Integrated Tracking and Recognition of Human Activities in Shape Space

Bi Song, Amit K. Roy-Chowdhury, and N. Vaswani

University of California, Riverside, USA and
Iowa State University, USA

Abstract. Activity recognition consists of two fundamental tasks: tracking the features/objects of interest, and recognizing the activities. In this paper, we show that these two tasks can be integrated within the framework of a dynamical feedback system. In our proposed method, the recognized activity is continuously adapted based on the output of the tracking algorithm, which in turn is driven by the identity of the recognized activity. A non-linear, non-stationary stochastic dynamical model on the "shape" of the objects participating in the activities is used to represent their motion, and forms the basis of the tracking algorithm. The tracked observations are used to recognize the activities by comparing against a prior database. Measures designed to evaluate the performance of the tracking algorithm serve as a feedback signal. The method is able to automatically detect changes and switch between activities happening one after another, which is akin to segmenting a long sequence into homogeneous parts. The entire process of tracking, recognition, change detection and model switching happens recursively as new video frames become available. We demonstrate the effectiveness of the method on real-life video and analyze its performance based on such metrics as detection delay and false alarm.

1 Introduction

The problem of event analysis from video consists of the related issues of recognizing different activities and keeping track of the objects participating in the activities. In many practical applications, all we have is a video sequence consisting of a number of activities, and we have to track, *as well as* recognize, the various events taking place in the video. Often we have time critical applications where the option of first completing the tracking and then recognizing is not available. Thus it is important to design methods that can *simultaneously* track and recognize a sequence of human activities from a video sequence.

In this paper, we present a novel framework for *integrated* tracking and recognition of human activities consisting of the following steps which take place in a loop: (i) modeling the appearance and motion of single activity sequences and tracking them, (ii) detecting a change from one sequence to the next, and (iii) classifying which is the next activity to change to and start tracking it. This is achieved in a recursive manner as new video frames become available. Human activities are represented by non-linear, non-stationary dynamical models,

P. Kalra and S. Peleg (Eds.): ICVGIP 2006, LNCS 4338, pp. 468–479, 2006.

learned from training data. These models represent the change in the *shape* of the human body in the course of the activity. Given a video sequence, the model parameters are used for recognition, with the recognized parameters then driving the tracking algorithm. The method is able to automatically detect changes from one activity to another and switch accordingly. Switching between models occurs when the tracking error [1], which serves as a feedback signal, exceeds a certain threshold. Thus our proposed system is able to persistently track and recognize a *sequence* of multiple activities. A diagrammatic representation of this framework is shown in Fig. 1. We present experimental results on real life video of different activities and analyze the issues of recognition delay and tracking accuracy.

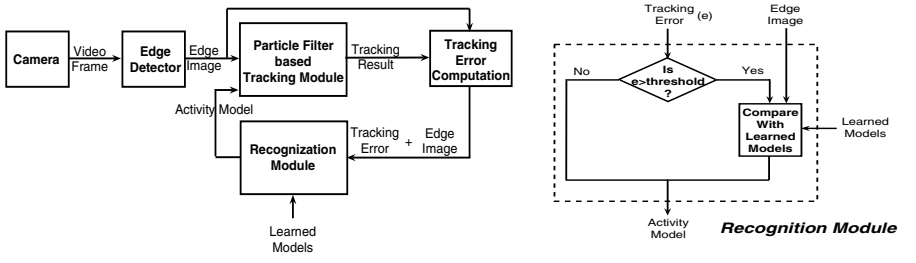


Fig. 1. (a): Framework of dynamical feedback system for simultaneous tracking and recognition. (b): Recognition module incorporating change detection and model switching.

1.1 Relation to Previous Work

A review of recent work is given in [7]. Based on the conclusions of [7], we find that most existing methods handle events of short duration with moderate changes in illumination, scene clutter and occlusion. In most video surveillance methods, the tracks are obtained first followed by recognition [10,5,18]. Integrated tracking and recognition is very promising becasue of its ability to track and recognize activities in a long video seuqnece, where switching between different activities will usually occur.

A few techniques have studied the problem of simultaneous tracking and recognition, though not always in the context of activity recognition. In [20,21], the authors presented methods whereby the identity of a person, based on face recognition, is obtained after tracking the face over the whole sequence. However, the identity of a face in a video sequence is a static parameter which can be estimated by integrating over the entire sequence, whereas activities are inherently dynamic and hence the recognition needs to evolve in time. In [15,13], the idea of integrated tracking and recognition of activities was proposed. However, their method requires a-priori knowledge of the transition probability matrix for switching between different activity models. While this is feasible in some

applications, designing such a transition matrix for uncontrolled environments like video surveillance may be difficult (since the class of possible activities is large and can change arbitrarily from one to another) and the process has to repeat for each application scenario. In contrast to this open-loop approach, we propose to use change detection measures to detect slow or sudden transitions between activities, and use these as a feedback signal in a closed-loop system.

Simultaneous tracking and recognition was also the theme in [6], but here the authors used color and depth information to create a "plan-view" map based on which tracking is done, and activity recognition was carried out using pose estimates; they did not consider the dynamics inherent in any activity. Simultaneous tracking of moving people and recognition of their activities has been performed in many applications using a Dynamic Bayesian Network (DBN) model tracked by a Rao-Blackwellized particle filter [11,3,2]. In [2], the authors perform figure tracking by defining a DBN to switch between various linear dynamical systems (also called Switched Linear Dynamical System (SLDS)). However, these methods also require knowledge of a state transition pdf for a seqeuence of changes, which implies learning what sequences are likely to occur. Key-frame segmentation methods [19] can achieve some of the goals of this research (i.e., find the switching instances), but they usually require the entire video to be available a-priori rather than simultaneously tracking, recognizing and detecting changes.

We use a discrete shape representation of the human body which is different from level set representations of shapes such as those described in [14,9,12]. The level set approach is theoretically infinite (and in practice large time varying finite) dimensional, and hence defining dynamics on and sampling from such a large dimensional space is computationally expensive. This is overcome through the use of various approximate tracking solutions. Level sets, however, have the advantage that they can adjust to large changes in the shape and topology, which is usually problematic for discrete representations. For large changes in shape, we show that it is possible to overcome this problem for many activity recognition applications by using a piecewise stationary dynamical model. We do not encounter topology changes in our application. Moreover, a discrete representation allows adoption of the overall framework to different descriptions of human body structure, like stick figures, cylindrical models, etc.

2 State Space Model for Shape Dynamics

We model the motion/deformation of a deforming shape as scaled Euclidean motion of a "mean shape" (i.e., translation, rotation, isotropic scaling) plus its non-rigid deformation. The term "shape activity" is used to denote a particular stochastic model for shape deformation. We define a "stationary shape activity" (SSA) as one for which the mean shape remains constant with time and the deformation model is stationary. We define a piecewise stationary shape activity (PSSA) model [17] as one that models a shape activity with slowly varying "mean shape" (approximated as piecewise constant). The SSA model is accurate for activities where the shape of the body does not change significantly in the

course of the activity. The PSSA model deals with the case where the shape changes appreciably in the course of the activity. This allows us to handle large shape deformation using a discrete shape descriptor.

2.1 Shape Representation

We briefly review Kendall's statistical shape theory, details of which can be found in [4]. We use a discrete representation of shape for a group of k **landmarks**. The **configuration** is the set of landmarks: in the 2D case it is the x and y coordinates of the landmarks which can be represented as a k dimensional complex vector, Y_{raw}. This raw configuration can be normalized for translation and then for scale to yield the **pre-shape**, denoted by w. A configuration of k points after translation normalization, denoted by Y, lies in $\mathcal{C}^{k-1}$ (a (k-1)-dimensional complex space), while the pre-shape, w, lies on a hyper-sphere in $\mathcal{C}^{k-1}$. A pre-shape w_1 can be aligned with another pre-shape w_0 by finding the rotation angle for the best fit (minimum mean square error fit) and this gives the **Procrustes fit** of w_1 onto w_0. This is the **shape** of w_1 with respect to w_0. The **Procrustes distance** between preshapes w_1 and w_0 is the Euclidean distance between the Procrustes fit of w_1 onto w_0. The **Procrustes mean** of a set of preshapes $\{w_i\}$ is the minimizer of the sum of squares of Procrustes distances from each w_i to an unknown unit size mean configuration μ. Any pre-shape of the set can then be aligned with respect to this Procrustes mean to return the **shape** (denoted by z) with respect to the mean shape, μ.

The shape space, $\mathcal{M}$, is a manifold in $\mathcal{C}^{k-1}$ and hence its actual dimension is $\mathcal{C}^{k-2}$. Thus the tangent plane at any point of the shape space is a $\mathcal{C}^{k-2}$ dimensional hyperplane in $\mathcal{C}^k$. The tangent coordinate (denoted by v) with respect to μ, of a configuration, Y_{raw}, is evaluated as follows:

$$
\begin{aligned}
Y &= CY_{raw}, \quad \text{where } C \triangleq I_k - 1_k 1_k^T/k \\
s &\triangleq s(Y) = ||Y||, \quad w = Y/s, \\
\theta &\triangleq \theta(Y,\mu) = -\arg(w^T\mu), \qquad z(Y,\mu) = we^{j\theta}, \qquad (1) \\
v &\triangleq v(Y,\mu) = [I_k - \mu\mu^T]z = [I_k - \mu\mu^T]\frac{Ye^{j\theta}}{s}. \qquad (2)
\end{aligned}
$$

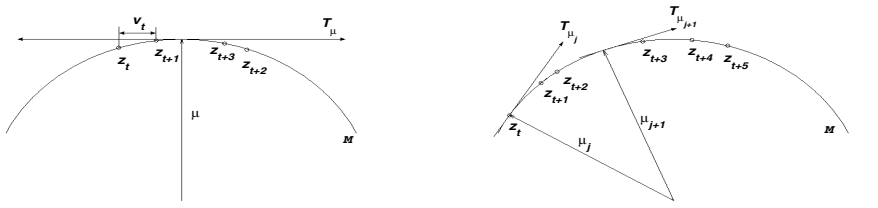
s is the scale of the centered configuration and θ is the rotation of the scaled configuration with respect to the mean shape.

The inverse mapping of (2) (tangent space to centered configuration space) is:

$$
\begin{aligned}
z(v,\mu) &= (1 - v^*v)^{1/2}\mu + v, \qquad (3) \\
Y(v,\theta,s,\mu) &= zse^{-j\theta} = [(1 - v^*v)^{1/2}\mu + v]se^{-j\theta}.
\end{aligned}
$$

2.2 System Model

The observed configuration of landmarks, in a single frame at time t, after translation normalization, is defined by Y_t, and forms the observation vector. Let μ_t



(a) Stationary Shape Sequence (SSS) (b) Piecewise Stationary Shape Sequence (PSSS)

Fig. 2. Stationary and "Piecewise-Stationary" Shape Sequences on the shape manifold which is depicted using a circle ($\mathcal{M}$), instead of a complex $\mathcal{C}^{k-1}$ sphere. In (a), we show a stationary sequence of shapes; at all times the shapes are close to the mean shape and hence the dynamics can be approximated in T_μ (tangent space at μ). In (b), we show a piecewise-stationary sequence of shapes; the shapes move on the shape manifold.

denote mean shape associated with this frame. Denote the tangent space at μ_t by T_{μ_t}. Since the tangent plane is a $(k-2)$-dim hyperplane in $\mathcal{C}^k$, a tangent vector has only $(k-2)$ independent (complex) coefficients. We perform an SVD (Singular Value Decomposition) of the tangent projection matrix, $[I_k - \mu_t \mu_t^T]C$ (from (1), to obtain a $(k-2)$-dim orthogonal basis for T_{μ_t}. The basis vectors of the SVD, $\{\underline{u}_{t,i}\}_{i=1}^{k-2}$, are arranged as column vectors of a matrix, $U_t(\mu_t)$, i.e., ${U_t}^{k\times(k-2)} = [\underline{u}_{t,1}, \underline{u}_{t,2} ... \underline{u}_{t,k-2}]$. The vector of coefficients ($(k-2)$-dim) along these basis directions, $c_t(z_t, \mu_t)$, is thus a canonical representation of the tangent coordinate of z_t in T_{μ_t}. The tangent coordinate is given by $v_t(z_t, \mu_t) = U_t c_t$. The coefficients vector of the tangent coordinate of shape with respect to the current mean shape, c_t, and the motion parameters (scale s_t, rotation θ_t) form the state vector, i.e., $X_t = [c_t, s_t, \theta_t]$.

For a stationary shape activity, the "mean shape" is constant with time, i.e., $\mu_t = \mu_0$, and the shape sequence is clustered around the "mean shape" (see figure 2(a)). Hence the shape deformation dynamics can be defined in a single tangent space at the mean (which can be learnt as the Procrustes mean of the training data). The dynamics on c_t is defined by the autoregression model, $c_t = A_c c_{t-1} + n_t$.

PSSA Model for Shape Deformation. When the shape is not stationary but is slowly varying, one could model the "mean shape" as being piecewise constant [17]. Thus unlike SSA, the dynamics can be described in a single tangent space. Let the "mean shape" change times be $t_{\mu_1}, t_{\mu_2}, t_{\mu_3}, ...$ and the corresponding means be $\mu_1, \mu_2, \mu_3, ...$ Then we have the following dynamics: between $t_{\mu_{j-1}} \le t < t_{\mu_j}$, $\mu_t = \mu_{j-1}$ and so $c_{t-1}(z_{t-1}, \mu_t) = c_{t-1}(z_{t-1}, \mu_{j-1})$. Hence in this interval, the dynamics is similar to that for an SSA, i.e.,

$$\begin{aligned} c_t(z_t, \mu_{j-1}) &= A_{c,j-1} c_{t-1}(z_{t-1}, \mu_{j-1}) + n_t,\ n_t \sim \mathcal{N}(0, \Sigma_{c,t}) \\ v_t &= U(\mu_{j-1}) c_t, \\ z_t &= (1 - v_t^* v_t)^{1/2} \mu_{j-1} + v_t. \quad \text{(from (3))} \end{aligned} \tag{4}$$

At the change time instant, $t = t_{\mu_j}$, $\mu_t = \mu_j$ and so the tangent coefficient c_{t-1} needs to be recalculated in the new tangent space with respect to $\mu_t = \mu_j$. This is achieved as follows[17]:

$$\begin{aligned} c_{t-1}(z_{t-1}, \mu_{j-1}) &= U(\mu_t)^* z_{t-1} e^{j\theta(z_{t-1}, \mu_{j-1})} \\ c_t(z_t, \mu_{j-1}) &= A_{c,j} c_{t-1}(z_{t-1}, \mu_{j-1}) + n_t, \\ v_t &= U(\mu_{j-1}) c_t, \\ z_t &= (1 - v_t^* v_t)^{1/2} \mu_{j-1} + v_t. \end{aligned} \tag{5}$$

Global Motion Dynamics. We use the same global motion model as in [18] to represent the Euclidean motion of the mean shape. We use a Gauss-Markov model for log-scale, $\log s_t$, and a Markov uniform model for θ_t, i.e.,

$$\begin{aligned} \log s_t &= \alpha_s \log s_{t-1} + (1 - \alpha_s)\mu_s + n_{s,t} \\ \log s_0 &\sim \mathcal{N}(\mu_s, \sigma_s^2), \quad n_{s,t} \sim \mathcal{N}(0, \sigma_r^2) \\ \theta_t &= \alpha_\theta \theta_{t-1} + n_{\theta,t}, \quad n_{\theta,t} \sim Unif(-a, a) \end{aligned} \tag{6}$$

Training. Given a training sequence of centered (translation normalized) configurations, $\{Y_t\}_{t=1}^T$, for a particular activity, we first evaluate $\{c_t, v_t, s_t, \theta_t\}_{t=1}^T$ for each stationary sub-model (i.e., $t_{\mu_{j-1}} \leq t < t_{\mu_j}$) as follows [1] :

$$\begin{aligned} \mu_{j-1} &= \text{Procrustes mean of } Y_t, t_{\mu_{j-1}} \leq t < t_{\mu_j} \\ s_t &= ||Y_t||, \quad w_t = Y_t / s_t, \\ \theta_t(Y_t, \mu_{j-1}) &= -angle(w_t^T \mu_{j-1}), \quad z_t(Y_t, \mu_{j-1}) = w_t e^{j\theta_t}, \\ v_t(Y_t, \mu_{j-1}) &= [I_k - z_{t-1} z_{t-1}^T] z_t, \\ c_t(Y_t, \mu_{j-1}) &= U_t(z_{t-1})^T z_t. \end{aligned} \tag{7}$$

If we assume a time invariant Markov model on c_t, we can use $\{c_t\}_{t=1}^T$ to learn its parameters [18].

2.3 Observation Model

In practice, the landmarks are not easy to extract directly from a given image, while an edge image is convenient to obtain by edge detection algorithms (e.g. Canny detector). Our observation is the edge image, $G_t = \Upsilon(I_t)$, (where Υ denotes the edge extraction operator) and I_t is the image at t. The observation likelihood describes the probability of a set of landmark points, Γ_t, on the edge image with $\Gamma_t \subset G_t$, given the predicted state vector, X_t. Let $\hat{Y}_t = h(X_t) = s_t z_t e^{-j\theta_t}$ be the predicted configuration of landmarks. It is assumed that a mapping, f, is known that associates each predicted landmark of $\hat{Y}_t$ with a point on the edges.

[1] Note, the last equation, $c_t = {U_t}^T z_t$, holds because $c_t = {U_t}^T v_t = {U_t}^T [I - \mu_{j-1}\mu_{j-1}^T] z_t = {U_t}^T [I - \mu_{j-1}\mu_{j-1}^T] C z_t = {U_t}^T U_t {U_t}^T z_t = {U_t}^T z_t$.

In practice this mapping is set up by searching for the closest edge along the normal of the predicted configuration (as in [8]) and this is treated as the observed landmark, Γ_t. Thus the observation likelihood is

$$p(\Gamma_t|X_t) \propto exp\{-\sum_{k=1}^{K} \frac{1}{2r_k K}||q_k - f(q_k, G_t)||^2\}, \tag{8}$$

where K is the shape vector dimension, r_k is the variance of the k^{th} component, q_k is the k^{th} predicted landmark, i.e., $q_k = \hat{Y}_{t,k}$ and $f(q_k, G_t) = \Gamma_t$ is the nearest edge point of q_k along its norm direction.

3 Tracking, Change Point Detection and Recognition

3.1 Tracking Using Particle Filters

In this paper, we use a particle filter for "tracking", i.e., for obtaining observations on the fly by tracing along the normals of the predicted configuration, $\hat{Y}_t$, to search for the closest edge (as described in Section 2.3). The particle filter is a sequential Monte Carlo method (sequential importance sampling plus resampling) which provides at each t, an N sample Monte Carlo approximation to the prediction distribution, $\pi_{t|t-1}(dx) = Pr(X_t \in dx|Y_{1:t-1})$, which is used to search for new observed landmarks. These are then used to update $\pi_{t|t-1}$ to get the filtering (posterior) distribution, $\pi_{t|t}(dx) = Pr(X_t \in dx|Y_{1:t})$. We use a particle filter because the observation model is nonlinear and the posterior can temporarily become multi-model when there are false edges due to background clutter.

3.2 Change Point Detection

Activities will change in the course of a long video sequence. The activity changes will cause the PF, with a large enough number of particles, and tuned to the dynamical model of a particular activity, to lose track when the activity changes. This is because under the existing activity model with which the particle filter operates, the new observations would appear to have very large observation noise. Thus the tracking error will increase when the activity changes and this can be used to detect the change times. The tracking error or prediction error is the distance between the current observation and its prediction based on past observations. When observation is an edge image, TE is calculated by

$$TE = \sum_{k=1}^{K} ||q_k - f(q_k, G_t)||^2.$$

For the case when the switch from one activity to another is a slow one, the PF dose not lose track very quickly (the tracking error increases slowly). The tracking error will take long to detect the change, and then we use the *Expected (negative) Log Likelihood (ELL)*, i.e., $ELL = E[-\log p(v_t)]$ [16]. ELL is approximated by

$$ELL^N = \frac{1}{N}\sum_{i=1}^{N} {v_t^{(i)}}^T \Sigma_v^{-1} v_t^{(i)} + K,$$
$$\text{where} \quad K \triangleq -\log\sqrt{(2\pi)^{2k-4}|\Sigma_v|},$$

and N is the number of particles, Σ_v is the covariance matrix of v.

3.3 Model Switching to a New Activity

Once the change time detection has happened successfully, the next problem is to determine the correct activity from the class of previously learned activity models. This is known as the problem of *model switching*. This is done by projecting the observed shape in a frame onto the mean shape for each of the learned activities and choosing the one with the largest projection. In practice, this is done for a few frames before a final decision is made, since individual frames of different activities may be similar. In order to initialize the shape after a model-switch, we use motion segmentation to isolate the person and re-estimate the scale and translation parameters (note that background information is not required). The autoregression matrix, A_c, is extremely sensitive to the training data, and is not used in the recognition experiments.

3.4 Simultaneous Tracking, Change Detection and Recognition (Simul-TraCR) Algorithm

We now outline the main steps of the simultaneous tracking and recognition algorithm, incorporating change detection and model switching. For simplicity, let us assume that there are two activities in the sequence, A_1 and A_2. For the first frame in A_1, the region of interest (a person or a group of people) is detected based on the application requirements (not part of this paper) and the corresponding model for the activity is determined as in Section 3.3. After this initialization, the algorithm now proceeds as follows.

Track. Based on the detected region and the chosen dynamical model, the particle filter is used to track the activity. Measures for determining the accuracy of the tracking algorithm (TE and ELL) are computed for each frame.

Change Detection. When the fidelity measures exceed a certain threshold (details in Section 4.1) for a few consecutive frames, a change is detected.

Model Switching. Once the change is detected, the new shape vector is obtained from the edge map of image frame and a search is initiated for the correct activity model. Given an observed image I_t, we label this frame as the activity that minimizes $||\Gamma_t - se^{j\theta}\mu_m + (a+jb)||^2, m = 1, ..., M$, where s, θ and $a + jb$ are the scale, rotation and translation parameters respectively, M is the number of all candidate activities, and Γ_t is obtained from I_t as explained in Section 2.3. If the distance is above a certain threshold for all m, we decide that the activity is not within the learned database and this is also indicated. Once the correct activity model is identified, we use this and go back to Track.

Note that change detection and switching may be between different portions of the same activity, specifically, for those activities in which a non-stationary dynamical model is needed.

4 Experimental Results

4.1 Indoor Activity Sequence

We now show examples of our Simul-TraCR algorithm on 10 different activities captured in video. The training and testing sequences were captured separately on different days. The binarized silhouette denoting the contour of the person in every frame of the *training* sequence is obtained using background subtraction. We extracted the shape from this binary image by uniformly sampling on the edge of the silhouette. Once the landmarks are obtained, the shape is extracted using the procedure described in Section 2.1. Using the training data, the parameters of the dynamical models for each activity were learnt using these shape sequences and as explained in Section 2.2. In the *testing* sequence, the silhouette is pre-computed only in the first frame if the background information is available; otherwise we use motion segmentation over a few initial frames to obtain the siluhouette. Thereafter it is obtained as the output of the tracking algorithm, as explained above. The database we collected consists of 10 activities (whose composition make up a number of normal everyday activities), bending across, walking towards camera and bending down, leaning forward and backward, leaning sideward, looking around, turning head, turning upper body, squatting, bending with hands outstretched, and walking. We will refer to the n_{th} activity as Act$n$.

Figure 3 shows the tracking results of several activities, along with the edge image observations for each of them. Activiies 1 and 2 are tracked with PSSA model composed of three and two "stationary" sub-models respectively. Activities 3-10 are tracked with the "stationary shape activity" model.

Figure 4 shows a plot of the tracking error of a multi-activity sequence which includes one slow change and some sudden changes. The order of activities is Act3, Act4, Act8, Act9 and Act7. From Act3 to Act4 the change happens slowly, other changes happen suddenly. There is a delay involved in detecting this change, which should not be confused with the one mentioned above for switching to the correct model. The total delay is the sum of the delays due to change detection and model switching. There is a long delay in the case of slowly changing activities, because the tracking error increases slowly, while for other changes, the delays for change detection are very short.

4.2 Experiments with Outdoor Data

The sequence on which we show our results consists of activities of two people: Person 1 walking with a package in hand and doing this multiple times, and Person 2 first walking towards the camera, and then walking parallel to the camera. There are three activities in this case: walking towards camera, walking

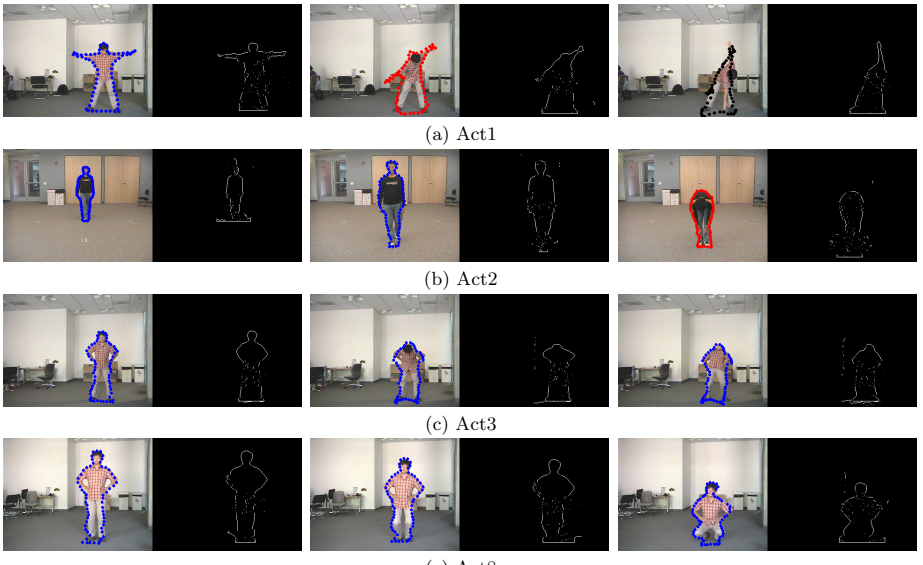


Fig. 3. Tracking results on video data. On the right is the edge image which is used as the observation.

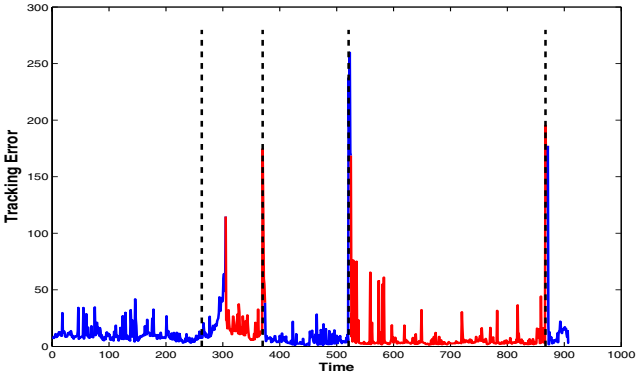


Fig. 4. Tracking error of multi-activity sequence which includes slow and sudden change. The order of activities is Act3, Act4, Act8, Act9 and Act7. From Act3 to Act4, the change happens slowly, other changes happen suddenly. The tracking error increases when an activity transition happens. Once the model switch occurs and the new model is able to track properly, the tracking error goes down.

parallel to camera, walking with small package in hand. The tracking results, along with the recognized activity, is shown in Figure 5. The recognition results for each frame for the two different people are shown in Fig. 6.

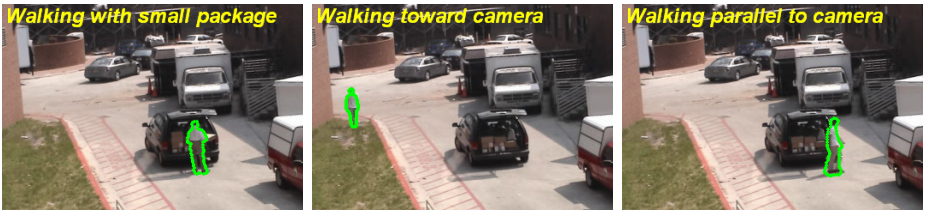
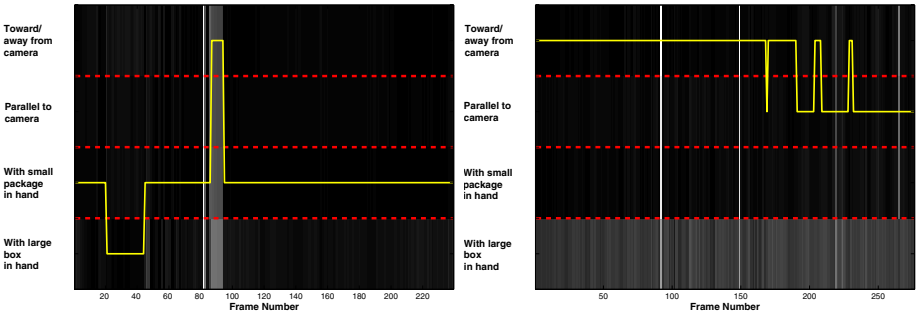


Fig. 5. Tracking and recognition results on an outdoor sequence



(a) Person 1 (b) Person 2

Fig. 6. Similarity Matrices, shown for the activities of of Person 1 in Figure 5(a), and Person 2 in Figure 5(b)-(c), respectively. The thick yellow line represents recognized activity for each frame.

5 Conclusion and Future Work

In this paper, we proposed a novel dynamical feedback system for simultaneous and persistent tracking, recognition and segmentation of human activities from video sequences. We use a non-linear, non-stationary model defined on the shape of human body contour to represent activities. The activities are recognized by comparing the tracked observations against a prior database. At the same time, the performance of our tracking algorithm is analyzed using feedback signals and this helps in segmenting the shots of different activities. We demonstrate the effectiveness of our system by showing experimental results on real life video of different activities. As a part of future work, we will address the problems of recognizing complex multi-person activities in networks of video cameras.

References

1. Y. Bar-Shalom and T. E. Fortmann. *Tracking and Data Association.* Academic Press, 1988.
2. T.-J. Cham and J. M. Rehg. A multiple hypothesis approach to figure tracking. In *Computer Vision and Pattern Recognition*, 1999.

3. A. Doucet, N. Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice.* Springer, 2001.
4. I. Dryden and K. Mardia. *Statistical Shape Analysis.* John Wiley and Sons, 1998.
5. W. Grimson, L. Lee, R. Romano, and C. Stauffer. Using Adaptive Tracking to Classify and Monitor Activities in a Site. In *Computer Vision and Pattern Recognition*, pages 22–31, 1998.
6. M. Harville and D. Li. Fast, integrated person tracking and activity recognition with plan-view templates from a single stereo camera. In *Computer Vision and Pattern Recognition*, pages II: 398–405, 2004.
7. W. Hu, T. Tan, L. Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *IEEE Trans. on Systems, Man, Cybernetics - Part C: Applications and Reviews*, 34(3), 2004.
8. M. Isard and A. Blake. Condensation: Conditional Density Propagation for Visual Tracking. *International Journal of Computer Vision*, pages 5–28, 1998.
9. J. Jackson, A. Yezzi, and S. Soatto. Tracking deformable moving objects under severe occlusions. In *IEEE Conference on Decision and control*, Dec, 2004.
10. Y. Li and T. Boult. Understanding Images of Graphical User Interfaces: A new approach to activity recognition for visual surveillance. In *ACM UIST*, 2003.
11. L. Liao, D. Fox, and H. Kautz. Location-based activity recognition using relational markov networks. In *Proc. of the International Joint Conference on Artificial Intelligence*, 2005.
12. M. Niethammer and A. Tannenbaum. Dynamic level sets for visual tracking. In *IEEE Conference on Decision and Control*, 2004.
13. B. North, A. Blake, M. Isard, and J. Rittscher. Learning and classification of complex dynamics. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(9):1016–1034, 2000.
14. Y. Rathi, N. Vaswani, A. Tannenbaum, and A. Yezzi. Particle Filtering for Geometric Active Contours with Application to Tracking Moving and Deforming Objects. In *Computer Vision and Pattern Recognition*, 2005.
15. J. Rittscher and A. Blake. Classification of human body motion. In *International Conf. on Computer Vision*, volume 2, pages 634–639, 1999.
16. N. Vaswani. Change Detection in Partially Observed Nonlinear Dynamic Systems with Unknown Change Parameters. In *American Control Conference*, 2004.
17. N. Vaswani and R. Chellappa. NonStationary Shape Activities. In *Proc. of IEEE Conf. on Decision and Control*, 2005.
18. N. Vaswani, A. Roy-Chowdhury, and R. Chellappa. Shape Activities: A Continuous State HMM for Moving/Deforming Shapes with Application to Abnormal Activity Detection. *IEEE Trans. on Image Processing*, October 2005.
19. Y. Zhai and M. Shah. A general framework for temporal video scene segmentation. In *International Conf. on Computer Vision*, 2005.
20. S. Zhou, R. Chellappa, and B. Moghaddam. Visual Tracking and Recognition Using Appearance-Adaptive Models in Particle Filters. *IEEE Trans. on Image Processing*, 13(11):1491–1506, November 2004.
21. S. Zhou, V. Krueger, and R. Chellappa. Probabilistic recognition of human faces from video. *Computer Vision and Image Understanding*, 91:214–245, July-August 2003.