

# Object Trajectory-Based Activity Classification and Recognition using Hidden Markov Models

Faisal I. Bashir, Ashfaq A. Khokhar, and Dan Schonfeld

Department of Electrical and Computer Engineering  
University of Illinois at Chicago  
851 S. Morgan Street  
Chicago, IL, 60607  
{fbashir,ashfaq,ds}@ece.uic.edu

**Abstract**— Motion trajectories provide rich spatio-temporal information about an object’s activity. Developing scalable activity recognition algorithms based on this high dimensionality cue is an extremely challenging task. This paper presents novel classification algorithms for recognizing object activity using object motion trajectory. The trajectory information can be obtained using a tracking algorithm on data streams available from a range of devices including motion sensors, video cameras, haptic devices, etc. In the proposed classification system, trajectories are segmented at points of change in curvature and the *subtrajectories* are represented by their Principal Component Analysis (PCA) coefficients. We first present a framework to robustly estimate the multivariate probability density function (PDF) based on PCA coefficients of the subtrajectories using Gaussian Mixture Models (GMM). We show that GMM-based modeling alone cannot capture the temporal relations and ordering between underlying entities. To address this issue, we use Hidden Markov Models (HMM) with a data-driven design in terms of number of states and topology (e.g. left-right versus ergodic). Different classes of object motions are modeled by a Continuous HMM (i.e. state observations are drawn from a continuous PDF) per class, where the state PDFs are represented by GMMs. Experiments using a database of over 5700 complex trajectories (obtained from UCI-KDD data archives and Columbia University Vision Group) subdivided into 85 different classes demonstrate the superiority of our proposed HMM-based scheme using PCA coefficients of subtrajectories in comparison with other techniques in the literature.

**Index Terms**— Trajectory Modeling, Activity Recognition, Principal Component Analysis, Hidden Markov Models, Gaussian Mixture Models.

## 1 INTRODUCTION

Object motion trajectory-based analysis and recognition has gained significant interest in scientific circles lately. This trend can be attributed mainly to two different reasons. First, spatio-temporal data derived from object motion is becoming more easily available due to advances in sensor technology and computing techniques [11]. On the hardware side, advancements in sensor technology are resulting in low-cost versatile sensors. On the software side, advancements in computer vision have led to the design of robust object trackers that can handle occlusions, shape deformations and intensity changes in single- and multi- camera settings. Second, novel applications employing analysis of motion trajectory are emerging due to enhanced interest in homeland security as well as due to prevalence of multimedia gadgets in commercial and scientific endeavors. Examples of the motion trajectory include tracking results from video trackers, sign language data measurements gathered from wired glove interfaces fitted with sensors, Global Positioning System (GPS) coordinates of satellite phones, cars using Car Navigation Systems (CNS), animal mobility experiments, etc. This spatio-temporal data embodies semantically rich information about the behavior of the object of interest, the action performed and the interaction among groups of objects [1], [11]. For example, in sign and gesture recognition, the signer moves his hands in specific pattern for a particular word. In sports video trajectory analysis and understanding can assist the players, coaches and sports analysts with strategies used on the field based on the motion patterns of players and their mutual interaction. Another important area is automatic video surveillance which is used, for example, in real-time observation of people and vehicles, in a busy environment, leading to a description of actions and mutual interactions. This application arises in scenarios as diverse as indoor and outdoor home and office scenes, railway and subway stations, parking lots, elevator and retail store videos, highway videos, etc. The complexity of the problem is exacerbated by low-resolution, weather-dependant video capture and the presence of multi-camera surveillance systems. The research challenge here is to quickly learn the permitted activities and set an alarm at any illegal or abnormal activity being performed. We emphasize that object motion plays the key role in the domain of activity analysis in general and in video surveillance in particular [41]. Psychological studies have shown that human beings can routinely discriminate and recognize this kind of object motion using motion

pattern, even in large viewing distances or poor visibility conditions; whereas, other cues such as clothes, appearance, or hair style tend to vanish at large distances or poor visibility conditions [25].

Nevertheless, developing high-accuracy activity classification and recognition algorithms using motion trajectories is still an extremely challenging task particularly when the number of activities to be recognized is relatively large ( $> 10$ ). The object trajectory is typically modeled as a sequence of consecutive locations of the object on a coordinate system resulting in a vector in 2-D or 3-D Euclidean space. The measurement parameters, at each point in time, needed for object localization can be arbitrarily high-dimensional vectors including x and y-projections (latitude and longitude), distance (height or depth), silhouette of the object shape, and other metadata corresponding to object appearance and environment.

In different trajectory-based applications, as with multimedia analysis in the broad sense, there are two major cornerstones for successful system development: a compact and robust representation of the trajectories to capture the spatio-temporal movement patterns; and a semantically meaningful high-level description of the activities, actions and events based on this trajectory data [1]. This paper is focused on both of these issues and introduces a novel method employing Gaussian mixture models (GMM)-based representations and hidden Markov model (HMM)-based classifiers for motion trajectory representation and analysis. We rely on arbitrary object tracking sensors or systems to provide successive spatio-temporal coordinates of objects. We use Principal Component Analysis (PCA) to represent trajectories using a compact set of features to obtain a reduced-dimensional space [26]. We apply this approach to segments of object trajectories which form perceptually similar atomic units that we shall refer to as *subtrajectories*. A statistically robust mechanism for motion representation based on trajectory segmentation is presented. The set of these subtrajectories in PCA subspace is then used to learn the statistical models for each class. PDF estimation of multidimensional subtrajectory data is proposed using Gaussian mixture models (GMM). Finally, we represent trajectories as temporal sequences of subtrajectories analogous to the characterization of words as a sequence of phonemes. We subsequently propose the representation of motion trajectories using ergodic Hidden Markov models (HMM). All HMM parameters including the topology are learnt from training datasets. Here the topology refers to the interconnections between states of the HMM which can be either left-to-right (i.e. state transitions are only allowed from states on left to states on the right), or ergodic (i.e.

state transitions between any two states are allowed) [38]. The performance of classification of motion trajectories using GMM and HMM-based models is analyzed and computer simulation experiments are used to compare the proposed classifiers with various existing methods. The comparisons are performed in terms of *receiver operating characteristics* (ROC) as well as model-based distances using posterior likelihoods [35]. Experiments are conducted on two datasets: the Australian Sign Language (ASL) dataset obtained from University of California at Irvine’s Knowledge Discovery in Databases archive [23], and the sports video dataset (HJSL) provided by Columbia University’s Digital Video and Multimedia Group (DVMM) [14]. The ASL dataset used in our experiments has trajectories for 83 word classes as signed by 5 signers of varying skill levels. Each word class has 69 trajectories for a total of 5,727 samples. The HJSL dataset is obtained from video sources after tracking and has 40 trajectories of high jumpers and 68 trajectories of slalom skiing objects for a total of 108 trajectories.

The remaining sections of this paper are organized as follows: Section 2 surveys related work on low-level and semantics-based trajectory representation and analysis. Section 3 briefly describes our trajectory segmentation and PCA-based representation. Section 4 presents the model-based representation of object trajectories for complex action recognition using both Gaussian mixture models and Hidden Markov models. In order to analyze the quality of the proposed classifier, we also discuss the Kullback-Leibler divergence between GMMs and HMMs. Section 5 provides a comparison of the model-based trajectory representation methods described above with other methods reported in the literature in connection with image and video genre-recognition. The details of the datasets used and the experiments conducted are also provided in Section 5. Section 6 provides a discussion of the results of the computer experiments. Finally, in Section 7, we present a brief summary and conclusion and outline future research in this area.

## 2 RELATED WORK

This section provides a survey of the related work from recent literature in the areas of trajectory representation, statistical modeling and applications of trajectory-based representation and learning. Studies into human psychology have shown the extra-ordinary ability of human beings to recognize object motion even from minimal information system such as Moving Light Displays (MLDs). Such displays are obtained by making a video of

moving subjects wearing reflective pads/light bulbs on their body joints in almost dark conditions. In spite of the paucity of information, human observers easily perceive not only motion but also the kind of motion; e.g., walking, running, dancing, cycling, etc. [25]. Based on this understanding, object motion has been an important feature for the representation and discrimination of one object from another in video applications. Earlier approaches in motion-based methods focused on object tracking from raw and compressed domain videos [1], [18], [20], [24], [42], [43]. Indexing and searching based on object motion as the dominant cue has attracted a lot of research activity in the past few years [15]. Chen et. al. [14] segment each trajectory into subtrajectories using fine-scale wavelet coefficients at high levels of decomposition. A feature vector is then extracted from each subtrajectory comprising of features like acceleration, velocity, subtrajectory length, etc. Finally, distances between each subtrajectory in query trajectory and all the indexed subtrajectories are computed to generate a list of similar trajectories in the database. This approach suffers from the fact that the representation is based on ad hoc features which are not tolerant to affine transformations of the trajectories. Also, the feature vectors lie in a non-uniform space, so the matching process has to compute the overall distance based on weighted average of individual features. Our previous work on trajectory indexing and retrieval [1] segments the trajectories based on dominant sign changes in curvature data. We represent the subtrajectories using PCA coefficients. We have addressed the view-invariant representation of trajectories for scenarios where similar trajectories are captured from different view points [2]. View-invariant representation has also been addressed in [39] for modeling and recognizing actions performed by individuals in video sequences. The representation is based on dynamic instants (segmentation points) of the trajectories. For each dynamic instant in the trajectory, frame number, location of the hand and ‘sign’ of the instant (-ve for counter clockwise turn and +ve for clockwise turn) is stored. The matching is performed on trajectories with the same number of dynamic instants and same sign permutations. This approach, though compact in representation, cannot be used for partial trajectory processing or generic trajectory representation.

Semantics-based processing of trajectory data to extract high-level information has gained interest quite recently [7]. Yacoob et. al. [53] have presented a framework for modeling and recognition of human motions based on principal components. Each activity is represented by eight motion parameters recovered from five body parts of

the human walking scenario. In [40], a semantic event detection technique for snooker videos is presented. Trajectory of the white ball is generated using a color-based particle filter. The implementation of the particle filter allows for ball collision detection and ball pot detection. A separate ball track is instantiated upon detection of a collision and the state of the new ball can be monitored. The evolution of the white ball position is modeled using a discrete HMM. In [1], the issue of recognizing a set of plays from American football videos is considered. Using a set of classes each representing a particular game plan and computation of perceptual features from trajectories, the propagation of uncertainty paradigm is implemented using automatically generated Bayesian network. The problem with above approaches is that they are highly domain-dependant, with domain knowledge and sensor dependence on video data being intimately woven into the systems. A sensor-independent approach towards modeling activity performed by a group of objects (persons, cars, etc.) is presented in [48]. Objects in scene are taken as points and they consider the 'shape' formed by a configuration of point objects at a given time instant. This 'shape' is tracked over time, normal shape is learnt and abnormality is detected as perturbation in this shape. Although robust for multi-agent abnormal activity detection, this approach can not be applied for single object trajectories. Vinciarelli et al [49] have used PCA and ICA (Independent Component Analysis) along with HMMs for word recognition in hand writing recognition application. De la Torre et al. [16] use PCA and HMM for tracking and recognition for lip-tracking and eye-tracking. Martin et al. [29] model the trajectories for gesture recognition using multidimensional histogram of gestures. In their approach, no segmentation to obtain subtrajectories is performed; only the recent history is taken into account. The results are reported in terms of head movements for two gestures of 'Yes' and 'No'; four single stroke letters from graffiti characters 'A', 'H', 'L', 'O'; five expressions for facial expression analysis from gray scale images of size 44x60. Starner and Pentland [45] address the issue of American sign language recognition from video sequences. An 8-element feature vector is obtained consisting of each hand's x and y positions, angle of axis of least inertia, and eccentricity of bounding ellipse is used. Bettinger et al [5] address the problems of learning a person's facial behaviors from video sequences and synthesizing sequences demonstrating the same behavior. A sequence of a face is represented as a parameter sequence labeled as a trajectory in parameter space, which is then segmented into subtrajectories. HMMs are then trained on this data to learn the facial behavior models. It is important to

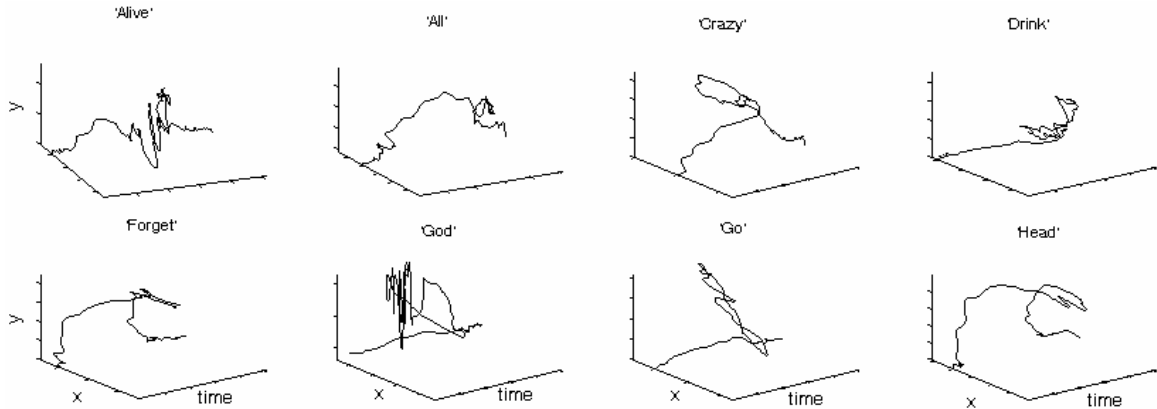
point out that the notion of trajectory, the process of segmentation and representation used in [5] are entirely different than the method presented in this paper. Another approach, which has originally been proposed in the context of face recognition by Moghaddam et. al. [30], can be directly modified for trajectory processing as outlined in Section 5.1. We have reported some preliminary findings towards trajectory-based activity recognition using Gaussian mixture models [3], hidden Markov models [4] and neural networks [37]. In the forthcoming sections, we present our model-based recognition system that uses GMMs and HMMs for trajectory modeling based on optimal representation provided by PCA. While GMM and HMM have been used as a tool in recognition tasks such as speech recognition, face recognition, etc., their use in motion trajectory representation and classification through PCA of subtrajectories presented in this paper is novel.

### 3 PCA-BASED SUBTRAJECTORY REPRESENTATION

A moving object registers its location in the three-dimensional space-time at each successive time instant. These locations can be recorded using a suitable set of sensors and localization algorithms. In the case of video data, typically the object of interest is tracked in successive frames of video clips. The result of this trajectory generation process is a two-dimensional N-tuple corresponding to the x and y-axes projections of the object's centroid at each instant of time,  $\{(X_k, Y_k), k = 1, \dots, N\}$ . We classify the trajectories into separate classes. The word "class" refers to a type of activity (represented by its full trajectory) for which we have a sufficient number of samples to train the system. As an example, we provide a small snapshot of the ASL dataset we are using in this presentation. Figure 1 depicts one representative trajectory per class for 8 different classes in this dataset. In this setting, a class refers to a single word signed by a few signers (e.g. the word 'Alive' is a class, for which we have 69 sample trajectories signed by various signers).

This section provides a very brief overview of our trajectory representation scheme based on trajectory segmentation and PCA. We recognize that most often full trajectory information is unavailable in video tracking applications due to occlusions. This limitation requires trajectory representation methods that can perform well even in the case of partial trajectory information. We address this problem by segmenting the trajectories at the

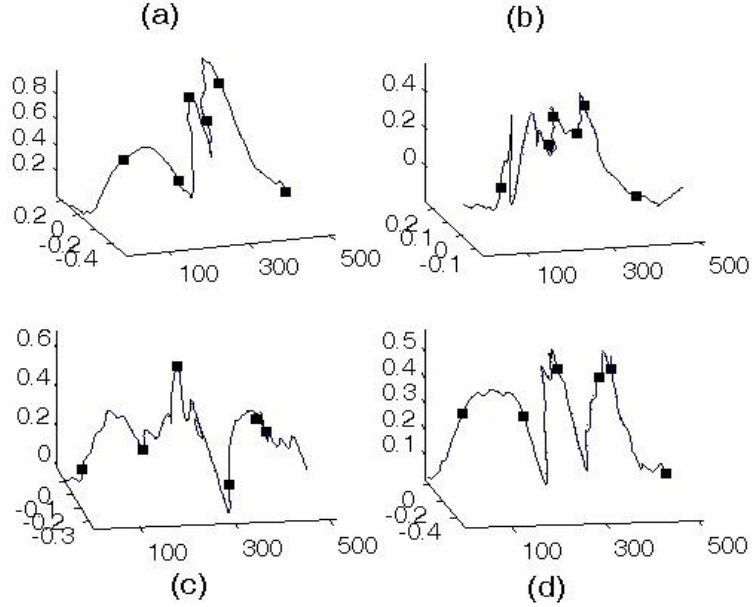
points of perceptual discontinuities. The other concern in trajectory modeling is its compact representation for efficient distance computation. For this purpose, we use the PCA-based representation of subtrajectories.



**Figure 1: One representative trajectory per class for 8 different classes in Australian Sign Language (ASL) dataset.**

Our subtrajectory-based representation has several advantages: Firstly, it is motivated by motion perception in humans which is highly dependent on piecewise segments based on atomic units of actions [25]. Second, it facilitates the modeling and recognition of trajectories which only have partial trajectory information available. And, finally, this process decomposes the large trajectory data to a few temporally-sequenced state vectors which obey the first-order Markovian property. We elaborate more on this point in the following section. For the purpose of segmentation, the discontinuities in the trajectory are detected with the help of velocity (1<sup>st</sup>-derivative) and acceleration (2<sup>nd</sup>-derivative). From the  $x$  and  $y$ -projections of the trajectory data, we compute the curvature which measures the sharpness of a bend in a 2-D curve and captures derivatives up to 2<sup>nd</sup>-order. The curvature is given by:

$$\kappa[k] = \frac{x'[k]y''[k] - y'[k]x''[k]}{[x'[k]^2 + y'[k]^2]^{3/2}} \quad (1)$$



**Figure 2:** Trajectory Segmentation (a) & (b): Segmentation of trajectories for ‘Norway’ signed by two different signers. (c) & (d): ‘Alive’ signed by same two signers.

Here,  $x'[k]$  refers to first derivative of the x-projection of the trajectory, and similar self-explanatory notation is used for the other variables in eq. (1). We perform a hypothesis testing-based process to locate the points of maximum change of the curvature data. These inflection points are detected using a likelihood ratio test-based approach. Figure 2 illustrates the results of segmentation on trajectories of two different word classes signed by two different signers. A more complete discussion of the trajectory segmentation scheme used can be found in [2]. We represent the subtrajectories using PCA because of its optimal energy compaction properties resulting from custom bases derived from the data [26]. The median segment size of the set of subtrajectories is computed from the segmentation results. The x- and y- data of each subtrajectory are concatenated into a single vector which is then resampled to twice the median segment size determined before. The resampling is done using a poly-phase filter implementation. During resampling, an anti-aliasing (low pass) filter is applied to the data which uses a Kaiser window. These xy-subtrajectory vectors are then stacked to form a dataset of individual subtrajectories corresponding to all the trajectories in the entire dataset. We concatenate the resampled x- and y-data of each subtrajectory into a single xy-vector for a combined representation. All the vectors of trajectories from all the classes are stacked to form one data matrix. The principal components of this data matrix are then estimated using eigenspace decomposition of the estimated covariance matrix [26]. To achieve dimensionality reduction, only the

first  $M$  principal components (PCs) are retained to form the transformation matrix  $\Phi_M$ . The pool of subtrajectories is finally represented by their PCA coefficients using the transformation:

$$Y = \Phi_M^T [X - \bar{X}] \quad (2)$$

where  $X$  denotes the data matrix of subtrajectories,  $\bar{X}$  is the vector containing the mean of the dataset, and  $Y$  is the matrix containing the PCA coefficients of all subtrajectories. The set of PCA coefficients of all subtrajectories for each class are subsequently used to train a stochastic model for each class as explained in the next section.

## 4 MODEL-BASED RECOGNITION OF OBJECT TRAJECTORIES

Model-based recognition and classification has been extensively used in applications such as sign language recognition, action recognition [7], sports video analysis [52], speech/speaker recognition [12], accent classification [12], etc. In this section, we build on the PCA-based subtrajectory representation outlined in the previous section for modeling different classes of object motion patterns. In Section 4.1, we present the GMM-based modeling, wherein the multimodal PDF of each class is represented using Gaussian mixtures. The successful static PDF estimation using GMMs is further extended to robustly model temporal variations in a probabilistic setting in Section 4.2. This is achieved with the help of continuous-density HMMs where observations in each state are modeled using mixture of Gaussians as in Section 4.1.

### 4.1 Gaussian Mixture Models

Gaussian mixtures have been used in speech modeling for speaker identification, accent classification among other tasks. In [51] the issue of speech and cross-talk detection in multi-channel audio is addressed. GMM-based classifier is used to classify the speech in different combinations of local speech and cross-talk for multi-speakers multi-microphones setting. Chen et. al. [12] propose an enhancement of speech/speaker recognition by modeling the speaker variability. They use PCA and independent component analysis (ICA) to extract the sources of dominant speaker variability. This variability is then modeled by GMM which results in superior performance as compared to those systems that don't take this variability in account. In [13], they address another dimension of

the same problem, namely the effect of accent on speech/speaker identification. They use GMMs for Mandarin accent identification. Next, we formulate our problem in terms of the GMM framework.

Given the PCA-based representation of subtrajectories for each class, we wish to model the underlying class probability distribution function (PDF) from the training set data. The training set is made as diverse as possible so the recognition system learns all possible data variations. This diversity in training set results in the underlying PDF to be increasingly complex. Hence, the statistical properties of PDF of the class become increasingly non-trivial to model. It is the goal of this subsection to model these complex class PDFs using mixture of Gaussians. In the training phase, we estimate the parameters of Gaussian mixtures using the expectation-maximization (EM) algorithm. Once the training phase has been completed, new trajectories are classified as one of the learnt classes of object motion based on the maximum likelihood (ML) principle.

#### 4.1.1 PDF Estimation using Gaussian Mixtures

For the parameter estimation problem, we first form the set of training set PCA feature vectors of subtrajectories for an individual class. Note that since the PCA feature vectors for each subtrajectory are M-dimensional, all of the individual multivariate Gaussian distributions will be M-dimensional. Let the set of PCA coefficients of subtrajectories for the  $c^{\text{th}}$  class be denoted by  $Y_c$ , as in Section 3. The class PDF  $P(Y_c|\Theta_c)$  can be modeled to an arbitrary accuracy using a mixture of Gaussians:

$$P(Y_c|\Theta_c) = \sum_{i=1}^{N_c} c_i \mathbb{N}(Y_c; \mu_i, \Sigma_i) \quad (3)$$

where  $\mathbb{N}(Y_c; \mu_i, \Sigma_i)$  is the M-dimensional Gaussian density with mean vector  $\mu_i$  and covariance matrix  $\Sigma_i$ , and  $c_i$  are the mixing parameters of the Gaussian components, satisfying  $\sum_{i=1}^{N_c} c_i = 1$ . The mixture is completely specified by the parameter  $\Theta_c = \{c_i, \mu_i, \Sigma_i\}_{i=1}^{N_c}$ . Since the parameter estimation phase is identical for each class and the training is performed on the disjoint dataset of these classes, we drop the class indexing subscript from our

notation for brevity. Now, given a training set of subtrajectories for a particular class  $\{y_t\}_{t=1}^{N_T}$ , represented by their M-dimensional PCA coefficients, the mixture parameters can be estimated using the ML principal; i.e.,

$$\Theta^* = \operatorname{argmax} \left[ \prod_{t=1}^{N_T} P(y^t | \Theta) \right] \quad (4)$$

This estimation problem can be solved using the Expectation-Maximization algorithm [17] which consists of the following two-step iterative process:

- E-Step:

$$h_i^k(t) = \frac{c_i^k \mathbb{N}(y^t; \mu_i^k, \Sigma_i^k)}{\sum_{j=1}^{N_c} c_j^k \mathbb{N}(y^t; \mu_j^k, \Sigma_j^k)} \quad (5)$$

- M-Step:

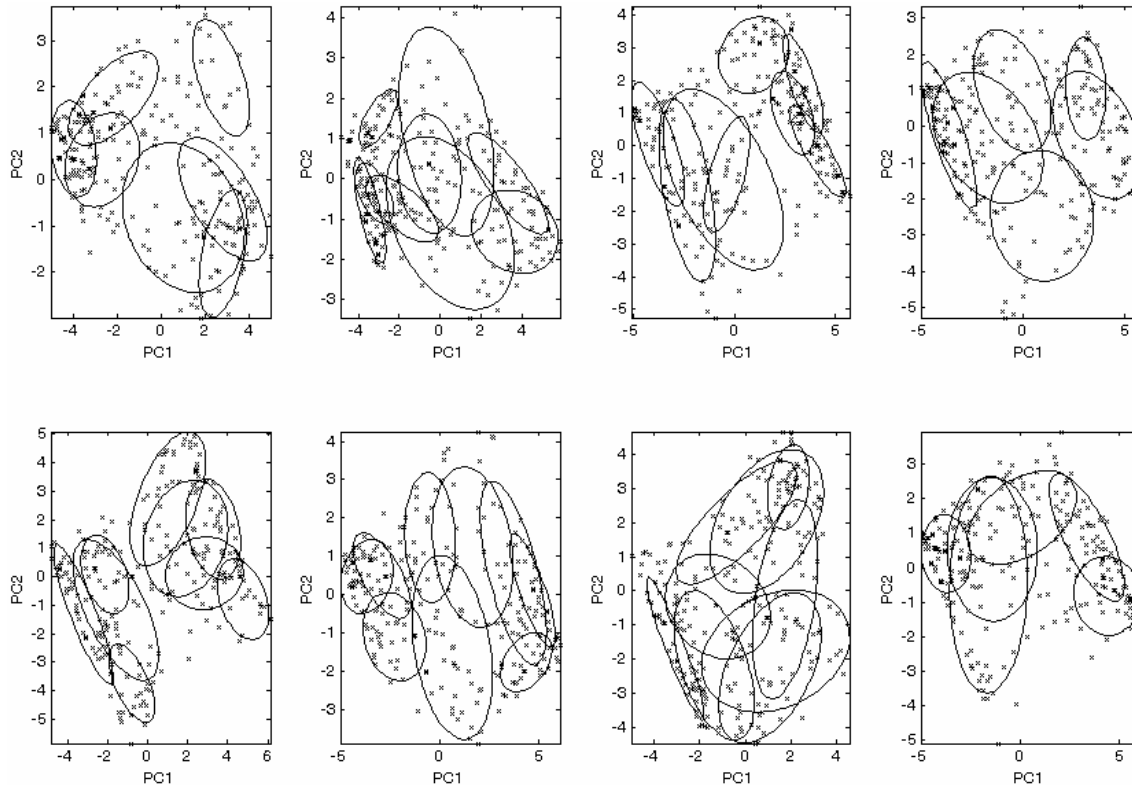
$$c_i^{k+1} = \frac{\sum_{t=1}^{N_T} h_i^k(t)}{\sum_{i=1}^{N_c} \sum_{t=1}^{N_T} h_i^k(t)} \quad (6)$$

$$\mu_i^{k+1} = \frac{\sum_{t=1}^{N_T} h_i^k(t) y^t}{\sum_{t=1}^{N_T} h_i^k(t)} \quad (7)$$

$$\Sigma_i^{k+1} = \frac{\sum_{t=1}^{N_T} h_i^k(t) (y^t - \mu_i^{k+1})(y^t - \mu_i^{k+1})^T}{\sum_{t=1}^{N_T} h_i^k(t)} \quad (8)$$

Here,  $\Sigma_i^{k+1}$  represents the estimate of  $i^{\text{th}}$  covariance matrix at  $k+1^{\text{st}}$  iteration, and similar notation holds for the other items. To initialize the EM algorithm, GMM component means  $\mu_i^1$  are set to randomly selected input data samples from PCA coefficients of the subtrajectories. The covariance matrices  $\Sigma_i^1$  are set to diagonal matrices with large variances of the order of maximum variance of the data samples. Iterations of the EM algorithm yield monotonically non-decreasing likelihood and thus converge in likelihood to a local maximum (or saddle point) in the total likelihood of training data [17], [32].

A major problem in GMM-based modeling is the reliable estimation of the number of modes to be used. We automatically estimate the number of modes from training set data using a string of pruning, merging and mode-splitting processes. We initialize the number of modes as twice the maximum number of subtrajectories in all of the trajectories for the class. The mixing weight of a mode  $c_i$  multiplied by the number of input data samples  $N$  determines how many input data samples are effectively used to estimate the mode parameters. This is the simple measure of ‘value’ of each mode. As long as this product is sufficiently high, the mode is estimated accurately. If  $c_i$  is too low, the mode is eliminated or merged with another. The weighted skew (3<sup>rd</sup>-order moment) and kurtosis (4<sup>th</sup>-order moment) for each mode are also monitored. If the sum of these values exceeds a threshold for any mode, that mode is split in two. Finally, we keep track of the distances between all the modes in order to keep the modes far apart from each other. For this purpose, a distance between the individual modes is computed. If two modes are found to be too close to each other, they are merged. Merging involves forming a weighted sum of the two modes (weighted by  $c_1$  and  $c_2$ ). Figure 3 provides a visualization of the PDF estimation results using Gaussian mixtures. The ASL dataset is used with 8 classes and 50% (35) of trajectories from each class are used for training the individual GMMs.



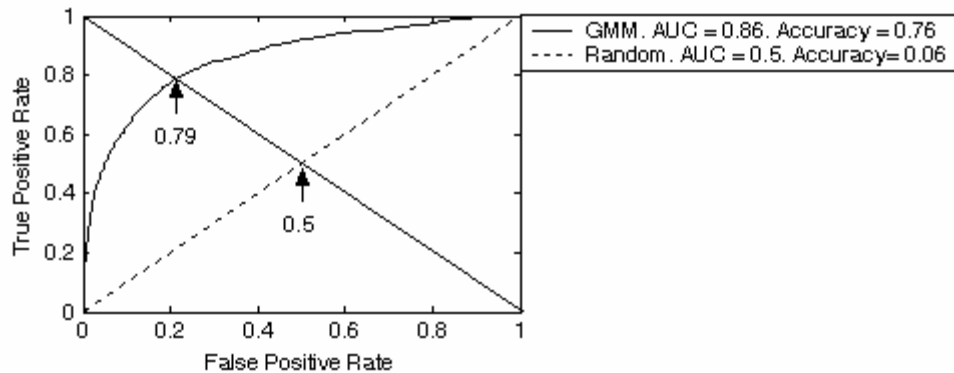
**Figure 3: PDF estimation using Gaussian mixture models. 1-sigma contours of mixture components are superimposed on first two components of PCA representation of data from 8 classes in the ASL dataset.**

#### 4.1.2 Trajectory Classification

Once the GMMs for all of the classes have been trained, the classification of new trajectories can be performed by computing the likelihood for each GMM. For this purpose, the PCA coefficient vectors of the input trajectory after segmentation are posed as an observation sequence to each GMM. During this computation, the likelihood is computed for each individual mode, and the corresponding weights are applied to generate the likelihood of the Gaussian mixture. The trajectory is declared to belong to the class represented by the GMM with the highest likelihood.

#### 4.1.3 Results

We have tested the GMM-based trajectory classification algorithm on a subset of the Australian Sign Language (ASL) dataset. This subset contains trajectories from 16 word classes with 69 sample trajectories per word class signed by 5 different signers for a total of 1104 trajectories. Due to speed variations in signing by different signers, the lengths of individual trajectories vary a lot. Since the speed variation is not very significant as



**Figure 4: ROC curves showing the performance of our GMM-based classifier against random classifier. Area under curve, optimal operating point and accuracy are shown for the two classifiers.**

opposed to the overall spatio-temporal ‘shape’ of the trajectory, we normalize the trajectories by resampling them to 512 points each for x and y-projections. We will provide additional details about the dataset and experimental set up in the next section. We train the GMM using 50% of the trajectory samples from all 16 classes, while testing is performed on all the trajectories of each class. The results are reported in terms of the average receiver operating characteristics (ROC) curve. The ROC curve captures the trade-off between false positive rate versus the true positive rate as the threshold on likelihood at the output of the classifier is varied. The resulting ROC curve is shown in Figure 4. As a baseline case, the performance of a uniformly distributed random classifier is also presented in terms of its ROC curve. Apart from the ROC curve itself, three related scalar metrics of performance are also presented in the figure. The *Area under curve* is a convenient way of comparing classifiers, and varies from 0.5 (random classifier) to 1.0 (ideal classifier) [21]. The other metric is the *optimal operating point* based on *equal error rate* criterion. This metric yields an optimal trade-off between false positives and true positives under the equal cost assumption between false acceptance and correct acceptance. This point is represented with a pointed arrow on the ROC curves for the two classifiers. The last scalar metric used to compare between classifiers is the classification accuracy across all the classes defined as:

$$P_{Accuracy} = 1 - \frac{|F|}{|S|} \quad (9)$$

where  $|F|$  represents the cardinality of the false positives set, while  $|S|$  represents the cardinality of the total dataset. As shown in the figure, the GMM-based classifier results in a 70% increase in accuracy as compared to

the baseline random classifier. It should be noted here that the ROC curve in Figure 4 represents an average of 16 individual curves for a total of 1104 queries for classification. The same comment applies to the probability of accuracy value as well.

## 4.2 Hidden Markov Models

Hidden Markov models have been used successfully in speech recognition[33][38], shape representation [9], gesture recognition [50], sports video structure analysis [52], etc. The Gaussian mixture-based modeling, as outlined in the previous subsection, represents a robust way of estimating the PDF for each motion pattern class. This method can be helpful in modeling classes where contents are time-invariant and don't have a strong dependence on temporal ordering. Our subtrajectory-based representation approach models the trajectories as a sequence of subtrajectories. This approach requires a scheme that takes the temporal dependence among subtrajectories into account. The trajectory model proposed can be viewed as analogous to the representation of words using a sequence of phonemes for speech recognition. It is therefore expected that the use of techniques developed for speech recognition would perform well in our approach to trajectory classification. Specifically, we propose to adopt the use of HMMs for trajectory classification and recognition applications. HMMs are finite state stochastic machines that robustly model temporal variations in time series data which satisfies the Markovian property. Simply put, the first-order Markovian property assumes the independence of the current state from all past states given the previous state. HMMs allow the system to stay in the same state or to transit to the next state at any given time according to state transition probabilities learnt from training data. This allows modeling of temporal variations, where the duration of the state is a variable. For example, in speech signals, different acoustic tokens of the same or similar speech utterance are rarely realized at the same speaking rate across the entire utterance. The HMM-based framework allows us to remove the effect of speaking rate and duration of utterance from the computation of distance measures. Object trajectory data is a stochastic process with temporal continuity, just like speech signals, which has been successfully modeled using HMMs for the past several decades [38]. Trajectory data, just like speech signals, can be modeled using a first-order Markov process. The segmented trajectory-based representation scheme models trajectory data along exactly the same lines as speech signals. In

this context, we are interested in modeling a class of object motions (words) based on the temporal ordering of subtrajectories (phonemes). In this model, a subtrajectory can be used to model the state of the HMM. Since subtrajectories represent segments of atomic motions between points of change in motion pattern, the resulting process can be modeled as first-order Markov chain. We also observe that mixture of Gaussians is a robust method of estimating the PDF in the absence of temporal variations. We therefore propose to use continuous density HMMs, where each state of the HMM is modeled by a mixture of Gaussians. A major problem in HMM design is the topology (left-to-right or ergodic), and the number of states in the HMM. We propose a data-driven design of the HMM with no restriction of HMM topology and number of states. In the following subsections, we outline the process of initializing and estimating the parameters of HMMs as well as the classification process.

#### 4.2.1 HMM Training and Parameter Estimation

The first parameter specified for an HMM is the number of states. For each class, represented by a separate HMM, we set the number of states equal to the maximum number of subtrajectories in all the training set trajectories for that class. Once the number of states is fixed, the complete set of model parameters describing the HMM are given by the triplet:

$$\lambda = \{ \pi_j, a_{ij}, b_j \} \quad (10)$$

where  $\pi_j$  is the probability of the  $j^{\text{th}}$  subtrajectory being the first subtrajectory among all the trajectories,  $a_{ij}$  denotes the probability of the  $j^{\text{th}}$  subtrajectory occurring immediately after the  $i^{\text{th}}$  subtrajectory, and  $b_j$  denotes the PDF of  $j^{\text{th}}$  state. We use a Gaussian Mixture-based representation for the state PDF.

Once the set of training trajectories for a class are segmented and the number of states decided, the HMM's parameter triplet in eq. (10) can be estimated. For a given trajectory, let there be  $T$  subtrajectories. Then, the state variable  $q_t$  which corresponds to the  $t^{\text{th}}$  subtrajectory, takes one of  $N$  values  $q_t \in \{S_1, \dots, S_N\}$ . Since we assume a Markovian process, the probability distribution of  $q_{t+1}$  depends only on  $q_t$ . This is described by the state transition probability matrix  $A$  whose elements  $a_{ij}$  represent the probability that  $q_{t+1}$  corresponds to state  $S_j$  given that  $q_t$  corresponds to  $S_i$ . The initial state probabilities are denoted by  $\pi_i$ , the probability that  $q_1$  equals  $S_i$ . The

observational data  $O_t$  from each state of the HMM is generated according to a PDF dependent on the state at the instant of  $t^{\text{th}}$  subtrajectory, denoted by  $b_j(O_t)$ . This state-conditional observation PDF is modeled as a Gaussian mixture given by

$$P(O_t = O | q_t = S_j) = b_j(O_t) = \sum_{m=1}^M c_{jm} \mathbb{N}(\mu_{jm}, \Sigma_{jm}) = \sum_{m=1}^M c_{jm} \frac{1}{(2\pi)^{P/2} |\Sigma_{jm}|^{1/2}} \exp\left\{-\frac{1}{2}(O - \mu_{jm})^T \Sigma_{jm}^{-1} (O - \mu_{jm})\right\} \quad (11)$$

where  $c_{jm}$ ,  $\mu_{jm}$  and  $\Sigma_{jm}$  denote the scalar mixing parameter,  $P$ -dimensional mean vector and  $P \times P$  covariance matrix of the  $m^{\text{th}}$  Gaussian component in the  $j^{\text{th}}$  state. Here, each Gaussian component is a multivariate normal distribution of the same dimensionality as the PCA coefficients representing the subtrajectories. The parameters of the HMM are initialized to random values and the Baum-Welch algorithm is used to estimate the parameters using the forward-backward procedure [38]. The above discussion relates to training a sequence of subtrajectories resulting from one trajectory. Given a set of trajectories corresponding to each class, we extend the training to multiple training set trajectories. At each iteration of the Baum-Welch estimation, the contribution from all of the individual training set trajectories are summed up in the forward-backward estimation parameters. Once the change in parameter values is less than a prefixed threshold for 10 successive iterations, the algorithm is declared to have converged. One problem with this form of parameter estimation is that it can get stuck in a local maximum no matter how many iterations are performed. To improve the performance in this situation, we use a process which is motivated by *simulated annealing*. After one set of iterations for parameter estimation, we expand (i.e. multiply by a factor) the covariance matrices of the PDF estimates; thus, pushing the state transition matrix and prior state probabilities closer to ‘uniform’. Expansion of the covariance matrix is equivalent to perturbing the current point in the search space with a stochastic noise process. This step is similar to increasing the ‘temperature’ in simulated annealing and forcing the current solution away from local peaks. The effect of this procedure is to increase the search space; thereby, allowing convergence towards the global maximum as the number of iterations increase, provided that the proper annealing schedule (i.e. expansion factor) has been maintained. A simulated annealing-based EM algorithm can also be used to guarantee almost-sure convergence to a global maximum [10].

### 4.2.2 Trajectory Classification

Once the HMMs for all classes have been trained, the classification of new trajectories can be performed by computing the likelihood that HMM  $i$  best describes the test trajectory. For this purpose, the PCA coefficient vectors of input trajectories after segmentation are posed as an observation sequence to each HMM. Given HMMs for the  $L$  classes,  $\lambda_1, \lambda_2, \dots, \lambda_L$ , and the set of PCA coefficient vectors of input subtrajectories (i.e. the observation sequence)  $O_1, O_2, \dots, O_m$ , we compute the maximum likelihood (ML) estimate of the test trajectory (sequence of subtrajectories) for each of the individual HMMs [38][32]:

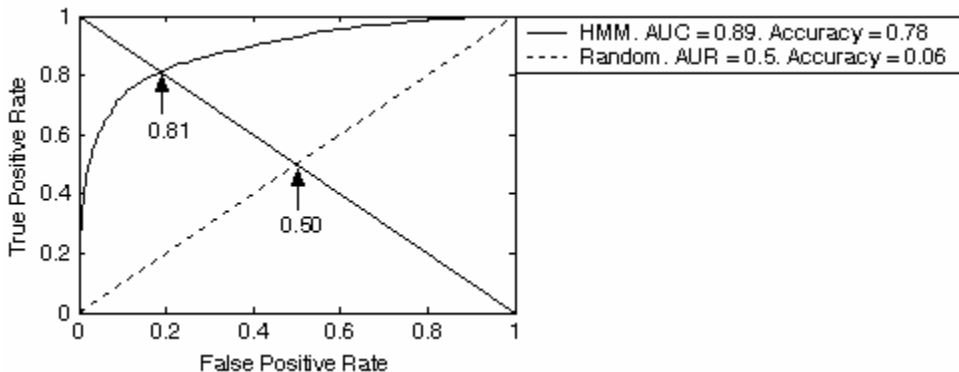
$$class = \arg \max_{i \in [1, \dots, L]} P(O_{1:m} | \lambda_i) = \arg \max_{i \in [1, \dots, L]} \sum_j P(O_{t+1:m} | q_t^i = j, O_{1:t}) P(q_t^i = j, O_{1:m}) \quad (12)$$

This computation is efficiently performed using the forward recursion procedure in the Baum-Welch algorithm [38]. Observe that the ML estimate is equivalent to the maximum a posteriori (MAP) estimate for the same model parameters when the prior distribution of the HMMs is uniform. The model parameters have already been learnt in the training phase.

The HMM-based classifier is used in the same experimental setup as outlined in subsection 4.1.3. Training is performed on half of the trajectories from the 16-word classes. The 1104 trajectories from the 16 classes are posed to the HMM-based system for classification. The resulting ROC curves are then compared to a random classifier in terms of the area under the curve, optimal operating point and classification accuracy, as depicted in Figure 5. As shown in the figure, the HMM-based classification system results in 72% improvement in classification accuracy as compared to the random classifier.

### 4.3 Analysis of GMM and HMM Classifiers

Results in the above subsections are presented in terms of receiver operating characteristic (ROC) curves. The ROC curves depict the performance of a classification system based on varying the threshold of the likelihood. The results, as shown in the previous subsections are reported in terms of probability of false alarm and probability of detection. ROC curves of classifiers with large number of classes (i.e., more than two) fail to



**Figure 5: ROC curves showing the performance of our HMM-based classifier against random classifier. Area under curve, optimal operating point and accuracy are shown for the two classifiers.**

succinctly capture the tradeoffs between classifiers. We therefore also provide the average probability of accuracy of the classifiers.

Recently, investigators have chosen to compare between classifiers by measuring the average distance between the classes [36], [46], [47]. This method is used to gain an insight into the geometrical structure of the classifier and hope that a wide distance between classes will translate to improved probability of accuracy. In this section, we compare the GMM and HMM-based methods in terms of the interclass separations. It is widely known in detection theory and pattern recognition literature that as the distances between individual classes increase, the probability of error at the detector decreases. It is therefore prudent to design a successful classifier by spreading out the class representation in the feature space as widely as possible in order to minimize the probability of error in detection.

Many possible distance measures have been employed in classification analysis. The classes are generally represented as density functions distributed in a feature space. The distance between classes is therefore characterized as the measure of difference between density functions. Some of the most common measures used in classification analysis include likelihood-based measures [47], Kullback-Leibler distance measure [28], etc.

The Kullback-Leibler distance (KLD), or relative entropy, is the most common distance measure between density functions [28]. It is defined as the average discrimination information per observation between PDFs  $f_1$  and  $f_2$ :

$$D_{KLD}(f_1 \| f_2) = \int f_1(x) \log \frac{f_1(x)}{f_2(x)} dx \quad (13)$$

Closed-form expressions of KLD exist for a small family of distributions such as Gaussian and generalized Gaussian density functions. However, exact closed-form expressions for Gaussian mixture models (GMMs) and hidden Markov models (HMMs) are still not known. Vasconcelos [47] has addressed the problem of finding an approximate closed-form expression for the KLD between Gaussian mixture models (GMMs) called *asymptotic likelihood approximation* (ALA). Conditions under which it converges to KLD are also given in [47]. Do [19] derives an upper-bound on the Kullback-Leibler distance rate (KLDL) between HMMs and computes the asymptotic approximation based on their parametric representation. Silva et. al. [44] propose the *average divergence distance* (ADD) based on a representation of the divergence calculated at the observation distribution level of the models.

In the following, we first give the ALA-based approximation of KLD for GMMs and HMMs. Let us consider a GMM-based representation of the PDF of a class, given by

$$P_i(x) = \sum_{k=1}^{M_i} c_{ik} \cdot \mathbb{N}(x, \mu_{i,k}, \Sigma_{i,k}) \quad (14)$$

We use the ALA-based approximation of KLD between GMMs [47] given by

$$KLD_G(P_i \| P_j) = - \sum_{k=1}^{M_i} c_{ik} \left[ \log c_{j,\beta(k)} + \log \mathbb{N}(\mu_{i,k}, \mu_{j,\beta(k)}, \Sigma_{j,\beta(k)}) - \frac{1}{2} \text{trace} \left\{ \Sigma_{j,\beta(k)}^{-1} \Sigma_{i,k} \right\} \right] \quad (15)$$

where

$$\beta(k) = r \Leftrightarrow \left\| \mu_{i,k} - \mu_{j,r} \right\|_{\Sigma_{j,r}}^2 - \log c_{jr} < \left\| \mu_{i,k} - \mu_{j,s} \right\|_{\Sigma_{j,s}}^2 - \log c_{js}, \forall s \neq r$$

Next we formulate the KLD between continuous density hidden Markov models (HMMs) where each state is modeled using mixture of Gaussians. From [44] we know that the KLD between continuous density HMMs is given by

$$KLD_H(\lambda_i \| \lambda_j) = \sum_{l=1}^m E_{s_l^i s_l^j} \left[ KLD_G \left( P_i^{s_l^i} \| P_j^{s_l^j} \right) \right] \quad (16)$$

where  $\lambda_i$  and  $\lambda_j$  denote the two HMMs;  $s_l^i \in \langle V_1^i, \dots, V_N^i \rangle$  is the  $l^{\text{th}}$  state in the state sequence of  $\lambda_i$  generated according to the model's initial- and transition- probabilities;  $P_i^{s_l^i}$  represents the Gaussian mixture for state  $s_l^i$  of

$\lambda_i$ ; and  $E_{s_i^l, s_j^l}(\cdot)$  denotes the expectation operator used to compute the expected value under all possible state mappings. Here, a key assumption is that the two HMMs have the same number of states,  $m$  which in our case is the number of subtrajectories. We retain this assumption to simplify the ensuing analysis. We can now extend the ALA-based approximation of the KLD between continuous density HMMs where each state is modeled using mixture of Gaussians (see eqs. (15) and (16)) to obtain:

$$KLD_H(\lambda_i \parallel \lambda_j) = -\sum_{l=1}^m E \left[ \sum_{k=1}^{M_l^i} c_{ik}^{s_l^i} \left\{ \log c_{j\beta(k)}^{s_l^j} + \log \mathbb{N} \left( \mu_{i,k}^{s_l^i}, \mu_{j,\beta(k)}^{s_l^j}, \Sigma_{j,\beta(k)}^{s_l^j} \right) - \frac{1}{2} \text{trace} \left( \begin{matrix} s_l^{j-1} & s_l^i \\ \Sigma_{j,\beta(k)} & \Sigma_{i,k} \end{matrix} \right) \right\} \right] \quad (17)$$

where

$$\beta(k) = r \Leftrightarrow \left\| \mu_{i,k}^{s_l^i} - \mu_{j,r}^{s_l^j} \right\|_{\Sigma_{j,r}^{s_l^j}}^2 - \log c_{jr}^{s_l^j} < \left\| \mu_{i,k}^{s_l^i} - \mu_{j,t}^{s_l^j} \right\|_{\Sigma_{j,t}^{s_l^j}}^2 - \log c_{jt}^{s_l^j}, \forall t \neq r$$

Here,  $M_l^i$  refers to the number of Gaussian components (modes) in the GMM of the  $i^{\text{th}}$  HMM's  $l^{\text{th}}$  state;  $c_{ik}^{s_l^i}$ ,

$\mu_{i,k}^{s_l^i}$  and  $\Sigma_{i,k}^{s_l^i}$  represent the weight, mean and variance of the  $k^{\text{th}}$  Gaussian component in the  $l^{\text{th}}$  state of the  $i^{\text{th}}$

HMM;  $c_{j\beta(k)}^{s_l^j}$ ,  $\mu_{j,\beta(k)}^{s_l^j}$  and  $\Sigma_{j,\beta(k)}^{s_l^j}$  represent the weight, mean and variance of the Gaussian component in

the  $l^{\text{th}}$  state of the  $j^{\text{th}}$  HMM, which is closest to the  $k^{\text{th}}$  component in the  $i^{\text{th}}$  HMM, as defined by the mapping  $\beta(k)$

given in eq. (17). We note here that our approximation of the KLD depends on the transition probabilities through expectation of all possible states. We shall now present sufficient conditions under which the inter-class distances generated by HMMs are greater than those using GMMs. Subsequently, we validate this claim under the operating conditions in our problem domain and present numerical experiments confirming this result.

It can be easily shown that the ALA-based approximation of KLD for HMMs where each state is modeled using mixture of Gaussians is greater than that based on GMMs; i.e.,

$$KLD_H(\lambda_i \parallel \lambda_j) > KLD_G(P_i \parallel P_j) \quad (18)$$

whenever the following inequalities hold:

$$\begin{aligned}
& \sum_{l=1}^m E \left[ \sum_{k=1}^{M_l^i} c_{ik}^{s_l^i} \cdot \log c_{j\beta(k)}^{s_l^j} \right] < \sum_{k=1}^{M_i} c_{ik} \cdot \log c_{j\beta(k)} \\
& \sum_{l=1}^m E \left[ \sum_{k=1}^{M_l^i} c_{ik}^{s_l^i} \cdot \left\| \mu_{i,k}^{s_l^i} - \mu_{j,\beta(k)}^{s_l^j} \right\|_{\Sigma_{j,\beta(k)}^{s_l^j}}^2 \right] > \sum_{k=1}^{M_i} c_{ik} \cdot \left\| \mu_{i,k} - \mu_{j,\beta(k)} \right\|_{\Sigma_{j,\beta(k)}^{s_l^j}}^2 \\
& \sum_{l=1}^m E \left[ \sum_{k=1}^{M_l^i} c_{ik}^{s_l^i} \cdot \log \left| \Sigma_{j,\beta(k)}^{s_l^j} \right| \right] > \sum_{k=1}^{M_i} c_{ik} \cdot \log \left| \Sigma_{j,\beta(k)} \right| \\
& \sum_{l=1}^m E \left[ \sum_{k=1}^{M_l^i} c_{ik}^{s_l^i} \cdot \text{trace} \left\{ \Sigma_{j,\beta(k)}^{s_l^j} \Sigma_{i,k} \right\} \right] > \sum_{k=1}^{M_i} c_{ik} \cdot \text{trace} \left\{ \Sigma_{j,\beta(k)}^{-1} \Sigma_{i,k} \right\}
\end{aligned} \tag{19}$$

We further note here that in our PCA-based trajectory representation approach used to build the GMM and HMM models, the covariance matrices are set to be diagonal. For PCA-based trajectory representation, the conditions provided by eq. (19), which establish that the ALA-based approximation of KLD of HMMs where each state is modeled using mixture of Gaussians is greater than that based on GMMs, simplify to the following conditions:

$$\begin{aligned}
& \sum_{l=1}^m E \left[ \sum_{k=1}^{M_l^i} c_{ik}^{s_l^i} \cdot \log c_{j\beta(k)}^{s_l^j} \right] < \sum_{k=1}^{M_i} c_{ik} \cdot \log c_{j\beta(k)} \\
& \sum_{l=1}^m E \left[ \sum_{k=1}^{M_l^i} c_{ik}^{s_l^i} \cdot \sum_{l=1}^D \frac{\left( \mu_{i,k,l}^{s_l^i} - \mu_{j,\beta(k),l}^{s_l^j} \right)^2}{\sigma_{j,\beta(k),l}^{s_l^j}} \right] > \sum_{k=1}^{M_i} c_{ik} \cdot \sum_{l=1}^D \frac{\left( \mu_{i,k,l} - \mu_{j,\beta(k),l} \right)^2}{\sigma_{j,\beta(k),l}} \\
& \sum_{l=1}^m E \left[ \sum_{k=1}^{M_l^i} c_{ik}^{s_l^i} \cdot \sum_{l=1}^D \log \sigma_{j,\beta(k),l}^{s_l^j} \right] > \sum_{k=1}^{M_i} c_{ik} \cdot \sum_{l=1}^D \log \sigma_{j,\beta(k),l} \\
& \sum_{l=1}^m E \left[ \sum_{k=1}^{M_l^i} c_{ik}^{s_l^i} \cdot \sum_{l=1}^D \frac{\sigma_{i,k,l}^{s_l^i}}{\sigma_{j,\beta(k),l}^{s_l^j}} \right] > \sum_{k=1}^{M_i} c_{ik} \cdot \sum_{l=1}^D \frac{\sigma_{i,k,l}}{\sigma_{j,\beta(k),l}}
\end{aligned} \tag{20}$$

The circumstances under which these conditions are satisfied can be motivated by simple geometrical arguments when using circles to represent the variance of Gaussian density functions. The second distance measure we compute is the *posterior likelihood-based distance* (PLD) between classes for each input trajectory from the test set given the model. Given two trajectories  $X_i$  and  $X_j$  and their corresponding models  $P_i$  and  $P_j$ , we compute two

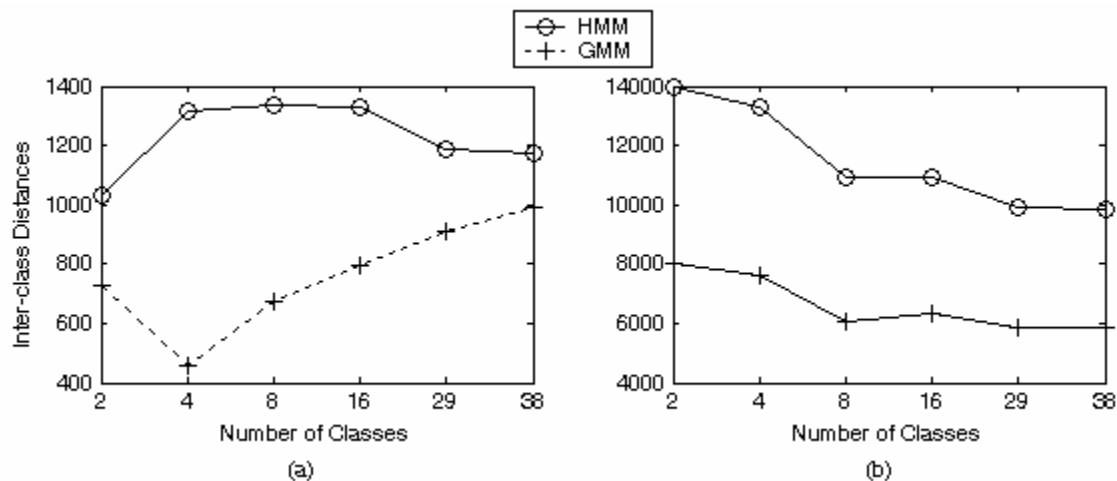
terms: self-fitness and cross-fitness. We then sum up the individual contributions by all trajectories for a distance measure between the two models as proposed in [36].

$$D_{PLD}(X_i, X_j) = \left| L(X_i | P_i) + L(X_j | P_j) - L(X_i | P_j) - L(X_j | P_i) \right| \quad (21)$$

Whereas the KLD-based distance computation, as defined in eqs. (15) and (17), takes into account the model parameters after training, the PLD measure is computed based on the performance of classification system on a test corpus. This way we ensure that the results are reported accurately for both training and test phases of the classification system. We have computed both the KL and posterior likelihood-based distances for GMM and HMM-based representation. This experiment is performed on the ASL dataset with several different class sizes and the results are displayed in Figure 6. As shown in the figure, the HMM-based representation results in wider inter-class distances as compared to its GMM counterpart. This fact explains the improved performance of HMM-based classifiers compared to GMM-based systems as captured in the ROC curves in Figure 4 and Figure 5.

## 5 COMPARISON AND ANALYSIS

This section compares the performance of our GMM and HMM-based trajectory modeling approaches proposed in the previous section. We also compare our results with an adaptation of the PCA-based density estimation approach outlined in [30]. The approach in [30] deals with face detection and recognition and has been adapted to



**Figure 6: Inter-class distances for HMM and GMM- based representations using (a) KLD (b) PLD.**

fit in our system of PCA-based trajectory representation without trajectory segmentation. In the following subsections, we first outline the implementation of the system that we use for comparison and then we detail the results of the comparison.

### 5.1 PCA-Based Density Estimation

This technique is derived for visual learning based on density estimation in high-dimensional spaces using eigenspace decomposition. The training phase comprises estimation of the mean  $\bar{X}$  and covariance  $\Sigma$  of the distribution from the given training set  $\{X^t\}$ . The sufficient statistic for characterizing the likelihood based on Gaussian density assumption uses the *Mahalanobis* distance which is estimated from the  $M$  principal components. Based on this estimation, the likelihood estimate can be written as:

$$\hat{P}(X|\Omega) = \left[ \frac{\exp\left(-\frac{1}{2} \sum_{i=1}^M \frac{y_i^2}{\lambda_i}\right)}{(2\pi)^{M/2} \prod_{i=1}^M \lambda_i^{1/2}} \right] \left[ \frac{\exp\left(-\frac{\varepsilon^2(x)}{2\rho}\right)}{(2\pi\rho)^{(N-M)/2}} \right] \quad (22)$$

where the first term is the true marginal density in the principal feature space and the second term is the estimated marginal density in the orthogonal complement space. Here, the residual reconstruction error is defined as:

$$\varepsilon^2(x) = \|\tilde{x}\|^2 - \sum_{i=1}^M y_i^2 \quad (23)$$

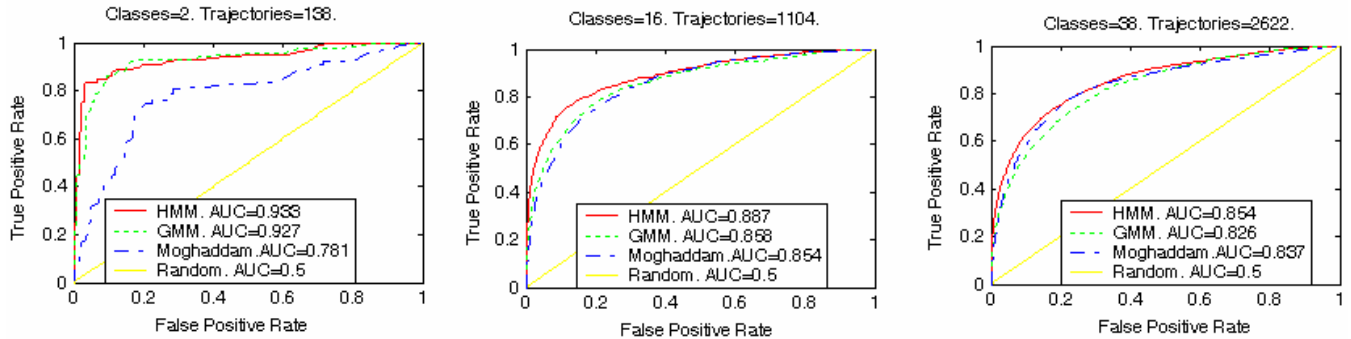
where  $\tilde{x}$  is the mean-normalized trajectory vector and  $y_i$  is the  $i^{\text{th}}$  principal component. Also, the optimal weight  $\rho$  is defined as:

$$\rho^* = \frac{1}{N-M} \sum_{i=M+1}^N \lambda_i \quad (24)$$

where the eigenvalues beyond  $M$  are found by fitting an exponential distribution to the first  $M$  eigenvalues. For the classification of a new trajectory  $X$ , eq. (22) is computed for all classes  $\Omega$ . The trajectory is declared to belong to the class generating the maximum likelihood.

### 5.2 Datasets

In this section, we outline the details of the datasets used in our experiments. We use two datasets in our simulations: The first dataset, the Australian Sign Language (ASL) dataset, is obtained from UCI’s KDD archives.<sup>1</sup> These trajectories are obtained by registration of the hand coordinates at each successive instant of time by using a Power Glove interfaced to the system. The system registers 3-D positions of the hand, palm orientation, and several other features at each sampling instant as 5 professional signers sign around 95 words in multiple sessions. For each word, there are 69 recordings of signing activity across all signers. Out of these set of trajectories, we extract the x and y-locations corresponding to around 83 classes for training and recognition. This corresponds to a dataset of 5,727 trajectories. The dataset shows a lot of variation in trajectory data for the same words and even for the same signer. The spatial and temporal variation arises from various sources including: noise in the sensors, variation in skill levels among different signers, and fatigue on signers performing the task. The second dataset in our experiments has been provided to us by Columbia University’s Digital Video and Multimedia Group (DVMM) [14] and contains object trajectories tracked from video clips of sports activities, like high jump, slalom skiing, etc. This dataset, HJSL (108), contains around 40 trajectories of high jump, and 68 trajectories of slalom skiing objects.



**Figure 7: ROC curves showing the performance of our GMM and HMM-based classifiers against an adaptation of Moghaddam’s PCA-based density estimation approach and random classifier. ROC curves with area under the curve are shown for datasets of various sizes.**

### 5.3 Simulation Results

This section summarizes the results of our computer simulation experiments. We have already compared the GMM and HMM-based models in terms of their inter-class separations using KLD and PLD in subsection 4.3.

<sup>1</sup> KDD archive: <http://kdd.ics.uci.edu/databases/auslan/auslan.html> . Original donor: <http://www.cse.unsw.edu.au/~waleed/tml/data/>

We have also reported the performance of the two systems in terms of ROC curves and accuracy for a dataset of 1104 trajectories with 16 classes in subsections 4.1.3 and 4.2.2. In this section, we compare our proposed GMM and HMM-based classification systems with the PCA-based density estimation approach outlined in 5.1 and a random classifier. We report the results on a wide range of dataset sizes from the ASL dataset, as well as the HJSL dataset. The results are reported in terms of the ROC curves in Figure 7 which also shows the area under the ROC curves as a scalar measure of classifier performance. As noted earlier, ROC curves of classifiers with large number of classes (i.e., more than two) fail to succinctly capture the tradeoffs between classifiers. We therefore also provide the average probability of accuracy of the classifiers. This result is provided in terms of scalar values of accuracy for different dataset sizes from the ASL dataset as well as the HJSL dataset in Table 1.

**Table 1: Probability of accuracy values for various class sizes from the ASL and the HJSL datasets. Column headings indicate the number of classes and number of trajectories used for the ASL dataset.**

Datasets	ASL						HJSL
	#Classes : #Trajectories						
	2:138	4:276	8:552	16:1104	29:2001	38:2622	
<b>HMM</b>	0.9638	0.9167	0.8587	0.7790	0.6882	0.6609	0.9074
<b>GMM</b>	0.9855	0.8949	0.8514	0.7455	0.6672	0.6400	0.8981
<b>Moghaddam</b>	0.9420	0.9312	0.8297	0.7283	0.5592	0.6175	0.4537

Based on these results, we see that the PCA-based approaches yield a superior representation for trajectory modeling. From Table 1 we also note that the relative accuracy of the HMM-based classifier compared with GMM and Moghaddam et. al. [31] increases with an increase in the number of classes; thus making it more scalable for large number of classes. Moreover, we note that much higher probability of accuracy values would have been attained provided the size of the training datasets would have been increased proportionally. Furthermore, the HMM-based trajectory modeling system proposed in this paper where individual states are modeled by mixtures of Gaussians has been shown to perform consistently better than the other trajectory modeling techniques used in all of our experiments.

## 6 SUMMARY AND CONCLUSIONS

In this paper, we have presented a novel framework for motion trajectory-based statistical modeling and classification of data captured from any form of object tracking. One of our main aims in this presentation has

been to motivate the need for, and understand the challenges involved in, the classification and recognition of temporal data resulting from object tracking. We first outline a PDF representation approach using Gaussian mixture models for motion trajectory identification. The strength of this technique is robust time-invariant modeling of the PDF, but its drawback is the lack of temporal modeling in the formulation. We keep the good part of GMMs and alleviate their limitation by proposing the use of Gaussian mixture-based HMMs for trajectory modeling. While GMMs and HMMs have been used in various recognition tasks, their use in motion trajectory representation and classification in this paper presents a novel new approach to trajectory analysis. We have based our experiments on various measures of performance. The ROC curves with their area under the curve are used to compare the classifiers by varying the threshold on likelihood at the output of the classifier. The classification effectiveness is also measured in terms of the inter-class distances for GMM and HMM-based modeling techniques. The classification systems were tested on two standard datasets in different application domains. The ASL dataset is captured through non-visual sensors and is intended for the gesture recognition domain. The results are generated for various sizes from 138 to 2,622 trajectories in this dataset. The HJSL dataset contains the trajectories obtained from tracking high jumpers and slalom skiers in sports video clips. The accuracy of the classification systems is tested on the 108 trajectories in this relatively small video dataset.

Future research must focus on motion trajectory-based modeling and classification of video sequences that are robust to camera orientation and movement. Generalization of the proposed PCA representation to nonlinear transformations (e.g., Kernel PCA or Kernel Discriminant Analysis), are needed to deal with nonlinear classification. We plan to explore the estimation of the number of modes for GMMs using the method presented by Gassiat [22] as an alternative to the pruning, merging and mode-splitting process. On theoretical analysis part, the HMM-based formulation proposed in this presentation can be proved to be a particular case of the so-called Triplet Markov Chain (TMC) [34]. A completely unsupervised method of learning parameters of all state conditional PDFs can be envisaged in this setting. An important extension of our approach would be required to perform multiple motion trajectory-based classification for ‘semantic’ retrieval from video sequences. It is also possible that the basis of our approach could be used for video sequence mining by detection and identification of motion trajectories in the video query.

## ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their comments and suggestions which have greatly improved the presentation of this paper.

## REFERENCES

- [1] Bashir F., Khanvilkar S., Schonfeld D., Khokhar A., “Multimedia Systems: Content-Based Indexing and Retrieval,” Sec. 4, Chapter 6, *The Electrical Engineering Handbook*. Ed. W.K. Chen. Academic Press, 2004.
- [2] Bashir F., Khokhar A., Schonfeld D., “A Hybrid System for Affine-Invariant Trajectory Retrieval,” *ACM SIGMM Multimedia Information Retrieval workshop*, New York, NY, 2004.
- [3] Bashir F., Khokhar A., Schonfeld D., “Automatic Object Trajectory-Based Motion Recognition using Gaussian Mixture Models,” *IEEE International Conference on Multimedia & Expo (ICME 2005)*, July. 6 - July 8, 2005. Amsterdam, the Netherlands.
- [4] Bashir F., Qu W., Khokhar A., Schonfeld D., “HMM-Based Motion Recognition System using Segmented PCA,” *IEEE International Conference on Image Processing (ICIP 2005)*, Sept. 11 - Sept. 14, 2005. Genoa, Italy.
- [5] Bettinger F., Cootes J., Taylor C.J., “Modelling Facial Behaviours,” *British Machine Vision Conference, (BMVC) 2002*.
- [6] Braffort A., Gherbi R., “Video Tracking and Recognition of Pointing Gestures using Hidden Markov Models,” *IEEE International Conference on Intelligent Engineering Systems (INES'98)*, 1998.
- [7] Brand M., Oliver N., Pentland A., “Coupled Hidden Markov Models for Complex Action Recognition,” *Proceedings Conference on Computer Vision and Pattern Recognition*, p. 994, 1997.
- [8] Burnham K.P., Anderson D.R., “Model Selection and Multi-Model Inference – A Practical Information Theoretic Approach,” *Springer Science+Business Media Inc.* 2002.
- [9] Caelli T., McCabe A., Briscoe G., “Shape Tracking and Production using Hidden Markov Models,” *International Journal of Pattern Recognition and Artificial Intelligence*. Vol. 15(1), 197-221.

- [10] Celeux G., Chauveau D., Diebolt J., "Some Stochastic versions of the EM Algorithm," *Journal of Statistical Computation and Simulation*, Vol. 55, pages 287-314, 2002.
- [11] Chang S.F., Chen W., Meng H.J., Sundaram H., Zhong D., "A Fully Automated Content-Based Video Search Engine Supporting Spatiotemporal Queries," *IEEE Trans. on Circ. & Sys. for Video Techn.*, Vol. 8(5), 1998.
- [12] Chen T., Huang C., Chang C., Wang J., "On the Use of Gaussian Mixture Model for Speaker Variability Analysis," *ICSLP*, Denver, Colorado, 2002.
- [13] Chen T., Huang C., Chang C., Wang J., "Automatic Accent Identification using Gaussian Mixture Model," *IEEE workshop on ASRU*, Italy, 2001.
- [14] Chen W., Chang S.F., "Motion Trajectory Matching of Video Objects," *SPIE*, San Jose, CA, Jan. 2000.
- [15] Cheung S., Zakhor A., "Fast Similarity Search on Video Sequences," *Proceedings IEEE International Conference on Image Processing*, 2003.
- [16] De La Torre F., Yacoob Y., Davis L., "A Probabilistic Framework for Rigid and Non-Rigid Appearance based Tracking and Recognition," *Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, 2000. pp. 491-498.
- [17] Dempster A., Laird N., Rubin D., "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of Royal Statistical Society, Series B*, Vol. 39(1). pp. 1-38, 1977.
- [18] Dimitrova N., Golshani F., "Motion Recovery for Video Content Classification," *ACM Transactions on Information Systems*. Vol. 13(4). pp. 408-439.
- [19] Do M. N., "Fast Approximation of Kullback-Leibler Distance for Dependence Trees and Hidden Markov Models," *IEEE Signal Processing Letters*, Vol. 10(4). pp. 115-118, April 2003.
- [20] Fablet R., Boutheymy P., "Motion recognition using spatio-temporal Random Walks in Sequence of 2D Motion-Related Measurements," *IEEE Int. Conf. on Image Processing*, pp. 652-655, Greece, Oct. 2001.
- [21] Fawcett T., "ROC Graphs: Notes and Practical Considerations for Researchers," Technical Report, *HP Labs*, HPL-2003-4, April 2004.

- [22] Gassiat E., “Likelihood Ratio Inequalities with Applications to various Mixtures,” *Ann. Inst. Henri Poincare*, Vol. 38, pages 897-906, 2002.
- [23] Hettich S., Bay S.D. “The UCI KDD Archive [<http://kdd.ics.uci.edu>],” University of California, Department of Information and Computer Science, Irvine, California, 1999.
- [24] Isard M., Blake A., “A Mixed-State CONDENSATION Tracker with Automatic Model-Switching,” *Proceedings of the International Conference on Computer Vision*, pp. 107-112, 1998.
- [25] Johansson G., “Visual Perception of Biological Motion and a Model for its Analysis,” *Perception and Psychophysics*. Vol. 14(2), pp. 201-211, 1973,
- [26] Jolliffe, I.T., *Principal Component Analysis*. Springer-Verlag, New York, 1986.
- [27] Kass R., Raferty A., “Bayes Factors,” *Journal of American Statistical Association*, Vol. 90, p 773-795, 1995.
- [28] Kullback S., *Information Theory and Statistics*. New York: Wiley, 1958.
- [29] Martin J., Hall D., Crowley J., “Statistical Gesture Recognition through Modeling of Parameter Trajectories,” *International Gesture Workshop on Gesture-based Communication in Human-Computer Interaction*, 1999. pp. 129-140.
- [30] Moghaddam B., Pentland A., “Probabilistic Visual Learning for Object Representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):696-710, July 1997.
- [31] Moghaddam B., Wahid W., Pentland A., “Beyond EigenFaces: Probabilistic Matching for Face Recognition,” *Inter. Conf. on Automatic Face and Gesture Recognition*, Nara, Japan, April 1998.
- [32] Murphy K. P., “Learning Markov Processes,” *The Encyclopedia of Cognitive Science*, L. Nadel et al (Eds.), Nature Macmillan, 2002.
- [33] Nouza J., “Feature Selection Methods for Hidden Markov Model-based Speech Recognition,” 13<sup>th</sup> *International Conference on Pattern Recognition*, Vol. 2, p 186-190, 1996.
- [34] Pieczynski W., Desbouvries F., “On Triplet Markov Chains,” *International Symposium on Applied Stochastic Models and Data Analysis (ASMDA 2005)*. Brest, France, May 2005.
- [35] Porikli F., Haga T., “Event Detection by Eigenvector Decomposition using Object and Frame Features,” *Intern. Conf. on Computer Vision and Pattern Recognition*, 2004.

- [36] Porikli F.M., "Trajectory Distance Metric using Hidden Markov Model Based Representation," *European Conference on Computer Vision*, May 2004.
- [37] Qu W., Bashir F., Khokhar A., Schonfeld D., "A Motion Trajectory Based Video Retrieval System Using Parallel Adaptive Self Organizing Maps," *International Joint Conference on Neural Networks*, July 31 - Aug. 4, 2005. Montreal, Canada.
- [38] Rabiner L.R., "A tutorial on Hidden Markov Models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, pp. 257-286, 1989.
- [39] Rao C., Yilmaz A., Shah M., "View-Invariant Representation and Recognition of Actions," *International Journal of Computer Vision*, Vol. 50 (2), pp. 203-226, 2002.
- [40] Rea N., Dahyot R., Kokaram A., "Semantic Event Detection in Sports through motion understanding," *Proceedings of Conference on Image and Video Retrieval*, Dublin, Ireland, July 21-23, 2004.
- [41] Rubin J.M., Richards W.A., "Boundaries of Visual Motion," Technical Report: AIM-835., *Massachusetts Institute of Technology, Artificial Intelligence Laboratory*, p.149, 1985.
- [42] Sahouria E., Zakhor A., "A Trajectory Based Video Indexing System For Street Surveillance," *IEEE Int. Conf. on Image Processing*, 1999.
- [43] Schonfeld D., Lelescu D., "VORTEX: Video retrieval and tracking from compressed multimedia databases—multiple object tracking from MPEG-2 bitstream," (Invited Paper). *Journal of Visual Communications and Image Representation*, vol. 11, pp. 154-182, 2000.
- [44] Silva J., Narayanan S., "A Statistical Discrimination Measure for Hidden Markov Models based on Divergence," *International Conference on Spoken Language Processing*, Korea, Oct. 4-8, 2004,
- [45] Starner T., Pentland A., "Visual Recognition of American Sign Language using Hidden Markov Models," *International Workshop on Automatic Face and Gesture Recognition*, Zurich, Switzerland, 1995.
- [46] Taskiran C.M., Bouman C.A., Delp E.J., "Discovering Video Structure Using the Pseudo-Semantic Trace", *Proceedings of SPIE*, San Jose, CA. Vol. 4315, pp. 571-578, 2001.
- [47] Vasconcelos N., "On the Efficient Evaluation of Probabilistic Similarity Functions for Image Retrieval," *IEEE Transactions on Information Theory*, Vol. 50(7). pp. 1482-1496, July 2004.

- [48] Vaswani N., Chowdhury A.R., Chellappa R., "Shape Activity: A Continuous State HMM for Moving/Deforming Shapes with Application to Abnormal Activity Detection," *IEEE Trans. on Image Proc.*, to appear.
- [49] Vinciarelli A., Bengio S., "Offline Cursive Word Recognition using Continuous Density Hidden Markov Models trained with PCA or ICA Features," *Sixth International Conference on Pattern Recognition (ICPR 2002)*. Vol. 3, pp. 81-84.
- [50] Wilson A.A., Bobick A.F., "Hidden Markov Models for Modelling and Recognizing Gesture under Variation," *Hidden Markov Models: Applications in Compute Vision*, pp. 123-160, 2001.
- [51] Wrigley S.N., Brown G.J., Wan V., Renals S., "Speech and Crosstalk Detection in Multichannel Audio," *IEE Transactions on speech and audio processing*, Vol. 13 (1), Jan. 2005.
- [52] Xie L., Chang S.F., Divakaran A., Sun H., "Structure Analysis of Soccer Video with Hidden Markov Models," *IEEE Inter. Conf. on Acoustic, Speech and Signal Processing*, Orlando, FL, May 2002.
- [53] Yacoob Y., Black M. J., "Parameterized Modeling and Recognition of Activities," *Computer Vision and Image Understanding*, Vol. 73 (2), pp. 232-247, Feb. 1999.