Multi-Cue Exemplar-Based Nonparametric Model for Gesture Recognition

Vinay D. Shet[†] V. Shiv Naga Prasad[†] Ahmed Elgammal[‡] Yaser Yacoob[†]
Larry S. Davis[†]

[†]Computer Vision Laboratory,
University of Maryland,
College Park, MD, USA
[†]{vinay,shiv,yaser,lsd}@cs.umd.edu

[‡]Department of Computer Science,
Rutgers University,
Piscataway, NJ, USA
[‡]elgammal@cs.rutgers.edu

Abstract

This paper presents an approach for a multi-cue, view-based recognition of gestures. We describe an exemplar-based technique that combines two different forms of exemplars - shape exemplars and motion exemplars - in a unified probabilistic framework. Each gesture is represented as a sequence of learned body poses as well as a sequence of learned motion parameters. The shape exemplars are comprised of pose contours, and the motion exemplars are represented as affine motion parameters extracted using a robust estimation approach. The probabilistic framework learns by employing a nonparametric estimation technique to model the exemplar distributions. It imposes temporal constraints between different exemplars through a learned Hidden Markov Model (HMM) for each gesture. We use the proposed multi-cue approach to recognize a set of fourteen gestures and contrast it against a shape only, single-cue based system.

1. Introduction

Visual recognition of arm and hand gestures has a variety of applications in human machine interfaces, virtual reality and robotics. In the last decade there has been an extensive interest in gesture recognition in the computer vision community [6, 7, 20, 23, 24, 3] as part of a wider interest in the analysis of human motion. 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 the three-dimensional configuration of articulated body parts, e.g. [19]. Appearance-based approaches use two dimensional information such as gray scale images or body silhouettes and edges, e.g. [20]. In contrast, motion based approaches attempt to recognize the gesture directly from the motion without any structural information about the physical body, e.g. [17, 3]. We refer the reader to [11, 16, 14] for extensive surveys of related work. In all these approaches, the temporal properties of the gesture are typically handled using Dynamic Time Warping (DTW) or using Hidden Markov Models (HMM) [20, 15, 24, 23].

Computer vision researchers have addressed integrating multiple cues in different contexts. In the context of gesture recognition and body parts tracking, the use of multiple image cues and therefore multiple object representations facilitates robust exploitation of the rich visual information contained in image sequences. This leads to trackers that are more robust to background clutter [1]. Typically, the different cues are used to provide independent object representations such that even if one object representation fails to distinguish the object from the cluttered scene, other cues might enable discrimination.

The gesture recognition system proposed by us in [8] employed an HMM whose observation likelihoods were captured in a nonparametric manner using pose (shape)exemplars. The exemplars were sets of points along the body contours extracted during training. As we used only shape information, the system's discriminative capability was limited, confining us to recognize only 6 gestures. Here, we extend the work in [8] to include motion exemplars as an additional cue which permits us to increase our gesture vocabulary. These motion exemplars are affine motion flow parameters extracted during training. We show the performance of the system when used for recognizing a set of 14 arm gestures used for military signalling. We get a recognition rate of about 83%, as opposed to about 23% when using only shape features.

The organization of the paper is as follows: Section 2 briefly gives an overview of our gesture recognition system and the particular application of interest. Section 3 describes the proposed probabilistic model to handle multiple cues within the exemplar-based paradigm and the learning approach used. Sections 4, 5 describe details about the motion and shape observation models. Section 6 describes the experimental results and finally we conclude in Section 7.

2. Problem Definition

This system is designed to operate a robot driven vehicle by recognizing human hand gestures performed by a subject standing in front of the vehicle. In such a set up, the background is typically very complex and dynamic. Moreover, the camera mounted on the vehicle, as well as the subject could be moving. Because of these aspects of our application, several traditional approaches for gesture recognition, based on background subtraction, silhouette extraction [20], motion history image [3] etc. are impractical. At present this work solves the problem of gesture recognition from a static camera with a stationary subject, albeit in a way that the system should scale to handle both camera and subject motion. Addressing these issues will be part of our future work.

The proposed approach was used to classify a subset of arm gestures used for military signaling to control vehicle drivers and/or crews [22]. The gesture set contains fourteen different gestures as shown in Figure 1. The gestures are: *Turn-left*, *Turn-right*, *Flap*, *Stop-left*, *Stop-right*, *Stop both*, *Attention left*, *Attention right*, *Attention both*, *Start Engines*, *Speed Up*, *Come Forward*, *Go back*, *Close Distance*. The left and right in the gesture notation is with respect to the vehicle.

In order to handle complex backgrounds, we avoid extracting silhouettes of the subjects directly from the video. Instead we try to match the pose-shape using edge contours. Many of the gestures have similar body poses at different phases of the gesticulation as can be seen from Figure 1. Moreover, the last three gestures take place in front of the person and therefore shape information alone is not discriminative enough in these cases. Motion information, therefore, will play a crucial role in classifying these gestures.

The system learns temporal models of each individual gesture as a sequence of the learned body poses and motions through a multi-cue, nonparametric HMM. We then use the Maximum Likelihood criterion to classify between different gestures.

3. Gesture Probabilistic Model

Following the definition of [10, 9]: 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 ρ that measures the distortion between any two points in the space. In [9, 21, 10] exemplars were used to create a feature space and the temporal constraints were imposed using Markov chains. The states of the Markov chains were coupled with the exemplars, i.e., it was assumed that only a particular exemplar (or its noisy version) could be produced in any given state. In [8], we introduced a decoupled approach, wherein the observed label produced by any state of an HMM could be a mixture of a set of exemplars, the exemplar set being common for all the states.

In our previous work, we used the shape information of the poses as the single cue for the exemplars. Here, we consider the case of multi-cue exemplars, each cue c being represented by an exemplar set $\mathbf{X}^c$ and a distance function ρ^c. Different cues, generally have different representations, therefore we need a unified way of combining them.

Suppose we are given m different cues in the form of exemplar sets $\mathbf{X}^{c_1}, \mathbf{X}^{c_2}, \cdots, \mathbf{X}^{c_m}$, where the number of exemplars in each set is $N_1, N_2, \cdots, N_m$, and distance functions are $\rho^{c_1}, \rho^{c_2}, \cdots, \rho^{c_m}$. The observation at time t, $\mathbf{z}_t$ is an m-tuple i.e. $\mathbf{z}_t = \langle \mathbf{z}_t^1, \mathbf{z}_t^2, \ldots, \mathbf{z}_t^m \rangle$. If we follow the coupled dynamics, as presented in [9, 21, 10], then the number of states will be $N_1 \times N_2 \times \cdots \times N_m$. With this exponential increase, learning dynamics of the form $P(X_t|X_{t-1})$ would be intractable. This difficulty can be partially alleviated by making the dynamics for cues independent of one another, i.e., by learning the dynamics in terms of transitions between the same cue states $P(X_t^c|X_{t-1}^c)$ (we are assuming that the first order Markovian assumption suffices).

In our approach, by decoupling the states from the exemplars we are able to avoid an exponential increase in the number of states even with additional number of cues. The state variable q_t at time t is an abstract variable that is independent of the exemplars as in a traditional HMM, while the exemplars are intermediate observations that are being emitted by the underlying process. The final observation, $\mathbf{z}_t$, is a probabilistic mixture of the exemplars.

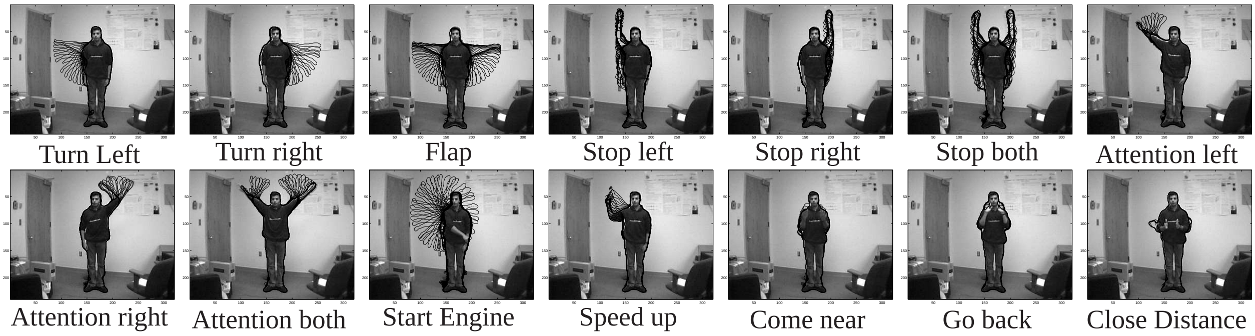
$$P(\mathbf{z}_t|q_t) = \prod_{i=1}^{m} P(\mathbf{z}_t^i|q_t) \tag{1}$$

Given the decoupled model, the dynamics are defined in terms of the transitions $P(q_t|q_{t-1})$ and the intermediate observation probabilities for each cue given the states $P(X_t^c|q_t)$.

3.1. Decoupled Model

In our model, without loss of generality, we will use two cues: shape cue, denoted by superscript s, and motion cue, denoted by superscript m.

At each discrete time, t, the system state is denoted by the hidden variable q_t, which can take any value from a set of M distinct abstract states, $S = \{s_1, s_2, \cdots, s_M\}$, representing a Markov stochastic process. The R.V. X_t^s represents the shape exemplar at time t which can be any one of the exemplars from the set of shape exemplars $\mathbf{X}^s =$



Turn Left, Turn right, Flap, Stop left, Stop right, Stop both, Attention left

Attention right, Attention both, Start Engine, Speed up, Come near, Go back, Close Distance

Figure 1. Shape exemplars for each gesture overlayed over the images

$\{\mathbf{x}_k^s, k = 1, \cdots, K^s\}$. The R.V. X_t^m represents the motion exemplar at time t which can be any one of the exemplars from the set of motion exemplars $\mathbf{X}^m = \{\mathbf{x}_k^m, k = 1, \cdots, K^m\}$. Thus, there is no coupling between the states and the exemplars for each of the cues. The system dynamics is now defined by the transitions $P(q_t|q_{t-1})$. Therefore, the dimensionality of the state space does not depend on the number of cues and, consequently, the number of possible states is independent of the number of exemplars for each cue and no longer increases exponentially with the number of cues.

The shape and motion observations $\mathbf{z}_t^s$,$\mathbf{z}_t^m$ at time t are probabilistic mixtures from the shape and motion exemplars, respectively, and can be calculated using

$$P(\mathbf{z}_t^s|q_t) = \sum_{k=1}^{K^s} P(\mathbf{z}_t^s|X_t^s = \mathbf{x}_k^s)P(X_t^s = \mathbf{x}_k^s|q_t) \quad (2)$$

$$P(\mathbf{z}_t^m|q_t) = \sum_{k=1}^{K^m} P(\mathbf{z}_t^m|X_t^m = \mathbf{x}_k^m)P(X_t^m = \mathbf{x}_k^m|q_t) \quad (3)$$

We will call the term $P(X_t^s = \mathbf{x}_k^s|q_t)$ and $P(X_t^m = \mathbf{x}_k^m|q_t)$ the intermediate observation probability for shape and motion respectively.

3.2. Learning Approach

Training involves learning the transition probabilities, $P(q_t|q_{t-1})$, the initial state distribution, and the intermediate observation (exemplar) probabilities for both shape and motion given the states, $P(X_t^s = \mathbf{x}_k^s|q_t)$ and $P(X_t^m = \mathbf{x}_k^m|q_t)$. The approach used for learning the parameters is a modified version of the Baum-Welch method [18] that utilizes nonparametric density estimation of the observation model PDF. The advantage of using nonparametric density estimation is that we do not need to design a "space" for the poses (e.g. parameters of an articulated body model) [4]. We simply use the exemplars themselves to create a basis. We introduced this approach in [8] and we extend it here to handle multi-cue observations.

Given an exemplar space for cue, c, defined by a set of exemplars $\mathbf{X}^c = \{\mathbf{x}_k^c, k = 1, \cdots, N_c\}$ and a distance function ρ^c, an estimate of the probability density function at any point, $\mathbf{x}$, can be obtained using a nonparametric estimator

$$\hat{P}(\mathbf{x}) = \frac{1}{N_c}\sum_{i=1}^{N_c} \psi_{h_c}(\rho(\mathbf{x}, \mathbf{x}_k^c))$$

where ψ_{h_c} is a kernel function (typically a Gaussian) with bandwidth h_c applied on the exemplar distance function ρ^c.

Let the set of shape exemplars be $\mathbf{X}^s = \{\mathbf{x}_k^s, k = 1, \cdots, N_s\}$ and the set of motion exemplars be $\mathbf{X}^m = \{\mathbf{x}_k^m, k = 1, \cdots, N_m\}$. We define the two R.V.s Y_j^s and Y_j^m, denoting the observed shape and motion exemplars respectively, at state j during the training and let $C_{ji}^s = P(Y_j^s = \mathbf{x}_i^s)$ and $C_{ji}^m = P(Y_j^m = \mathbf{x}_i^m)$. We can obtain estimates for the exemplar probabilities for both shape and motion denoted as, $\hat{b}_{kj}^s$ and $\hat{b}_{kj}^s$ respectively, using

$$\begin{aligned}\hat{b}_{kj}^s &= \hat{P}(X_t^s = \mathbf{x}_k^s|q_t = j) = \\ &\sum_{i=1}^{N} C_{ji}^s \cdot \psi_{h_s}(\rho(\mathbf{x}_k^s, \mathbf{x}_i^s))) \quad (4)\\ \hat{b}_{kj}^m &= \hat{P}(X_t^m = \mathbf{x}_k^m|q_t = j) = \\ &\sum_{i=1}^{N} C_{ji}^m \cdot \psi_{h_m}(\rho(\mathbf{x}_k^m, \mathbf{x}_i^m))) \quad (5)\end{aligned}$$

where ψ_{h_s} and ψ_{h_m} are kernel functions with bandwidths h_s and h_m, applied on the exemplar distance functions ρ^s and ρ^m. We call C_{ji}^s and C_{ji}^m the occupancy coefficients, which can be computed during the training by counting. For example for the shape case as:

$$C_{ji}^s = \frac{\sharp(j, i)}{\sharp(j)} \quad (6)$$

where $\sharp(j, i)$ is the expected number of times in state j and observing shape exemplar i and $\sharp(j)$ is the expected number of times in state j during the training. Similarly, we can compute C_{ji}^m for the motion case.

As a summary, we need to modify the Baum-Welch learning approach as follows:

Expectation Step: Use the estimate $\hat{P}(X_t^s = \mathbf{x}_k^s | q_t = j)$, $\hat{P}(X_t^m = \mathbf{x}_k^m | q_t = j)$ from equation 4, 5 to evaluate the observation probabilities of the training sequences.

Maximization Step: Update only the coefficient matrices $C^s = \{C_{ji}^s\}$, $C^m = \{C_{ji}^m\}$ as in the traditional Baum-Welch using equation 6.

4. Motion Observation Model

4.1. Motion Estimation

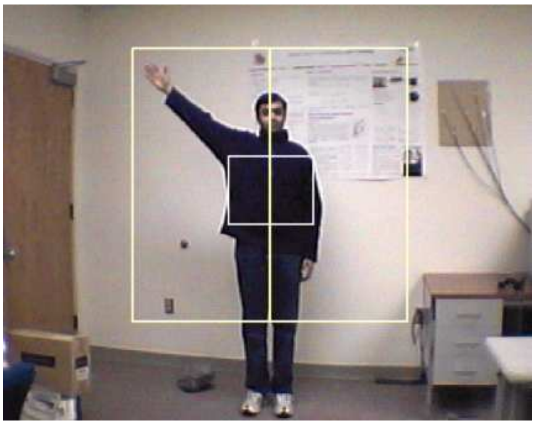


Figure 2. Image windows used to extract motion parameters

This section describes the estimation of the motion parameters and the motions observation probabilistic model. The objective is to parameterize the motion observation corresponding to the gesture and to obtain estimates for the probability of such observation given the learned exemplars.

Given the subject's location in the image, we divide the space around the subject into three motion spaces, one at each side of the body to the full extent of the arm and the third centered on the chest. Each window represents a separate motion space, and the motion is parameterized in each of these windows. Figure 2 illustrates these motion spaces. Since the arm motion in each window is not the dominant motion, the motion estimation approach should be able to estimate multiple motions in each window. To accomplish this, we use the robust motion estimator proposed by Black and Anandan [2]. A motion flow field is computed for each window using a brightness constancy constraint. We assume an affine flow model to characterize the motion flow, $\mathbf{u}(.)$, computed within a window.

$$\mathbf{u}(x, y; \mathbf{a}) = \begin{bmatrix} u(x,y) \\ v(x,y) \end{bmatrix} = \begin{bmatrix} a_o + a_1 x + a_2 y \\ a_3 + a_4 x + a_5 y \end{bmatrix}$$

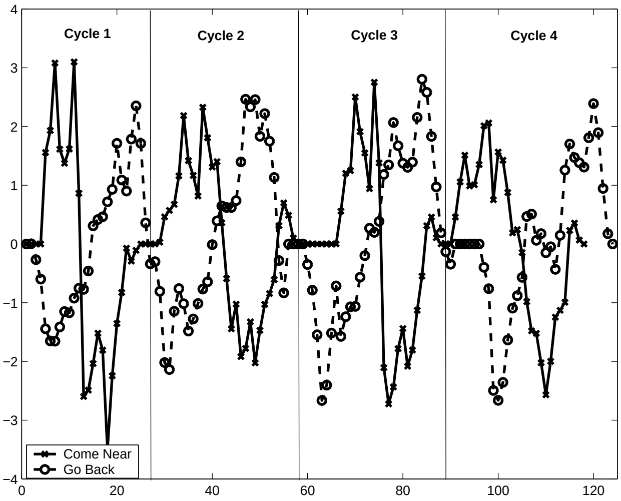


Figure 3. Plots of m_1 (vertical component of motion parameter) computed in the center window for four cycles of Come-Near and Go-Back gestures.

Given the recovered affine motion parameters, **a**, corresponding to the arm motion, the motion can be described in terms of another set of parameters **m** with geometric interpretations. We use five parameters to describe the motion [13, 5] : *horizontal* and *vertical* translations (a_o, a_3), *divergence* representing change in scale, *curl* representing change in orientation, and pure shear or *deformation* representing distortion (squashing and stretching in two perpendicular directions with the area unchanged)[1]. These parameters can be described in terms of the affine parameters as:

$$\begin{array}{lll} \text{horizontal} & = m_o = & a_o \\ \text{vertical} & = m_1 = & a_3 \\ \text{divergence} & = m_2 = & a_1 + a_5 \\ \text{curl} & = m_3 = & -(a_2 - a_4) \\ \text{deformation magn.} & = m_4 = & \sqrt{(a_1 - a_5)^2 + (a_2 - a_4)^2} \end{array} \quad (7)$$

Figure 3 shows plots for parameter m_1 obtained for the center motion window, for four cycles of Come-Near and Go-Back gestures. As can be seen, motion parameters serve to disambiguate gestures that a solely pose based system might confuse.

4.2. Motion Likelihood

The motion is parameterized using a fifteen dimensional vector consisting of the five estimated motion parameters

[1] deformation is a two dimensional vector described by its magnitude and the orientation of the axis of expansion with horizontal projection $a_1 - a_5$ and vertical projection $a2 + a4$

from each of the three motion windows. This motion parameterization represents the motion observation $\mathbf{z}_t^m$. The training data is used to obtain representative motion exemplars $\mathbf{X}^m$ for each gesture. Both the motion exemplars and the motion observations have the same representation using the fifteen dimensional space described above and, therefore, the distance function can be defined as $\rho^m(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T \Lambda^{-1} (\mathbf{x}_i - \mathbf{x}_j)$ with diagonal scale matrix Λ to scale each motion parameter.

Given a motion observation $\mathbf{z}_t^m$, the observation likelihood $P(\mathbf{z}_t^m | X_t^m)$ given the motion exemplar X_t^m at time t is estimated using a Gaussian PDF centered around each motion exemplars with a diagonal covariance matrix Λ, i.e.,

$$P(\mathbf{z}_t^m | X_t^m = \mathbf{x}_k^m) = \frac{1}{\sqrt{2\pi}|\Lambda|} e^{(\mathbf{z}_t^m - \mathbf{x}_k^m)^T \Lambda^{-1} (\mathbf{z}_t^m - \mathbf{x}_k^m)}$$

5. Shape Observation Model

The shape exemplars are sequences of body contours representing each gesture. These contours are used to compute $P(\mathbf{z}_t^s | X_t^s = \mathbf{x}_k^s)$. We use a probabilistic form of Chamfer matching [12] to compute this term. Further details for obtaining shape observation likelihood can be found in [8]. Figure 4 shows registered poses for some exemplars.

6. Experimental Evaluation

For the training, shape exemplars and motion exemplars were obtained from training sequences. We trained the HMMs using the scheme described in section 3.2. Figure 4 shows some pose classification results for different people in indoor and outdoor setups. The figures show the shape exemplar with the highest likelihood score overlaid over the original image. The *gesture segmentation* was performed by detecting a pause at the end of each gesture. Figure 5 shows how log likelihoods for each HMM changes as the observed Come-Near gesture progresses over time.

To evaluate the performance of the approach, an evaluation data set was obtained consisting of video sequences taken for five different subjects performing fourteen gestures . Each subject performed the gesture five times. That is, a total of $5 \times 5 \times 14 = 350$ sequences are used for the evaluation (25 for each gesture). The results were generated by following the leave-one-out paradigm: we trained the HMMs on 4 subjects and evaluated them on the 5^{th}. This was done for each of the 5 subjects and the results were combined into a single confusion matrix, shown in table 6.

As can be seen from the confusion matrix, the system gives us an accuracy of about 83.7% with a large number of gestures being classified correctly. The misclassifications are usually due to the similarity between some gestures in shape or in motion. The Turn-Left gesture, for instance, was classified once as the Stop-Left gesture in this case because the individual performing the Turn-Left gesture raised his hand well above the horizontal, therefore making his pose looks similar to the Stop-Left gesture. Most of the other misclassifications were also a result of considerable deviations from the generic guidelines given to the individuals on how to perform the gestures. We believe good user training can significantly better these results. These results can also be further improved by choosing the motion windows differently. e.g. by having two more windows above the shoulder, we could more efficiently discriminate between above shoulder and below shoulder gestures. This is pending further experimentation.

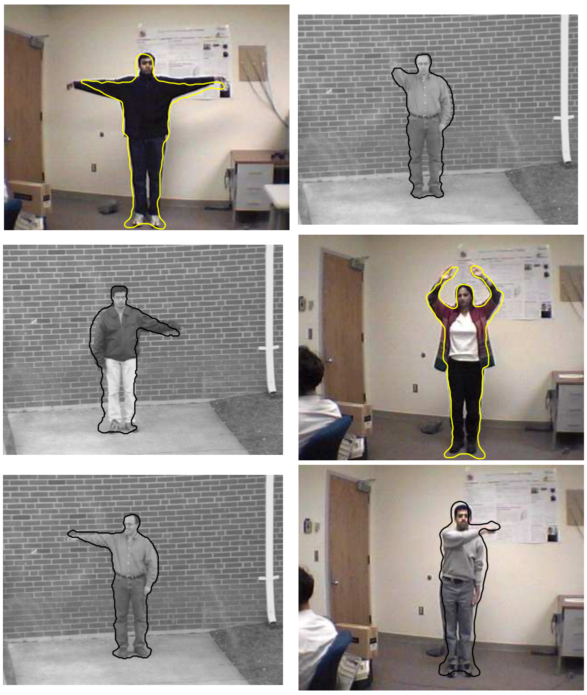


Figure 4. Pose matching results

Gestures such as Come-Near, Go-Away, and Close-Distance are performed entirely in front of the chest and do not stick out of the profile like the other gestures. This makes the pose (which is merely a contour of the body shape) a poor discriminator for classification. Motion however provides the necessary cues to discriminate between these three gestures as is evident from the high values for these gestures in the matrix.

In order to evaluate the additional information being provided by the motion features, we classified the 14 gestures using only pose information. The average classification accuracy observed in this case was about 23%. The low recognition rate can be attributed to the following reasons:

- Shape information by itself is not discriminative

	Turn-Left	Turn-Right	Flap	Stop-Left	Stop-Right	Stop-Both	Attention-Left	Attention-Right	Attention-Both	Start Engines	Speed Up	Come Near	Go Back	Close Distance
Turn-Left	**24**	0	0	1	0	0	0	0	0	0	0	0	0	0
Turn-Right	0	**21**	0	0	4	0	0	0	0	0	0	0	0	0
Flap	0	0	**22**	0	0	3	0	0	0	0	0	0	0	0
Stop-Left	1	0	0	**24**	0	0	0	0	0	0	0	0	0	0
Stop-Right	0	0	0	0	**25**	0	0	0	0	0	0	0	0	0
Stop-Both	0	0	0	0	0	**22**	0	0	0	0	0	3	0	0
Attention-Left	0	0	0	0	0	0	**18**	0	0	0	0	0	7	0
Attention-Right	0	0	0	0	0	0	0	**17**	0	0	0	0	8	0
Attention-Both	0	0	0	0	0	0	0	0	**16**	0	0	1	0	8
Start Engines	0	0	0	8	0	0	0	0	0	**17**	0	0	0	0
Speed Up	0	0	0	5	0	0	0	0	0	0	**19**	0	1	0
Come Near	0	0	0	0	0	2	0	0	0	0	0	**19**	4	0
Go Back	0	0	0	0	0	0	0	0	0	0	0	0	**25**	0
Close Distance	0	0	0	0	0	0	0	0	0	0	0	0	1	**24**

Table 1. Confusion Matrix

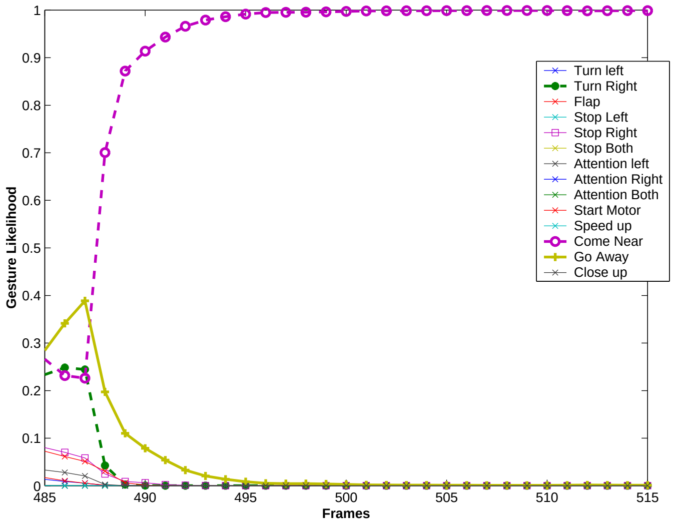


Figure 5. Log Likelihood for Come-Near gesture.

enough for gestures where the hand motion occurs in front of the torso. Motion information on the other hand serves well to discriminate these gestures.

- The Chamfer distance scheme is limited as it only allows us to match the exemplar contours to the image but not vice versa, thus potentially ignoring some discriminative foreground edges. The motion parameters help in capturing the overall movement of the subject and thus remove some of the ambiguity in the pose matching.

However, motion parameters by themselves may not be sufficient to disambiguate between gestures. As the motion parameters are extracted over windows in the image, they cannot distinguish between gestures that involve similar motions in different areas within the same region. Decreasing the window size and increasing the number of windows is not a solution as this will increase the number of parameters to be learned by the HMMs, making training difficult. Use of a multi-cue framework serves to alleviate this problem by falling back on one cue when the other fails to discriminate and vice versa, thereby boosting overall recognition scores. Further enhancement may be possible by weighing the contributions of each cue for each gesture.

7. Summary

This paper presented a multi-cue, exemplar-based non-parametric approach for gesture recognition. The key contribution of this paper was the extension of the nonparametric exemplar density estimation approach [8] to handle multiple cues thereby enabling the expansion of the system's gesture vocabulary. Using nonparametric exemplar density estimation, helps us to learn the dynamics from large exemplar spaces, which is not feasible with conventional HMMs. Using motion as a second cue not only lets us discriminate further among various gestures, it also allows us to classify gestures that cannot be characterized solely based on their contour information thus boosting overall recognition scores.

References

[1] S. Birchfield. Elliptical head tracking using intensity gradients and color histograms. In *IEEE Conference on Computer Vision and Pattern Recognition*, Jun 1998.

[2] M. J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. In *Computer Vision and Image Understanding, CVIU*, volume 63(1), page 75104, 1996.

[3] 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.

[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] R. Cipolla and A. Blake. Surface orientation and time to contact from image divergence and deformation. In *ECCV92*, pages 187–202, 1992.

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

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

[8] A. Elgammal, V. Shet, Y. Yacoob, and L. Davis. Learning dynamics for exemplar-based gesture recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 571–578, 2003.

[9] 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.

[10] 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.

[11] D. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–98, Jan 1999.

[12] D. Gavrila and V. Philomin. Real-time object detection for "smart" vehicles. In *ICCV99*, pages 87–93, 1999.

[13] J. Koenderink and A. van Doorn. Invariant properties of the motion parallax field due to the movement of rigid bodies relative to an observer. *Optica Acta*, 22(9):773–791, 1975.

[14] T. B. Moeslund and E. Granum. A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding: CVIU*, 81(3):231–268, 2001.

[15] 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.

[16] V. Pavlovic, R. Sharma, and T. S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):677–695, 1997.

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

[18] 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.

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

[20] 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.

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

[22] US-ARMY. Visual signals, field manual fm 21-60, 30 1987.

[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), 1999.