Machine Vision and Applications
DOI 10.1007/s00138-008-0176-5

SHORT PAPER

Pedestrian tracking by fusion of thermal-visible surveillance videos

Alex Leykin · Riad Hammoud

Received: 26 September 2007 / Accepted: 10 October 2008

Abstract In this paper we introduce a system to track pedestrians using a combined input from RGB and thermal cameras. Two major contributions are presented here. First is the novel probabilistic model of the scene background where each pixel is represented as a multi-modal distribution with the changing number of modalities for both color and thermal input. We demonstrate how to eliminate the influence of shadows with this type of fusion. Second, based on our background model we introduce a pedestrian tracker designed as a particle filter. We further develop a number of informed reversible transformations to sample the model probability space in order to maximize our model posterior probability. The novelty of our tracking approach also comes from a way we formulate observation likelihoods to account for 3D locations of the bodies with respect to the camera and occlusions by other tracked human bodies as well as static objects. The results of tracking on color and thermal sequences demonstrate that our algorithm is robust to illumination noise and performs well in the outdoor environments.

Keywords Visual tracking · Thermal cameras · Fusion of color and thermal imagery

A. Leykin (✉)
Computer Science Department, Indiana University,
Bloomington, IN, USA
e-mail: oleykin@indiana.edu

R. Hammoud
Delphi Corporation, Electronics and Safety World Headquarters,
Kokomo, IN, USA
e-mail: riad.hammoud@delphi.com

1 Introduction

The problem of automatic real-time pedestrian recognition has gained a lot of attention in the machine vision community and is identified as one of the key issues in numerous applications ranging from collision avoidance with pedestrians in the automotive world, through border surveillance, and situation awareness in autonomous vehicles and robotic systems [3,22] through human activity recognition [14,26]. Sensor fusion has become an increasingly important direction in computer vision and in particular human tracking systems in recent years. Human motion tracking based on the input from RGB camera already has been producing reliable results for the in-door scenes with the constant illumination and steady backgrounds. Scenes with significant background clutter due to illumination changes, however, still appear to be challenging to handle using inputs from a conventional CCD camera. In our work we propose a method of utilizing an additional source of information—a thermal camera/sensor which produces for each pixel a gray scale mapping of the infrared radiation at the corresponding location.
Related work. Substantial research has been accumulated in detection and tracking of people. The majority of the studies address tracking of isolated people in a well controlled environment, but increasingly there is more attention to tracking specifically in *crowded environments* [4,11,13–15,23,29]. It is worth noting that many works assume the luxury of multiple well-positioned cameras or stereo vision, which are to a certain extent not present in most establishments and/or do not have the desired overlapping fields of view. In contrast, cheap low-resolution digital monocular color cameras are becoming more and more readily available in stores, airports and other public places as well as the hardware for capturing compressed real-time streams provided by these cameras.

Recently, a flurry of contributions on pedestrian localization and tracking in visible and infrared videos have appeared in the literature [1,5,6,12,25,31]. In [32], the P-tile method is developed to detect human head first and then include human torso and legs by performing a local search. Nanda [25] builds a probabilistic shape hierarchy to achieve efficient detection at different scales. In [27], a particle swarm optimization algorithm is proposed for human detection in infrared imagery. Dai et al. [6] proposed a hybrid (shape and appearance) algorithm for pedestrian detection, in which shape cue is first used to eliminate non-pedestrian moving objects and appearance cue is then used to pin down the location of pedestrians. A generalized Expectation Maximization algorithm has been employed by the authors to decompose infrared images into background and foreground layers. These approaches rely on the assumption that the person region has a much hotter appearance than the background. Davis et al. [8] proposed to fuse thermal and color sensors in a fusion-based background-subtraction framework using contour saliency map in urban settings. Information including object locations and contours from both synchronized sensors are fused together to extract the object silhouette. A higher performance is reported by fusing both sensors over visible-only and thermal-only imagery. This method is however computationally expensive as it attempts to construct a complete object contour, which does not seem to be a requirement in various applications like surveillance or crash-avoidance system. In [31], support vector machine and Kalman filtering are adopted for detection and tracking, respectively. In [28], two pedestrian tracking approaches, pixel-periodicity and model-fitting, are proposed based on gait. The first employs computationally efficient periodicity measurements. Unlike other methods, it estimates a periodic motion frequency using two cascading hypothesis testing steps to filter out non-cyclic pixels so that it works well for both radial and lateral walking directions. The extraction of period is efficient and robust with respect to sensor noise and cluttered background. In the second method, they integrate shape and motion by converting the cyclic pattern into a binary sequence by maximal principal gait angle (MPGA) fitting.

We developed our generative tracking framework encouraged by recently found implementations for particle filtering. Random sampling was shown not only to successfully overcome singularities in articulated motion [9,24], but the particle filtering approach applied to human tracking has also demonstrated potential in resolving ambiguities while dealing with crowded environments [16,33]. Working under the Bayesian framework it has been shown that particle filters can efficiently infer both the number of objects and their parameters. Another advantage is that in dealing with distributions of mostly unknown nature, particle filters do not make Gaussianity assumptions, unlike Kalman filters [17,30].

Contribution. Despite these efforts, the challenges still remain both for the stationary and moving imaging systems. This is due to a number of key factors like lighting changes (shadow vs. sunny day, indoor/night vs. outdoor), cluttered backgrounds (trees, vehicles, animals), artificial appearances (clothing, portable objects), non-rigid kinematics of pedestrians, camera and object motions, depth and scale changes (child vs. adult), and low video resolution and image quality. This paper proposes a pedestrian detection and tracking approach that combines both thermal and visible information (see Fig. 1) and subsequently models the motion in the scene using a Bayesian framework. We define a set of jump-diffuse transitions for a particle filter operating within Bayesian formulation, such that, these transitions reflect the nature of the motion in the scene. This is an enhancement of blind multivariate optimization to incorporate prior information of the real world.

We built our tracking system to operate in a two-camera setup, assuming one of the cameras can sense in the thermal part of the spectrum and is calibrated to give the view matching that of the visible camera. Also we assume that the location of the floor plane in the frame is established, either by supervised or automated calibration techniques. The goal of our tracking system is twofold: first attempt to employ all available information to achieve the noise free blob-map and second, subsequently use the blob-map to perform reliable pedestrian tracking to minimize two types of tracking errors—falsely detected people and people missed by the system. Our system segments foreground regions out of each frame by using a dynamically adapting background model presented here (see Fig. 2). Because each foreground region may contain multiple people, we further hypothesize about the number of human bodies within each such region by using the head-candidate selection algorithm. The head is chosen as the most distinguishable and pronounced part of the human body, especially when observing the scene with a highly elevated monocular camera. As the next step, our system constructs a Bayesian inference model, based on the a priori knowledge of the human parameters and scene layout and geometry. Observations of the body appearances at each frame are a second driving force in our probabilistic scheme.

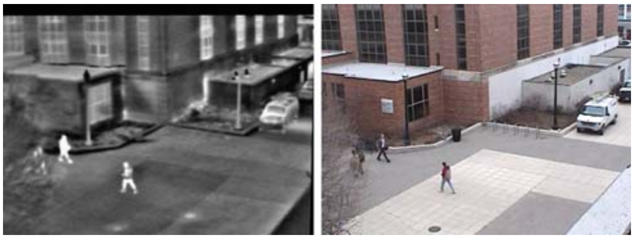

Fig. 1 *Left* Thermal image of the scene. *Right* Color image of the same scene

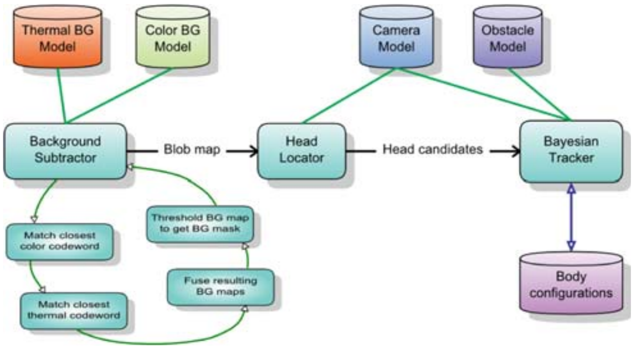

Fig. 2 Tracking system flowchart

2 Background model

Video sequences from the surveillance cameras are frequently compressed with MPEG-like algorithms, which normally create a periodic noise on the level of a single pixel. Moreover periodic changes in the illumination of the scene such as cloud cover or wind effects produce similar effects. We implemented a multi-modal adaptive background model based on the codebook approach. Variable length multi-modal distribution was previously used to represent background model for a single RGB input in [18]. Here we extend our previous model [20,21] by adding a probabilistic weight to color and thermal components of the background.

2.1 Acquiring multi-modal pixel representation

Each pixel in the image is modeled as two dynamically growing vectors of codewords, so-called codebooks as shown in Fig. 3.

For the RGB input a codeword is represented by: the average pixel RGB value and by the luminance range I_{low} and I_{hi} allowed for this particular codeword. If an incoming pixel p is within the luminance range and the dot product of p_{RGB} and RGB of the codeword is less than a predefined threshold it is considered to belong to the background.

For the thermal monochromatic input a codeword is represented by: intensity range T_{low} and T_{hi} occurring at the pixel location. Unlike for the color codewords the matching of the incoming pixel's approximate temperature p_T is done by comparing the ratios of p_T/T_{low} and p_T/T_{hi} to the empirically set thresholds. This way we can hard limit the percentage of temperature change allowed to happen at each location. By observing several thermal sequences we have established that changes in cloud cover or shadows produced by other moving objects do not typically cause the temperature change of more than 10%.

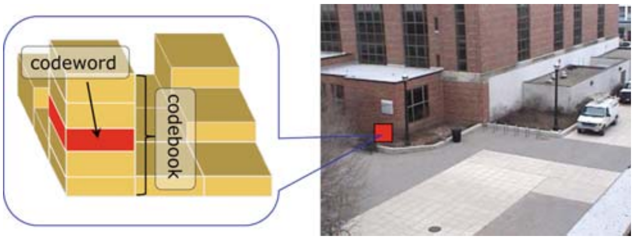

Fig. 3 Codebook creation

During the model acquisition stage the values are added to the background model at each new frame if there is no match found in the already existing vector. Otherwise the matching codeword is updated to account for the information from the new pixel. Empirically, we have established that there is seldom an overlap between the codewords. In the situation when this is the case, i.e. more than one match has been established for the new pixel, we merge the overlapping codewords. We assume that the background changes due to compression and illumination noise are of re-occurring nature. Therefore, at the end of training we clean up the values ("stale" codewords) that have not appeared for periods of time greater than some predefined percentage of frames in the learning stage as not belonging to the background. We keep in each codeword a so-called "maximum negative run-length (MNRL)" which is the longest interval during the period that the codeword has not occurred. One additional benefit of this modeling approach is that, given a significant learning period, it is not essential that the frames be free of moving foreground object. The background model can be learned on the fly and is helpful when tracking and model acquisition are done simultaneously.

As a further enhancement we eliminated the background learning stage as such to enable our system to operate dynamically. This was done by adding the *age* parameter to each codeword as the count of all the frames in which the codeword has appeared. Now, we can start background subtraction as soon as the majority of the codewords contain "old-enough" modalities by setting a threshold for *age* variable. Typically, around 100 frames in our test sequences (see Sect. 4) were enough for reliable detection of the foreground objects. This improvement also allows us to perform the removal of "stale" codewords periodically and not as a one-time event. To determine the "staleness" of a codeword we consider the ratio between its MNRL and it overall *age*. We have found that when employing "stale" pixel cleanup for heavily compressed sequences the length of the codebook required to encapsulate the background complexity within one pixel is usually under 20 codewords.

Additionally, we store the number of the last frame number f_{last} in which the codeword was activated (i.e. it matched a pixel). To make our model dynamic, we *remove the codewords* that have not appeared for long periods of time, based on their MNRL. Instances of such codewords are indicating that the interior has changed, due to possibly a stationary object placed or removed from the scene, thus causing our model to restructure dynamically.

Background modeling using a fixed acquisition stage is dependent on approximated environment temperature and thus has to start within the short time interval preceding the tracking. Our enhanced adaptive background acquisition method will sample every n-th frame and update the background model to reflect changing environment.

2.2 Probabilistic background segmentation

To perform background subtraction step for each color x_C or temperature x_T pixel the distance to the closest color v_C and thermal codeword v_T is computed correspondingly to get the RGB and IR background map.

$$\begin{aligned} p_C &= \frac{\langle x_C, v_C \rangle}{|x_C||v_C|} \\ p_T &= \min\left(1, \frac{|x_T - v_T|}{\sigma_T}\right) \end{aligned} \tag{1}$$

where σ_T is the normalizing constant for thermal mode. It can be thought of as the maximum allowed deviation for the temperature in each location. Parameter p_C has a geometric interpretation of a cosine between two corresponding color vectors and therefore scales nicely from 0 to 1. One has to take special care in choosing the parameter σ_t in order for p_T to scale properly with respect to p_C.

Knowing for each pixel the p_C and p_T we combine them into the aggregate probabilistic foreground map.

$$p_{\mathrm{BG}} = \frac{p_C + p_T}{2} \tag{2}$$

This fusion model allows to compensate for the instances where one modality is performing poorly (see Fig. 6). For example when there is a shadow from the cloud cover in RGB mode, or there is a halo created by the heated surface in the thermal mode.

2.3 Computing foreground mask

The probabilistic background map obtained in the previous step is further thresholded to the binary background mask $p_{\mathrm{BG}} > T$. The threshold T can be preset based on the desired sensitivity or chosen adaptively with the assumption of the known average percentage of foreground pixels in the scene. For instance, we found for medium crowded sequences shown in Fig. 7 initial value of $T = 0.5$ to be optimal. Furthermore, we sampled the percentage of foreground pixels every 15 frames. If this percentage fell below 0.5% we would increase the threshold by 0.05, if it went above 2% we would correspondingly decrease the threshold by 0.05.

The binary mask after background subtraction is filtered with morphological operators (2 pixel erosion followed by 2 pixel dilation) to remove standalone noise pixels and to bridge the small gaps that may exist in otherwise connected blobs. This results in an array of blobs created where each blob b is represented as an array of vertices $b_i, i = 1, \ldots, n$ in two-dimensional image space. The vertices describe the contour of b in which each adjacent pair of vertices b_j and b_i is connected by a straight line.

3 Tracking

We presume the projection matrix $P_{3\times4}$ of the camera and coordinates of head candidates h_{xy} within each blob b_i detected in the scene are known as described in detail in [19]. These are essential parts of our system that make it possible to convert from simple pixel motion in image plane to the tracking of the pedestrians in a 3D scene. To back-project from $2D$ to $3D$ coordinates we assume that the bodies move along the floor plane with the zero vertical coordinate (i.e. for each head candidate h_{xy}, there is a corresponding foot-candidate f_{xy} for which $Z(h_{xy}) = 0$).

3.1 Bayesian model: observations and states

We formulate the tracking problem as the maximization of posteriori probability of the Markov chain state. To implement Bayesian inference process efficiently we model our system as a Markov chain $M = \{x, z, x_0\}$ and employ a variant of Metropolis–Hastings particle filtering algorithm. The choice of this particular approach can be justified whenever the number of system parameters to be optimized is great and presents a large computational load for conventional optimization techniques (see [10]). The state of the system at each frame is an aggregate of the state of each body $x_t = \{B_1, \ldots, B_n\}$. Each body, in order, is parametrically characterized as $B_i = \{x, y, h, w, c\}$, where x, y are coordinates of the body on the floor map, h, w its width and height measured in centimeters and c is a 2D color histogram, represented as 32 × 32 bins in hue-saturation space. The body is modeled by the ellipsoid with the axes h and w. An additional implicit variable of the model state is the number of tracked bodies n.

3.2 Computing posterior probability

The goal of our tracking system is to find the candidate state x' (a set of bodies along with their parameters) which, given the last known state x, will best fit the current observation z. Therefore, at each frame we aim to maximize the posterior probability $P(x'|z, x)$, which, according to the rules of

Bayesian inference, and knowing that z and x are not independent, can be formulated by Eq. 3.

$$P(x'|z, x) = P(z|x', x) \cdot P(x'|x)/P(z|x) \tag{3}$$

From Eq. (3) we formulate our goal as finding the candidate state x' resulting in a maximum a-posteriori probability:

$$\begin{aligned} x' &= \text{argmax}_{x'}(P(z|x', x) \cdot P(x'|x)/P(z|x)) \\ &= \text{argmax}_{x'}(P(z|x', x) \cdot P(x'|x)) \end{aligned} \tag{4}$$

The right hand side of Eq. (4) is comprised of the observation likelihood and the state prior probability. The likelihoods are computed jointly for all bodies present in the scene as described below.

3.2.1 Priors

In creating a probabilistic model of a body we considered two types of prior probabilities.

The first type of priors imposes physical constraints on the body parameters. Human body width and height are weighted according to a normal truncated distributions as $P_{\text{height}} = N(h_\mu, h_{\sigma^2})$ and $P_{\text{width}} = N(w_\mu, w_{\sigma^2})$, with the corresponding means and variances reflecting the dimensions of a normal human body, estimated over a number of independent video tracking sequences. We truncated the distributions for both height and width and the value of $\pm 3\sigma$ to exclude physiologically impossible body configurations. This results in the body size prior:

$$P_{\text{size}} = P_{\text{height}} \cdot P_{\text{width}} \tag{5}$$

The second type of priors reflects the dependency between the candidate state at time t and the accepted state at time $t-1$. Firstly, the difference between body width and height w_t, h_t and w_{t-1}, h_{t-1} lowers the prior probability. As another factor, we use the distance between proposed body position $pos_t = (x_t, y_t)$ and $\hat{pos}_{t-1} = (\hat{x}_{t-1}, \hat{y}_{t-1})$—the prediction from the constant velocity Kalman filter. The state of Kalman filter consists of the location of the body on the floor and its velocity. Although tracking the head seems like a first reasonable solution, we have established empirically that the perceived human body height varies as a result of walking, thus the position of the feet on the floor was chosen as a more stable reference point. Furthermore, first order Kalman filter was chosen as a MAP estimator under linear velocity assumption, which, we have observed, holds in majority of indoor/outdoor pedestrian motion patterns.

$$P_{\text{temporal}} = P(w_t|w_{t-1})P(h_t|h_{t-1}) * P(pos_t|\hat{pos}_{t-1}) \tag{6}$$

When new body is created it does not have a correspondence, we use a normally distributed prior $N(d_0, \sigma)$, where d_0 is the location of the closest door (designated on the floor plan) and σ is chosen empirically to account for image noise. The same process is taking place when one of the existing bodies is being deleted. In the outdoor sequences the doors can be designated at the boundaries of the scene view.

$$P_{\text{temporal}} = N(d_0) \tag{7}$$

The resulting prior probability of the state x' given previous state x is computed in Eq. 8.

$$P(x'|x) = \prod_{\forall B \in x'} P_{\text{size}} * P_{\text{temporal}} \tag{8}$$

3.2.2 Likelihoods

The second component in forming proposal probability relates the observation to the model state. First, for each existing body model the color histogram c is formed by the process of weighted accumulation, with more recent realizations of c given more weight. We then compute Bhattacharyya distance Bh between proposed c'_t and corresponding c_{t-1} as part of the observation likelihood.

$$P_{\text{color}} = 1 - w_{\text{color}} \cdot Bh(c'_t, c_{t-1}), \tag{9}$$

where w_{color} is an importance weight of the color matching, experimentally set to 0.8 to accommodate for gradual changes in the color representation of the object.

To guide the tracking process by the background map at hand, we use two more components while computing model likelihood: the amount of blob pixels not matching any body pixels P^+ and the amount of body pixels not matching blob pixels P^-. Note that we use a Z-buffer Z for these as well as for computing the color histogram of the current observation in order to detect occlusions. In this buffer all the body pixels are marked according to their distance from the camera (i.e. 0 = background, 1 = furthermost body, 2 = next closest body, etc.), which we obtain during the calibration process. This way only visible pixels are considered when computing the likelihood (see Fig. 4). The Z-buffer is updated after each transition to reflect the new occlusion map.

In computing the likelihood as outlined above, there is one major shortcoming overlooked in previous works [16,33]. If the computation is done in terms of the amounts of image pixels it causes the bodies closer to the camera influence

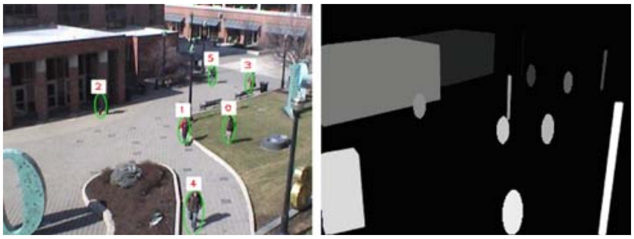

Fig. 4 *Left* Original frame with tracked pedestrians. *Right* Z-buffer (*lighter shades of gray* are closer to the camera)

the overall configuration much more, and the bodies further away are being mostly neglected. This becomes particularly evident when the camera covers a large area, where pedestrian image presentations can vary from under 20 pixels of overall area in the back of the scene to more than 200 in front. In addition, such neglect makes the system absolutely tied to the current scene configuration and not portable to a different camera model.

To avoid these shortcomings we have utilized a "distance weight plane" D which is the image of the same dimensions as the input frame and $D_{xy} = |P_{XYZ}, C_{XYZ}|$, where $||$—is the Euclidean distance, C_{XYZ}—camera world coordinates and P_{XYZ}—world coordinates of the hypothetical point in space located at a height $z = \frac{h_\mu}{2}$ and corresponding to the image coordinates (x, y). To prevent noise in the areas with large D_{xy} (i.e. areas near the horizon) from affecting the tracking process we imposed an upper limit $D_{\max} = 20$ m. The map produced in this manner is a rough assessment of the actual size to image size ratio (see Fig. 5).

To summarize, the implementation of **z-buffer** and **distance weight plane** allows to compute multiple-body configuration with one computationally efficient step. Let I be the set of all the blob pixels and O the set of all the pixels corresponding to the bodies currently modeled, then

$$\begin{aligned} P^+ &= \sum \frac{(I - O \bigcap Z_{(Z_{xy}>0)}) \cdot D}{|I|} \\ P^- &= \sum \frac{(O \bigcap Z_{(Z_{xy}>0)} - I) \cdot D}{|O|} \end{aligned} \tag{10}$$

where '$\bigcap$' is set intersection, '$\cdot$' is element-wise multiplication; '$-$' is set difference and '$||$' is set size (number of pixels).

The resulting joint observation likelihood is computed in Eq. 11 where w_+ and w_- are scalar weight to give more priority to either overcompensating blobs or overcompensating bodies, with $w_+ + w_- = 1$.

$$P(z|x') = \exp -(w_+ * P^+ + w_- * P^-) \cdot \prod_{\forall \text{bodies}} P_{\text{color}} \tag{11}$$

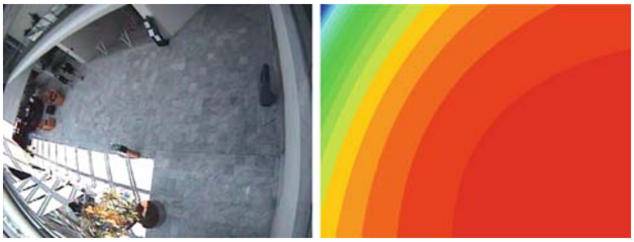

Fig. 5 *Left* Original frame with tracked pedestrians *Right* Distance weight plane (weights increase from blue to red)

3.3 Jump-diffusion dynamics

Particle filtering approach can be considered as a type of a non-deterministic multi-variate optimization method. As such it inherits the problems to which other, classical optimization methods can be prone [10]. Here we present a way to overcome one such problem—traversing valleys in the optimization space by utilizing task specific information. On the other hand, particle filtering methods are robust because they do not require any assumptions about the probability distributions of the data.

Our joint distribution is not known explicitly, so we have chosen to use Metropolis–Hastings sampling algorithm.

$$\alpha(x, x') = \min \left(1, \frac{P(x')}{P(x_t)} \cdot \frac{m_t(x|x')}{m_t(x'|x)}\right). \tag{12}$$

where x' is the candidate state, $P(x)$ is the stationary distribution of our Markov chain, m_t is the proposal distribution. In Eq. (12), the first part is the likelihood ratio between the proposed sample x' and the previous sample x_t. The second part is the ratio of the proposal density in both directions (1 if the proposal density is symmetric).

This proposal density would generate samples centered around the current state. We draw a new proposal state x' with probability $m_t(x'|x)$ and then accept it with the probability $\alpha(x, x')$. Notice that the proposal distribution is a time function, that is at each frame it will be formed based on the rules outlined below.

To form the proposal distribution we have implemented a number of reversible operators. After a number of such operators (mutations) are applied to the current state of the system, the resulting candidate state is accepted with the probability $\alpha(x, x')$. There are two types of jump transitions and five types of diffuse transitions implemented in our system:

Adding a body To generate human hypotheses within a blob detected in the scene we have used a principle similar to that of the vertical projection histogram of the blob. Our method utilizes information about the vanishing point location we obtain from the camera during the calibration stage. The projection of the blob is done along the rays going through the vanishing point instead of the parallel lines projecting onto the horizontal axis of the image. Search for local maxima is performed on the resulting histogram. A more detailed description of the "vanishing point projection histogram" can be found in [21]. This transformation draws a random head candidate and adds a new body using its head and foot coordinates. At this point the actual height and floor coordinates of the body are estimated.

Deleting a body A randomly selected body is removed from the system and excluded from further tracking. As with the creation of new bodies, door proximity is a factor. We have experimented with three types of modeling the doors, 2D points on the floor; 2D polygon, representing the floor projection of the interior of the tracked part of the scene—the proximity is computed then as the distance from the floor projection of the body to the closest edge; and a uniform model, where every location is considered equally distant from the hypothetical door. The last type of door model is useful in dealing with videos in which pedestrians are already present in the scene at the start of the tracking sequence.

Recovering a recently deleted body This step is similar to the new body creation. The body id is taken from the list of recently deleted bodies which were in some spatial proximity to the newly created body. If a strong match is found with respect to color descriptors we assign the id of the deleted body to the new one. This step is essential to overcome short-lived full occlusions and requires maintaining a history of deleted objects to pool from.

Changing body dimensions Changes the height or width of a random body probabilistically drawn from normal truncated distribution. Dimensions are viewed as normally distributed around mean human body height and width (set empirically at 170 and 70 cm correspondingly) and truncated to zero at the extremes (130–200 cm for body height and 50–90 cm for body width).

Changing body position One out of two position changes is applied at random. **First type**: move one of the existing bodies by applying the mean-shift operator with weighted anisotropic Gaussian kernel. The kernel is formed as a Gaussian, elliptic mask, where the weights increase with increased Mahalanobis distance. Additionally, if a pixel value of the foreground mask (corresponding to the background) is zero or the same pixel value from the Z-buffer is greater (i.e located further from the camera) than the current body, the weight in the kernel is effectively zeroed out. This, in essence, performs a standard color-based mean shift, but accounts only for the pixels belonging to the hypothesized body model. **Second type**: move the body to a random "initial head candidate" in some proximity from the current body position. It allows for the head candidates not initially revealed (possibly due to image noise) to be considered in the subsequent frames.

Notice that we use a set of controllable weight probabilities to add more emphasis to one or another transition type. In our application normally around 100 jump-diffuse iterations are required for each frame to reach convergence.

4 Experimental results

For testing and validation purposes we used thermal and color dataset from OTCBVS [7], with short outdoor pedestrian sequences in two locations. Each scene is filmed both with a RGB and thermal camera at the identical resolution, and calibrated to provide an approximate pixel to pixel correspondence between two types of sensors.

We assess the performance of color-thermal background model and conclude that it significantly reduces and in most cases fully eliminates two types of false foreground regions: (1) shadows as the result of a moving cloud cover (2) shadows cast by moving pedestrians (see Fig. 6).

We performed preliminary evaluation of our tracking system for the presence of three major types of inconsistencies: misses, false hits and identity switches. A *miss* is when the body is not detected or detected but tracked for an insignificant portion of its path (<30%). A *false hit* is when a new body is created where there is no actual person present. An *identity switch* is when two or more bodies exchange their IDs once within the close proximity from each other. By counting the number of each of types of errors on a number of sequences of overall 6,000 frames we have obtained results summarized in Table 1.

The most common mistakes made by the tracker were false hits. We have observed that the majority of false hits (more than 50%) are short lived, i.e. typically last for only several frames. False hits are created when the system recognizes a blob as containing more people than the actual number present in the scene. This can be explained by the distortion in shape of the blob due to pixel level noise.

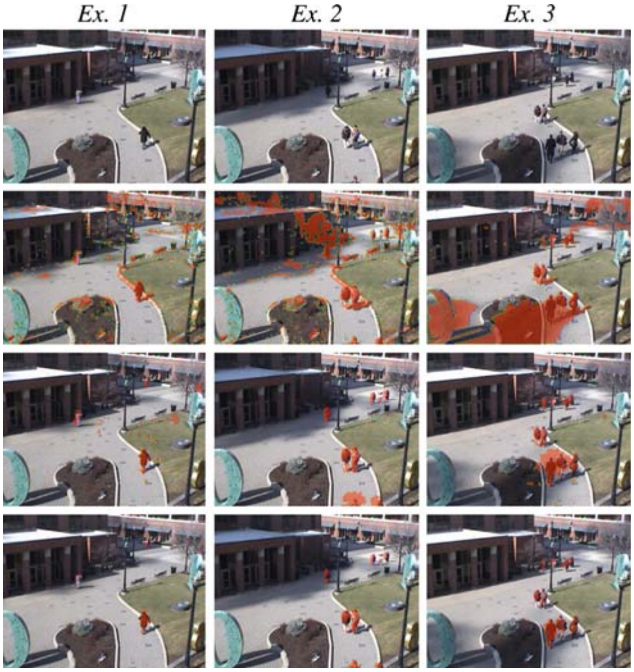

Fig. 6 Original frame (*row 1*). Probabilistic heatmap of the foreground (red representing higher probability) is shown using color channel only (*row 2*), thermal channel only (*row 3*), and using a combined model (*row 4*)

Table 1 Tracking results based on the manually observed ground truth

Sequence	Frames	People	−ppl	−Frames	+ppl	+Frames	Identity switches
1	1,054	15	3	20	1	1	3
2	0601	8	0	0	3	2	4
3	1,700	16[a]	5	10	14	5	15
4	1,506	3[b]	0	0	0	0	1
5	2,031	2	0	0	0	0	2
6	1,652	4	0	0	0	0	1
ALL	8,544	48	8	5	3	1.3	4.3
%	100	100	16.6	N/A	6.2	N/A	8.9

$+/-$ *frames* Percentage of frame misses/false-hits out of all sequence frames; $+/-$ *ppl* is missed/false-hit pedestrians

[a] Two infants, below the tracked height limit, not counted

[b] 2 pedestrian covered by trees not counted

Misses typically occurred due to partial or total occlusions by the scene objects or due to the clothing color being too close to the background. The first type of misses is usually promptly recovered by the tracker, but if the recovery took place in a different location, a new **id** is assigned, that way resulting in a "switch". For example in the first frame from sequence one in Fig. 7, pedestrian with $ID = 5$ is missed for a number of frames because of his color proximity to acquired background and insufficient number of pixels to recover with the help the color histogram.

Overall performance of the tracker shows space for improvement, it produces satisfactory detection and prolonged tracking in the crowded scenes (as shown in Fig. 7), although the ratio of pedestrian id switches has to be reduced. As it becomes apparent the complexity of the scene, i.e. the number of pedestrians, decrease the performance of the tracker.

5 Future work

One way to increase the accuracy of tracking is to enhance binary foreground map to include probabilities of each pixel belonging to foreground. Since this requires higher computational load, we intend to develop an efficient implementation for this method.

We intend to investigate the application of temporal filtering to remove insignificantly short paths. Sometimes, however, false detections are accompanied by ID switches, when the body tracked for a long time is substituted for a false hit. This presents a more complicated case and deserves further study.

As of now the tracked body is modeled by the vertical spheroid. This, in itself is quite limiting when it comes to modeling complex body transformations and interactions between people and objects. With this in mind we plan to extend the model to include a separate modality (parametrized by the size as well as the rotation angles) for each prominent body part: torso, head, arms and legs.

Although we have provided some preliminary evaluation of our tracking method which shows promising results, we still plan to extensively validate the accuracy of our algorithm using the manually marked ground truth dataset of more than 30,000 frames provided by CAVIAR project [2].

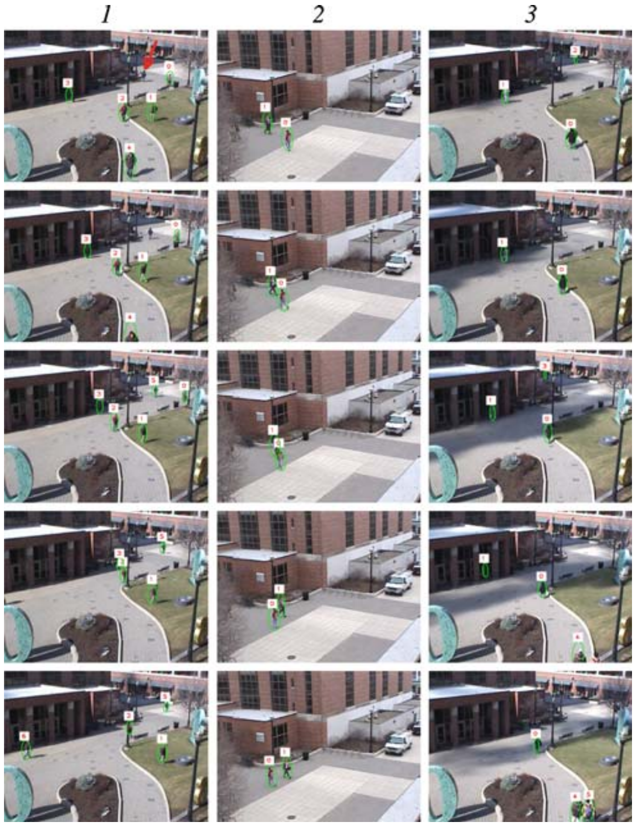

Fig. 7 Sample frames showing tracking in three sequences (each tracked body is indicated by a *green light ellipse with white tag* above showing a unique ID of a body; missed ID = 5 in sequence 1 is identified with a *red arrow*)

References

1. Bhanu, B., Han, J.: Kinematic-based human motion analysis in infrared sequences. In: IEEE WS. on Applications of Computer Vision, pp. 208–212 (2002)
2. CAVIAR: Ist 37540. Found at http://homepages.inf.ed.ac.uk/rbf/CAVIAR/ (2001)

3. Collins, R., Lipton, A., Kanade, T.: Introduction to the special section on video surveillance. IEEE Trans. Pattern Anal. Mach. Intell. **22**(8), 745–746 (2000)
4. Collins, R., Lipton, A., Kanade, T., Fujiyoshi, H., Duggins, D., Tsin, Y., Tolliver, D., Enomoto, N., Hasegawa, O., Burt, P., Wixson, L.: A system for video surveillance and monitoring. Tech. Rep. CMU-RI-TR-00-12. Carnegie Mellon University, Pittsburgh (2000)
5. Connaire, C.O., O'Connor, N., Smeaton., A.F.: Thermo-visual feature fusion for object tracking using multiple spatiogram trackers. Mach. Vis. Appl., pp. 1–12 (2007). doi:10.1007/s00138-007-0078-y
6. Dai, C., Zheng, Y., Li, X.: Layered representation for pedestrian detection and tracking in infrared imagery. In: IEEE CVPR WS on OTCBVS (2005)
7. Davis, J., Sharma, V.: Fusion-based background-subtraction using contour saliency. In: IEEE International Workshop on Object Tracking and Classification Beyond the Visible Spectrum, IEEE OTCBVS WS Series Bench (2005)
8. Davis, J.W., Sharma, V.: Fusion-based background-subtraction using contour saliency. In: IEEE CVPR WS on Object Tracking and Classification Beyond the Visible Spectrum, pp. 19–26 (2005)
9. Deutscher, J., North, B., Bascle, B., Blake, A.b.: Tracking through singularities and discontinuities by random sampling. In: International Conference on Computer Vision (1999)
10. Doucet, A., de Freitas, N., Gordon, N.: Sequential Monte Carlo Methods in Practice. Springer, Heidelberg (2001)
11. Elgammal, A., Davis, L.: Probabilistic framework for segmenting people under occlusion. In: International Conference on Computer Vision (2001)
12. Gavrila, D.: The visual analysis of human movement: A survey. Comp. Vis. Image Understanding **73**(1), 82–98 (1999)
13. Haritaoglu, I., Flickner, M.: Detection and tracking of shopping groups in stores. In: International Conference on Computer Vision and Pattern Recognition (2001)
14. Haritaoglu, I., Harwood, D., Davis, L.: W-4: Real-time surveillance of people and their activities. IEEE Trans. Pattern Anal. Mach. Intell. **22**(8), 809–830 (2000)
15. Havasi, L., Sziranyi, T.: Motion tracking through grouped transient feature points. In: Advanced Concepts for Intelligent Vision Systems (2004)
16. Isard, M., MacCormick, J.: Bramble: A bayesian multiple-blob tracker. In: International Conference on Computer Vision (2001)
17. Kemp, C., Drummond, T.: Multi-modal tracking using texture changes. In: British Machine Vision Conference (2004)
18. Kim, K., Chalidabhongse, T., Harwood, D., Davis, L.: Background modeling and subtraction by codebook construction. In: International Conference on Image Processing (2004)
19. Leykin, A.: Visual human tracking and group activity analysis: A video mining system for retail marketing. PhD Thesis, Indiana University (2007)
20. Leykin, A., Hammoud, R.: Robust multi-pedestrian tracking in thermal-visible surveillance videos. In: IEEE CVPR WS on Object Tracking and Classification Beyond the Visible Spectrum, p. 136. IEEE Computer Society, Los Alamitos (2006)
21. Leykin, A., Tuceryan, M.: A vision system for automated customer tracking for marketing analysis: Low level feature extraction. In: Human Activity Recognition and Modelling Workshop (2005)
22. Maybank, S., Tan, T.: Introduction to special section on visual surveillance. Int. J. Comp. Vis. **37**(2), 173–173 (2000)
23. Mittal, A., Davis, L.: M2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene using region-based stereo. In: European Conference on Computer Vision (2002)
24. Morris, D., Rehg, J.: Singularity analysis for articulated object tracking. In: International Conference on Computer Vision and Pattern Recognition (1998)
25. Nanda, H., Davis, L.: Probabilistic template based pedestrian detection in infrared videos. In: IEEE Intell. Vehicles Symp. (2002)
26. Oliver, N., Rosario, B., Pentland, A.: A bayesian computer vision system for modeling human interactions. IEEE Trans. Pattern Anal. Mach. Intell. **22**(8), 831–843 (2000)
27. Owechko, Y., Medasani, S., Srinivasa, N.: Classifier swarms for human detection in infrared imagery. In: IEEE CVPR WS on Object Tracking and Classification Beyond the Visible Spectrum (2004)
28. Rana, Y., Weiss, I., Zheng, Q., Davis, L.S.: Pedestrian detection via periodic motion analysis. Int. J. Comp. Vis. **71**(2), 143–160 (2007)
29. Rittscher, J., Tu, P., Krahnstoever, N.: Simultaneous estimation of segmentation and shape. In: International Conference on Computer Vision and Pattern Recognition, vol. II, pp. 486–493 (2005)
30. Sminchisescu, C., Triggs, B.: Kinematic jump processes for monocular 3d human tracking. In: International Conference on Computer Vision and Pattern Recognition (2003)
31. Xu, F., Fujimura, K.: Pedestrian detection and tracking with night vision. In: IEEE Intell. Vehicles Symp. (2002)
32. Yasuno, M., Yasuda, N., Aoki, M.: Pedestrian detection and tracking in far infrared images. In: IEEE CVPR WS on Object Tracking and Classification Beyond the Visible Spectrum (2004)
33. Zhao, T., Nevatia, R.: Tracking multiple humans in crowded environment. In: International Conference on Computer Vision and Pattern Recognition (2004)