2005, pp. 202–209.
[54] M. Fahle, T. Poggio, Visual hyperacuity: Spatiotemporal interpolation in human vision, Proceedings of Royal Society of London B 213 (1981) 451–477.
[55] A. Watson, A. Ahumada, Model of human visual-motion sensing, Journal of the Optical Society of America A 2 (2) (1985) 322–342.
[56] W. Freeman, E. Adelson, The design and use of steerable filters, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (9) (1991) 891–906.
[57] K. Derpanis, J. Gryn, Three-dimensional *n*th derivative of Gaussian separable steerable filters, in: Proceedings of the IEEE International Conference on Image Processing, vol. III, 2005, pp. 553–556.
[58] P. Pearce, S. Pearce, Polyhedra Primer, Van Nostrand, New York, 1978.
[59] G. Granlund, H. Knutsson, Signal Processing for Computer Vision, Kluwer Academic, Dordrecht, The Netherlands, 1995.
[60] D. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2) (2004) 91–110.
[61] L. Christopher, W. Mayweather, S. Perlman, A VLSI median filter for impulse noise elimination in composite or component TV signals, IEEE Transactions on Consumer Electronics 34 (1) (1988) 262–267.
[62] N. Macmillan, C. Creelman, Detection Theory: A User's Guide, sixth ed., Cambridge University Press, Cambridge, 1991.
[63] D. Heeger, Model for the extraction of image flow, Journal of the Optical Society of America A 2 (2) (1987) 1455–1471.
[64] B. Buxton, H. Buxton, Monocular depth perception from optical flow by space time signal processing, Proceedings of the Royal Society of London Series B 218 (1210) (1983) 27–47.
[65] L. Haglund, Adaptive multidimensional filtering, Ph.D. thesis, Linkoping University, 1992.