"Shape Activity": A Continuous State HMM for Moving/Deforming Shapes with Application to Abnormal Activity Detection

Namrata Vaswani, Amit Roy Chowdhury, Rama Chellappa

***Abstract*— The aim is to model "activity" performed by a group of moving and interacting objects (which can be people or cars or different rigid components of the human body) and use the models for abnormal activity detection. Previous approaches to modeling group activity include co-occurrence statistics (individual and joint histograms) and Dynamic Bayesian Networks, neither of which is applicable when the number of interacting objects is large. We treat the objects as point objects (referred to as "landmarks") and propose to model their changing configuration as a moving and deforming "shape" (using Kendall's shape theory for discrete landmarks). A continuous state Hidden Markov Model (HMM) is defined for landmark shape dynamics in an activity. The configuration of landmarks at a given time forms the observation vector and the corresponding shape and the scaled Euclidean motion parameters form the hidden state vector. An abnormal activity is then defined as a change in the shape activity model, which could be slow or drastic and whose parameters are unknown. Results are shown on a real abnormal activity detection problem involving multiple moving objects.**

I. Introduction

In this paper, we develop models for the configuration dynamics of a group of moving landmarks (point objects) in shape space. Shape of a group of discrete points (known as 'landmarks') is defined by Kendall [1] as all the geometric information that remains when location, scale and rotational effects (referred to as "motion parameters" in this paper) are filtered out. There has been a lot of work in learning the statistics of a dataset of similar shapes and defining probability distributions in shape and pre-shape space, [2] provides a good overview. Statistical shape theory began in the late 1970s and has evolved into viable statistical approaches for modeling the shape of an object with applications in object recognition and matching. In this work, we extend these static classification approaches to defining dynamical models for landmark shape deformation. Also, we consider here the shape formed by a configuration of point objects instead of that of a single object.

For a dataset of similar shapes, the shape variability can be modeled in the tangent hyperplane to the shape space at the mean shape [2]. The tangent hyperplane is a linearized version of the shape space linearized at a particular point known as the pole of tangent projection. Typically one uses the Procrustes mean [2] of the dataset as the pole. The tangent plane is a vector space and hence techniques from linear multivariate statistics can be used to model shape variability in tangent space. In this work, we model shape dynamics by defining an autoregressive (AR) model in the tangent plane at the mean shape. To model the configuration dynamics, we also define motion models (models for translation, isotropic scaling and rotation). We use the term *"shape activity"* to denote a continuous state HMM (also referred to as a "partially observed nonlinear dynamical model" or a "stochastic state space model" in different contexts) for the shape deformation and motion in the activity.

Previous approaches to modeling activity performed by groups of point objects include co-occurrence statistics (e.g. [3]) and discrete state Dynamic Bayesian Networks (DBNs) (e.g. [4]). Co-occurrence statistics involves learning individual and joint histograms of the objects. Joint histograms for modeling interactions is feasible only when the number of interacting objects is small. Our approach on the other hand implicitly models interactions and independent motion of a group of objects with any number of interacting objects. DBNs define high level relations between different events and typically use heuristics for event detection. Our algorithms can be used to provide a more principled strategy for event detection using DBNs. Another advantage of our framework is that using shape and its dynamics makes the representation invariant to translation, in-plane rotation or sensor zoom. The idea of using "shape" to model activities performed by groups of moving objects is similar to recent work in literature on controlling formations of groups of robots using shape (e.g. [5]).

One example of a stationary shape activity, that we discuss in this paper, is that of people (treated as point objects) deplaning and moving towards the terminal at an airport (See figure 2(a)). Our framework can be used to model normal activity and detect abnormal activity as a deviation from the normalcy model. We are able to detect both spatial and temporal abnormalities (terminology borrowed from [3]). The "landmark" could also be a moving vehicle and one could model traffic in a certain region as the normal activity and define lane change as the "abnormality". Our framework can also be used to model the dynamics of articulated shapes like the human body (the different rigid parts of the human body forming the landmarks) and thus represent different actions [6]. This has application in classifying or tracking a sequence of actions and also in detecting motion disorders. Also, our approach is sensor independent. The same framework could be used for point location observations obtained from other sensors, for e.g. infrared, acoustic, radar or seismic, and only the observation model would change.

Manuscript received August 22, 2003; revised April 5, 2004. This work was partially supported by the DARPA/ONR Grant N00014-02-1-0809.

Namrata Vaswani is now with the School of Electrical and Computer Engineering at the Georgia Institute of Technology, Atlanta, GA 30332 (email: namrata@ece.gatech.edu).

Amit K. Roy-Chowdhury is now with the Dept. of Electrical Engineering at University of California, Riverside, CA 92521 (email: amitrc@ee.ucr.edu).

Rama Chellappa is with the Dept. of Electrical and Computer Engineering and the Center for Automation Research at the University of Maryland, College Park, MD 20742 (email: rama@cfar.umd.edu).

A. Organization of the Paper

The paper is organized as follows: We discuss related work in the next subsection. Some definitions and methods for shape analysis are presented in Section II. The shape dynamics for stationary shape activity and the training algorithm to learn its parameters is described in Section III-A. The noise in the observed configuration makes the state (shape, motion) partially observed (or hidden). The partially observed model is discussed in Section III-B. The non-stationary shape activity model is given in Section III-C. The particle filtering algorithm to estimate the hidden state from the observations is discussed in Section III-D and its advantages are discussed in III-E. The abnormality detection problem and its formulation as a change detection problem is discussed in Section IV. The strategy to deal with time-varying number of landmarks is given in Section V. Experimental results on the airport terminal abnormal activity detection problem are presented in Section VI. Extensions of our framework to tracking observations and to activity sequence identification and tracking are discussed in Section VII. Conclusions are given in Section VIII.

B. Related Work

Shape Representations: Some of the commonly used representations for shape are Fourier descriptors [7], splines [8] and deformable snakes all of which model the shape of continuous curves. But in our work we are attempting to model the dynamics of a group of discrete landmarks (which could be moving point objects or moving parts of an articulated object like the human body). Since the data is inherently finite dimensional, using infinite dimensional representations of a continuous curve is not necessary and hence we look only at the representation of shape in $\Re^n$ (modulo Euclidean similarity transformations) which was first defined by Kendall in 1977. Active Shape Models introduced by Cootes et al [9] also consider the shape of points in $\Re^n$. In [10], they define 'Point Distribution Models' which are principal component models for shape variation using Procrustes residuals.

Modeling Shape Change: There has been a lot of work in defining probability distributions in (Kendall's) shape and preshape space and also in analyzing datasets of similar shapes in the tangent space at the mean (discussed in chapter 6, 7 and 11 of [2], and in [11], [12], [13] and references therein). Many models for shape deformation of one shape into another have been proposed which include affine deformation, thin plate splines, and principal and partial warp deformations (discussed in chapter 10 of [2]). But none of these define dynamical models for time seqeunces of shapes. We propose in this paper, a partially observed dynamical model (which also satisfies the Hidden Markov Model property and hence we refer to it as an HMM in the rest of the paper) for stationary and non-stationary shape activities. Our model for non-stationary shape activities is similar in spirit to those in [14] and [15] where the authors define dynamical models for motion on Lie groups and Grassmann manifolds, respectively, using piecewise geodesic priors and track them using particle filtering.

Modeling Activity: There is a huge body of work in computer vision on modeling and recognition of activities, human actions and events. The work can be classified (based on the formalisms used) as Bayesian networks (BNs) and Dynamic Bayesian networks (DBNs) [16], [4]; finite state HMMs for representing activity [17], [18] ; stochastic grammars [19]; and factorization method based approaches [20], [21]. In [3], the authors perform clustering to learn the co-occurrence statistics of individual objects and their interactions with other objects. [22] is another work which treats events as long spatio-temporal objects and clusters them based on their behavioral content. In [23], action "objects" are represented using generalized cylinders with time forming the cylinder axis. Now, [3], [20], [21], [22], [23] are non-parametric approaches to activity/event recognition, while HMMs, stochastic grammars, BNs and DBNs are model based approaches. Our work also defines a parametric model (but it is a continuous state HMM) for activity performed by a group of objects and there are some other differences. First, we treat objects as point objects and hence we can get our observations from low resolution video or even from other sensors like radar, acoustic or infra-red. Second, we provide a single global framework for modeling the interactions and independent motion of multiple moving objects by treating them as a deformable shape.

Particle Filters and Change Detection: Particle filters [24] have been used extensively in computer vision for tracking a single moving object in conjunction with a measurement algorithm to obtain observations [25], [26], [27]. In [28], particle filtering is used to track multiple moving objects but they use separate state vectors for each object and define data association events to associate the state and observation vectors. In our work, we represent the combined state of all moving objects using the shape and global motion of their configuration and define a dynamic model for both shape and motion. We use a particle filter to filter out the shape from noisy observations of the object locations and use the filtered shape for abnormal activity detection. We define an abnormal activity as a change which could be slow or drastic and whose parameters are unknown. An algorithm for change detection in nonlinear systems using particle filters is given in [29]. But it assumes that the changed system's parameters are known and it deals only with sudden changes. In this paper we use a statistic called ELL for detecting slow changes, with unknown parameters [30], [31].

II. Preliminaries and Notation

We would first like to clarify that the terms partially observed dynamical model and HMM are used interchangeably for "shape activity" models since the partially observed dynamic model that we define is also an HMM. We use "$\arg$" to denote the angle of a complex scalar as well as in "$\arg\min$" for the argument minimizing a function, but the meaning is clear from the context. * is used to denote conjugate transpose. $||.||$ is used for the Euclidean norm of a complex or real vector and $|.|$ for the absolute value of a complex scalar. I_k denotes the $k \times k$ identity matrix and 1_k denotes a k dimensional vector of ones. Also note that to simplify notation we do not distinguish between a random process and its realization. We review below the tools for statistical shape analysis as described in [2].

Definition 1: [2] The **Configuration** is an ordered set (k-tuple) of landmarks (which in our case is the k-tuple of point object locations). The **configuration matrix** is a $k \times m$ matrix of Cartesian coordinates of the k landmarks in m dimensions. For 2D data ($m = 2$), a more compact representation is a k dimensional complex vector with x and y coordinates forming the real and imaginary parts. The **configuration space** is the space of all k-tuples of landmarks i.e. $\Re^{km}$.

Translation Normalization: The complex vector of the configuration (Y_{raw}) can be centered by subtracting out the centroid of the vector, thus yielding a **centered configuration**, i.e.

$$Y = CY_{raw} \quad \textit{where} \quad C = I_k - \frac{1_k 1_k{}^T}{k}. \tag{1}$$

Definition 2: [2] The **pre-shape** of a configuration matrix (or complex vector), Y_{raw}, is all the geometric information about Y_{raw} that is invariant under location and isotropic scaling. The **pre-shape space**, S_m^k, is the space of all possible pre-shapes. S_m^k is a hyper-sphere of unit radius in $\Re^{(k-1)m}$ and hence its dimension is $(k-1)m-1$ (a unit hyper-sphere in $\Re^P$ has dimension $P-1$).

Scale Normalization: The pre-shape is obtained by normalizing the centered configuration, Y, by its Euclidean norm, $s(Y) = ||Y||$ (known as **scale or size** of the configuration), i.e. $w(Y) = Y/s(Y)$.

Definition 3: [2] The **shape** of a configuration matrix (or complex vector), Y_{raw}, is all the geometric information about Y_{raw} that is invariant under location, isotropic scaling and rotation i.e. $[z] = \{sY_{raw}R + 1_k\alpha^T : s \in \Re^+, R \in SO(m), \alpha \in \Re^m\}$. The **shape space** is the set of all possible shapes. Formally, the shape space, Σ_m^k, is the orbit space of the non-coincident k point set configurations in $\Re^m$ under the action of Euclidean similarity transformations. The dimension of shape space is $M = (k-1)m - 1 - m(m-1)/2$. It is easy to see that $\Sigma_m^k = S_m^k/SO(m)$, i.e. Σ_m^k is the quotient space of S_m^k under the action of the special orthogonal group of rotations, $SO(m)$.

Rotation Normalization: Shape, z, is obtained from a pre-shape, w, by rotating it in order to align it to a reference pre-shape γ. The optimal rotation angle is given by $\theta(Y,\gamma) = \arg(w^*\gamma) = \arg(Y^*\gamma)$, and the shape, $z(Y,\gamma) = we^{j\theta(Y,\gamma)} = \frac{Y}{s(Y)}e^{j\theta(Y,\gamma)}$.

In this work we deal with $m = 2$ dimensional shapes and hence the configuration vector is represented as a k dimensional complex vector and the shape space dimension is $(2k-4)$.

Distance between shapes: A concept of distance between shapes is required to fully define the non-Euclidean shape metric space. We use the Procrustes distance which is defined below.

Definition 4: [2] The **full Procrustes fit** of w onto y is

$$\begin{aligned} w^P(y) &= \hat{\beta}e^{j\hat{\theta}}w + \hat{a} + j\hat{b} \quad \text{where} \\ \hat{\beta},\hat{\theta},\hat{a},\hat{b} &= \arg\min_{(\beta,\theta,a,b)} D(y,w), \\ D(y,w) &= ||y - (\beta e^{i\theta}w + a + jb)||. \end{aligned}$$

If y and w are preshapes, it is easy to see that the matching parameters are (result 3.1 of [2])

$$\hat{a} + j\hat{b} = 0, \quad \hat{\theta} = \arg(w^*y), \quad \hat{\beta} = |w^*y| = (y^*ww^*y)^{1/2}.$$

Definition 5: [2] The **full Procrustes distance** between preshapes w and y is the Euclidean distance between the Procrustes fit of w onto y, i.e.

$$\begin{aligned} D_F(w,y) &= \inf_{\beta,\theta,a,b} D(y,w) = ||y - w^P(y)|| \\ &= \sqrt{1 - y^*ww^*y} \end{aligned} \tag{2}$$

Definition 6: [2] The **full Procrustes estimate of mean shape** (commonly referred to as **full Procrustes mean**), of a set of preshapes $\{w_i\}$ is the minimizer of the sum of squares of full Procrustes distances from each w_i to an unknown unit size mean configuration μ, i.e.

$$\begin{aligned} [\hat{\mu}] &= \arg\min_{\mu:||\mu||=1} \sum_{i=1}^n \min_{\beta_i,\theta_i,a_i,b_i} D^2(w_i,\mu) \\ &= \arg\min_{\mu:||\mu||=1} \sum_{i=1}^n D_F^2(w_i,\mu) \\ &= \arg\min_{\mu:||\mu||=1} \sum_{i=1}^n (1 - \mu^* w_i w_i^* \mu) \\ &= \arg\max_{\mu:||\mu||=1} \mu^* [\sum_{i=1}^n w_i w_i^*]\mu \end{aligned} \tag{3}$$

i.e. $[\hat{\mu}]$ is given by the set of complex eigenvectors corresponding to the largest eigenvalue of $S \stackrel{\triangle}{=} \sum_{i=1}^n w_i w_i^*$ (Result 3.2 of [2]).

Shape Variability in Tangent to Shape Space: The structure of shape variability of a dataset of similar shapes can be studied in the tangent space to the shape space. We shall consider the tangent projections to the preshape sphere after normalizing for rotation (w.r.t. the pole), which form a suitable tangent coordinate system for shape. The tangent space is a linearized local approximation of shape space at a particular point in shape space which is called the pole of tangent projection. Thus Euclidean distance in tangent space is a good approximation to Procrustes distance, for points in the vicinity of the pole. See chapter 4 of [2] for more details.

Definition 7: [2] The **Procrustes tangent coordinates** of a centered configuration, Y, taking μ as the pole, are obtained by projecting $z(Y,\mu)$ (the shape of Y aligned to μ) into the tangent space at μ, i.e.

$$v(Y,\mu) = [I_k - \mu\mu^*]z(Y,\mu) = [I_k - \mu\mu^*]\frac{Y}{s(Y)}e^{j\theta(Y,\mu)}. \tag{4}$$

The inverse of the above mapping (tangent space to centered configuration space) is

$$Y(v,\theta,s,\mu) = [(1 - v^*v)^{1/2}\mu + v]se^{-j\theta}. \tag{5}$$

The shape space is a manifold in $\mathcal{C}^{k-1}$ and hence its dimension is $k-2$. Thus the tangent space at any point of the shape space is a $k-2$ dimensional hyperplane in $\mathcal{C}^k$ (or equivalently, a $(2k-4)$-dim hyperplane in $\Re^{2k}$) [2].

III. Modeling Shape Dynamics

The distinction between motion and deformation of a deformable shape is not clear. We separate the dynamics

of a deforming configuration into scaled Euclidean motion (translation, rotation, uniform scaling) of the mean shape and non-rigid deformations. This idea is similar to that suggested in [32] for continuous curves. We define a continuous state HMM for the changing configuration of a group of moving landmarks (point objects) with the shape and scaled Euclidean motion parameters being the hidden state variables and the noisy configuration vector forming the observation. We refer to it as a "shape activity". A *"stationary shape activity"* is defined as one for which the shape vector is stationary i.e. the mean shape[1] remains constant with time and the deformation model is stationary while in a *"non-stationary shape activity"*, the mean shape changes with time.

We discuss below the stationary and non-stationary shape activity models and also the particle filtering algorithm to estimate the shape from the noisy configuration observations. The entire discussion assumes a fixed number of landmarks. But in certain applications like the airport scenario with people deplaning, the number of landmarks varies with time. We deal with this by resampling the curve formed by joining the landmarks to a fixed number of points. This is discussed in Section V. Also, note that in this representation of the shape of discrete landmarks, correspondences between landmarks are assumed to be known across frames. Since the number of landmarks is usually small ($k = 8$ in this case), this is easy to ensure.

A. Stationary Shape Activity: Shape Deformation Model in Tangent Space

A sequence of point configurations from a stationary shape activity (SSA), with small system noise variance, would lie close to each other and to their mean shape (see figure 1(a)). Hence a single tangent space at the mean is a good approximate linear space to learn the shape deformation dynamics for a SSA. We represent a configuration of landmarks by a complex vector with the x and y coordinates of a landmark forming the real and imaginary parts[2]. We discuss the training algorithm i.e. how to learn the shape dynamics given a single training sequence of configurations. Given a sequence of configurations with negligible observation noise, $\{Y_{raw,t}\}$, we learn its Procrustes mean and evaluate the tangent coordinates of shape (using the Procrustes mean as the pole), as

$$\begin{aligned} Y_t &= CY_{raw,t}, \\ s_t &\triangleq s(Y_t) = ||Y_t||, \qquad w_t = Y_t/s_t, \\ \mu &= \arg \max_{\mu:||\mu||=1} \mu^* [\sum_{t=1}^{T} w_t w_t^*] \mu \\ \theta_t &\triangleq \theta(Y_t, \mu) = \arg(w_t^* \mu), \qquad z_t = w_t e^{j\theta_t} \end{aligned} \tag{6}$$

$$v_t \triangleq v(Y_t, \mu) = [I_k - \mu\mu^*] z_t = [I_k - \mu\mu^*] \frac{Y_t e^{j\theta_t}}{s_t} \tag{7}$$

Since the tangent coordinates are evaluated w.r.t. the mean shape of the data, assuming that they have zero mean is a valid assumption. We string the complex tangent vector components as a $2k$ dimensional real vector and define a linear Gauss Markov model on it to model the shape deformation dynamics. Note that since we are assuming small variations about a mean shape, a first order Gauss Markov model is sufficient to model the shape dynamics in this case, i.e.

$$\begin{aligned} v_t &= A_t v_{t-1} + n_t \\ v_0 &\sim \mathcal{N}(0, \Sigma_{v,0}), \quad n_t \sim \mathcal{N}(0, \Sigma_{n,t}) \end{aligned} \tag{8}$$

where $\{n_t\}$ is i.i.d. Gaussian system noise. The deformation process is assumed to be stationary and ergodic. Under this assumption the above is a first order autoregressive (AR) model. Thus, $\Sigma_{v,0} = \Sigma_{v,t} = \Sigma_v$, $\Sigma_{n,t} = \Sigma_n$ and $A_t = A$ is the autoregression matrix with $A < I$. Thus all the three parameters can be *learnt using a single training sequence* of tangent coordinates, $\{v_t\}$, as follows [33]

$$\begin{aligned} A &= R_v(1)\Sigma_v^{-1} \quad \text{where} \\ \Sigma_v &= \frac{1}{T}\sum_{t=1}^{T} v_t v_t^T \quad \text{and} \quad R_v(1) = \frac{1}{T-1}\sum_{t=2}^{T} v_t v_{t-1}^T \\ \Sigma_n &= \frac{1}{T}\sum_{t=1}^{T} (v_t - Av_{t-1})(v_t - Av_{t-1})^T \end{aligned} \tag{9}$$

and the joint pdf of v_t is given by

$$\begin{aligned} p(v_t) &= \mathcal{N}(0, \Sigma_v), \quad \forall t \\ p(v_t|v_{t-1}) &= \mathcal{N}(Av_{t-1}, \Sigma_n), \quad \forall t. \end{aligned} \tag{10}$$

Note that the asymptotically stationary case where $A < I$ but $\Sigma_{v,0} \neq \Sigma_v$ so that $\Sigma_{v,t} \to \Sigma_v$ only for large time instants ($t \to \infty$), can also be dealt with in the above framework. In that case $\Sigma_{v,0}$ is defined using a-priori knowledge, Σ_n can be learnt exactly as in (9), and $\Sigma_v, R_v(1)$ can also be learnt as in (9) but by excluding the summation over the initial (transient) time instants.

B. Stationary Shape Activity: Partially Observed (Hidden) Shape Dynamics

In the previous subsection we defined a dynamic model on the shape of a configuration of moving points. We assumed that the observation sequence used for learning the shape dynamics has zero (negligible) observation noise associated with it (e.g. if it were hand-picked). But a test sequence of point configurations, $\{Y_{raw,t}\}$, will usually be obtained automatically using a measurement algorithm (e.g. a motion detection algorithm [34]). It will thus have large observation noise associated with it, i.e. $Y_{raw,t} = Y_{raw,t}^{actual} + \zeta_{raw,t}$ where $\zeta_{raw,t}$ is zero mean i.i.d. Gaussian noise, $\zeta_{raw,t} \sim \mathcal{N}(0, \Sigma_{obs,raw,t})$. If the different landmarks are far apart, the noise can be assumed to be i.i.d. over the different landmarks as well (i.e. white $\Sigma_{obs,raw,t}$). Now translation normalization is a linear process and hence $Y_t = CY_{raw,t}$ is also Gaussian[3]

[1]In the entire paper, "mean shape" refers to the Expected Procrustes estimate of mean shape

[2]Note that all transformations between the configuration space to shape space and tangent to shape space are defined in $\mathcal{C}^k$ (k-dim complex space) but the dynamical model on tangent coordinates is defined in $\Re^{2k}$ by vectorizing the complex vector. This is done only for compactness of representation. The entire analysis could instead have been done in $\Re^{2k}$.

[3]Note that here we have assumed Gaussian observation noise, $\zeta_{raw,t}$, but in general a PF can track with any kind of noise. But for non-Gaussian $\zeta_{raw,t}$, it is in general not possible to define a distribution for ζ_t and one would have to treat the translation as part of the state vector.

with observation noise, ζ_t, given by

$$\Sigma_{obs,t} = C\Sigma_{obs,raw,t}C^T \tag{11}$$

(C is the centering matrix defined in (1)). But the mapping from centered configuration space to the tangent space is nonlinear (scaling by $||Y_t||$ followed by rotation to align with mean) and hence it is not possible to obtain a closed form expression for the pdf of the tangent coordinates given that there is observation noise in the configuration vector. To deal with this, one solution is to define a partially observed dynamical model which can then be tracked using a particle filter (PF) to estimate the distribution of the tangent coordinates of shape given the noisy observations. The observed centered configuration, Y_t, forms the observation vector and the shape, scale and rotation form the hidden state vector. We discuss the PF in Section III-D and its advantage over an Extended Kalman Filter in Section III-E.

Now, we have the following *observation model* for a "stationary shape activity" with the observation vector Y_t being the centered configuration vector and the state vector $X_t = [v_t, s_t, \theta_t]$:

$$\begin{aligned} Y_t &= h(X_t) + \zeta_t, \qquad \zeta_t \sim \mathcal{N}(0, \Sigma_{obs,t}) \\ h(X_t) &= z_t s_t e^{-j\theta_t}, \text{ where } z_t = (1 - v_t * v_t)^{1/2}]\mu + v_t \end{aligned} \tag{12}$$

Defining scale and rotation (motion parameters) as part of the state vector implies that we need to define prior dynamic models for them (motion model). The *motion model* can be defined based on either the motion of the shape if it is a moving configuration or based on motion of the measurement sensor if the sensor is moving (for e.g. a moving camera or just an unstable camera undergoing a slight random motion) or a combined effect of both. A camera on an unstable platform, like an unmanned air vehicle (UAV), will have small random x-y motion (translation), motion in z direction (scale change) and rotation about the z axis (rotation angle change). The translation gets removed when centering $Y_{raw,t}$. The scale and rotation can be modeled in this case by using an AR model both for log of scale and for the unwrapped rotation angle[4], i.e.,

$$\begin{aligned} \log s_t &= \alpha_s \log s_{t-1} + (1 - \alpha_s)\mu_s + n_{s,t} \\ \log s_0 &\sim \mathcal{N}(\mu_s, \sigma_s^2), \quad n_{s,t} \sim \mathcal{N}(0, \sigma_r^2) \\ \theta_t &= \alpha_\theta \theta_{t-1} + (1 - \alpha_\theta)\mu_\theta + n_{\theta,t} \\ \theta_0 &\sim \mathcal{N}(0, \sigma_\theta^2), \quad n_{\theta,t} \sim \mathcal{N}(\mu_\theta, \sigma_\theta^2) \end{aligned} \tag{13}$$

The motion model parameters can be learnt using the training sequence values of $\{s_t\}_{t=1}^T$ and $\{\theta_t\}_{t=1}^T$ given by (6). $\{\theta_t\}_{t=1}^T$ will have to be the unwrapped value of the rotation angle to learn a Gaussian model. Also, one can either assume wide sense stationarity, in which case $\mu_s, \sigma_s^2, \sigma_r^2, \alpha_s$ and $\mu_\theta, \sigma_\theta^2, \sigma_u^2, \alpha_\theta$ can be learnt using Yule-Walker equations [33], or assume a random walk motion model(set $\alpha_s = 1$ and $\alpha_\theta = 1$), depending on the application.

[4]Since we are modeling only random motion of a camera, a first order linear Markov model for log of scale and rotation is sufficient in this case.

The *shape deformation dynamics* (equation (8) in Section III-A) and the *motion model* defined above (equation (13)) form the *system model* while equation (12) defines the *observation model*. Thus we have defined *a continuous state HMM (partially observed dynamic model)* for a "stationary shape activity". The model is non-linear since the mapping $h(X_t)$ is nonlinear.

C. Non-Stationary Shape Dynamics

For a "non-stationary shape activity" model (details in [6]), the mean shape is time-varying and hence modeling the shape dynamics requires a time-varying tangent space (see figure 1(b)) defined with the current shape as the pole. Note that, modulo reflections, there is a one to one mapping between the tangent space at any point on the shape manifold and the shape manifold. But the distance between two points on a tangent plane is a good approximation to the distance on the shape manifold only for points close to the pole of the tangent plane. Hence the assumption of i.i.d. system noise to go from shape at t to shape at $t+1$ is valid only for shapes in the vicinity of the pole. Thus when the shape variation is large (for NSSA), there is a need to define a tangent space with the current shape being the pole.

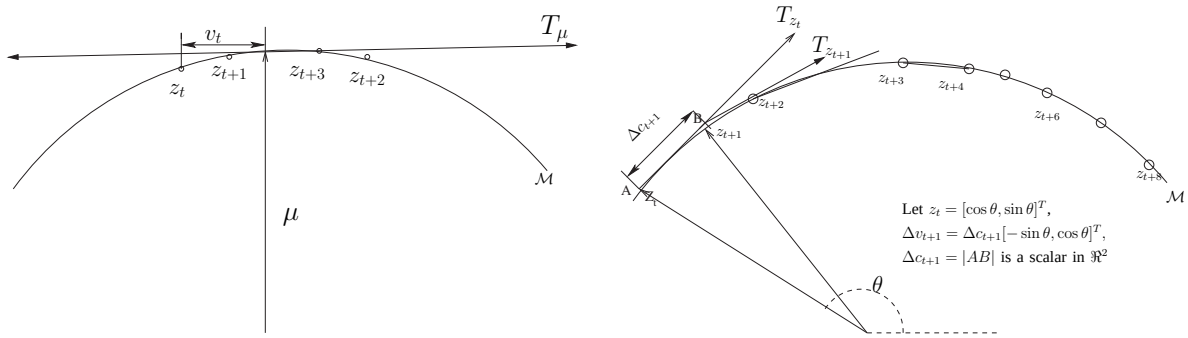
The state space now consists of the mean shape at time t, z_t, the "shape velocity coefficients" vector, c_t, and the motion parameters (scale s_t, rotation θ_t) i.e. state $X_t = [z_t, c_t, s_t, \theta_t]$. Denote the tangent space at z_t by T_{z_t}. We then have the following dynamics: The tangent coordinate of z_t in $T_{z_{t-1}}$ (denoted by $v_t(z_t, z_{t-1})$) defines a *"shape velocity"* (time derivative of shape) vector. We perform a Singular Value Decomposition [33] of the tangent projection matrix, $[I_k - z_{t-1}z_{t-1}^*]C$, to obtain an orthogonal basis for the $(k-2)$-dim tangent hyperplane $T_{z_{t-1}}$. Denote the orthogonal basis matrix for $T_{z_{t-1}}$ by $U(z_{t-1})$ [5]. The $(k-2)$-dim vector of coefficients along these basis directions, denoted by $c_t(z_t, z_{t-1})$, is a coefficients vector for the "shape velocity", v_t, i.e. $v_t = U(z_{t-1})c_t$. The shape at t, z_t is obtained by "moving" z_{t-1} on the shape manifold as follows: "Move" an amount v_t (from origin) in $T_{z_{t-1}}$ and then project back onto shape space. This is done as follows: $z_t = (1 - v_t^* v_t)^{1/2} z_{t-1} + v_t$.

We define a linear Gauss-Markov model on shape velocity v_t which corresponds to a linear Gauss Markov model for c_t. We can then summarize the shape dynamics as follows:

$$\begin{aligned} c_t &= A_{c,2,t}c_{t-1} + n_t, \quad n_t \sim \mathcal{N}(0, \Sigma_{n,c,2,t}) \\ v_t &= U(z_{t-1})c_t, \quad U(z_{t-1}) = \text{orthogonal basis}(T_{z_{t-1}}) \\ z_t &= (1 - v_t^* v_t)^{1/2} z_{t-1} + v_t. \end{aligned} \tag{14}$$

If we assume a time invariant AR model on $\{v_t\}$, i.e. $v_t = A_{v,2}v_{t-1} + n_{v,t}$ then we have a time varying Gauss-Markov

[5]The basis vectors, $\{\underline{u}_{t,i}\}_{i=1}^{k-2}$, are arranged as column vectors of a matrix, $U(z_{t-1})$, i.e. $U_t^{k\times(k-2)} = [\underline{u}_{t,1}, \underline{u}_{t,2} \cdots \underline{u}_{t,k-2}]$. $U_t^{k\times(k-2)}$ = orthogonal basis$(T_{z_{t-1}})$ is evaluated as : $U_t = U_{full,t}Q$ where $U_{full,t}SU_{full,t}^* = [I_k - z_{t-1}z_{t-1}^*]C$, and $Q = [I_{(k-2)\times(k-2)}, 0_{(k-2)\times 2}]^T$



(a) Stationary Shape Activity (SSA) (b) Nonstationary Shape Activity (NSSA)

Fig. 1. SSA & NSSA 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 sequence of shapes from a SSA; 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 sequence of shapes from an NSSA, the shapes move on the shape manifold and hence we need to define a new tangent space at every time instant.

model on c_t with

$$\begin{aligned} A_{c,2,t} &= U(z_{t-1})^* A_{v,2} U(z_{t-2}), \quad \text{and} \\ \Sigma_{n,c,2,t} &= U(z_{t-1})^* \Sigma_{n,v,2} U(z_{t-2}). \end{aligned} \tag{15}$$

Note that a Markov model on the shape velocity corresponds to a second order Markov model on shape, z_t (hence the subscript '2' on the parameters). Some special cases are $A_{v,2} = 0$ or i.i.d. velocity (first order Markov model on shape); $A_{v,2} = I$ which corresponds to i.i.d. shape acceleration and $A_{v,2} = A_{AR}$ or stationary shape velocity.

The *motion model* (model on s_t, θ_t) can be defined exactly as in equation (13) but now θ_t is the rotation angle of current configuration w.r.t. the current mean shape $\mu_t = z_{t-1}$ and hence is a measure of rotation speed. As before, one can assume the motion model to be stationary or non-stationary. The shape and motion model, (14) and (13)), form the *system model*. The *observation model* is as follows:

$$Y_t = \tilde{h}(X_t) + \zeta_t, \quad \text{where} \quad \tilde{h}(X_t) = z_t s_t e^{-j\theta_t}. \tag{16}$$

1) Training: Given a training sequence of centered (translation normalized) configurations, $\{Y_t\}_{t=1}^T$, we first evaluate $\{c_t, v_t, s_t, \theta_t\}_{t=1}^T$ as follows [6] :

$$\begin{aligned} s_t &= ||Y_t||, \quad w_t = Y_t/s_t, \\ \theta_t(Y_t, z_{t-1}) &= arg(w_t^* z_{t-1}), \quad z_t(Y_t, z_{t-1}) = w_t e^{j\theta_t}, \\ v_t(Y_t, z_{t-1}) &= [I_k - z_{t-1} z_{t-1}^*] z_t, \\ c_t(Y_t, z_{t-1}) &= U(z_{t-1})^* z_t. \end{aligned} \tag{17}$$

Assuming a time invariant AR model on shape velocity, v_t, one can learn its parameters $(A_{v,2}, \Sigma_{n,v,2})$ as in (9) and then define the time-varying Markov model for c_t using (15).

D. The Particle Filtering Algorithm

The problem of nonlinear filtering is to compute at each time t, the conditional probability distribution, of the state X_t given the observation sequence $Y_{1:t} = (Y_1, Y_2, ...Y_t)$, $\pi_t(dx) = Pr(X_t \in dx | Y_{1:t})$. Now if the system and observation models are linear Gaussian, the posteriors would also be Gaussian and can be evaluated in closed form using a Kalman filter. For nonlinear or nonGaussian system or observation model, except in very special cases, the filter is infinite dimensional. Particle Filtering is a sequential Monte Carlo technique for approximate nonlinear filtering which was first introduced in [24] as Bayesian Bootstrap Filtering.

Let the initial state distribution be denoted by $\pi_0(dx)$, the state transition kernel by $K_t(x_t, dx_{t+1})$ and the observation likelihood given the state, by $g_t(Y_t|x_t)$. For the SSA model, the state $X_t = [v_t, s_t, \theta_t]$, the transition kernel K_t is defined by (8) and (13) and g_t is defined by (12). For NSSA, $X_t = [\mu_t, v_t, s_t, \theta_t]$ and K_t is given by (14) and (13). The particle filter (PF) [24] is a recursive algorithm which produces at each time t, a cloud of N particles, $\{x_t^{(i)}\}_{i=1}^N$, whose empirical measure closely "follows" $\pi_t(dx_t)$. It also produces an approximation of the prediction distribution, $\pi_{t|t-1}(dx) = Pr(X_t \in dx | Y_{1:t-1})$.

It starts with sampling N times from the initial state distribution $\pi_0(dx)$ to approximate it by $\pi_0^N(dx) = \frac{1}{N}\sum_{i=1}^N \delta_{x_0^{(i)}}(dx)$ and then *implements the Bayes' recursion* at each time step. Now given that the distribution of X_{t-1} given observations upto time $t-1$ has been approximated as $\pi_{t-1}^N(dx) = \frac{1}{N}\sum_{i=1}^N \delta_{x_{t-1}^{(i)}}(dx)$, the prediction step samples the new state $\bar{x}_t^{(i)}$ from the distribution $K_{t-1}(x_{t-1}^{(i)}, .)$. The empirical distribution of this new cloud of particles, $\pi_{t|t-1}^N(dx) = \frac{1}{N}\sum_{i=1}^N \delta_{\bar{x}_t^{(i)}}(dx)$ is an approximation to the conditional probability distribution of X_t given observations upto time $t-1$. For each particle, its weight is proportional to the likelihood of the observation given that particle, i.e. $w_t^{(i)} = \frac{N g_t(Y_t|\bar{x}_t^{(i)})}{\sum_{i=1}^N g_t(Y_t|\bar{x}_t^{(i)})}$. $\bar{\pi}_t^N(dx) = \frac{1}{N}\sum_{i=1}^N w_t^{(i)} \delta_{\bar{x}_t^{(i)}}(dx)$ is then an estimate of the probability distribution of the state at time t given observations uptil time t. We sample N times with replacement from $\bar{\pi}_t^N(dx)$ to obtain the empirical estimate $\pi_t^N(dx) = \frac{1}{N}\sum_{i=1}^N \delta_{x_t^{(i)}}(dx)$. Note that both $\bar{\pi}_t^N$ and π_t^N approximate π_t but the last step is used because it increases the sampling efficiency by eliminating samples with very low weights.

[6] Note, the last equation, $c_t = U_t{}^* z_t$, holds because $c_t = U_t{}^* v_t = U_t{}^*[I - z_{t-1} z_{t-1}^*] z_t = U_t{}^*[I - z_{t-1} z_{t-1}^*] C z_t = U_t{}^* U_t U_t{}^* z_t = U_t{}^* z_t$.

E. Particle Filtering versus Extended Kalman Filtering

We discuss here the need for a PF and why it is better than an extended Kalman filter. An Extended Kalman Filter (EKF) [35] linearizes the non-linear system at each time instant using Taylor series and runs a Kalman filter for the linearized system. For the Taylor series approximation to be accurate, one requires the initial guess (point about which you linearize) to be close to the actual value at every time instant. Typically linearization is done about the predicted state. This means that one poorly estimated state will cause more error in the linearization matrices for the next prediction and this error will propagate (thus an EKF cannot recover once it loses track). Loss of track can occur due to an outlier observation, modeling error, large system noise or large linearization error. A PF on the other hand is stable under mild assumptions [36], [37] and hence it gets back in track more easily after losing track.

Also an EKF is unable to track non-Gaussian systems, in particular systems with multi-modal priors or posteriors, while a PF can. Multi-modal system models are required to model a sequence of activities or multiple simultaneous activities. Also in particle filtering, the number of particles, N, required to achieve a certain performance guarantee on estimation error, does not increase with increasing dimension of the state space [25], it depends only on the total randomness in the system. So for a system which is more random (larger system noise or observation noise), the PF performance can be improved by increasing N.

IV. Abnormal Activity Detection

An abnormal activity (suspicious behavior in our case) is defined as a change in the system model, which could be slow or drastic, and whose parameters are unknown. Given a test sequence of observations and a "shape activity" model, we use the change detection strategy discussed in [30], [6] to detect a change (observations stop following the given shape activity model). The cases of negligible observation noise (Fully Observed) and non-negligible observation noise (Partially observed) are discussed separately. We consider only stationary shape activities in this work.

A. Fully Observed Case

The system is said to be fully observed when the function $h(.)$ is invertible and the observation noise is zero (negligible compared to the system noise, n_t). For such a test sequence, the shape dynamics of Section III-A fully defines the "shape activity model". We can evaluate the tangent coordinates of shape (v_t) directly from the observations using (7). We use log-likelihood to test for abnormality. A given test sequence is said to be generated by a *normal activity iff* the probability of occurrence of its tangent coordinates using the pdf defined by (10) is large (greater than a certain threshold). Thus the distance to activity statistic for an '$L+1$' length observation sequence ending at time t, $d_{L+1}(t)$, is the negative log likelihood of the sequence of tangent coordinates of the shape of the observations (first used by us in [38]). We can test for abnormality at any time t by evaluating $d_{L+1}(t)$ for the past $L+1$ frames. $d_{L+1}(t)$ is defined as follows: (K is a constant defined in equation (19))

$$\begin{aligned} d_{L+1}(t) &= -2\log p(v_{t-L}, v_{t-L+1}, ...v_t) \\ &= v_{t-L}^T \Sigma_v^{-1} v_{t-L} \\ &\quad + \sum_{\tau=t-L+1}^{t} (v_\tau - A v_{\tau-1})^T \Sigma_n^{-1} (v_\tau - A v_{\tau-1}) \end{aligned} \tag{18}$$

Note here that, Σ_v is always rank deficient since $\{v_t\}$ lie in a $(2k-4)$-dim hyperplane of $\Re^{2k}$ and hence the inverse defined above actually represents the pseudo-inverse.

B. Partially Observed Case

In a partially observed system, the observation noise in the configuration landmarks' measurements is non-negligible and it is defined by the observation model discussed in Section III-B. The PF is used to estimate the posterior distribution of shape at time t given observations upto $t-1$ (prediction) and upto t (filtering). We use the change detection strategy described in [30], [6].

1) If the abnormality is a drastic one it will cause the PF, with N large enough to accurately track only normal activities, to lose track. This is because under the normal activity model (equations (8) and (13)), the abnormal activity observations (which do not follow this model) would appear to have a very large observation noise. Thus the tracking error will increase for an abnormal activity (very quickly for a drastic one) and this can be used to detect it. The *tracking error (TE)* or prediction error is the distance between the current observation and its prediction based on past observations, i.e.

$$\begin{aligned} \text{TE} \triangleq ||Y_t - \hat{Y}_t||^2 &= ||Y_t - E[Y_t|Y_{0:t-1}]||^2 \\ &= ||Y_t - E_{\pi_{t|t-1}}[h(X_t)]||^2 \end{aligned}$$

   Also, instead of tracking error, observation likelihood (OL) can also be used and as discussed in chapter 2 of [6], $OL \approx TE$ for white Gaussian noise.
2) For the case when the abnormality is a slow change (say a person walking away slowly in a wrong direction), the PF does not lose track very quickly (the tracking error increases slowly) or if it is a short duration change it may not lose track at all. The tracking error will thus take longer to detect the change or it may not detect it at all. For such a case, we use the *Expected (negative) Log Likelihood (ELL)* [31], [30], $ELL = E_{\pi_t}[-logp(v_t)]$. Note that the ELL is a posterior expectation of the right hand side of (18) with $L = 0$. In general, one could use a sequence of past shapes ($L > 0$) in this case as well. The expression for ELL is approximated by ELL^N as follows

$$\begin{aligned} ELL^N \triangleq E_{\pi_t^N}[-logp(v_t)] &= \frac{1}{N}\sum_{i=1}^{N} {v_t^{(i)}}^T \Sigma_v^{-1} v_t^{(i)} + K, \\ \text{where } K &\triangleq -\log\sqrt{(2\pi)^{2k-4}|\Sigma_v|} \end{aligned}$$

Now since the PF loses track slowly, the estimated posterior $\pi_t^{c,0,N}$ remains a good approximation of the true posterior $\pi_t^{c,c}$ for a long time. But a slowly changing shape introduces a systematically increasing bias in the tangent coordinates of shape (they no longer remain zero mean) and hence ELL would increase. These intuitive idea is analyzed rigorously in [6], [39].

Thus to *detect any kind of abnormality (slow or drastic) without knowing its rate of change, we use a combination of ELL and tracking error. We declare a sequence of observations to be abnormal when either ELL or tracking error exceeds its corresponding threshold.*

V. Time-Varying Number of Landmarks

All the analysis until now assumes that a configuration of points is represented as an element of $\Re^{2k}$ where k is a fixed number of landmarks. Now we consider what happens when the number of landmarks (here the point objects) is time-varying even though the curve formed by joining their locations remains similar. For example, a group of people (or also a group of vehicles) moving on a certain path with fixed initial and final points but number of people on the path decreases by one when a person leaves and increases by one when someone enters. In such a case, we linearly interpolate the curve by joining the landmark points in a predefined order and then re-sample the interpolated curve to get a fixed number of landmarks. The interpolation depends on the parametrization of the curve, which is an ill-posed problem when the data is inherently discrete. We have attempted to use two different schemes which exist in the literature - "arc-length re-sampling" (also known as "equidistant sampling") and "uniform re-sampling" which use two different parameterizations.

In **"arc-length resampling"**, one looks at the curve formed by joining the landmarks in a predefined order, and parameterizes the x and y coordinates by the length, l, of the curve, upto that landmark. Let $[x_t(l), y_t(l)]$ be one-dimensional functions of the curve length and seen this way the discrete landmarks $x_{t,j} = x_t(l_j), y_{t,j} = y_t(l_j), j = 0, 1, ..k_t - 1$ are non-uniformly sampled points from the function $[x_t(l), y_t(l)]$ with $l_0 = 0, l_j^2 = l_{j-1}^2 + (x_{t,j} - x_{t,j-1})^2 + (y_{t,j} - y_{t,j-1})^2$. We linearly interpolate using these discrete points to estimate the function $[\hat{x}_t(l), \hat{y}_t(l)]$ and then re-sample it uniformly at points $\tilde{l}_j = (j-1)L/k, j = 0, 1, ..k-1$ (L is the total length, $L^2 = \sum_j l_j^2$) to get a fixed number, k, of uniformly spaced landmarks. Thus, for every configuration of k_t landmarks, we get a new configuration of uniformly sampled (and hence uniformly spaced) k landmarks. The linear interpolation and resampling stages can be approximated as a linear transformation, B_t (a $k_t \times k$ matrix), applied to the original points. The covariance of observation noise in the re-sampled points becomes $\Sigma_{obs,t}^k = B_t \Sigma_{obs,t}^{k_t} B_t^T = B_t C^{k_t} \Sigma_{obs,raw,t}^{k_t} {C^{k_t}}^T B_t^T$.

"Uniform resampling", on the other hand, assumes that the observed points are uniformly sampled from some process, $[x_t(s), y_t(s)]$, i.e. it assumes that the observed points are parameterized as $x_{t,j} = x_t(s_j), y_{t,j} = y_t(s_j)$ with $s_j = (j-1)/k_t$. We linearly interpolate to estimate $[\hat{x}_t(s), \hat{y}_t(s)]$ and re-sample it uniformly at points $\tilde{s}_j = (j-1)/k$, to get a fixed number of landmarks, k. Assuming the observed points to be uniformly sampled makes this scheme very sensitive to the changing number of landmarks. Whenever the number of landmarks changes, there is a large change in the re-sampled points' configuration. This leads to more false alarms while performing abnormal activity detection. But unlike "arc-length resampling", this scheme gives equal importance to all observed points irrespective of the distance between consecutive points and so is more quick to detect abnormalities in shape caused even by two closely spaced points. We discuss an example in Section VI-D.

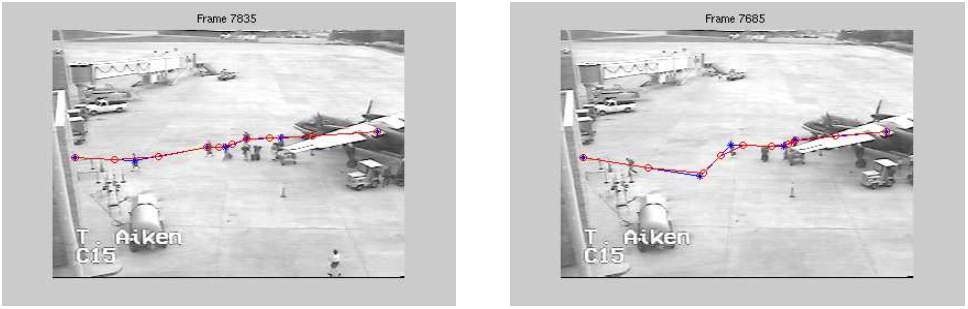
VI. Experimental and Simulation Results

A. Dataset and Experiments

We have used a video sequence of passengers deplaning and walking towards the airport terminal as an example of a "stationary shape activity". The number of people in the scene varies with time. We have resampled the curve formed by joining their locations using "arc-length resampling" (described in Section V) in all experiments except the temporal abnormality [3] detection where we use "uniform resampling". As we needed observation noise-free data to learn the system model, we used hand-marked passenger locations for training. The mean shape, μ, and the tangent space Gauss Markov model parameters, A, Σ_v, Σ_n, were learnt using this data (as discussed in Section III-A). Also the motion model parameters (which in this case model random motion of the camera) were estimated with this data. Simulated test sequences were produced by adding observation noise to the hand-marked data. We did this to study robustness of the method to increasing observation noise. We also tested with real observations obtained using a motion detection algorithm [34]. Both real and simulated observation sequences were tracked using the PF described in Section III-D with the number of particles, $N = 1000$.

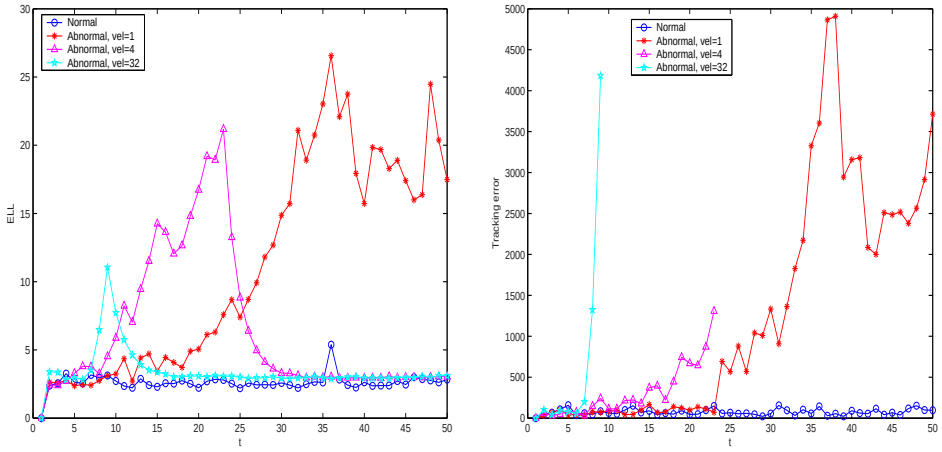
This video was provided to us by the Transport Security Administration (TSA) and did not have any instances of abnormal behavior. Abnormal behavior was simulated in software by making one of the persons walk away in an abnormal direction (in the results shown one person was made to walk away at an angle of 45^o to the X-axis, see figure 2(b); 2(a) shows a normal activity frame). Now, the person could be moving away at any speed which will make the abnormality a slow or a drastic change. We have simulated this by testing for walk away speeds of $1, 2, 4, 16, 32$ pixels per time step in both x and y directions. The average speed of any person in the normal sequence is about 1 pixel per time step. Thus walk-away velocity of 1 pixel per time step, denoted as $vel. = 1$, corresponds to a slow change which does not go out of track for a long time while $vel. = 32$ is a drastic change that causes the PF to lose track immediately.

We show change detection results and tracks using real observations of the passengers' locations in each frame obtained using a motion detection algorithm described in [34]. The ability of our algorithm to deal with temporal abnormalities [3] is demonstrated as well. We also plot the ROC curves for change detection using the ELL, the tracking error (TE) and a combination of both.



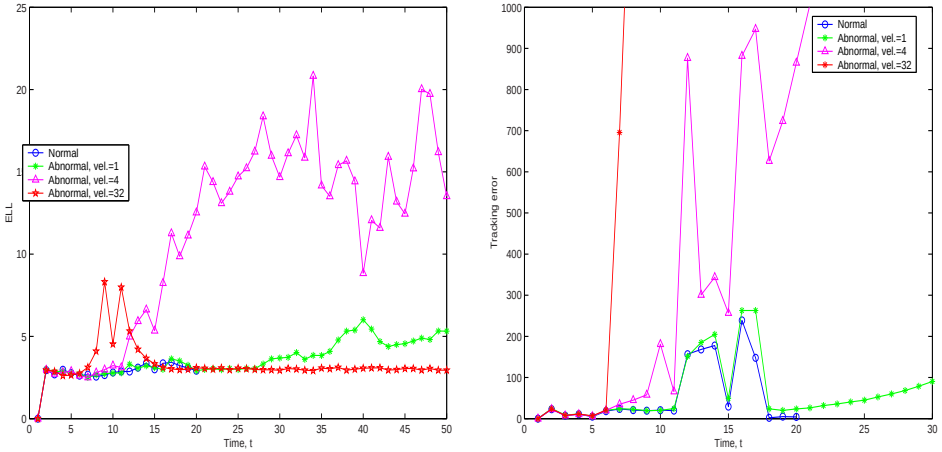
(a) A 'normal activity' frame with 4 people (b)Abnormality introduced by making one person walk-away in an abnormal direction

Fig. 2. Airport example: Passengers deplaning



(a) ELL (b) Tracking error

Fig. 3. ELL and Tracking error (TE) plots: Simulated Observation noise, $\sigma^2_{obs} = 9$ (3-pixel noise).

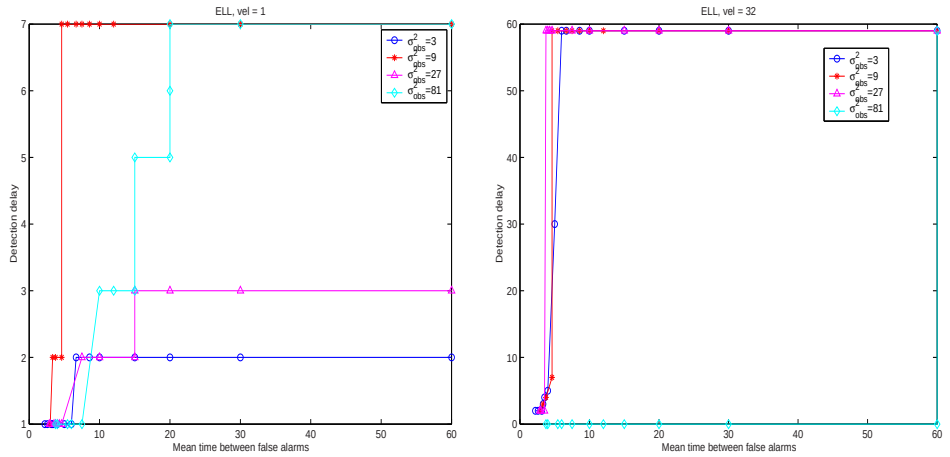


(a) ELL (b) Tracking error

Fig. 4. ELL and Tracking error plots: Real Observations. Abnormality was introduced at $t = 5$. The ELL is able to detect slow changes better while the tracking error works better for drastic changes. The plots are discussed in Section VI-B.

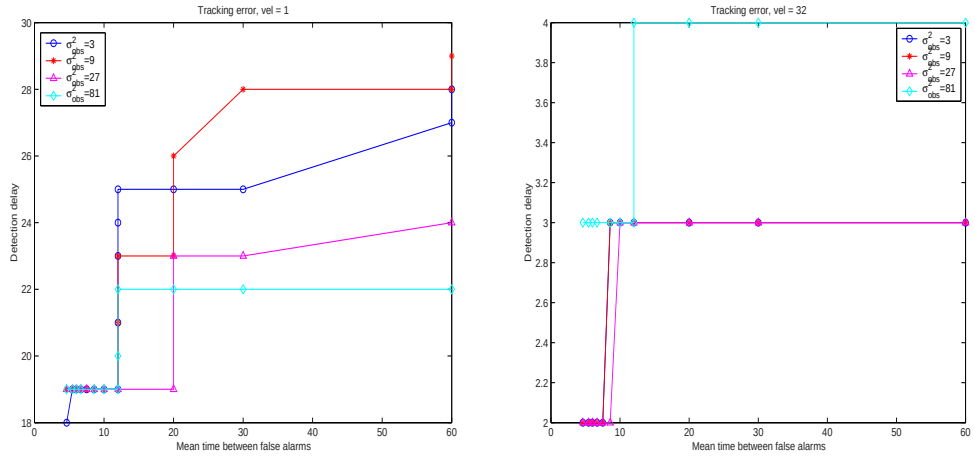
B. ELL versus Tracking Error: Slow and Drastic Changes

Figure 3 shows ELL and tracking error plots for simulated observation noise and figure 4 shows the plots for real observations. Real observations are obtained using a motion detector [34]. Observation noise is because of the sensor noise and motion detection error. Now, figure 9(b) shows a slow abnormality (vel.=1) introduced at $t = 5$ which is tracked correctly for a long time (tracking error plot is shown in figure 4(b)) and hence we need to use ELL to detect it (ELL plot is shown in figure 4(a)). Figure 9(c) shows a drastic abnormality ($vel. = 32$) which was also introduced at $t = 5$ but loses track immediately. In this case the abnormal observations are



(a) Slow change (vel.=1): **WORKS** (b) Drastic change (vel.=32): FAILS

Fig. 5. ROCs for Change detection using ELL. Blue circles, red stars, majenta triangles and cyan diamonds plots are for $\sigma^2_{obs} = 3, 9, 27, 81$ respectively. Note that the two plots have different y axis ranges. The ELL completely fails for drastic changes. Detection delays in (b) are very large (60 time units) while for the slow change maximum detection delay is only 7 time units. Plots are discussed in Section VI-C.



(a) Slow change (vel.=1): DOES NOT WORK (b) Drastic change (vel.=32): **WORKS**

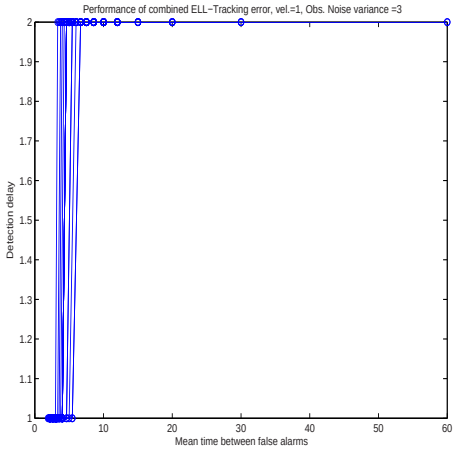
Fig. 6. ROCs for Change detection using Tracking error. Blue circles, red stars, majenta triangles and cyan diamonds plots are for $\sigma^2_{obs} = 3, 9, 27, 81$ respectively. Please note that the two plots have different y axis ranges. Tracking error does not detect slow changes easily. Detection delays in (a) are large (maximum delay is 28 time units) while drastic changes are detected almost immediately with delay ≤ 4 time units. Plots are discussed in Section VI-C.

ignored and the PF continues to follow the system model. As a result, the ELL (plot shown in figure 4(a)) confuses it for a normal sequence and fails completely, while tracking error (plot shown in figure 4(b)) detects it immediately. In figure 4(a), we show the ELL plot for increasing rates of change. With $vel. = 1$, the abnormality (introduced at $t = 5$) gets detected at $t = 27$ and with $vel. = 4$ it gets detected at $t = 12$. For $vel. = 32$, the ELL is unable to detect the abnormality. The tracking error (figure 4(b)) detects this abnormality immediately (at $t = 6$) while it misses detecting the slow abnormality ($vel. = 1$).

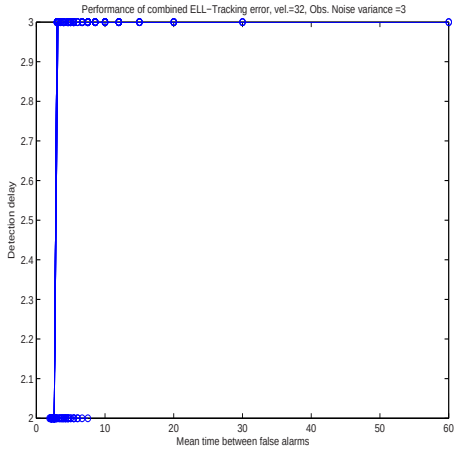
This demonstrates the need to use a combination of ELL and tracking error to detect both slow and drastic changes (since the aim is to be able to detect any kind of abnormality with rate of change not known). As explained earlier, we declare an abnormality if either the ELL or the tracking error exceeds its corresponding thresholds. The ROC curves for this combined ELL/TE strategy are shown in Figure 7. As is discussed below, by combining ELL and TE we are able to detect all slow and drastic changes with detection delay less than 7 time units.

C. ROC curves and Performance Degradation with increasing Observation Noise

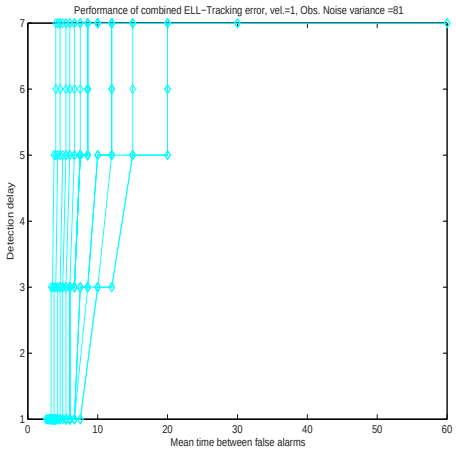
The intuition discussed above is captured numerically in the ROC (Receiver Operating Characteristic) curves [33], [40] for change detection using ELL (figure 5(a) and (b) for slow and drastic changes respectively), using tracking error (figure 6(a) and (b)) and using a combination of both (figure 7(a),(b),(c),(d)). Please note that every figure in the ROC plot has a different y axis range. The blue circles, red stars, magenta triangles and cyan diamonds are the ROC plots for simulated observation noise with increasing variances of $3, 9, 27, 81$ square pixels. The ROC for a change detection problem [40] plots the average detection delay against the



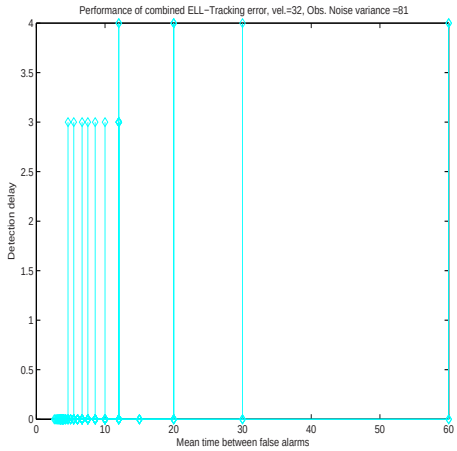
(a) $\sigma^2_{obs} = 3$, Slow change (vel.=1): **WORKS**



(b) $\sigma^2_{obs} = 3$, Drastic change (vel.=32): **WORKS**



(c) $\sigma^2_{obs} = 81$, Slow change (vel.=1): **WORKS**



(d) $\sigma^2_{obs} = 81$, Drastic change (vel.=32): **WORKS**

Fig. 7. ROCs for Change detection using the combined ELL-Tracking error. In this case, for each observation noise variance, there are multiple curves, since one needs to vary thresholds for both ELL and tracking error to get the ROC. A single curve is for the ELL threshold fixed and tracking error threshold varying. We have a set of curves for varying ELL thresholds. The maximum detection delay is 2 and 3 time units for $\sigma^2_{obs} = 3$ ((a) and (b)), and 7 and 4 time units for $\sigma^2_{obs} = 81$ ((c) and (d)). Plots are discussed in Section VI-C.

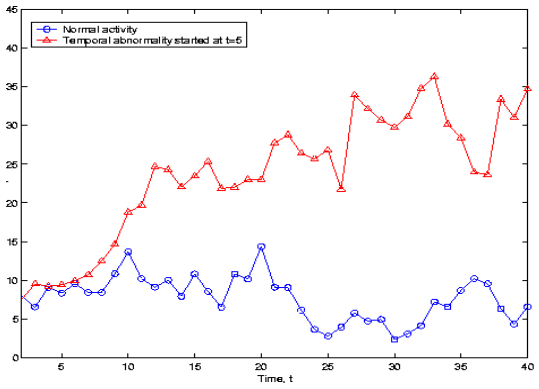
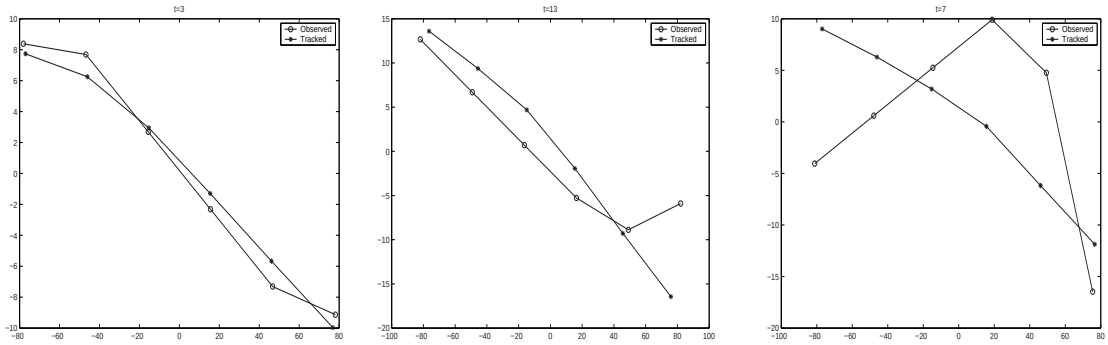


Fig. 8. ELL plot for Temporal abnormality detection. Abnormality was introduced at $t = 5$. The plot is discussed in Section VI-D.

mean time between false alarms by varying the detection threshold. The aim of an ROC plot is to choose an *operating point* threshold which minimizes detection delay for a given value of mean time between false alarms.

For the slow change ($vel. = 1$), the detection delay is much lesser using ELL than using the tracking error while the opposite is true for the drastic change ($vel. = 32$). The detection performance degradation of ELL for slow change and of tracking error for drastic change with increasing observation noise is slow. In figure 5(a) (ELL for slow change), detection delay is less than or equal to 2 time units for $\sigma^2_{obs} = 3$ and 7 time units for $\sigma^2_{obs} = 81$. In figure 6(b) (tracking error for drastic change), the detection delay is less than or equal to 3 time units for $\sigma^2_{obs} = 3$ and 4 time units for $\sigma^2_{obs} = 81$.



(a)Normal frame: In track (b)Slow abnormality ($vel = 1$): Still in track (c)Drastic abnormality ($vel = 32$): Loses track

Fig. 9. Tracks: Real Observations. Plotting the observed and tracked positions of the landmarks (passengers) on the x-y plane. The plots are discussed in Section VI-E.

Since the aim is to be able to detect all kinds of abnormalities (abnormality parameters are assumed not known), we propose to use a combination of the ELL and the tracking error and declare a change when either exceeds its threshold. In figure 7, we plot the ROC curves for slow and drastic change detection using a combination of ELL and tracking error. In this case, for each observation noise variance, there are multiple curves, since one needs to vary thresholds for both the ELL and the tracking error to get the ROC. A single curve is for the ELL threshold fixed and tracking error threshold varying. We have a set of curves for varying ELL thresholds. We plot the low and high observation noise cases in two separate plots. As can be seen, the combined strategy has better performance than either ELL or tracking error for all rates of change and for all observation noises (detection delay less than 7 time units in all cases).

D. Temporal abnormality [3] detection

We also tested our method for detecting what is referred to in [3] as a temporal abnormality (one person stopped in his or her normal path). It gets detected in this framework because there is a change in shape when the person behind the stopped person goes ahead of him (curve becomes concave). We used "uniform resampling" (discussed in Section V) which detected temporal abnormality easily using ELL (figure 8). "Arc-length resampling" does not work too well in this case. This is because it tends to average out the locations of two closely spaced points, thus smoothing out the concavity which needs to be detected. "Uniform resampling", on the other hand, assumes the observed points are uniformly sampled and hence gives equal weight to all the observed points irrespective of the distances between them. Thus it is able to detect concavity caused even by two closely spaced points. Another way to detect temporal abnormality would be to use a NSSA model and look at deviations from the expected value of shape velocity.

E. Tracks

Figure 9(a) shows a normal observation frame (circles) and the corresponding tracked configuration (stars), for real observations obtained using a motion detector [34] on the image sequences. The observation noise was modeled to be Gaussian (although the PF can filter non-Gaussian noise as well) and its covariance was learnt from a training sequence of observations obtained using the motion detector. This shows the ability of our model to potentially be used for "tracking to obtain observations". Figures 9(b),(c) show tracking of a slow ($vel = 1$) and drastic ($vel = 32$) abnormality both introduced at $t = 5$. As can be seen, the drastic abnormality has lost track at $t = 7$ while the slow one is not totally out of track even at $t = 13$. The NSSA model tracks abnormality better [6]. Note that since we use only a point object abstraction for moving objects (here persons), we show observed and tracked point object locations only without showing the actual images.

VII. Extensions

A. Tracking to Obtain Observations

In the entire discussion till now, we used a PF in the filtering mode to estimate the probability distribution of shape from noisy observations and used this distribution for abnormality detection. But the PF also provides at each time instant the prediction distribution, $\pi_t(X_t|Y_{1:t-1})$, which can be used to predict the expected configuration at the next time instant using past observations, i.e. $\hat{Y}_t \stackrel{\triangle}{=} E[Y_t|Y_{0:t-1}] = E_{\pi_{t|t-1}}[h(X_t)]$. We can use this information to improve the measurement algorithm used for obtaining the observations (a motion detector [34] in our case). Its computational complexity can be reduced and its ability to ignore outliers can be improved by using the predicted configuration and searching only locally around it for the current observation[7]. As we show in Section VI-E, the observed configuration is close to its prediction when there is no abnormality or change and hence the prediction can be used to obtain the observation. An SSA model can track a normal activity while the NSSA is able to track abnormality as well (shown in [6]).

[7]One thing to note here is that in certain cases (for example, if the posterior of any state variable is multimodal), evaluating the posterior expectation as a prediction of the current observation is not the correct thing to do. In such a case, one can track the observations using the CONDENSATION algorithm [26] which searches for the current observation around each of the possible $h(\bar{x}_t^i), i = 1, 2...N$.

If used in this "tracking observations and filtering" framework, a lot of drastic abnormalities can be detected at the measurement stage itself because no observations will be found in the "vicinity" (region of search defined using observation noise variance) of the predicted position. But an outlier might get confused with a drastic abnormality since even for an outlier we will not find any observation in the "vicinity". The difference is that outliers would be temporary (one or two time instants and then the PF comes back in track), while a drastic abnormality will appear to be an outlier for a sequence of frames. Thus by averaging the number of detects over a sequence of past time instants, we can separate outliers from real abnormalities.

Also, if the configuration is a moving one, then the predicted motion information can be used to translate, zoom or rotate the camera (or any other sensor) to better capture the scene but in this case, one would have to alter the motion model to include a control input.

B. Activity Sequence Identification and Tracking

Consider two possible situations for tracking a sequence of activities. Assume each activity is represented by an SSA so that the sequence of activities is characterized by a PSSA (discussed in [6]). The mean shape of each SSA component is known but the transition times are assumed unknown.

1) First consider the simple case when there are just two possible activities and their order of occurrence is known, only the change time is unknown. In this case, one can detect the change using ELL (before the particle filter loses track) and then start tracking it with the second activity's transition model.
2) Now consider the general case when a sequence of activities occur, and we do not know the order in which they occur. In this case, we can use a discrete mode variable as part of the state vector to denote each activity type. We make the state transition model a mixture distribution and keep the mode variable as a state. Whenever a change occurs, it takes the mode variable a few time instants to stabilize to the correct mode. One could replace the multimodal dynamics with that of the detected mode once the mode variable has stabilized. Also, in this case we can declare an activity to be abnormal (i.e. neither of the known activity types) if the ELL w.r.t all known models exceeds a threshold.

VIII. Conclusions and Future Directions

In this paper we have presented a "shape activity model", which is a continuous state Hidden Markov Model for the changing configuration of a set of moving landmarks. The shape and global motion parameters constitute the hidden state vector and the observed landmark locations form the observation vector. The state dynamics and the mapping between the state and the observation is nonlinear and hence the shape is estimated from the noisy observations using a particle filter. Abnormal activity detection is formulated as a change detection problem with change parameters being unknown and change being slow or drastic. We have used a change detection strategy using particle filters which has been proposed and analyzed by us in past work [30], [31], [41]. Experimental results have been shown for abnormal activity detection in an airport scenario.

As part of future work, we hope to implement joint tracking and abnormality detection and tracking a sequence of activities (discussed in Section VII). Also, in this work, we have experimented only with stationary shape activities. We are currently studying the non-stationary case (discussed in Section III-C) in more detail. We hope to characterize (define a pdf for) specific instances of a normal activity in the non-stationary case and to define the abnormality detection problem. The non-stationary shape activity model provides the flexibility to model and track a much larger class of group activities. We are also experimenting with a piecewise stationary shape activity model which can be used along with ELL for activity sequence segmentation and tracking.

The issue of time-varying number of landmarks needs to be studied more rigorously by first defining the optimality criterion to make the interpolation problem well-posed and then deciding the best strategy. Also, the current shape space ($\Re^{km}$ modulo Euclidean similarity transformations) can be replaced by general shape spaces, for example, the affine shape space (chapter 12 of [2]) would be useful to make the activity invariant to an affine camera's motion. Finally, we plan to apply our framework to many other applications (discussed in the introduction).

Acknowledgement

We would like to acknowledge Mr. Fumin Zhang and Prof. Andre Tits of the ECE dept. at the University of Maryland, College Park for interesting discussions on the work.

References

[1] D. Kendall, D. Barden, T. Carne, and H. Le, *Shape and Shape Theory*. John Wiley and Sons, 1999.
[2] I. Dryden and K. Mardia, *Statistical Shape Analysis*. John Wiley and Sons, 1998.
[3] W. Grimson, L. Lee, R. Romano, and C. Stauffer, "Using adaptive tracking to classify and monitor activities in a site," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Santa Barabara, CA, 1998, pp. 22–31.
[4] T. Huang, D. Koller, J. Malik, G. Ogasawara, B. Rao, S. Russell, and J. Weber, "Automatic symbolic traffic scene analysis using belief networks," in *American Association for Artificial Intelligence Conference*, 1994, pp. 966–972.
[5] J. Spletzer, A. Das, R. Fierro, C. Taylor, V. Humar, and J. Ostrowski, "Cooperative localization and control for multi-robot manipulation," in *Proceedings of the Conference on Intelligent Robots and Systems (IROS)*, Hawaii, USA, 2001.
[6] N. Vaswani, *Change Detection in Stochastic Shape Dynamical Models with Applications in Activity Modeling and Abnormality Detection*. Ph.D. Thesis, ECE Dept, University of Maryland at College Park, August 2004.
[7] C. Zahn and R. Roskies, "Fourier descriptors for plane closed curves," *IEEE Transactions on Computers*, vol. C-21, pp. 269–281, March 1972.
[8] D. F. Rogers and J. A. Adams, *Mathematical Elements for Computer Graphics*. WCB/McGraw-Hill, 1990.
[9] T. Cootes, C. Taylor, D. Cooper, and J. Graham, "Active shape models: Their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, January 1995.
[10] ——, "Training models of shape from sets of examples," in *British Machine Vision Conference (BMVC)*, 1992, pp. 9–18.

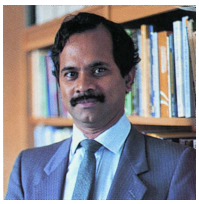
[11] J. Kent, "The complex bingham distribution and shape analysis," in *Journal of the Royal Statistical Society, Series B*, 1994, pp. 56:285–299.
[12] Y. Zhou, L. Gu, and H. Zhang, "Bayesian tangent space model: Estimating shape and pose parameters via bayesian inference," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Madison, WI, June 2003.
[13] C. Small, *The Statistical Theory of Shape*. Springer, New York, 1996.
[14] A. Srivastava and E. Klassen, "Geometric filtering for subspace tracking," *Advances in Applied Probability*, vol. 36(1), March 2004.
[15] A. Chiuso and S. Soatto, "Monte-Carlo filtering on Lie groups," in *IEEE Conference on Decision and Control (CDC)*, Sydney, Australia, December 2000.
[16] S. Kurakake and R. Nevatia, "Description and tracking of moving articulated objects," in *International Conference on Pattern Recognition (ICPR)*, The Hague, Netherlands, August 1992, pp. I:491–495.
[17] T. Starner and A. Pentland, "Visual recognition of american sign language using hidden Markov models," in *Proc. Intl. Workshop on Face and Gesture Recognition*, 1995.
[18] C. Bregler, "Learning and recognizing human dynamics in video sequences," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1997, pp. 568–574.
[19] A. Bobick and Y. Ivanov, "Action recognition using probabilistic parsing," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Santa Barbara, California, 1998, pp. 196–202.
[20] A. Roy Chowdhury and R. Chellappa, "A factorization approach for event recognition," in *CVPR Event Mining Workshop*, Madison, WI, June 2003.
[21] L. Torresani and C. Bregler, "Space-time tracking," in *European Conference on Computer Vision (ECCV)*, Copenhagen, Denmark, May 2002.
[22] L. Zelnik-Manor and M. Irani, "Event based analysis of video," in *IEEE International Conference on Computer Vision (ICCV)*, Vancouver, Canada, 2001.
[23] D. P. T. Syeda-Mahmood, "Recognizing action events from multiple viewpoints," in *IEEE Workshop on Detection and Recognition of Events in Video*, Vancouver, Canada, July 2001.
[24] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/nongaussian bayesian state estimation," *IEE Proceedings-F (Radar and Signal Processing)*, pp. 140(2):107–113, 1993.
[25] A. Doucet, N. deFreitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. Springer, 2001.
[26] J. MacCormick and A. Blake, "A probabilistic contour discriminant for object localisation," in *IEEE International Conference on Computer Vision (ICCV)*, Mumbai, India, January 1998.
[27] S. Zhou and R. Chellappa, "Probabilistic human recognition from video," in *European Conference on Computer Vision (ECCV)*, Copenhagen, Denmark, May 2002, pp. 681–697.
[28] D. Schulz, W. Burgard, D. Fox, and A. Cremers, "Tracking multiple moving targets with a mobile robot using particle filters and statistical data association," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, Seoul, Korea, May 2001.
[29] B. Azimi-Sadjadi and P. Krishnaprasad, "Change detection for nonlinear systems: A particle filtering approach," in *American Control Conference (ACC)*, Anchorage, Alaska, May 2002.
[30] N. Vaswani, "Change detection in partially observed nonlinear dynamic systems with unknown change parameters," in *American Control Conference (ACC)*, Boston, MA, June 2004.
[31] N. Vaswani, A. Roychowdhury, and R. Chellappa, "Activity recognition using the dynamics of the configuration of interacting objects," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Madison, WI, June 2003.
[32] A. J. Yezzi and S. Soatto, "Deformotion: Deforming motion, shape average and the joint registration and approximation of structures in images," *Int. J. Comput. Vision*, vol. 53, no. 2, pp. 153–167, 2003.
[33] A. Papoulis, *Probability, Random Variables and Stochastic Processes*. McGraw-Hill, Inc., 1991.
[34] Q. Zheng and S. Der, "Moving target indication in LRAS3 sequences," in *5th Annual Fedlab Symposium College Park MD*, 2001.
[35] T. Kailath, A. Sayed, and B. Hassibi, *Linear Estimation*. Prentice Hall, 2000.
[36] D. Crisan and A. Doucet, "A survey of convergence results on particle filtering for practitioners," *IEEE Trans. Signal Processing*, vol. 50, no. 3, pp. 736–746, 2002.
[37] F. LeGland and N. Oudjane, "Stability and Uniform Approximation of Nonlinear Filters using the Hilbert Metric, and Application to Particle Filters," *Technical report, RR-4215, INRIA*, 2002.
[38] N. Vaswani, A. RoyChowdhury, and R. Chellappa, "Statistical shape theory for activity modeling," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2003.
[39] N. Vaswani, "Slow and drastic change detection in general HMMs using particle filters with unknown change parameters," *Submitted to IEEE Trans. on Signal Processing*, 2004.
[40] M. Basseville and I. Nikiforov, *Detection of Abrupt Changes: Theory and Application*. Prentice Hall, 1993.
[41] N. Vaswani, "Bound on errors in particle filtering with incorrect model assumptions and its implication for change detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Montreal, Canada, May 2004.

Namrata Vaswani received the B.Tech. degree in Electrical Engineering from the Indian Institute of Technology (I.I.T.), Delhi in 1999 and the Ph.D. in Electrical and Computer Engineering from the University of Maryland, College Park in August 2004. Her Ph.D. thesis was on change detection in stochastic shape dynamical models and applications to activity modeling and abnormal activity detection.

Her research interests are in the broad area of signal and image processing, in particular change detection using particle filters (theoretical issues as well as applications) and in shape analysis and optimization. She has also worked on subspace methods for pattern classification. She is currently a Postdoctoral Fellow in the School of Electrical and Computer Engineering at Georgia Tech where she is studying variational calculus methods for curve and surface evolution and their application to shape segmentation, registration and tracking.

Amit K. Roy-Chowdhury received the B.S. degree in Electrical Engineering from Jadavpur University, India in 1995, the M.S. degree in Systems Science and Automation from the Indian Institute of Science, Bangalore in 1997 and Ph.D. from the Dept. of Electrical and Computer Engineering, University of Maryland, College Park in 2002. His PhD thesis was on statistical error characterization of 3D modeling from monocular video sequences.

He is an Assistant Professor in the Dept. of Electrical Engineering, University of California, Riverside. He was previously with the Center for Automation Research, University of Maryland as a Research Associate, where he worked in projects related to face, gait and activity recognition. He is presently coauthoring a research monograph on recognition of humans and their activities. His broad research interests are in signal, image and video processing, computer vision and pattern recognition.



Rama Chellappa received the B.E. (Hons.) degree from the University of Madras, India, in 1975 and the M.E. (Distinction) degree from the Indian Institute of Science, Bangalore, in 1977. He received the M.S.E.E. and Ph.D. Degrees in electrical engineering from Purdue University, West Lafayette, IN, in 1978 and 1981 respectively.

Since 1991, he has been a Professor of electrical engineering and an affiliate Professor of computer science at the University of Maryland, College Park. He is also affiliated with the Center for Automation Research (Director) and the Institute for Advanced Computer Studies (Permanent member). Prior to joining the University of Maryland, he was an Assistant (1981-1986) and Associate Professor (1986-1991) and Director of the Signal and Image Processing Institute (1988-1990) with the University of Southern California, Los Angeles. Over the last 21 years, he has published numerous book chapters, peer-reviewed journal and conference papers. He has edited a collection of Papers on Digital Image Processing (published by IEEE Computer Society Press), co-authored a research monograph on Artificial Neural Networks for Computer Vision (With Y.T. Zhou) published by Springer-Verlag, and co-edited a book on Markov Random fields (with A.K. Jain) published by Academic Press. His current research interests are face and gait analysis, 3D modeling from video, automatic target recognition from stationary and moving platforms, surveillance and monitoring, hyper spectral processing, image understanding, and commercial applications of image processing and understanding.

Dr. Chellappa has served as an associate editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IMAGE PROCESSING, and NEURAL NETWORKS. He was co-Editor-in- Chief of Graphical models and Image Processing. He is now serving as the Editor-in-Chief of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE. INTELLIGENCE. He served as a member of the IEEE Signal Processing Society Board of Governors during 1996-1999. Currently he is serving as the Vice President of Awards and Membership for the IEEE Signal Processing Society. He has received several awards, including NSF Presidential Young Investigator Award, an IBM Faculty Development Award, the 1990 Excellence in Teaching Award from School of Engineering at USC, and the 1992 Best Industry Related Paper Award from the International Association of Pattern Recognition (with Q. Zheng), the 2000 Technical Achievement Award from the IEEE Signal Processing Society. He was elected as a Distinguished Faculty Research Fellow (1996-1998) at the University of Maryland. He is a Fellow of the International Association for Pattern Recognition. He has served as a General the Technical Program Chair for Server IEEE international and national conferences and workshops.