# Learning Dynamics for Exemplar-based Gesture Recognition

Ahmed Elgammal [†] Vinay Shet [‡] Yaser Yacoob [‡] Larry S. Davis [‡]

[†] Department of Computer Science, Rutgers University, Piscataway, NJ, USA
[‡] Computer Vision Laboratory, University of Maryland, College Park, MD, USA

## Abstract

*This paper addresses the problem of capturing the dynamics for exemplar-based recognition systems. Traditional HMM provides a probabilistic tool to capture system dynamics and in exemplar paradigm, HMM states are typically coupled with the exemplars. Alternatively, we propose a non-parametric HMM approach that uses a discrete HMM with arbitrary states (decoupled from exemplars) to capture the dynamics over a large exemplar space where a nonparametric estimation approach is used to model the exemplar distribution. This reduces the need for lengthy and non-optimal training of the HMM observation model. We used the proposed approach for view-based recognition of gestures. The approach is based on representing each gesture as a sequence of learned body poses (exemplars). The gestures are recognized through a probabilistic framework for matching these body poses and for imposing temporal constraints between different poses using the proposed non-parametric HMM.*

## 1 Introduction

The recognition of human gestures has many applications in human computer interaction, virtual reality and in robotics. In the last decade there has been a notable extensive interest in gesture recognition in the computer vision community [5, 6, 21, 23, 24, 2] as part of a wider interest in the analysis of human motion in general. The approaches used for gesture recognition and analysis of human motion in general can be classified into three major categories: model-based, appearance-based, and motion-based. Model-based approaches focus on recovering three-dimensional configuration of articulated body parts, e.g., [17]. Appearance-based approaches uses two dimensional information such as gray scale images or body silhouettes and edges, e.g., [21]. In contrast, motion based approaches attempt to recognize the gesture directly from the motion without any structural information about the physical body, e.g., [15, 2]. In all these approaches, the temporal properties of the gesture are typically handled using Dynamic Time Warping (DTW) or statistically using Hidden Markov Models (HMM) such as [21, 3, 11, 24, 23]

This paper presents an exemplar-based approach for view-based recognition of gestures. The approach is based on representing each gesture as a sequence of body poses (exemplars) through a probabilistic framework for matching these body poses to the the image data. Previous exemplar-based approaches, such as [7, 22], couple the system dynamics with the exemplars where training data are used to learn both the system dynamics and the exemplar representation. Alternatively, we use a model where the dynamics of the system is decoupled from the exemplars in order to achieve orthogonality between the spatial and temporal domains. The main contribution of the paper is introducing a non-parametric estimation approach for learning the dynamics from large exemplar spaces where the exemplars are decoupled from the dynamics. This approach has advantages that will be pointed out through the paper.

The paper is organized as follows: Section 2 review the exemplar-based tracking paradigm and emphasizes some of the draw backs that arises when using it for gesture recognition. Section 3 introduced the decoupled model and the proposed learning approach. Sections 4 and 5 present details about the observation model and the gesture classification. Section 6 contains some experimental results with simulation data and results of the proposed approach for recognizing arm gestures.

## 2 Exemplar-based Model

We use the definition of [8, 7]: An exemplar space is specified by a set of "exemplars", $\mathbf{X} = \{\mathbf{x}^k, k = 1 \cdots K\}$, containing representatives of the training data, and a distance function $\rho$ that measures the distortion between any two points in the space. The work of [7] was a major step towards learning probabilistic models for exemplars and their spatial transformations where exemplars are considered to be centers of a probabilistic mixture. The work of [22] was another major step that introduces the use of exemplars in a metric space within the same framework. Figure 1 shows the probabilistic graphical model for exemplar-based tracking as introduced by [7, 22]. The observation $z_t$ at time $t$ is considered to be drawn from a probabilistic mixture, i.e., $z_t \approx T_\alpha x_t$ where $x_t \in \mathbf{X}$ is the exemplar at time $t$ and $T_\alpha$ is

Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'03)
1063-6919/03 $17.00 © 2003 IEEE

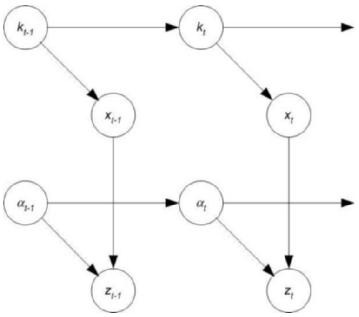


**Figure 1. Graphical model for exemplar-based tracking**

a geometric transformation with parameter $\alpha$. In this model, $k_t$ is the exemplar index at time $t$ and $\alpha_t$ is the transformation parameter at time $t$ for the geometric transformation $T_\alpha$. Therefore the system state $X_t = (k_t, \alpha_t)$ at time $t$ is both the exemplar index $k_t$ and transformation parameters $\alpha_t$. Learning this probabilistic model involves learning the exemplars as a small set of representatives from the training set, learning the dynamics in the form of $P(X_t|X_{t-1})$ which, typically, is assumed to be a Markovian process.

In gesture recognition, in general, there are two orthogonal types of style variability of performing the actions:

1. Temporal style variation: variations due to how fast or how slow individuals perform different parts of the gesture.

2. Spatial style variation: due to the physical constraints of the body, the appearance of the body at corresponding points of time is different between different individuals. Here we are concerned with the spatial variations that can not be modelled by the geometric transformation parameter $\alpha$.

The way the exemplar-based paradigm, as defined in [22, 7], handles this problem is to cluster the training data into representative exemplars in the spatial domain, assuming that the transformation parameters $\alpha$ will take care of the different spatial styles and that the learning will lead to compact clusters. The dynamics is learned through learning the transition $P(x_t|x_{t-1})$ between different exemplars. So, learning the dynamics is coupled with the exemplars, which are driven by spatial similarity, i.e., the orthogonality between the spatial domain and the temporal domain is not met. This presents a major draw back in the probabilistic model introduced by [22, 7] when used in recognition.

In order to solve this problem, we use an alternative probabilistic model that decouples the state variables from the exemplars as shown in figure 2. In this case, the state variable $q_t$ at time $t$ is an *abstract* variable that is independent of the exemplars as in the traditional sense of an Hidden Markov model state while the exemplars are intermediate observations that are being emitted by the underlying process. The final observation, $z_t$, remains as a probabilistic mixture of the exemplars. This way, the orthogonality between the spatial domain (the exemplars) and the temporal domain (the states) can be achieved. At each point of time, the system state is an independent abstract concept and therefore at certain point of time during the time span of the gesture different clusters (exemplars) might occur. On the other hand, the same cluster might occur at different points in time.

Given this decoupled model, a leaning approach is needed to learn $P(x_t|q_t)$, i.e., the intermediate observation (exemplar) probability given the state, and the dynamics $P(q_t|q_{t-1})$. If the number of exemplars is small, then this is the same as the traditional discrete output HMM learning, where each exemplar can be treated as a discrete symbol and the learning is done using the tradition Baum-Welch approach. Unfortunately, this is not the case that we are interested in. Instead, we are interested in the case where the number of exemplars is large for the reasons pointed below. This requires an alternative approach to learn the intermediate observation probabilities. This will be introduced in Section 3.2

Here we introduce three motivating scenarios for which our approach will be advantageous:

**Large Number of Clusters:** Consider the case where there are many different spatial styles of performing a gesture, where these variations in style can not be captured by the geometric transformation $T_\alpha$. This will leads to many exemplars, i.e., $K$ is large. Learning the dynamics in the form of $P(X_t|X_{t-1})$ or $P(x_t|x_{t-1})$ will be problematic because we will not have enough data per cluster to learn the transitions. Therefore the learned model of the dynamics is expected to specialize to the training sequences and to have poor generalization. The approach we propose is advantageous in this case because system state, $q_t$ is decoupled from the exemplars and therefore the number of possible states can be limited which will lead to better generalization.

**Data-driven Dynamics:** Consider the case where we want to learn the dynamics from the training data directly, i.e., no clustering is performed. This is necessary if it is not possible to cluster the data in a meaningful way due to the high dimensionality of the data. For example, in the metric mixture case [22], a one dimensional distance function $\rho$ is used to compute the exemplar distance in spite of the fact that the underlying dimensionality is high. This is because clustering the data in the original space will require complex intermediate representations such as parameterized contour models or a 3D articulated model as was indicated by [22]. As a result of using such a low dimensional function, clusters will not necessarily correspond to the real clusters in the original high dimensional space, specially if we have high variability in the data. So the approach provided in this paper moves one step forward to learn the dynamics from the

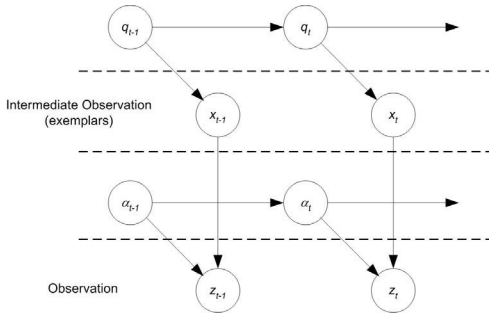



**Figure 2. Graphical model with hidden state decoupled from exemplars**

training data directly, without the need for clustering. We will use a low dimensional distance function, $\rho$, but in this case, since no clustering is needed, the effect of this function on learning the dynamics is minimal.

**Imposing Constraints on the Dynamics:** Dynamics in the form of $P(X_t|X_{t-1})$, as used in the exemplar-based paradigm as presented in [22], do not facilitate imposing constraints on the learned dynamics. For example, in many gestures, the progress of the gesture is moving forward in time and there is no meaning to going backward. To impose such a constraint, certain HMM topologies might be better. For example, a left-right model as shown in figure 5 or other topologies. By decoupling the states from the exemplars, this can be achieved as traditionally done in HMMs

# 3 Model Dynamics

## 3.1 Probabilistic Model

As illustrated in figure 2, at each discrete time, $t$, the system state is denoted by the pair $(q_t, \alpha_t)$ and $x_t$ denotes the exemplar at time $t$. The hidden variable $q_t$, representing a Markov stochastic process, can take any value from a set of $M$ distinct abstract states, $S = \{s_1, s_2, \cdots, s_M\}$, . The R.V. $x_t$ can be any exemplar from the set of exemplars $\mathbf{X} = \{\mathbf{x}^k, k = 1, \cdots K\}$. So, there is no coupling between the states and the exemplars. The system dynamics is now defined by the transitions $P(q_t|q_{t-1})$ and $P(\alpha_t|\alpha_{t-1})$.

The observation $z_t$ at time $t$ is a probabilistic mixture from all the exemplars and can be calculated using

$$P(z_t|q_t, \alpha_t) = \sum_{k=1}^{K} P(z_t|x_t = \mathbf{x}^k, \alpha_t) P(x_t = \mathbf{x}^k|q_t, \alpha_t) \tag{1}$$

We will drop the transformation parameter $\alpha$ from the equations since handling this parameter is well studied by the work of [7]. So the observation at time $z_t$ is

$$P(z_t|q_t) = \sum_{k=1}^{K} P(z_t|x_t = \mathbf{x}^k) P(x_t = \mathbf{x}^k|q_t) \tag{2}$$

We will call the term $P(x_t = \mathbf{x}^k|q_t)$ the intermediate observation probability.

## 3.2 Nonparametric Exemplar Density Estimation

Learning involves learning the transition matrix, i.e, $P(q_t|q_{t-1})$, learning the initial state distribution and learning the intermediate observation probabilities $P(x_t = \mathbf{x}^k|q_t)$. If the number of exemplars are small then this is the same setting as the traditional discrete output HMM and learning can be done using Baum-Welch method [16]. As was discussed above we are not interested in this case. Instead, we are interested in learning the dynamics directly from the whole training set, i.e., the set of exemplars is the whole set of training examples. There is one fundamental issue that needs to be addressed: In this case, the number of exemplars (discrete symbols) $K$ is much larger than the number of states $N$, i.e, $K \gg N$. In fact, if we use the whole training data as the exemplars set $\mathbf{X}$, then during the training each exemplar will be seen exactly once each iteration, i.e., the intermediate observation matrix $B = \{b_{kj} = P(x_t = \mathbf{x}^k|q_t = j)\}$ will have only one entry across each row. In general, if $K \gg N$ the probability of observing certain exemplar $\mathbf{x}^i$ might be very small although the probability of observing another exemplar $\mathbf{x}^k$ which is very similar might be high. This is because the exemplars are treated as discrete symbols without any similarity metric imposed.

This problem arises because of the discrete manipulation of the data. In fact, the actual observation is a continuous function. Treating the output as continuous requires estimating the pdf of the state output (observation model). Traditionally, continuous output HMM has been used to model speech signal for speech recognition and image signals for gesture recognition. Most of these systems use parametric forms for the observation model pdf. Typically a mixture of gaussian is used to model this density where the training data is used to estimate the Gaussian mixture parameters. This process in not optimal since both the Markovian assumption as well as the pdf form are not good models of the gesture. The same applies for speech signal as was noted by [18]

To overcome the problem of estimating observation model pdf, non-parametric density estimation can be used, given that large sets of observations can be obtained. In this case, the training data is not used to estimate pdf parameters, instead the data is used to obtain an estimate of the pdf directly. One main advantage of this approach is that new estimates of the pdf can obtained as more data is obtained. Different Non-parametric approaches have been used to estimate state output pdf in speech recognition as in the work of [20, 4]

Given the set of all training data $\mathbf{X} = \{\mathbf{x}_k, k = 1 \cdots K\}$ an estimate of any data point, $\mathbf{x}$ , (exemplar) can be obtained using the estimator

$$\hat{P}(\mathbf{x}) = \frac{1}{K} \sum_{i=1}^{K} \psi_h(\rho(\mathbf{x}, \mathbf{x}^i))$$

where $\psi_h$ is a kernel function with bandwidth $h$ applied on the exemplar distance function $\rho$. Let $\xi(j,i)$ be the expected number of times in state $j$ and observing exemplar $i$, and $\xi(j)$ be the expected number of times in state $j$ during one cycle of the training and let $M$ be the total number of observations [1]. We can obtain an estimate of the joint distribution as

$$\hat{P}(x_t = \mathbf{x}^k, q_t = j) = \frac{1}{M} \sum_{i=1}^{K} \psi_h(\rho(\mathbf{x}^k, \mathbf{x}^i))) \cdot \xi(j,i)$$

Therefore we can obtain an estimate for the exemplar observation probability $\hat{b}_{kj}$ using

$$\begin{aligned} \hat{b}_{kj} &= \hat{P}(x_t = \mathbf{x}^k | q_t = j) = \frac{\hat{P}(x_t = \mathbf{x}^k, q_t = j)}{\hat{P}(q_t = j)} \\ &= \frac{\frac{1}{M} \sum_{i=1}^{K} \psi_h(\rho(\mathbf{x}^k, \mathbf{x}^i))) \cdot \xi(j,i)}{\frac{\xi(j)}{M}} \\ &= \sum_{i=1}^{K} C_{ji} \cdot \psi_h(\rho(\mathbf{x}^k, \mathbf{x}^i))) \end{aligned} \quad (3)$$

We call $C_{ji}$ the occupancy coefficients which can be computed during the training as:

$$C_{ji} = \frac{\xi(j,i)}{\xi(j)} \quad (4)$$

Simply, the occupancy coefficients are the traditional discrete HMM observation probabilities.

We need to modify the Baum-Welch learning approach as follows:

**Expectation Step** use the estimate $\hat{P}(x_t = \mathbf{x}^k | q_t = j)$ from equation 3 to evaluate the observation probabilities of the training sequences.

**Maximization Step** update only the coefficient matrix $C = \{C_{ji}\}$ as in the traditional Baum-Welch using equation 4.

If we take a closer look at Equation 3 we can see that it resembles the observation probability of the continuous output HMM model, which is computed as a mixture of $m$ distributions

$$b_j(O) = \sum_{m=1}^{M} C_{jm} \phi(O; \mu_m, \Sigma_m) \quad (5)$$

where $\phi$ is typically a Guassian with mean $\mu_m$ and covariance $\Sigma_m$ for the $m$th mixture and $C_{jm}$ is the mixture coefficient for the $m$th mixture in state $j$[2]. This form of HMM requires estimation of the parameters $\mu_m$ and $\Sigma_m$ for each of the distributions as part of the training process, which is a lengthy and non-optimal process [18]. Instead, the non-parametric formalization of equation 3 avoids this parameter estimation process.

[1] $M = K$ if we use the whole training data as the exemplars

[2] This particular case of HMM where the mixture is shared among all states is called semi-continuous HMM

# 4 Observation Model

## 4.1 Pose Likelihood

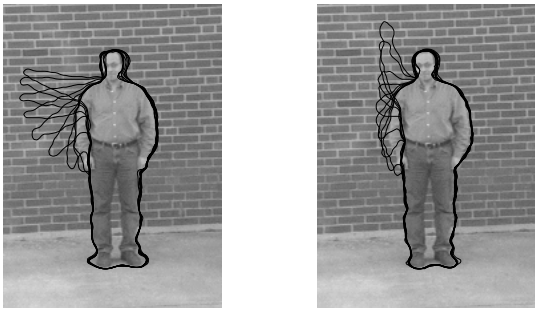


**Figure 3. Pose exemplars registered to an image**

We represent each gesture as a sequence of body poses (exemplars). The temporal relation between these body poses is enforced using the probabilistic model presented in section 3. This section focuses on matching individual body poses. The objective is to evaluate all different exemplars with respect to each new frame at time $t$ in order to obtain estimate of observation likelihood given each pose exemplar $P(z_t | x_t = \mathbf{x}^k)$.

Let the set of shape exemplars be $\mathbf{X} = \{\mathbf{x}^k, k = 1, \cdots, K\}$ which contains all learned body poses for all the gestures to be recognized. Each pose (exemplar) is an edge template representing the body silhouette, i.e., each pose is represented as a finite set of contour points image coordinates.

$$\mathbf{x}^k = \{y_1^k, y_2^k, \cdots, y_{m^k}^k\},$$

where $y_i^k \in R^2$ and $m^k$ is the number of points along the contour for pose exemplar $k$. Figure 4 shows example pose templates for two different gestures. All the poses are aligned to each other during the learning so aligning one pose to any new image will therefore align the rest of the poses. Registering these poses to the images is done while the person is not performing any gesture (idle). In this case, the matching is performed using an idle pose (shown in figure 4, first pose on top) through a coarse to fine search using an image pyramid. Figure 3 shows the registered poses to a new frame.

At each new image, $I$, it is desired to find a probabilistic matching score for each pose. Let $d_F(x)$ be the distance transformed image at image location, $x$, given the set of edge features, $F$, detected at image $I$. For each edge feature $y_i^k$ in pose model $\mathbf{x}^k$, the measurement $D_i^k = d_F(y_i^k)$

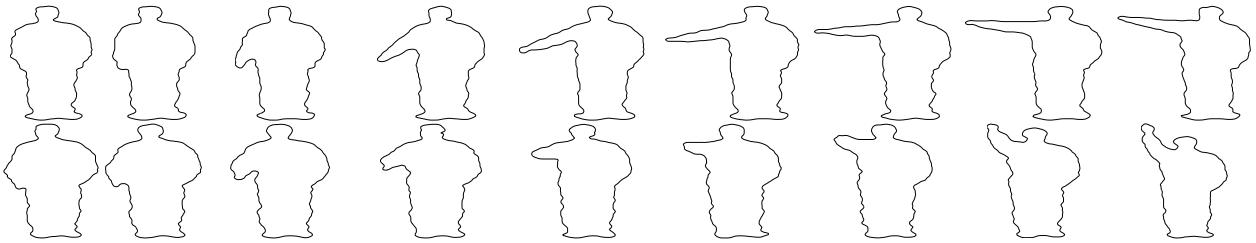


**Figure 4. Example body poses from two different Gesture**

is the distance to the nearest edge feature in the image. A perfect model to image match will have $D_i^k = 0$ for all model edge features. Consider the random variable associated with this distance measurement, and let the associated probability density function (PDF) be $p_i^k$. We assume that these random variables are independent. This assumption was used in [12, 14] based on the results obtained in [13]. Therefore, the observation likelihood (the probability of the observation given the exemplar $\mathbf{x}^k$) can be defined as the product of these PDFs as

$$P(z_t|x_t = \mathbf{x}^k) = \prod_{i=1}^{m^k} p_i^k(D_i^k) \tag{6}$$

The PDF $p_i^k$ for the distance between model features and nearest image feature location is defined for each feature $i$ in each pose model $k$. We use a PDF of the form

$$p_i^k(D) = c_1 + \frac{1}{\sigma\sqrt{2\pi}} e^{-D^2/2(\sigma)^2}$$

Since the distance $D$ can become arbitrary large, the probability can become very small and therefore the constant $c_1$ is used as a lower bound on the probability. This makes the likelihood function robust to outliers.

This probabilistic formulation was first introduced in [12] and was used in a Hausdorff matching context to find the best transformation of an edge template using maximum likelihood estimation. Equation 6 represents a probabilistic formulation for Chamfer matching. Chamfer distance has been used extensively in object detection, for example in [10]. In [9] the matching was generalized to include multiple feature types, (for example, oriented edges) by matching each individual feature template with its corresponding distance transformed image and combining the results. Also the matching was generalized in [9, 10] to match multiple templates through a hierarchical template structure.

### 4.2 Weighted Matching

Our objective is to match multiple pose exemplars to the same image location in order to evaluate the likelihood of the observation given each of these poses. Typically, the different pose templates are similar in some parts and different in another parts in the templates. For example, the head, torso and bottom parts of the body are likely to be similar in different pose templates, while articulated body parts that are involved in the gesture, such as the arm, will be at different positions at different pose templates. For example, see figure 3. Since the articulated part, such as the arm, is represented by a small number of features with respect to the whole pose templates, the matching is likely to be biased by the major body parts. Instead, it is desired to make the matching biased more by articulated parts involved in performing the gesture since these parts will be more discriminating between different poses templates.

To achieve this goal, different weights are assigned to different feature points in each pose exemplar. Therefore each pose exemplar, $\mathbf{x}^k$, is represented as a set of feature locations as well as a set of weights, $\{w_1^k, w_2^k, \cdots, w_{m^k}^k\}$, corresponding to each feature where $\sum_{i=1}^{m^k} w_i^k = 1$. The likelihood equation 6 can be written in terms of weighted log-likelihood as

$$\log P(z_t|\mathbf{x}^k) = \sum_{i=1}^{m^k} w_i^k \log p_i^k(D_i^k) \tag{7}$$

In our case, the set of all recognized poses does not have a common correspondence frame. For example, some features in one pose might not have corresponding features in another pose. Also we do not restrict the pose templates to have the same number of features. Therefore we drive the weights with respect to the image locations.

Let $X$ be the set of all features in all registered poses in the training data, i.e.,

$$X = \bigcup_k \mathbf{x}^k = \{x_1, x_2, \cdots x_m\}$$

where each $x_i$ is the image location of an edge feature. Given this sample of edge feature locations, the edge probability distribution $f(y)$ (the probability to see an edge at certain image location, $y$) can be estimated using kernel density estimation [19] as

$$\hat{f}(y) = \frac{1}{m} \sum_{i=1}^{m} K_h(y - x_i)$$

Where $K_h$ is a kernel function with a scale variable $h$. We

used a Gaussian kernel $K_h(t) = \frac{1}{\sqrt{2\pi}h} e^{-1/2(\frac{t}{h})^2}$ for this probability estimation.

The weight assigned to each feature point is based on the information this feature provides. Given the estimated edge probability distribution, $\hat{f}(y)$, at any image pixel, $y$, the weight for a certain feature $i$ at a certain pose $k$ is the ratio of the information given by this feature to the total information by that pose, i.e.,

$$w_i^k = \frac{\log \hat{f}(x_i^k)}{\sum_{j=1}^{m^k} \log \hat{f}(x_j^k)}$$

## 5 Gesture Classification

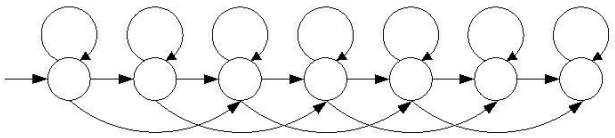


**Figure 5. Left-Right HMM**

This section summarizes the gesture classification procedure. We represent each gesture $g$ by an exemplar space, $(\mathbf{X}^g, \rho)$, and an HMM , $\lambda^g$. The exemplar space is defined by a set of shape exemplars $\mathbf{X}^g = \{\mathsf{x}^k, k = 1, \cdots, K\}$, representing different body poses during the gesture, and a distance function $\rho$ which is defined as the symmetric chamfer distance, i.e., given exemplar $x^k = \{y_1^k, \cdots, y_{m^k}^k\}$ and exemplar $x^l = \{y_1^l, \cdots, y_{m^l}^l\}$ the symmetric distance $\rho(x^k, x^l)$ is computed as,

$$\rho(x^k, x^l) = \frac{1}{m^l} \sum_i^{m^l} d_{x^k}(y_i^l) + \frac{1}{m^k} \sum_i^{m^k} d_{x^l}(y_i^k).$$

The HMM hidden states correspond to the progress of the gesture with time. We used a left-right model as in figure 5 to impose a constraint on the dynamics which leads to better generalization since there are less transitions to adjust. Note that the number of states is different from the number of poses as mentioned above. Training sequences of poses are used to learn the model parameters:

1. The state transition probabilities $A = \{a_{ij}\}$ where $a_{ij} = P[q_{t+1} = s_j | q_t = s_i] \quad \forall i, j = 1 \cdots N$.
2. The initial state distribution $\pi$ where $\pi_j = P[q_1 = s_j] \quad \forall j = 1 \cdots N$.
3. The probability of each pose $\mathsf{x}^k$ given the states, $B = \{b_{kj} = P(\mathsf{x}^k | s_j) \quad \forall j = 1 \cdots N, \forall k = 1 \cdots K\}$

The learning is performed using the nonparametric learning approach in section 3.2.

The actual observation $z_t$ is the detected edge features at each new frame, which is a probabilistic function of the current state of the gesture as defined in equation 2. The observation probabilities given the exemplar, $P(z_t|\mathsf{x}^k)$, are obtained using the likelihood equations 6 and 7 as was described in section 4

Given a set of observations $\mathcal{Z} = z_1 z_2, \cdots, z_T$ and given a set of HMM models $\lambda^g$ corresponding to different gesture, the objective is to determine the probability of that observation sequence given each of the models, i.e., $P(\mathcal{Z}|\lambda^g) \forall g$. This is a traditional problem for HMM and can be solved efficiently through the Forward-Backward procedure [16, 1].

## 6 Experimental results

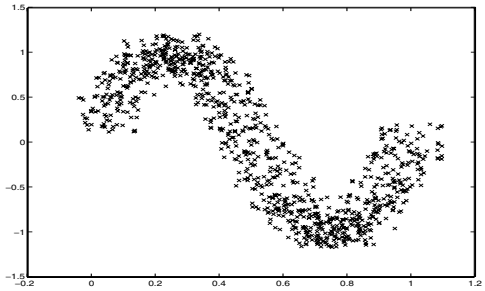
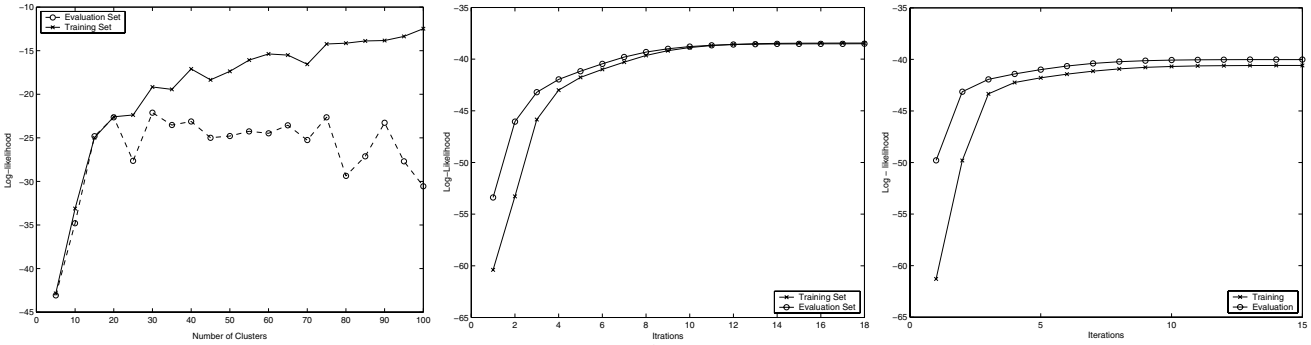



**Figure 6. Simulation data.**

We performed simulation experiments to evaluate the performance of the proposed learning approach. We used synthetic data to emulate a gesture progressing over time with spatial and temporal variations. We used data of the form $y = a\sin(bx) + c$ where the parameters $a, b, c$ were generated randomly. The variation parameters $a, b, c$ are meant to emulate actual gesture spatial variations that cannot be recovered through the geometric transformation parameter $\alpha$ as discussed in section 2. Figure 6-a illustrates the used data (50 sequences of 20 points each). We modelled the dynamics of the data using both the coupled exemplar-based model (shown in figure 1) and using an the uncoupled model (shown in figure 2). For the uncoupled model, we estimated the intermediate observations using the non-parametric approach presented in section 3.2. In both cases we performed cross-evaluation by splitting the data into training and test sets. For the coupled model, we varied the number of clusters at each learning experiment where the cluster centers are considered to be the exemplars. Figure 7-a shows the log-likelihood for both the training data and the test data for different number of clusters. As expected, as the number of clusters increases, the model specializes to the training data and yields poor generalization. For the uncoupled nonparametric model there is no clustering performed and the learning is done from the whole training data. Figures 7-b,c show the learning curves (in terms of log-likelihood) with two different kernels: a Gaussian kernel and a Parzen window respectively. Each figure shows the log-likelihood for both the training and test data. As can be noticed from the figures, in both cases the




(a) (b) (c)

**Figure 7. (a) Coupled model evaluation with different clustering. (b,c) Nonparametric learning of uncoupled model with Gaussian kernel and a Parzen window**

leaning curves converge which shows the generalization.

The proposed approach was experimented with real images for arm gesture classification. Here we show two different gesture classification experiments. In the first experiment, the proposed approach was used to classify eight arm gestures. Basically, the eight recognized gestures are similar to the ones shown in figure 3, performed with each arm, in upward motion and downward motion. Figure 8 shows some pose classification results for different people. The figures shows the pose with the highest likelihood score overlaid over the original image.

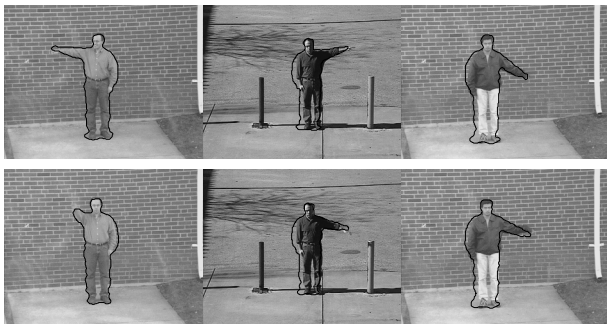


**Figure 8. Pose matching results**

Figure 9 shows the gesture likelihood probabilities for the eight gesture classes. from the graphs, all the gestures were close in likelihood at the beginning of the action but as the gesture progresses with time, the likelihood of the correct gesture increases, and the the likelihood of the other gesture decreases as a result of the temporal constrains imposed by the HMM for each gesture. Each plot shows two consecutive gestures performed and as one gesture was recognized all the HMMs were reset.

In the second experiment we used six gestures as shown in figure 10 where in this case each gesture consists of an upward followed by a downward arm motion. In this experiments, pose exemplars were obtained from five different people where the total number of exemplars were 203, 195, 194, 216, 232, 208 exemplars for each of the six gesture classes respectively. Six fixed topology HMMs with 13 states each were used (one for each gesture class) where learning the HMM parameters and the intermediate observation probabilities was performed using the approach described in section 3.2. We trained five sets of HMMs where each HMM set was trained with one person's exemplars left out. For the evaluation we used 150 gesture sequences (5 people $\times$ 5 cycles $\times$ 6 gestures. For classifying any gesture sequence for a particular person, we used an HMM set that was trained with this person's exemplars left out. The classification results are shown in figure 6 in terms of a confusion matrix.

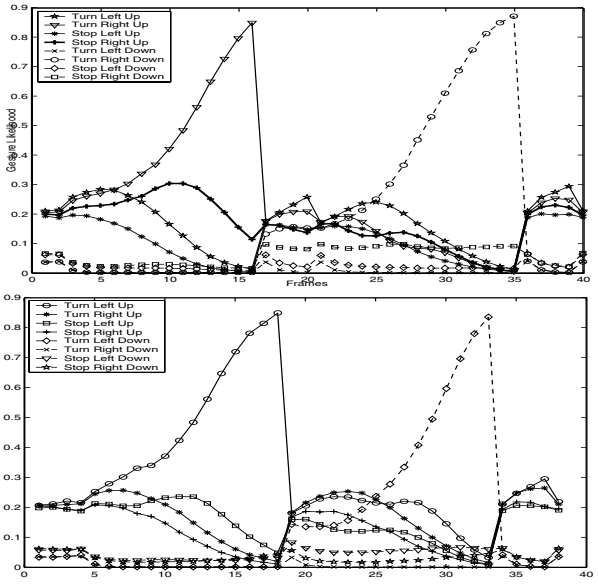



**Figure 9. Gesture classification results**

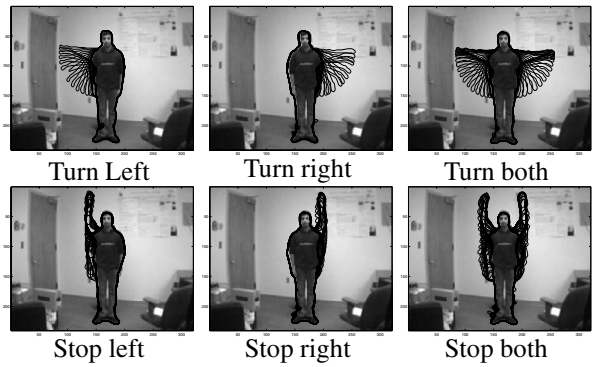



**Figure 10. Six gestures: Each gesture consists of an upward followed by a downward arm motion**

| | | | | | | |
|---|---|---|---|---|---|---|
| turn left | 23 | 0 | 0 | 2 | 0 | 0 |
| turn right | 0 | 19 | 0 | 0 | 6 | 0 |
| turn both | 2 | 0 | 21 | 0 | 2 | 0 |
| stop left | 2 | 0 | 0 | 23 | 0 | 0 |
| stop right | 0 | 2 | 0 | 0 | 23 | 0 |
| stop both | 0 | 0 | 2 | 2 | 2 | 19 |

**Figure 11. Confusion Matrix**

## 7 Conclusion

The paper presented an approach for learning the dynamics for exemplar-based recognition systems and its application to gesture recognition. The key contribution of the paper is an approach for learning HMM parameters that utilizes nonparametric density estimation for modeling the observation density. This facilitates learning the dynamics from large exemplar spaces where nonparametric estimation is used to model the exemplar distribution. The probabilistic model used decouples the system dynamics from the exemplar space in order to achieve orthogonality between the spatial and temporal domains. The approach was experimented with simulation data and was used to recognize simple arm gestures. The experiments showed the ability of the learning approach to capture the system dynamics.

## References

[1] Y. Bengio. Markovian models for sequential data. *Nueral Computing Suveys*, pages 129–162, 1999.

[2] A. F. Bobick and J. W. Davis. The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):257–267, 2001.

[3] M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. In *Proc. CVPR*, 1997.

[4] M. C., C. M.-J., and L. F. K-nn versus gaussian in a hmm-based recognition system. In *Eurospeech, Rhodes*, pages 529–532, 1997.

[5] T. Darrell and A. Pentland. Space-time gesture. In *Proc IEEE CVPR*, 1993.

[6] J. Davis and M. Shah. Visual gesture recognition. *Vision, Image and Signal Processing*, 141(2):101–106, 1994.

[7] B. J. Frey and N. Jojic. Learning graphical models of images, videos and their spatial transformation. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence - San Francisco, CA*, 2000.

[8] B. J. Frey and N. Jojic. Flexible models: A powerful alternative to exemplars and explicit models. In *IEEE Computer Society Workshop on Models vs. Exemplars in Computer Vision*, pages 34–41, 2001.

[9] D. Gavrila. Multi-feature hierarchical template matching using distance transforms. In *Proc. of the International Conference on Pattern Recognition, Brisbane, Australia,1998.*, pages 439–444.

[10] D. Gavrila and V. Philomin. Real-time object detection for "smart" vehicles. In *ICCV99*, pages 87–93.

[11] C. Morimoto, Y. Yacoob, and L. Davis. Recognition of head gestures using hidden markov models international conference on pattern recognition. In *International Conference on Pattern Recognition, Vienna, Austria, August 1996*, pages 461–465, 1996.

[12] C. Olson. A probabilistic formulation for hausdorff matching. In *CVPR98*, pages 150–156.

[13] C. Olson and D. Huttenlocher. Automatic target recognition by matching oriented edge pixels. *IEEE Transactions on Image Processing*, (1):103–113, 1997.

[14] C. F. Olson. Maximum-likelihood template matching. In *IEEE International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 52–57, 2000.

[15] R. Polana and R. C. Nelson. Detecting activities. *Journal of Visual Communication and Image Representation*, June 1994.

[16] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–285, February 1989.

[17] J. M. Rehg and T. Kanade. Model-based tracking of self-occluding articulated objects. In *ICCV*, pages 612–617, 1995.

[18] S. Renals, N. Morgan, H. Bourlard, M. Cohen, and H. Franco. Connectionist probability estimators in HMM speech recognition. *IEEE Transactions Speech and Audio Processing*, 1993.

[19] D. W. Scott. *Mulivariate Density Estimation*. Wiley-Interscience, 1992.

[20] S. Soudoplatoff. Markov modelling of continous parameters in speech recognition. In *Proceedings of IEEE ICASSP*, pages 45–48, 1986.

[21] T. Starner and A. Pentland. Real-time american sign language recognition from video using hidden markov models. In *SCV95*, page 5B Systems and Applications, 1995.

[22] K. Toyama and A. Blake. Probabilistic tracking in a metric space. In *ICCV*, pages 50–59, 2001.

[23] C. Vogler and D. N. Metaxas. Parallel hidden markov models for american sign language recognition. In *ICCV (1)*, pages 116–122, 1999.

[24] A. D. Wilson and A. F. Bobick. Parametric hidden markov models for gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9):884–900, 1999.