

Tracking Multiple Humans in Complex Situations

Tao Zhao, *Member, IEEE*, and Ram Nevatia, *Fellow, IEEE*

Abstract—Tracking multiple humans in complex situations is challenging. The difficulties are tackled with appropriate knowledge in the form of various models in our approach. Human motion is decomposed into its global motion and limb motion. In the first part, we show how multiple human objects are segmented and their global motions are tracked in 3D using ellipsoid human shape models. Experiments show that it successfully applies to the cases where a small number of people move together, have occlusion, and cast shadow or reflection. In the second part, we estimate the modes (e.g., walking, running, standing) of the locomotion and 3D body postures by making inference in a prior locomotion model. Camera model and ground plane assumptions provide geometric constraints in both parts. Robust results are shown on some difficult sequences.

Index Terms—Multiple-human segmentation, multiple-human tracking, visual surveillance, human shape model, human locomotion model.

1 INTRODUCTION

TRACKING multiple people in video sequences is important for a number of tasks, such as video surveillance and event inference, as humans are the principal actors in daily activities of interest, and locomotion (e.g., walking, running, standing) is a key class of human motion.

The goal of our work is to track multiple people in complex situations using a single stationary video camera. We decompose the human motion into a global motion (i.e., position and orientation) and limb motions (i.e., more detailed body postures). Some applications, such as large-scale video surveillance, may only require global motion while others require the knowledge of both. Our first objective is to segment multiple human objects and track their global motion in complex situations where they may move in small groups, have interocclusions, cast shadows on the ground, and reflections may exist. Our second objective is to estimate the locomotion *modes* (e.g., walking, running, standing) and the coarse 3D body postures. This also helps verify the human hypotheses generated in the first part. Earlier versions of our work were published in [45] and [46].

In a stationary camera setting, moving pixels can usually be extracted by change detection techniques (e.g., [44], [40]) after some noise filtering. Moving pixels are grouped into *blobs* according to their connectivity. Computational efficiency of the method is attractive for real-time applications. However, such blobs do not incorporate object shape constraints. In ideal situations, each blob corresponds to a single moving object, such as a human or a vehicle; such an

assumption has been common in past research (e.g., [13], [7], [40], etc.). However, in more complex and realistic scenarios, the blob-based analysis may face some difficulties. First, a single blob may contain multiple humans due to their physical proximity or due to the camera viewing angle; this may happen temporarily (e.g., passing-bys) or persistently (e.g., moving together). Second, a single object may be fragmented into several blobs due to low color contrast. Third, blobs may contain pixels corresponding to the shadows or reflections cast by the moving objects as well as noise. These problems may also be present simultaneously; some examples are shown in Fig. 1. When multiple people move in a scene, the blobs may go through frequent structural changes (i.e., splits and merges) due to the above problems. This causes the combinatorial search for temporal correspondence of blobs to be expensive. Even if the correspondence is established, what each trajectory corresponds to (e.g., an object, a part of an object, a few objects together) is still unknown.

We argue that analysis using explicit object shape models has advantages over blobs. First, detecting individual objects is a goal. Second, as an entity in the physical world, an object does not undergo structural changes such as split and merge. Furthermore, the constraints (e.g., shape, size, motion) of the physical objects can assist segmentation and tracking. The model-based approach is also less sensitive to noise and the parameters of lower-level processing. Here, we propose to use a coarse 3D human shape model (an ellipsoid). A camera model provides additional constraints on the geometric aspect.

Estimating human body postures from a single viewpoint is difficult especially when the image size of humans is small as is typical in surveillance videos. We employ a prior locomotion model to assist posture estimation since the motion pattern of human locomotion is highly structured. The coarse body postures are efficiently and robustly estimated by inferring an optimal path in the prior motion model given the observed data, resulting in a *tracking as*

• T. Zhao is with Sarnoff Corporation, 201 Washington Road, Princeton, NJ 08543. E-mail: tzhao@sarnoff.com.

• R. Nevatia is with the Institute for Robotics and Intelligent Systems, University of Southern California, Los Angeles, CA 90089. E-mail: nevatia@usc.edu.

Manuscript received 19 Sept. 2003; revised 26 Feb. 2004; accepted 1 Mar. 2004.

Recommended for acceptance by S. Sclaroff.

For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0282-0903.



Fig. 1. Example moving blobs by change detection from three images in our dataset. One blob may contain multiple objects due to their proximity or camera projection (a), (c). It may also contain nonobject pixels such as shadow (sun shadow in (a) and soft shadow in (c)) or reflection (b). The foreground corresponding to an object may get fragmented (c).

recognition approach. The motion pattern is also used to verify the human hypotheses generated by static frame analysis.

2 RELATED WORK

Most of the work on tracking for visual surveillance is based on change detection [44], [36], [40], [15], [13], [11], [21], [38] or frame differencing [23] if the camera is stationary. Additional stabilization is required if the camera is mobile [7], [42]. These methods usually infer global motion only and can be roughly grouped as follows:

- Perceptual grouping techniques are used to group the blobs in the spatio-temporal domain as in Cohen and Medioni [7] and Kornprobst and Medioni [20]. However, these methods still suffer from the deficiencies of blob-based analysis discussed earlier. In Lipton et al. [23], a moving blob is classified into a single human, multiple-human or a vehicle according to its shape. However, the positions of the people in a multihuman blob is not inferred.
- Some work (Rosales and Sclaroff [36], Elgammal and Davis [11], and McKenna et al. [25], etc.) assumes people are isolated when they enter the scene so that an appearance model can be initialized to help in tracking when occlusion happens. These methods cannot be applied where a few people are observed walking together in a group.
- Some methods try to segment multiple people in a blob. The W^4 system [15] uses blob vertical projection to help segment multiple humans in one blob. It only applies to data where multiple people distribute horizontally in the scene (“step on one’s head” does not happen, usually from a ground level camera). It handles shadows by use of stereo cameras [14]. Siebel and Maybank [38] extend the Leeds human tracker [1] by the use of a head detection method similar to the approach taken in our system.
- Tao et al. [41] and Isard and MacCormick [18] track multiple people using the CONDENSATION algorithm [17]. The system in [18] also uses a human shape model and the constraints given by camera calibration. It does not involve any object-specific representation; therefore, the identities of humans are likely to be confused when they overlap. Besides, the performance of particle filter is limited by the dimensionality of the state space, which is proportional to the number of objects.

Other related work includes Tao et al. [42] which use a dynamic layer representation to track objects. It combines

compact object shape, motion, and appearance in a Bayesian framework. However it does not explicitly handle occlusion of multiple objects since it was designed mainly for airborne video.

Much work has been done on estimating human body postures in the context of video motion capture (a recent review is available in [26]). This problem is difficult, especially from a single view because 3D pose may be underconstrained from one viewpoint. Most successful systems (e.g., [9]) employ multiple viewpoints, good image resolution, and heavy computation, which is not always feasible for applications such as video surveillance. Use of constrained motion models can reduce the search space, but it only works on the type of motion defined in the model. Rohr [35] describes pioneering work on motion recognition using motion captured data. In each frame, the joint angle values are searched for on the motion curves of a walking cycle. Results are shown only on an isolated human walking parallel to the image plane. Motion subspace is used in Sidenbladh et al. [37] to track human walking using a particle filter. Both [35] and [37] operate in an online mode. Bregler [4] uses HMMs (hidden Markov models) to recognize human motion (e.g., running), but the recognition is separated from tracking. Brand [3] maps 2D shadows into 3D body postures by inference in an HMM learnt from 3D motion captured data, but the observation model is for isolated objects only. In Krahnstover et al. [21], human tracking is treated as an inference problem in an HMM; however, this approach is appearance-based and works well only for the viewpoints for which the system was trained. For motion-based human detection, motion periodicity is an important feature since human locomotion is periodic; an overview of these approaches is given in [8]. Some of the techniques are view dependent, and usually require multiple cycles of observation. It should be noted that the motion of human shadow and reflection is also periodic. In Song et al. [39], human motion is detected by mapping the motion of some feature points to a learned probabilistic model of joint position and velocity of different body features, however, joints are required to be detected as features. Recently, an approach similar to ours has been proposed by Efros et al. [10] to recognize actions. It is also based on flow-based motion description and temporal integration.

3 OVERVIEW

We propose to solve the problem of human locomotion tracking in complex situations by taking advantage of the available camera, scene, and human models. We believe

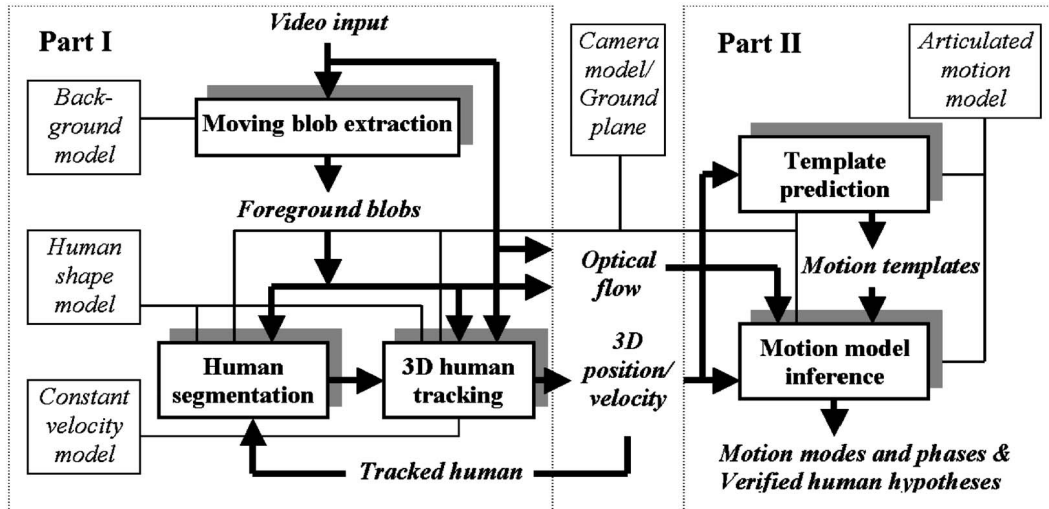


Fig. 2. The system diagram. Shaded box: program module; plain box: model; thick arrow: data flow; thin line: model association.

that the models we use are generic and applicable to a wide variety of situations. The models used are:

- A *statistical background appearance model* directs the system's attention to the regions showing difference from the background.
- A *camera model* to provide a transformation from the world to the image. In conjunction with the assumption that humans move on a known ground plane, it helps transform positions between the image and the physical world and allows reasoning with invariant 3D quantities (e.g., height and shape).
- A *3D coarse human shape model* to constrain the shape of an upright human. It is critical for human segmentation and tracking.
- A *3D human articulated locomotion model* to help recover the locomotion modes and phases and recognize walking humans to eliminate false hypotheses formed by the static analysis.

The overview block diagram of the system is shown in Fig. 2. First, the foreground blobs are extracted by a change detection method. Human hypotheses are computed by boundary analysis and shape analysis using the knowledge provided by the human shape model and the camera model. Each hypothesis is tracked in 3D in the subsequent frames with a Kalman filter using the object's appearance constrained by its shape. Two-dimensional positions are mapped onto the 3D ground plane and the trajectories are formed and filtered in 3D. Depth ordering can be inferred from the 3D information, which facilitates the tracking of multiple overlapping humans and occlusion analysis.

To estimate the locomotion modes and the body postures, we propose a *tracking as recognition* approach where tracking is accomplished by recognizing the motion (i.e., making inference) in a 3D articulated locomotion model. The locomotion model consists of walking, running, and standing. The model is constructed using 3D motion capture data and is represented as a hierarchical finite state machine. The higher-level states represent the modes while the lower-level states represent 3D body postures (phases).

The approach results in robust performance without the need for interactive initialization. Sometimes, the static shape information in one frame does not yield correct hypotheses in the segmentation. We verify the human hypotheses by their dynamical aspect, i.e., by checking if they exhibit a proper walking pattern.

4 SEGMENTATION AND TRACKING OF MULTIPLE HUMANS

4.1 Background Model, Camera/Scene Model, and Human Shape Model

We incorporate a statistical background model [44] where the color of each pixel in the image is modeled by a Gaussian distribution. The background model is first learnt in a period where there are no moving objects in the scene and then updated for each incoming frame with the nonmoving pixels. A single initial background frame is sufficient to start. The background model can be easily replaced with a more complex one (e.g., one with a multi-Gaussian model [40] or one which can start with moving objects in the scene [15]) if needed.

Change detection is performed on each incoming frame. The pixels whose values are sufficiently different from the corresponding background models are classified as foreground pixels. The binary map is filtered with a median filter and the morphology *close* operator to remove isolated noise, resulting in the foreground mask F . Connected components are then computed, resulting in the *moving blobs* (or, simply, *blobs*) of that frame. Some examples are shown in Fig. 1.

In contrast to the ground-level camera setup used in some of the previous work (e.g., [15], [25], etc.), we deploy the camera a few meters above the ground looking down. This allows a larger coverage and less occlusion, especially avoiding the situation where the entire scene is occluded by one object. Such a setup is also in accordance with most commercial surveillance systems.

To compute the camera calibration, the traditional approach requires enough 3D feature points (≥ 6 points with

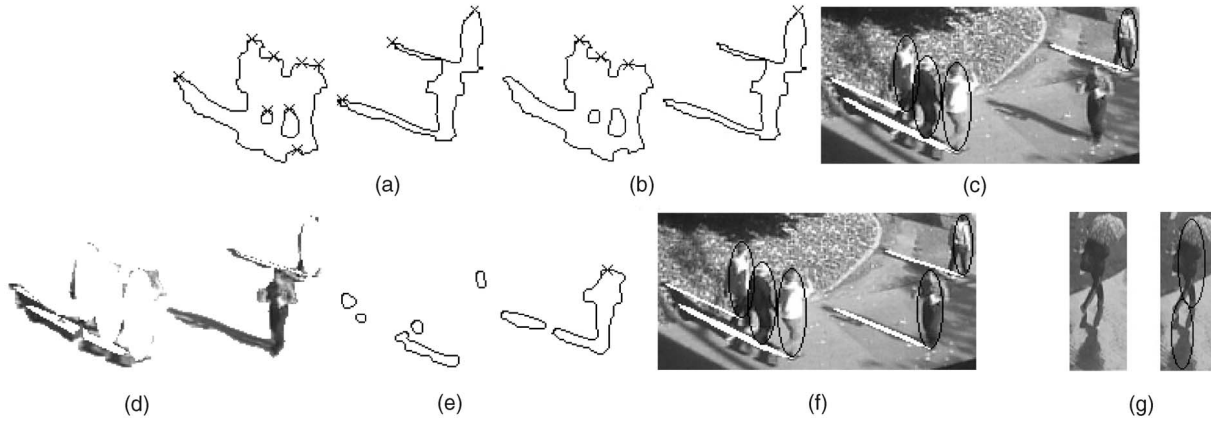


Fig. 3. The process of multihuman segmentation. (a) unscreened head top candidates; (b) screened head top candidates; (c) first four segmented people; (d) the foreground residue after first four people are segmented; (e) head top candidate after first four people are segmented; (f) the final segmentation; (g) an example of false hypothesis.

≥ 2 of them out of a plane) and their corresponding image points. A linear calibration method described in [12] works satisfactorily if the selected points are distributed evenly in the image. If the number of feature points is not enough or measurement of 3D points is not possible, methods based on the projective invariance (e.g., vanishing points) can be used (e.g., [22], [24]). It has also been shown in [24] that humans walking in more than one direction can provide enough information for an approximate camera calibration. Both methods have been used in our experiments.

We assume that people move on a known ground plane. The camera model and the ground plane together serve as a bridge to transform 2D and 3D quantities. Three-dimensional quantities can be projected into 2D quantities by the camera model. The camera model and the ground plane define a transformation (i.e., a *homography*) between the points on the image plane and the points on the ground plane. The measurements of the objects (such as position, velocity, and height) in the image can be transformed into 3D. Sometimes, we only know the position of a human's head instead of his/her feet. Then, the transformation can be carried out approximately by assuming that the humans are of an average height. The transformation degenerates when the projection of the reference plane is (or close to) a line in the image, i.e., when the optical axis is on the reference plane. Such a case does not occur in our camera setup.

We model gross human shape by a vertical 3D ellipsoid. The two short axes are of the same length and have a fixed ratio to the length of the long axis. The parameters of an object include its position on the ground plane and its height. Assuming an ellipsoid is represented by a 4 by 4 matrix, \mathbf{Q} , in homogenous coordinates, its image under camera projection \mathbf{P} (a 3 by 4 matrix) is an ellipse, represented by a 3 by 3 matrix, \mathbf{C} . Relation between them is given in [16] by $\mathbf{C}^{-1} = \mathbf{P}\mathbf{Q}^{-1}\mathbf{P}^T$. An object mask M is defined by the pixels inside the ellipse. The 3D human shape model also enables geometric shadow analysis.

4.2 Segmenting Multiple Humans

We attempt to interpret the foreground blobs with the ellipsoid shape model. Human hypotheses are generated by analyzing the boundary and the shape of the foreground

blobs. The process is described below and shown step by step graphically in Fig. 3.

4.2.1 Locating People by Head Top Candidates

In scenes with the camera placed several meters above the ground, the head of a human is less likely to be occluded; we find that recognizing the head top on the foreground boundary is a simple and effective way to locate multiple, possibly overlapping humans.

A point can be a head top candidate if it is a peak (i.e., the highest point in the vertical direction (the direction towards the vertical vanishing point) along the boundary within a range (Fig. 3a)) defined by the average size of a human head assuming an average height. A human model of an average height is placed at each peak. Those peaks which do not have sufficient foreground pixels within the model are discarded (Fig. 3b). If a head is not overlapped with the foreground region of other objects, it is usually detected with this method (Fig. 3c).

For each head top candidate, we find its potential height by finding the first point that turns to a background pixel along the vertical direction in the range determined by the minimum and the maximum human height. We do this for all points in the head area and take the maximum value; this enables finding the height of different human postures. Having head top position and the height, an ellipsoid human hypothesis is generated.

4.2.2 Geometrical Shadow Analysis

Assuming that the sun is the only light source and its direction is known (can be computed from the knowledge of time, date, and geographical location, e.g., using [29]), the shadow of an ellipsoid on the ground, which is an ellipse, can be easily determined. Any foreground pixel which lies in the shadow ellipse and whose intensity is lower than that of the corresponding pixel in the background by a threshold T_s is classified as a shadow pixel. Most of the current shadow removal approaches are based on an assumption that the shadow pixels have the same hue as the background but are of lower intensity (see [33] for a review) and ignore the shadow geometry. The color-based approaches are not expected to work well on very dark sun cast shadows, as hue computation will be highly inaccurate.

4.2.3 The Algorithm

Segmenting multiple humans is an iterative process. We denote the foreground mask after removing the existing human masks and their shadows as the *foreground residue map* Fr . At the beginning of the segmentation, Fr is initialized with F .¹ The head top candidate set Hc is computed from Fr . We choose one candidate, which has the minimum depth value (closest to the camera) to form a human hypothesis. Figs. 3c and 3d show the first four segmented humans and the foreground after their masks and shadow pixels are removed. As can be seen, a large portion of the shadow pixels is removed correctly. A morphological *open* operation is performed on Fr to remove the isolated small residues (Fig. 3e). This process iterates until no new head candidates are found (Fig. 3f).

This approach works well for a small number of overlapping people that do not have severe occlusion; a severely occluded object will be detected when it becomes more visible in a subsequent frame. This method is not sensitive to blob fragmentation if a large portion of the object still appears in the foreground. In our experiments, we found that this scheme tends to have a very low false alarm rate. The false alarms usually correspond to large foreground region not (directly) caused by a human. For example, when people move with their reflections, the reflections are also hypothesized as humans (as in Fig. 3g). We can verify the hypotheses with dynamical features to remove such false alarms, as is described in Section 5.5.2.

4.3 Tracking Multiple Humans

Once segmented, the objects are tracked in the subsequent frames. Tracking is a loop consisting of prediction of the positions from the previous frame, search for the best match, and update of the object representation. Multiple objects are matched one by one according to their depth order.

4.3.1 Object Representation for Tracking

An elliptic shape mask (M) projected from the ellipsoid model represents the gross human shape. The shape/scale of the mask changes automatically according to the human's position and the geometry. A texture template (T) is used to represent the appearance of a human by the *rgb* value of each pixel. Not every pixel inside the elliptic mask corresponds to the foreground; we also keep a foreground probability template (Fp) for each human object, which stores the probability of each pixel in the elliptic mask as foreground. It enables handling of some variations of body shape/pose. Fig. 4b shows examples of the representation.

Due to camera perspective effect, the elliptic masks of the same ellipsoid have different shape (i.e., orientations and lengths of the axes) when the human is at different locations. Therefore, a mapping is needed to align different ellipses for matching and updating. Suppose we have two ellipses $e_1 = \{\mathbf{u}_1, \alpha_1, \beta_1, \theta_1\}$ and $e_2 = \{\mathbf{u}_2, \alpha_2, \beta_2, \theta_2\}$ in their parametric forms where \mathbf{u} , α , β , and θ are the center, long axis, short axis, and the rotation, respectively. A mapping $\mathbf{u}' = W(\mathbf{u})$ transforms a point \mathbf{u} in e_1 to its corresponding point \mathbf{u}' in e_2 by

1. When segmentation is integrated in a tracking system, we remove the masks of the objects already being tracked from Fr before performing segmentation.

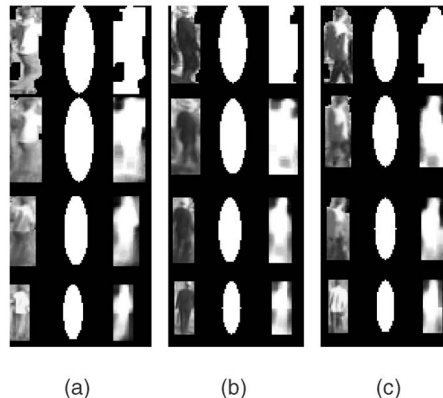


Fig. 4. Examples of object representation for tracking and its evolution: (a) texture template, (b) shape mask, and (c) foreground probability template. From top to bottom: 1st, 25th, 100th, 200th frame, respectively.

aligning e_1 and e_2 with their centers and corresponding axes through translation, rotation, and scaling by

$$\begin{aligned} \mathbf{u}' &= W(\mathbf{u}) \\ &= \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 \\ \sin\theta_2 & \cos\theta_2 \end{bmatrix} \begin{bmatrix} \frac{\alpha_2}{\alpha_1} & 0 \\ 0 & \frac{\beta_2}{\beta_1} \end{bmatrix} \begin{bmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos\theta_1 \end{bmatrix} \\ &\quad [\mathbf{u} - \mathbf{u}_1] + \mathbf{u}_2. \end{aligned} \quad (1)$$

4.4 Computing the Best Match

Suppose we have an estimate of the position of a human object in frame $n - 1$, and have a prediction of its position $\bar{\mathbf{u}}_n = (\bar{u}_n, \bar{v}_n)$ and a covariance matrix $\bar{\Sigma}_n$ (from a filter, such as one introduced in Section 4.5) in frame n . Starting from $\bar{\mathbf{u}}_n$, a search is conducted for all the positions in a neighborhood ($\mathbf{u} \in \Omega$) to locate each human object using the representations in Section 4.3.1.

In case of overlapping objects, a joint likelihood (e.g., [18]) should be used. However, it involves a large search space which is computationally expensive. Since their depth order from the camera can be inferred by their 3D positions and the camera model, multiple objects are matched one by one, starting from the one closest to the camera. An occupancy map O , where $O(\mathbf{u}) = 1$ if pixel location \mathbf{u} is occupied by another object (in front of the current object), is used to facilitate the multiobject matching. O is initialized with zeros and filled with an object mask once its position is fixed.

For each object, we determine the search range Ω using:

1. the knowledge of maximum human motion velocity,
2. the covariance of the predicted position $\bar{\Sigma}_n$, and
3. the physical occupancy constraint, i.e., a human object cannot occupy the space already taken by other objects.

Limiting Ω as much as possible both reduces the computation of matching and minimizes the potential of false matching.

The best estimate of the position of a human object in the current frame is given by

$$\mathbf{u}^* = \arg \min_{\mathbf{u} \in \Omega} \left(\sum_{i \in M^{n-1}} E_i(\mathbf{u}) \right), \quad (2)$$

where $E_i(\mathbf{u})$ is the penalty due to pixel location \mathbf{i} in the shape mask if the position of the human feet is at \mathbf{u} , which is computed by

$$E_i(\mathbf{u}) = \begin{cases} Fp_{n-1}(\mathbf{i})\|T_{n-1}(\mathbf{i}) - I(W_{\mathbf{u}}(\mathbf{i}) + \mathbf{u})\|, & \text{if } O(W_{\mathbf{u}}(\mathbf{i}) + \mathbf{u}) = 0 \\ E_o, & \text{otherwise,} \end{cases}$$

where T_{n-1} and Fp_{n-1} are the texture template and the foreground probability template, estimated in frame $n-1$, respectively, $W_{\mathbf{u}}(\mathbf{i})$ is the transformation from the ellipse of frame $n-1$ to the ellipse determined by feet position \mathbf{u} and $\|\cdot\|$ is the Euclidean distance of the color (RGB) of two pixels. E_o is a predefined penalty for a pixel occluded by other humans and set larger than the average error of a pair of correctly matched pixels. This penalty avoids the solution where an object “hides” behind another object unless it is necessary. This matching criterion is similar to template matching after ellipse warping and with consideration of occlusion.

4.5 Estimation with Kalman Filter

We estimate the state of each human object using a Kalman filter [19] with a constant velocity model (3). At each frame, the position is predicted by the Kalman filter, the best position is obtained as above and is used to update the filter parameters.

$$\begin{bmatrix} \mathbf{x}_n \\ \dot{\mathbf{x}}_n \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x}_{n-1} \\ \dot{\mathbf{x}}_{n-1} \end{bmatrix} + \mathbf{w}_n; \quad \hat{\mathbf{x}}_n = \begin{bmatrix} 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x}_n \\ \dot{\mathbf{x}}_n \end{bmatrix} + \mathbf{v}_n, \quad (3)$$

where $\mathbf{x}_n = [x_n, y_n]^T$ is the 3D position and $\dot{\mathbf{x}}$ is the velocity on the ground plane, $\hat{\mathbf{x}}_n = [\hat{x}_n, \hat{y}_n]^T$ is the measured 3D positions. $\mathbf{w}_n \sim \mathcal{N}(\mathbf{0}, \text{diag}(\sigma_x^2, \sigma_y^2, \sigma_{\dot{x}}^2, \sigma_{\dot{y}}^2))$ is the process noise, including the deviation from the constant velocity assumption. $\mathbf{v}_n \sim \mathcal{N}(\mathbf{0}, \hat{\Sigma}_n)$ is the measurement noise and Δt is the time interval since last update. Δt equals to the temporal sampling interval unless the object is completely occluded. Although the measurement $\hat{\mathbf{x}}$ only consists of position, the state being estimated also includes velocity.

$\hat{\mathbf{x}}_n$ is calculated from the image position of the human object $\mathbf{u}_n^* = [u_n, v_n]^T$ through the homography $\mathbf{x} = H(\mathbf{u})$. The measurement noise on the ground plane is different for an object at different positions due to the perspective effect of the camera projection. $\hat{\Sigma}_n$ can be computed as in (4) using a first order approximation of the homography. This gives us good filtering results (especially for the motion direction) in spite of the perspective effect.

$$\hat{\Sigma}_n = \mathbf{J}(\mathbf{u})_n^{-1} \cdot \begin{bmatrix} \sigma_u^2 & 0 \\ 0 & \sigma_v^2 \end{bmatrix} \cdot (\mathbf{J}(\mathbf{u})_n^{-1})^T, \quad (4)$$

where $\mathbf{J}(\mathbf{u})_n = \frac{\partial H(\mathbf{u})}{\partial \mathbf{u}}$ and $\text{diag}(\sigma_u^2, \sigma_v^2)$ is the covariance matrix of the 2D measurement noise.

4.6 Object Updating

The texture templates and foreground probability templates are initialized at the first frame where the human objects are hypothesized. They are updated using the new observations of each frame to take into account the change of

appearance and shape over time. Simple IIR (infinite impulse response) filters are used:

$$T_n(\mathbf{i}) = \begin{cases} T_{n-1}(W(\mathbf{i})), & \text{if } O(\mathbf{i} + \mathbf{u}) = 1 \text{ or } F_n(\mathbf{i} + \mathbf{u}) = 0; \\ (1 - \alpha)T_{n-1}(W(\mathbf{i})) \\ \quad + \alpha T_n(\mathbf{i} + \mathbf{u}), & \text{otherwise.} \end{cases} \quad (5)$$

Similarly,

$$Fp_n(\mathbf{i}) = \begin{cases} Fp_{n-1}(W(\mathbf{i})), & \text{if } O(\mathbf{i} + \mathbf{u}) = 1; \\ (1 - \alpha)Fp_{n-1}(W(\mathbf{i})) + \alpha F_n(\mathbf{i} + \mathbf{u}), & \text{otherwise,} \end{cases} \quad (6)$$

where \mathbf{u} is the feet position which is also the origin of the templates, and α is an updating factor. Note that the texture template is only updated with foreground pixels. Both maps are updated with the nonoccluded parts. Fig. 4 shows the evolution of the templates in an example where both the objects’ sizes and orientations changed.

4.7 Handling Occlusions

Occlusion of multiple objects has been addressed in several places in the algorithm, for example, in matching and updating. Furthermore, we compute r , the visible fraction of the object. r is defined by N_v/N_e , where N_v is the number of visible (i.e., unoccluded) foreground pixels in the elliptic mask and N_e is area, in pixel, of the elliptic mask of each object. The measurement noise σ_u, σ_v of the Kalman filter are set proportional to $1/r$. Using two thresholds T_{o1} and T_{o2} , if $T_{o1} > r > T_{o2}$, the object is said to be partially occluded. If $r < T_{o2}$, the object is said to be completely occluded. In case of complete occlusion, the object follows the prediction of the Kalman filter. If an object is completely occluded for a certain number of frames, it is discarded.

4.8 Implementation and Results

To handle the entry and exit of human objects, the user provides the information of entrances and exits besides the image boundaries (can be real entrances and exits, e.g., a gate, or occluding structures, e.g., a roof; marked on the image in the first pane of Fig. 5 and Fig. 6). We only generate a human hypothesis if its shape mask does not touch an entrance. This effectively causes a human object to be detected only after it is fully in the scene. A human object hypothesis is removed if it touches an exit.

We have tested the above approach on various datasets from different sources and obtained good results.² The system can segment and track multiple humans in small groups and temporary severe occlusion very well. It is insensitive to image noise by blob fragmentation or camera shaking. Due to space limits, we only show results of two representative sequences and report the performance evaluation in Section 4.9. The segmentation and tracking system runs in real-time (30 fps or more) on videos of similar complexity as below with frame size 360 by 240 on a Pentium 4 2.7G Hz PC.

2. The result video files in this paper are available online at <http://iris.usc.edu/~taozhao/papers/PAMI/>.

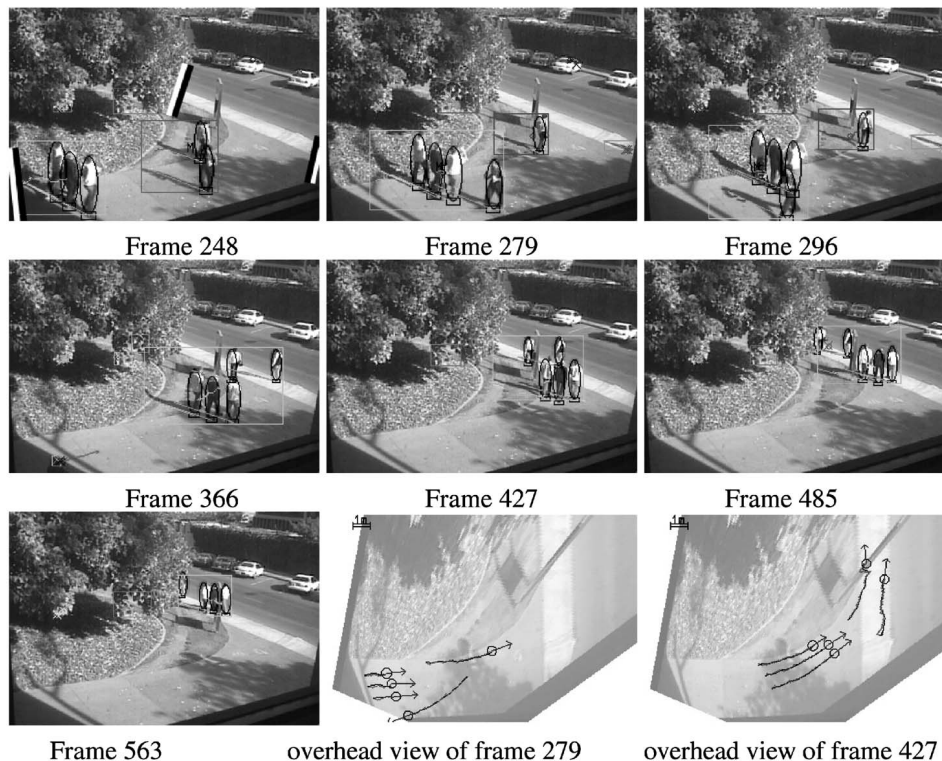


Fig. 5. Selected frames of human global motion tracking result of *seq1*. Human objects are shown in ellipses. Search mask sizes are shown at the feet of each human. Rectangles show the bounding boxes of the blobs. The last two panes show the tracking result projected onto the ground plane: object (in circle), velocity (in arrow), and trajectories. The entrances (in dark lines) and the exits (in bright lines) besides the image boundaries are marked in the first pane.

The first example (*seq1*, in Fig. 5) includes three humans walking in a close group, humans passing-by each other, single/multiple humans passing an obstacle, and strong sun cast shadows. All humans were tracked successfully and their trajectories were estimated accurately. As comparison, the bounding boxes of the blobs are also shown in the images. The blobs split and merge frequently though the humans move smoothly. The search region size changes when humans are in different positions and grows when passing the occluding object. The last two panes of Fig. 5 show the humans' position and velocity mapped onto the ground plane which reveals the real relationships (e.g., distance) of different humans.

The second example (*seq2*, in Fig. 6) is a segment containing dense traffic from a longer sequence. There are 13 people going through the scene in different directions resulting in many instances of their passing-by each other. In particular, a roller-skater (ID33) and a biker (ID34) pass through two walkers (ID35, 31) with similar clothing color at high speeds. Both their identities and their positions are tracked accurately.

4.9 Evaluation and Discussion

Besides observing the results visually, we evaluate the performance of the system quantitatively on a dataset. The sequences in the dataset are selected to be heterogeneous: They are obtained from different sources (including a sequence in PETS01 dataset [30]), captured on nine distinct sites, with the camera tilt angle ranging from 5 to 40 degrees and image height of a typical human ranging from about

25 pixels to about 80 pixels. The total length of the data is 61,890 frames (35 minutes) and the total number of humans that appear is 520 (counted by a human observer). The number of human-frames (i.e., the summation of the frames of each human) is 243,804 (135 minutes).

The performance of human segmentation and tracking is affected by the complexity of the data. Generally, the more the occlusion, the more challenges the data pose to the algorithms. It is known that track drifts or switches are more likely to happen when a few people walk in a group or pass-by each other. We describe the complexity of the dataset by the number of the events challenging to the system: "passing-by each other," "passing an obstacle," and "walking in a group." A "passing-by" means that two or more humans cross each other with the human farther away from the camera partially or completely occluded. A "passing an obstacle" means that a human moves behind a structure (e.g., a tree or a pole) in the scene. A "walking in a group" means that two or more humans walk together closely (e.g., side by side) and persistently. The counts of such events, obtained from a human observer, are summarized in Table 1a.

Quantitative evaluation of multiobject tracking is more difficult than object detection, object recognition or single object tracking due to the complex behaviors created by the system. These behaviors need various statistics to be computed in order to be meaningful to different applications. Some discussions have been given in [31]. We do not intend to address these general issues.

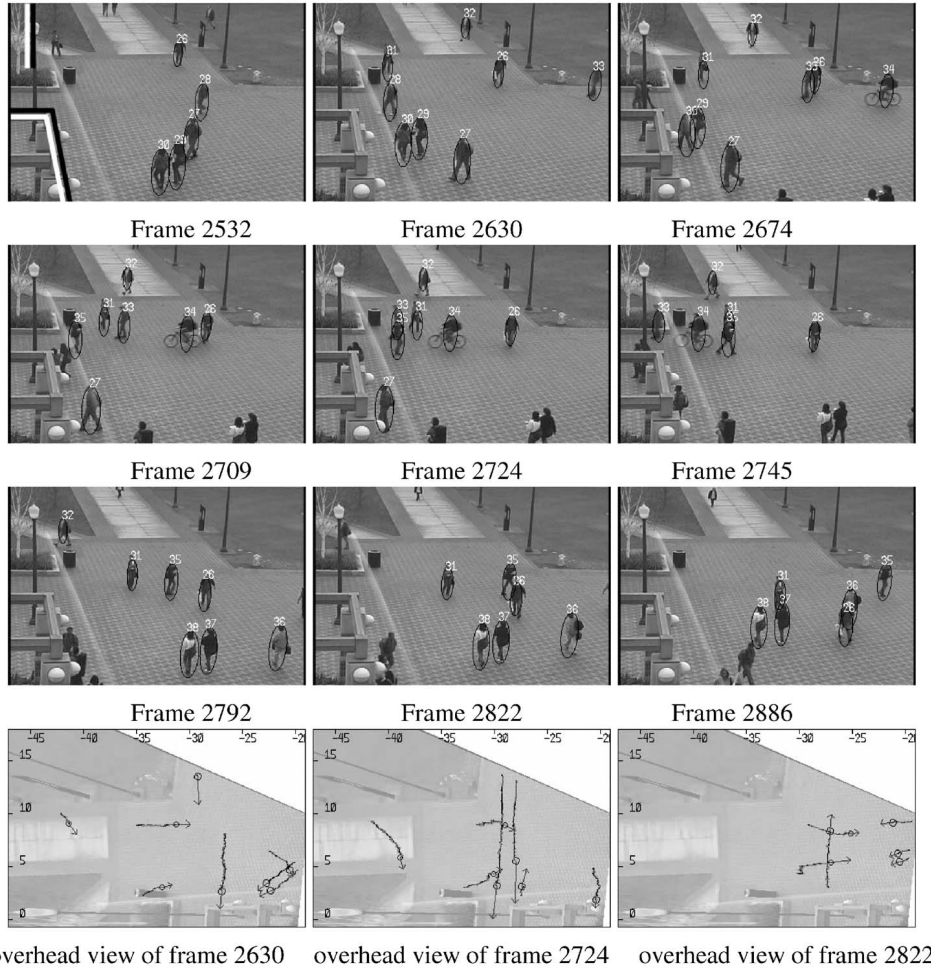


Fig. 6. Selected frames of human global motion tracking result of a segment containing heavy traffic seq2 from a longer sequence. The notation is the same as Fig. 5. Object IDs are shown on the top to help the readers differentiate people with similar clothing.

TABLE 1
The Performance Evaluation of the Segmentation and Tracking System

Event	Count	Event	Count
passing-bys involving 2 humans	171	walking in group of 2	48
passing-bys involving 3 humans	31	walking in group of 3	13
passing-bys involving 4 humans	16	walking in group of 4	1
passing-bys involving 5 humans	5	passing obstacles	42

(a)

detection errors			tracking errors		
Type of error	Count	Percentage	Type of error	Count	Percentage
Missed-detection (complete,partial)	13,16	2.5%,3%	Track drift onto other human	25	5%
False alarm	4	0.8%	Track switch	2	0.4%
Redundant detection	10	1.9%	Track lost	5	1%

(b)

(a) The occurrence of different types of events which pose challenges to the system. (b) The occurrence and percentage (with regard to the number of people) of different types of errors.

We use the occurrence of typical errors of the system (in relation to the size of the dataset and the occurrence of challenging events) as metrics for evaluation. The errors are based on trajectories, which are more meaningful than frame-based errors for a tracking system. The errors are first

classified into *detection errors* and *tracking errors*. Detection errors include *missed-detection* (partial or complete), *false alarm*, and *redundant detection*. A partial or complete missed-detection means a human’s trajectory is partially or completely not detected during its presence in the scene. The short

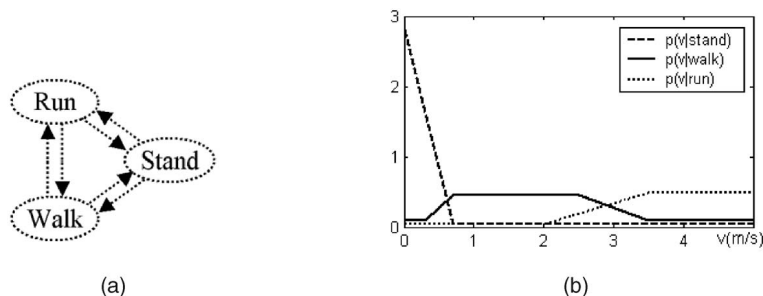


Fig. 7. (a) The first level of the locomotion model (modes). (b) The prior probability of speed given the locomotion modes.

delay right after a human entry is not counted as partial missed-detection. If a detection does not overlap with any human objects in the scene, we call it a false alarm; if multiple detections overlap with a human object, we call the most accurate one *correct detection* and the rest *redundant detections*. We separate redundant detections from false alarms because these two may have different influence in some applications. The tracking errors include *track drift onto other object*, *track switch between two objects*, and *other track lost*. The failures categories are by no means a complete classification, however, they cover most of the typical errors that have been observed.

The error counts are summarized in Table 1b, where we also derive the percentages with regard to the number of people to make the numbers more intuitive. These numbers should also be considered against the number of challenging events.

The main reason for missed-detection is change detection failure due to an object's lack of contrast with the background. It happens more often in side views since the projected human size is smaller than in the frontal/back views. Missed-detection also happens when a human is severely occluded by other humans. The number of false alarms is small since most of the image noise does not have shape and size comparable to a human. The main reason for redundant detection is the inclusion of nonhuman regions (e.g., soft shadows or bicycles) in the foreground. Incorporating shadow removal algorithms [33] should reduce redundant detections.

The typical scenario for the track of an object to drift on another object is when they have similar clothing and move closely and persistently. Since the object closer to the camera has priority for its position, it may erroneously shift onto the human at the back. Joint likelihood of all the overlapping objects and the background should be considered in future research. However, the increased computation due to the joint likelihood may limit the use of the system. Other track losses include the situations where the object exhibits fast change of velocity (biker or roller-skater).

5 A TRACKING AS RECOGNITION APPROACH TO BODY POSTURE ESTIMATION

Some applications require information of the body postures in addition to the positions. Three-dimensional human body posture is commonly represented by major joint angles in a kinematics model of over 20 degrees of freedom. A common formulation to estimate them from video sequences is to find the state at time $n + 1$ assuming the state at time n is known.

This formulation has a number of difficulties that make it sensitive to realistically noisy situations. First, the track may drift when the observations are ambiguous; such ambiguity is common when tracking a high dimension model using two-dimensional image observations. Second, the prior knowledge about the motion can only be used locally for prediction in tracking (e.g., [37]) and does not provide a global constraint. Finally, tracking requires knowledge of the initial state; this is a difficult problem and a user often needs to specify it manually.

These difficulties call for stronger prior knowledge on the motion being studied and stronger temporal integration. We propose here a *tracking as recognition* approach where the estimation of body postures is accomplished by recognition in a locomotion model over a period of time. By doing so, postures are inferred only in the prior motion model, which greatly constrains the solution space. Short-term ambiguities of measurements are resolved by considering all measurements together. Initial state is estimated in the same way as all the other frames. Besides, high-level descriptions (i.e., modes) are obtained concurrently. Mori and Malik [27] also used recognition to help tracking. It only involves recognition of body postures in a single frame, while we perform recognition in a motion (temporal) model.

We believe that our tracking as recognition approach is a general formulation and can also be applied to other kinds of motion, which can be described with 3D limb trajectories and for which a state-based motion model can be built. The object's global motion, if not fixed, needs to be available.

5.1 A Hierarchical Locomotion Model

Human locomotion has many *modes*, among which walking, running, and standing are the three most common ones. A human can switch between these modes. The modes are naturally represented as a finite state machine shown in Fig. 7a. The speed of the body is an important feature of these modes. The prior probabilities of the speed given the modes $P(v|m)$, $m \in \{walk, run, stand\}$ (Fig. 7b) are set according to previous research [2]. The transition probabilities between the modes are set according to their observed frequencies (we use 0.1 for the transitions to a different mode). These constitute the first level of the hierarchical locomotion model.

The second level of the model captures the limb motions (body postures) of walking and running. Walking and running are both periodic motions. We define a *cycle* to be the minimum repetitive unit, which equals two *steps*. Each cycle is represented by a number of characteristic *phases*. For

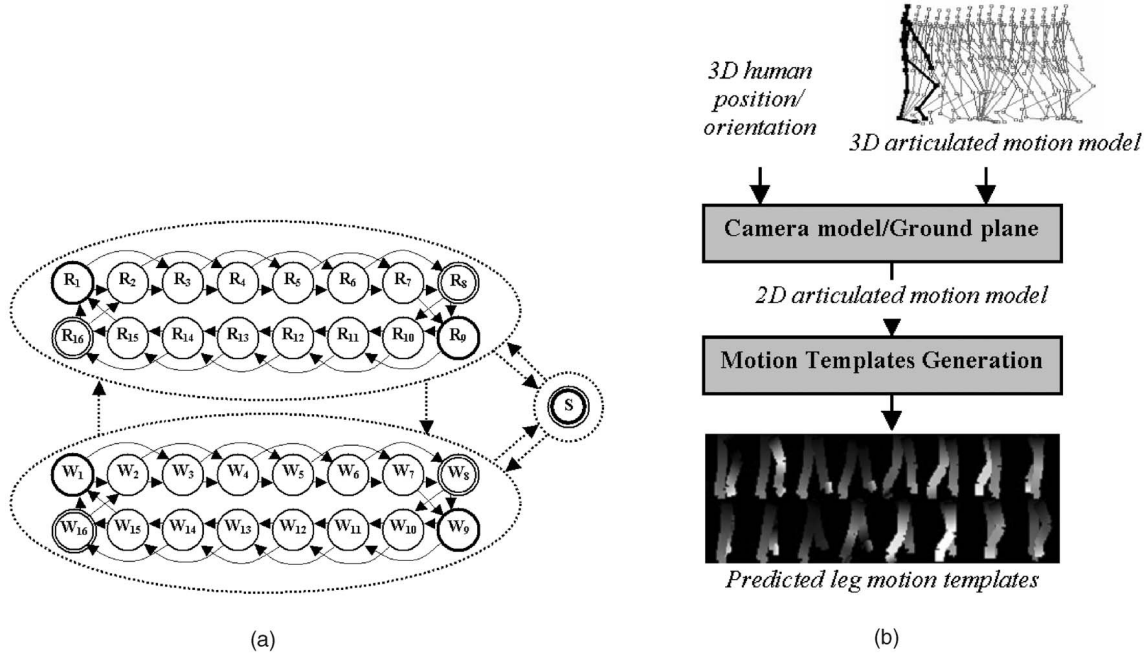


Fig. 8. (a) The second level of the locomotion model (phases); self-cycles omitted to save space, dark circle—starting state, double circle—ending state. (b) The diagram of computing model motion templates (showing the computation for R_1, \dots, R_{16} in example Fig. 10a).

each of the walking and running, several 3D motion capture sequences were gathered to compute an average. Three-dimensional motion capture data, consisting of a human kinematics (stick figure) model and a time series of joint angle values, is a concise representation of articulated motion and such data are now easily available (e.g., from [6]). The average cycle is uniformly quantized into 16 phases in time (W_1, \dots, W_{16} for walking and R_1, \dots, R_{16} for running, each being a state in the second-level state machine). See Fig. 8a. Each phase corresponds to a 3D body posture (top of Fig. 8b). The quantization levels are chosen considering the response property of the likelihood measurements.

A state has a link back to itself (i.e., a self-cycle, not shown in Fig. 8a for simplicity), a link to the state of the next phase and a link to the phase after the next (i.e., a bypass links, e.g., the link from R_1 to R_3) to handle fast motion. The start and end states in the low-level state machine are defined so that the transition between modes can only happen at the start and end of each step. The transition probabilities within each mode are set according to average walking/running cycle duration. In our implementation, the transition probabilities $A_{W_i, \overline{W}_{i+1}} = 0.3$, $A_{W_i, \overline{W}_{i+2}} = 0$, $A_{R_i, \overline{R}_{i+1}} = 0.4$, and $A_{R_i, \overline{R}_{i+2}} = 0.1$ (— means MOD 16). The transition probabilities between any two states are computed by combining the within-mode and between-mode transitions. The initial state probabilities are set to be uniform for all possible states.

5.2 Motion Template

A similarity measure for the states and the image observations is needed for inference. Here, we propose to use a *motion template* which is the template of model flow or image optical flow. It encodes both the shape and the motion information, and has advantages over other possibilities such as edges [35], foreground mask and moments of the foreground mask [3].

These static features are less discriminative since they do not contain motion information. Edges are sensitive to noises especially when the objects are small. Foreground mask and its moments are sensitive to the inclusion of background pixels. Moments are also sensitive to multiobject overlapping and blob fragmentation. Motion templates are more robust and their response fields are relatively wide so that fewer quantization steps (e.g., 16 for each locomotion mode) are needed. It is different from the Motion Energy Image (MEI) and Motion History Image (MHI) introduced in [5] which are templates of accumulated motion over a period of time. We choose to use only the motion of the legs since it is salient and stable compared to the motion of other parts (e.g., arms).

The motion templates are 2D models and, hence, are view-dependent. We can compensate for this dependency by generating the 2D models from the 3D locomotion model, given a camera model and the estimated 3D position and orientation of the human. We assume that the humans face forward in the direction of its motion. The stick figure model corresponding to each state is rescaled according to the human's physical height. Forward kinematics [28] is used to compute 3D positions of all joints and the camera model projects the 3D positions into 2D image positions. The motion vectors of the joint angles can be easily computed by differencing the neighboring phases; the motion vectors of the limb parts are computed by interpolation of the motion of the joints on the two ends. The process is shown in Fig. 8b. More details can be found in [47].

Thirty-two model motion templates (16 for walking and 16 for running) are generated this way in each frame. Image motion or optical flow is computed only for the foreground region using a block matching based optical flow algorithm (in [43]). The optical flow obtained is coarse; however, we find that it is sufficient for the estimation in the locomotion model. We define normalized template distance to be the

normalized sum of the vector differences of the two templates given by

$$D = \sum_{(x,y) \in \Omega} \frac{|\vec{v}_{xy} - \vec{m}_{xy}|}{\sqrt{|\vec{v}_{xy}| |\vec{m}_{xy}|}}, \quad (7)$$

where Ω is the area within the bounding box of both legs, \vec{v} is the image optical flow, and \vec{m} is the predicted model motion template computed in the previous paragraph. The similarities are computed for the 32 model motion templates and the image optical flow.

5.3 Inference in the Model

To estimate the modes and phases, we need to buffer the observations from frame 1 to frame T and to make inferences in the motion model to compute an optimal path which maximizes the likelihood of the observations—both body speed and the motion template responses.

Denote $V = \{v_1, \dots, v_T\}$ as the speed, $O = \{o_1, \dots, o_T\}$ as the motion template observations, and $Q = \{q_1, \dots, q_T\}$ as the states in each frame. q_t is further decomposed into $[m_t, p_t]$ in which m_t is the mode and p_t ($p_t \in \{1, \dots, 16\}$ for walking/running and $p_t = 1$ for standing) is the quantized phase in that state. We number q_t from 0 to 32 where states $0 \sim 15$ correspond to walking phases $W_1 \sim W_{16}$, states $16 \sim 31$ correspond to running phases $R_1 \sim R_{16}$, and state 32 is the standing state. $A_{i,j} = P(q_t = j | q_{t-1} = i)$ is the state transition probability and we use $A_{0,i}$ to denote the initial state distribution for simplicity of notation. The normalized distance of motion template $[m_t, p_t]$ with the image optical flow is denoted as $D_t^{[m_t, p_t]}$. The optimal path is given by $Q^* = \text{argmax} P(O, V | Q)$.

Because the speed and the motion template correlation (7) reflect global motion and limb motion, respectively, we assume that they are conditionally independent given the state q_t . We have

$$\begin{aligned} P(O, V | Q) &= \prod_{t=1}^T A_{q_{t-1}, q_t} P(o_t, v_t | q_t) \\ &= \prod_{t=1}^T A_{q_{t-1}, q_t} P(o_t | q_t) P(v_t | q_t) \\ &= \prod_{t=1}^T A_{q_{t-1}, q_t} P(o_t | [m_t, p_t]) P(v_t | m_t). \end{aligned} \quad (8)$$

By assuming a Gaussian noise model ($\mathcal{N}(0, \sigma^2)$, where σ is the standard deviation; it is identical for all states) for the model motion templates, the likelihood of walking/running motion templates is given by

$$P(o_t | [m_t, p_t]) \propto \exp \left\{ -\frac{D_t^{[m_t, p_t]^2}}{2\sigma^2} \right\}.$$

$P(o_t | m_t = \text{stand})$ is fixed to 1.

Computing the optimal path is similar to the problem of finding the best state sequence in an HMM. We employ the Viterbi algorithm [8] for inference. The resulting Q^* gives both the high-level modes and the detailed phases which corresponds to body postures of each frame.

5.4 Optional Postprocessing

In many cases, the phases of walking or running change approximately linearly with regards to time. This linear relationship can be enforced in a postprocessing stage and be used to deal with temporary missing or bad measurements due to occlusion and noise. For each segment of walking or running, the linear relationship is written as $p_t = (p'_t)_{\text{MOD}16}$ where $p'_t = k * t + b$. k is the phase speed and b is the initial phase.

Direct estimation of parameters k and b is not straightforward due to the nonlinear MOD operation. The structure of the circular state machine constrains the state to only move forward and to move at most one or two steps at a time. This property enables us to compute the p'_t before the MOD operation as $p'_t = p_t + n * 16$. n is initialized to zero and is increased by one whenever p_t is less than p_{t-1} (e.g., $p_{t-1} = W_{16}$ and $p_t = W_1$). Then, \hat{k} and \hat{b} are estimated from $\{p'_t, t\}$ pairs using LMS fitting. $\hat{p}_t = (\hat{k} * t + \hat{b})_{\text{MOD}16}$ is recomputed and the posture is interpolated by the postures corresponding to $\lfloor \hat{p}_t \rfloor$ and $\lceil \hat{p}_t \rceil$.

5.5 Experiment Results

5.5.1 Locomotion Results

We have tested the proposed method on a variety of datasets and found it to work very robustly despite the difficulties of small image size, change of motion direction, and temporary occlusion.

Figs. 9a, 9b, and 9c shows the result of a 477-frame sequence (seq3) containing a human walking, running, and standing. There are significant changes in orientation and size, and the trajectory is not linear. The observations of all the frames are used to estimate an optimal path shown in Fig. 9a. Fig. 9b shows the states after the postprocessing. The 3D human postures corresponding to the states are overlaid on the original frames in Fig. 9c (before postprocessing). The images (and the video) show that the estimated motion parameters are accurate and the transitions between modes are natural. The only inaccuracy occurs around frame 140 when the human's direction is almost aligned with the camera optical axis and the walking speed is low so that the motion of the legs is not very salient. This inaccuracy is resolved in the postprocessing step. One deficiency of our method is that the human orientation when standing is inferred poorly from its speed, which is close to zero. Fig. 9d shows an example (seq4) with two humans passing by each other. The result is not affected by the severe occlusion and the noisy blob due to the low contrast of one's pants to the background (Fig. 1c). We also show an example from the USF/NIST gait data set [32] in which a human walks along an elliptical path (Fig. 9e), with emphasis on the fast turn of the walker. Due to the nature of the state-based representation, the estimated postures are inherently of limited precision.

Besides the batch processing, we have also implemented an online system in which a sliding window of most recent observation of 50 frames is used. Compared to use of all frames, the sliding window occasionally does not show temporal coherence in the state estimates of adjacent frames when the observation is ambiguous. Specifically, the transition from p_t (based on observation from $t - 50$ to t)

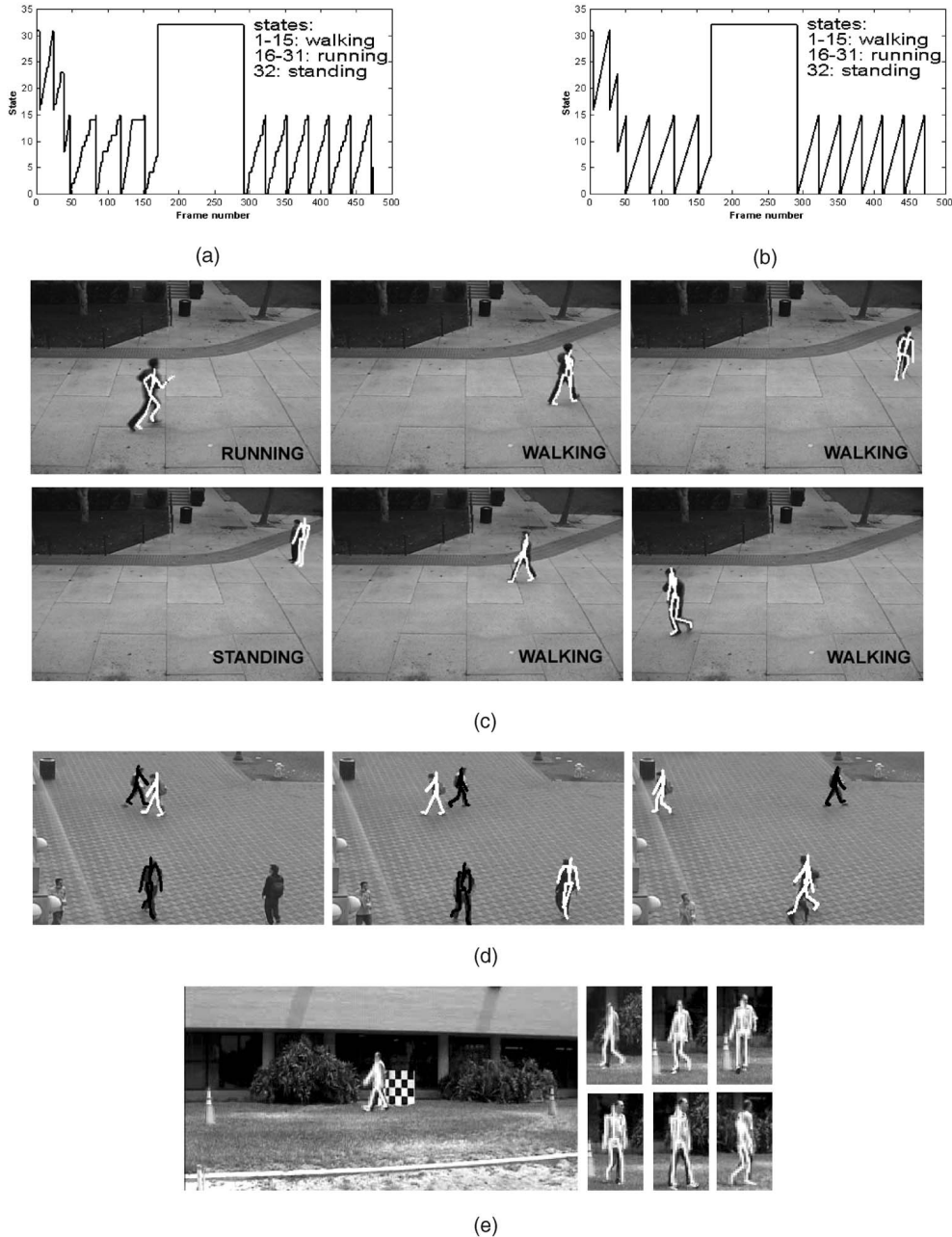


Fig. 9. Result of estimating locomotion modes and phases. Seq3: state output before (a) and after (b) postprocessing, and selected frames of the estimated modes and phases (overlaid stick figures) (c); (d) selected frames of the result on Seq4; (e) selected frames of result on sequence NSF/NIST, focusing on the turn at the left end.

to p_{t+1} (based on observation from $t - 49$ to $t + 1$) may not be valid.

The program is implemented in unoptimized C++ code and runs at about 8 Hz on a Pentium 4 2.7G Hz PC. The main computation effort is spent on computing image optical flow and generating model motion templates. As a trade off for computation, a simpler but also less powerful observation model (e.g., shape, edges, etc) could be used.

5.5.2 Hypothesis Verification by Walking

In Section 4.2, we described human hypothesis formation by shape analysis of the foreground blobs. The hypotheses may contain false alarms (e.g., Fig. 3i). The variation of human appearance makes static frame analysis difficult.

Instead, dynamical features can provide much more robust information. The above technique can be used to verify the hypotheses.

We assume that humans enter the scene walking. We only use the part of the locomotion model involved in walking (e.g., $W_1 \sim W_{16}$) in Fig. 7. We set the verification process to execute for T frames ($T = 40; \sim 1.3sec$), which is the average length of a walking cycle. The inference and postprocessing operations are performed. The phase speed k of a valid hypothesis matches its physical speed and the summed correlation value is high. The example in Fig. 10a shows the verification of two hypotheses—a human and his reflection. The computed phases match the image very well for the valid hypothesis, while the phase speed (close to 0) does not match

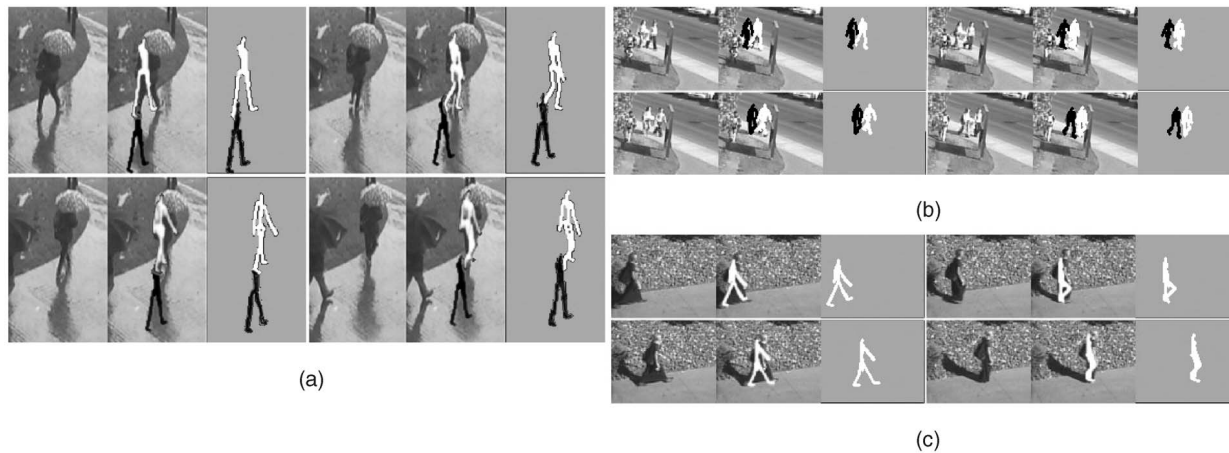


Fig. 10. Selected frames of human verification examples. (a) A real human was verified while his reflection was not; (b), (c) other verified examples.

the physical speed for the invalid one. Two more examples are shown in Fig. 10. Fig. 10b is an example of two close-by walkers. The human height is about 25 pixels in the image and there are frames in which the walkers are occluded by the map board. Fig. 10c shows an example of a woman in long dress where the legs are not visible. In both examples, the phase alignments are very accurate.

We have tested the verification on a number of sequences that contain 45 distinct people. Before the walking verification, there are 12 false alarms and 1 missed detection. The verification cuts down the number of false alarms to 2, however, it also rejects an extra five real humans. This happens mostly when the humans walk towards or away from the view direction so that the leg motion is not as salient as when walking in other directions.

6 CONCLUSION AND FUTURE WORK

We have described our methods for segmentation and tracking of multiple humans in complex situations and estimation of human locomotion modes and phases (coarse 3D body postures). We use explicit 3D shape model in segmentation and tracking of multiple humans. The shape model enables segmenting multiple-human with persistent occlusion (e.g., walking together) and provides a representation for tracking. Three-dimensional model combined with camera model and the assumption that people move on a ground plane makes the approach suitable for a wide range of viewpoints, automatically scales the model as people moves, facilitates occlusion reasoning, and provides 3D trajectories. Human body postures are inferred in a 3D locomotion model over a period of time. The prior model and the temporal integration help resolve the ambiguities in the estimation and contribute to robust results. Promising results are obtained in the presence of shadow, reflection, temporarily severe occlusion, and persistent occlusion.

There are a few interesting directions to be explored in the future. A joint likelihood might be needed in segmentation and tracking of more overlapping objects. Since a joint state increases the dimensionality of the search space, affordable computation needs to be investigated. The coarse 3D body postures obtained could be refined for more

detailed analysis (e.g., gait analysis). Motion parameters and body parameters can be optimized locally to best fit the images. Having the initial states recovered from a model of global structure, the optimization is more likely to converge to the global optimum.

ACKNOWLEDGMENTS

This work was done in USC/IRIS. This research was supported, in part, by the Advanced Research and Development Activity of the US Government under contract No. MDA-908-00-C-0036.

REFERENCES

- [1] A.M. Baumberg, "Learning Deformable Models for Tracking Human Motion," PhD thesis, Univ. of Leeds, 1995.
- [2] G.A. Bekey, "Walking," *The Handbook of Brain Theory and Neural Networks*, M.A. Arbib, ed., MIT press, 1995.
- [3] M. Brand, "Shadow Puppetry," *Proc. Int'l Conf. Computer Vision*, vol. 2, pp. 1237-1244, 1999.
- [4] C. Bregler, "Learning and Recognizing Human Dynamics in Video Sequences," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 568-574, 1997.
- [5] A.F. Bobick and J.W. Davis, "The Recognition of Human Movement Using Temporal Templates," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 3, Mar. 2001.
- [6] Character Studio: Software Package, <http://www.discreet.com/products/cs/>, 2002.
- [7] I. Cohen and G. Medioni, "Detecting and Tracking Moving Objects for Video Surveillance," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 319-325, 1999.
- [8] R. Cutler and L.S. Davis, "Robust Real-Time Periodic Motion Detection, Analysis, and Applications," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, Aug. 2001.
- [9] J. Deutscher, A. Davison, and I. Reid, "Automatic Partitioning of High Dimensional Search Spaces Associated with Articulated Body Motion Capture," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 669-676, 2001.
- [10] A.A. Efros, A.C. Berg, G. Mori, and J. Malik, "Recognizing Action at a Distance," *Proc. IEEE Int'l Conf. Computer Vision*, pp. 726-733, 2003.
- [11] A.M. Elgammal and L.S. Davis, "Probabilistic Framework for Segmenting People under Occlusion," *Proc. Int'l Conf. Computer Vision*, vol. 1, pp. 145-152, 2001.
- [12] D. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*. Prentice-Hall, 2001.
- [13] S. Hongeng and R. Nevatia, "Multi-Agent Event Recognition," *Proc. Int'l Conf. Computer Vision*, vol. 2, pp. 84-91, 2001.

- [14] I. Haritaoglu, D. Harwood, and L.S. Davis, "W4S: A Real-Time System for Detecting and Tracking People in 2 1/2 D," *Proc. European Conf. Computer Vision*, pp. 962-968, 1998.
- [15] S. Haritaoglu, D. Harwood, and L.S. Davis, "W4: Real-Time Surveillance of People and Their Activities," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, Aug. 2000.
- [16] R. Hartley and A. Zisserman, *Multi View Geometry*. Cambridge Press, 2000.
- [17] M. Isard and A. Blake, "Condensation-Conditional Density Propagation for Visual Tracking," *Int'l J. Computer Vision*, vol. 29, no. 1, pp. 5-28, 1998.
- [18] M. Isard and J. MacCormick, "BraMBLE: A Bayesian Multiple-Blob Tracker," *Proc. Int'l Conf. Computer Vision*, vol. 2, pp. 34-41, 2001.
- [19] R. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *J. Basic Eng.*, vol. 82, pp. 35-45, 1960.
- [20] P. Kornprobst and G. Medioni, "Tracking Segmented Objects Using Tensor Voting," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 118-125, 2000.
- [21] N. Krahnstover, M. Yeasin, and R. Sharma, "Towards a Unified Framework for Tracking and Analysis of Human Motion," *Proc. IEEE Workshop Detection and Recognition of Events in Video*, 2001.
- [22] D. Liebowitz, A. Criminisi, and A. Zisserman, "Creating Architectural Models from Images," *Proc. EUROGRAPH Conf.*, vol. 18, pp. 39-50, 1999.
- [23] A.J. Lipton, H. Fujiyoshi, and R.S. Patil, "Moving Target Classification and Tracking from Real-Time Video," *Proc. DARPA IU Workshop*, pp. 129-136, 1998.
- [24] F. Lv, T. Zhao, and R. Nevatia, "Self-Calibration of a Camera from a Walking Human," *Proc. Int'l Conf. Pattern Recognition*, vol. 1, pp. 562-567, 2002.
- [25] S.J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler, "Tracking Groups of People," *Computer Vision and Image Understanding*, vol. 80, no. 1, pp. 42-56, 2000.
- [26] T.B. Moeslund and E. Granum, "A Survey of Computer Vision-Based Human Motion Capture," *Computer Vision and Image Understanding*, vol. 81, pp. 231-268, 2001.
- [27] G. Mori and J. Malik, "Estimating Human Body Configurations Using Shape Context Matching," *Proc. European Conf. Computer Vision*, pp. 666-681, 2002.
- [28] R. Murry, Z.X. Li, and S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.
- [29] NOVAS—Naval Observatory Vector Astrometry Subroutines, http://aa.usno.navy.mil/software/novas/novas_info.html, 2003.
- [30] Data Set Provided by IEEE Workshop on Performance Evaluation of Tracking and Surveillance (PETS2001), 2001.
- [31] S. Pingali and J. Segen, "Performance Evaluation of People Tracking Systems," *Proc. Third IEEE Workshop Applications of Computer Vision*, pp. 33-38, 1996.
- [32] P.J. Phillips, S. Sarkar, I. Robledo, P. Grother, and K.W. Bowyer, "The Gait Identification Challenge Problem: Data Sets and Baseline Algorithm," *Proc. Int'l Conf. Pattern Recognition*, pp. 385-388, 2002.
- [33] A. Prati, R. Cucchiara, I. Mikic, and M.M. Trivedi, "Analysis and Detection of Shadows in Video Streams: A Comparative Evaluation," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 571-576, 2001.
- [34] L.R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proc. IEEE*, vol. 77, no. 2, 1989.
- [35] K. Rohr, "Towards Model-Based Recognition of Human Movements in Image Sequences," *CVGIP: Image Understanding*, vol. 59, no. 1, pp. 94-115, 1994.
- [36] R. Rosales and S. Sclaroff, "3D Trajectory Recovery for Tracking Multiple Objects and Trajectory Guided Recognition of Actions," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 117-123, 1999.
- [37] H. Sidenbladh, M.J. Black, and D.J. Fleet, "Stochastic Tracking of 3D Human Figures Using 2D Image Motion," *Proc. European Conf. Computer Vision*, pp. 702-718, 2000.
- [38] N.T. Siebel and S. Maybank, "Fusion of Multiple Tracking Algorithm for Robust People Tracking," *Proc. European Conf. Computer Vision*, pp. 373-387, 2002.
- [39] Y. Song, X. Feng, and P. Perona, "Towards Detection of Human Motion," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 810-817, 2000.
- [40] C. Stauffer and W.E.L. Grimson, "Learning Patterns of Activity Using Real-Time Tracking," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, Aug. 2000.
- [41] H. Tao, H.S. Sawhney, and R. Kumar, "A Sampling Algorithm for Tracking Multiple Objects," *Proc. IEEE Workshop Vision Algorithms*, 1999.
- [42] H. Tao, H.S. Sawhney, and R. Kumar, "Object Tracking with Bayesian Estimation of Dynamic Layer Representations," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, Jan. 2002.
- [43] A.M. Tekalp, *Digital Video Processing*. Prentice Hall, 1995.
- [44] C.R. Wren, A. Azarbayejani, T. Darrell, and A.P. Pentland, "Pfinder: Real-Time Tracking of the Human Body," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, July 1997.
- [45] T. Zhao, R. Nevatia, and F. Lv, "Segmentation and Tracking of Multiple Humans in Complex Situations," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 194-201, 2001.
- [46] T. Zhao and R. Nevatia, "3D Tracking of Human Locomotion: A Tracking as Recognition Approach," *Proc. Int'l Conf. Pattern Recognition*, vol. 1, pp. 546-551, 2002.
- [47] T. Zhao, "Model-Based Segmentation and Tracking of Multiple Humans in Complex Situations," PhD thesis, Univ. of Southern California, Los Angeles, 2003.



pattern recognition. He is a member of the IEEE and IEEE computer society.



to several others. He has been a regular contributor to the literature in computer vision. His research interests include computer vision, artificial intelligence, and robotics. Dr. Nevatia is a fellow of the American Association for Artificial Intelligence and a member of the ACM. He is an associate editor of *Pattern Recognition* and *Computer Vision and Image Understanding*. He has served as an associate editor of the *IEEE Transactions on Pattern Analysis and Machine Intelligence* and as a technical editor in the areas of robot vision and inspection systems for the *IEEE Journal of Robotics and Automation*. He also served as a co-general chair of the IEEE Conference on Computer Vision and Pattern Recognition in June 1997. He is a fellow of the IEEE and member of the IEEE Computer Society.

► For more information on this or any computing topic, please visit our Digital Library at www.computer.org/publications/dlib.