Clustering Facial Displays in Context

Jesse Hoey
Department of Computer Science
University of British Columbia
Vancouver, Canada, V6T 1Z4
jhoey@cs.ubc.ca

TR-01-17

November 14, 2001

Abstract

A computer user's facial displays will be context dependent, especially in the presence of an embodied agent. Furthermore, each interactant will use their face in different ways, for different purposes. These two hypotheses motivate a method for clustering patterns of motion in the human face. Facial motion is described using optical flow over the entire face, projected to the complete orthogonal basis of Zernike polynomials. A context-dependent mixture of hidden Markov models (cmHMM) clusters the resulting temporal sequences of feature vectors into facial display classes. We apply the clustering technique to sequences of continuous video, in which a single face is tracked and spatially segmented. We discuss the classes of patterns uncovered for a number of subjects.

1 Introduction

Recently, the notion that the primary function of facial displays is the expression of emotion has been challenged by psychologists, who have proposed a model of human facial displays as signals of social intent (the *Behavioral Ecology View*) [14]. They take the position that humans use their face as a way of communicating *through the medium of a social context*. For example, the reasons facial displays are used in normal conversations include both semantic and syntactic support of what the speaker is saying, as well as reactions in the listeners face to offer support of continuation of the dialog [8]. Human computer interaction researchers have also moved towards the ecological view, and have focused on interpreting the human face as a signaling mechanism [7]. That the same conclusions apply to both groups of researchers is not surprising given the media equation [23]: the principles which apply to inter-human communication should apply to human-computer communication.

Analyzing facial displays as communicative mechanisms has important ramifications for the design of facial display recognition systems. Which facial displays are observed will depend on the momentary context in which the display is shown [14]. Furthermore, the context gives meaning to the display. Context is defined as all circumstances relevant to the display, and may include concurrent or proximate speech and gestures of observer and performer, as well as other environmental factors. For example, eyebrows are sometimes raised during conversation when the speaker is thinking or remembering. However, eyebrows are also raised in backchannel displays of acknowledgment, or when an individual is taking turn in a conversation [7]. In any case, the observed facial display carries little meaning by itself, while the combination of the current context and the observed facial display is meaningful.

The particular configurations and motions observed in facial displays are individual dependent [8]. As gestures, the individuality of displays are governed in part by cultural, educational, situational and physical [1] factors [28]. Furthermore, although people may use similar facial displays, they use them in dissimilar situations. This implies that a recognition system will have to be able to *adapt* to the ways in which a particular human is using their face.

This paper presents a framework for adaptive and context dependent recognition of human facial displays. We present a recognition system which observes an active computer user's face, and learns salient display patterns which correlate with the current context. The particulars of what the human is doing is unimportant. For example, they could be playing a computer game, querying a database, or taking a lesson. However, the inclusion of an embodied face is important, in order to open a facial display communication channel (to give the human a reason to use their face).

> facial displays are more likely to be emitted when there is an available receiver, when they are useful in conveying the particular information, and when that information is pertinent or appropriate for the social interaction.([9], p.329)

In most human-computer interaction today, there is no incentive for the human to use their face because there is no appropriate reaction from the computer. Exceptions are systems such as Gandalf [26], and Rea [1], which react to a small, pre-defined, set of user hand gestures, head motions, and facial displays. Our work differs in that we make no assumptions about the facial displays a particular human will be using. We wish to examine only what patterns can be extracted without prior information about facial displays, in order to uncover techniques which can be used for adaptation of a facial display recognition system. Our Bayesian approach can be incorporated with prior knowledge of cross-individual statistics once we have analyzed the adaptive likelihood functions.

While researchers have examined unsupervised clustering of human gesture sequences [29, 27], and video sequences in general [11, 21], to our knowledge no work has focused on facial displays. Wren *et. al* [29] investigate understanding purposeful human gestures using a combination of a kinematic model of human motion, a mixture of hidden Markov models and a high-level classifier, similar to our approach. Related work [11] used the same techniques to cluster video from an ambulatory camera. The umbrella under which these different approaches fall is the general content-based video indexing problem, which will not be reviewed here.

In our approach, the human face in a color video sequence is tracked and facial motion is described using optical flow over the entire face, projected to the complete orthogonal basis of Zernike polynomials. This *a priori* basis is useful for adaptive recognition, because it efficiently describes the facial displays without any commitment to particular motions. The trajectories of the resulting feature vector describe the observed facial displays, and are modeled using a context dependent mixture of hidden Markov models (cmHMM). The cmHMM is a mixture of hidden Markov models [25] augmented with an (observed) context which conditions the mixture model. The cmHMM models the trajectories of feature vectors, simultaneously clustering them into facial display classes and modeling their dependence on the context.

Section 2 shows how we obtain a sequence of feature vectors representative of motion in the face from continuous video input. Section 3 presents the mixture of hidden Markov models as a clustering technique for temporal patterns of the extracted feature vectors, and shows how to incorporate context. We show how to use the EM algorithm to estimate the parameters and how to robustly initialize the model. Section 4 presents clustering results from a number of different subjects engaged in a simple facial display imitation task.

[1] facial structure, makeup, beards, glasses, jewelry, etc.

2 Facial motion representation

Extracting a meaningful and simple feature vector representing the human face and the motion thereof can be approached in many ways. In this work, we focus only on the motion of the face, and use a holistic representation over the entire facial region. We estimate optical flow between successive images, and project this flow field over a tracked facial region to the complete orthogonal basis of Zernike polynomials, yielding a feature vector, $\mathbf{Z} = Z_0 ... Z_T$. The track is updated using both the recovered optical flow and an independent estimate of the face region using skin color segmentation. The Zernike representation of the motion in the human face is simple, efficient and requires no training [18]. These characteristics make it a desirable approach for unsupervised classification of patterns of motion in the face. It is a generalization of the method of [3], in which optical flows over the whole face as well as over eye and mouth regions were projected to the three lowest orders of Zernike polynomials. Projections over eye and mouth regions allow for lower dimensional representations within each region, but require additional tracking and spatial segmentation. In this work, we only look at projections over the entire face. The Zernike representation differs from approaches such as Eigen-analysis [20], or facial action unit recognition [19] in that it makes no commitment to a particular type of motion, leading to a transportable classification system (e.g. usable for gesture clustering). Although the the recognition of facial action units [19] gives the ability to discriminate between very subtle differences in facial motion, it requires extensive training and domain specific knowledge. We prefer to use a representation which can be extended from simple to complex given the task to be accomplished. For example, a system which only needs to recognize nods of the head could use only the first order ZPs. More complex recognition tasks need only add as much representation as is necessary to distinguish the important facial displays of a particular user.

2.1 Optical Flow

We estimate optical flow between a successive pair of frames $\{I_t, I_{t+1}\}$ in a video sequence using the robust gradient-based regularization method of [2]. This method yields estimates of flow which are smooth over patches in the human face, preserving important discontinuities but removing high spatial frequency noise arising from violations of the brightness constancy assumption. After the flow is computed, the centroid and scale of the area of interest are estimated and a feature vector is obtained by projecting the flow onto the basis of Zernike polynomials.

2.2 Zernike Projection

Zernike polynomials are an orthogonal set of complex polynomials defined on the unit disk [22]. The lowest two orders of Zernike polynomials correspond to the standard affine basis. The next order polynomials correspond to extensions of the affine basis, roughly *yaw*, *pitch* and *roll*, as explored in [3]. Higher orders represent motions with higher spatial frequencies. The basis is orthogonal over the unit disk, such that each order can be used as an independent characterization of the flow, and each flow field has a unique decomposition in the basis. Zernike polynomials are expressed in polar coordinates as a radial function, $R_n^m(\rho)$, modulated by a complex exponential in the angle, ϕ, as follows:

$$U_n^m(\rho, \phi) = R_n^m(\rho) e^{im\phi} \tag{1}$$

with radial function,$R_n^m(\rho)$, given by

$$R_n^m(\rho) = \sum_{l=0}^{(n-|m|)/2} \frac{(-1)^l (n-l)!}{l![\frac{1}{2}(n+|m|)-l]![\frac{1}{2}(n-|m|)-l]!} \rho^{n-2l}$$

for n and m integers with $n \geq |m| \geq 0$ and $n - m$ even. The first few radial basis functions are therefore:

$$\begin{array}{ccc} R_0^0 = 1 & R_1^1 = \rho & R_2^0 = \rho^2 \\ R_2^2 = 2\rho^2 - 1 & R_3^1 = 3\rho^3 - 2\rho & R_3^3 = \rho^3 \end{array}$$

The Zernike polynomials are orthogonal on the unit disk. and obey the following orthogonality relation:

$$\int_0^1 \int_0^{2\pi} U_n^m(\rho, \phi) U_{n'}^{m'}(\rho, \phi) \rho \, d\phi \, d\rho = \frac{\pi}{(n+1)} \delta_{nn'} \delta_{mm'}, \tag{2}$$

where $\delta_{nn'} = 1$ if $n = n'$, and 0 otherwise.

The orthogonality of the basis allows the decomposition of an arbitrary function on the unit disk, $F(\rho, \phi)$, in terms of a unique combination of odd and even Zernike polynomials. That is, [22]

$$F(\rho, \phi) \approx \sum_{m=0}^{M} \sum_{n=m}^{N} \left[A_n^m cos(m\phi) + B_n^m sin(m\phi) \right] R_n^m(\rho), \tag{3}$$

The coefficients, A_n^m and B_n^m, of the decomposition of the horizontal and vertical flow estimates, $u(x,y)$ and $v(x,y)$, over the tracked facial region are thus obtained using:

$$\begin{matrix} {}^uA_n^m \\ {}^uB_n^m \end{matrix} = \frac{\epsilon_m(n+1)}{\pi} \sum_x \sum_y u(x,y) R_n^m(\rho) \begin{matrix} cos(m\phi) \\ sin(m\phi) \end{matrix} \tag{4}$$

where $\phi = \arctan(y'/x')$, $\rho = \sqrt{x'^2 + y'^2} \leq 1$, $x' = (x - x_0)/r_x$, $y' = (y - y_0)/r_y$, and $\{x_0, y_0\}$ and $\{r_x, r_y\}$ are the centroid and scales of the region of interest. The flows can be reconstructed from the coefficients using Equation 3. Feature vectors are sets of the coefficients from Equation 4. The choice of a particular set to represent the flow will depend on the types of flows being modeled [18]. This choice is currently made by the modeler, by removing the $n = 0$ component (translation) and then adding as many orders as can be supported by the data in the modeling process.

2.3 Tracking

The tracking problem is to update the facial region as described by centroid and scale parameters $c = \{x_c, y_c, r_x, r_y\}$ from one frame to the next. We assume that there is only one head present in all frames. We get a first estimate from the first and second order coefficients of the projected flow:

$$x_c' = x_c + {}^uA_0^0 \quad y_c' = y_c + {}^vA_0^0$$

$$r_x' = r_x + {}^uA_1^1 \quad r_y' = r_y + {}^vB_1^1$$

Updates using the only the flow are prone to significant drift over any sequence longer than roughly 600 frames (20 seconds). Furthermore, severe permanent tracker failure can be caused by *adaptors*, such as scratching the face. Therefore, we derive a correction term using skin color segmentation. We transform the RGB color images to HSV space, and segment using simple thresholding in hue and saturation. Median filtering removes noisy estimates, and the resulting binary image is projected along horizontal and vertical directions. The region of interest is then estimated by examining where the projected distributions fall below a threshold. The centroid and scale are then updated using a weighted sum of the skin and flow estimates, where the weights are given by the relative errors. The scale and centroid for the initial frame of a sequence is given by the skin segmentation procedure alone. The top row in Figure 1 shows an example track using only updates based on optical flow. The tracker performs well until errors are introduced by the tracked individual's hand, from which the tracker cannot recover. Correction using the skin segmentation (shown in the middle row in Figure 1) yields the track shown in the bottom row in Figure 1.

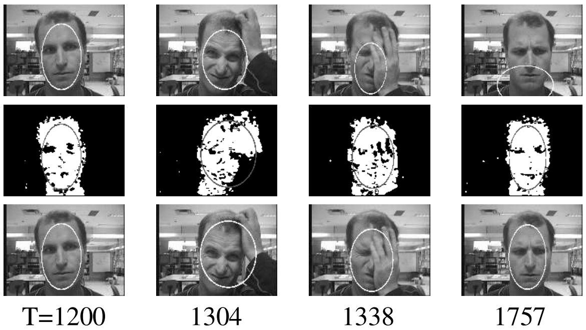


Figure 1: Tracking using only flow (top row). Skin segmentation (middle row) corrections yield corrected track (bottom row).

3 Clustering motion patterns

Once we have recovered a sequence of feature vectors $\mathbf{Z} = Z_0...Z_T$ we wish to find and classify the trajectories corresponding to salient facial displays. Simultaneously, we want to model the dependence on the current context. That is, we want to model the relationship between what is happening in a computer user's face, and what is happening on the computer screen they are using, what an embodied agent's face is doing, the agent or user's speech, or other environmental factors. In the simplest case, the user's facial displays will depend only on an embodied agent's facial displays (as it is in Section 4). That is, the communication channel is purely through facial displays, and involves no other factors.

In the following we will refer to the time indices on the input feature vectors, Z_t, as *frame time*, as they correspond exactly to video frames. *Frame time* is related to clock time by a nearly constant multiplicative factor. We assume the context is given at frame time t by a single observable discrete variable C_τ, from a temporal sequence $\mathbf{C} = C_0...C_\mathcal{T}$, where the temporal indices τ refer to event times. This variable encapsulates all information about the state of the human-computer system *other* than that coming from the user's face. It may include concurrent or proximate speech and gestures of both user and agent, as well as other environmental factors. In general, some components of C_τ will not be directly observed, but will be distributions over high-level descriptions of user states inferred from other modalities such as speech. We only consider the discrete observable case here.

The human user's facial display at frame time t is given by a hidden variable $D_{\tau'}$, which is a temporally abstract description of some subset of the input sequence, $\mathbf{Z}$, and comes from a sequence $\mathbf{D} = D_0...D_{\mathcal{T}'}$. Again, the temporal indices τ' refer to event times, which may not be the same as the event times of the context states, C. Our goal is to model the joint probability distribution of $\mathbf{Z}$ and $\mathbf{C}$, and we are hypothesising an intermediate set of (hidden) facial display states D:

$$P(\mathbf{ZC}) = \int_\mathbf{D} P(\mathbf{ZDC}).$$

The intermediate states, D, are important because facial displays do not occur at *frame time*. A typical display (such as raising the eyebrows) takes a few hundred milliseconds, or about 10 frames. Therefore, we want to extract a high level description of the display, instead of working directly with the individual frames. We are trying to model the dependencies between the current context C and the user's facial displays, which are observed as sequences of feature vectors. Modeling such dependencies directly would be problematic. However, we can model the context C as generating some user (re)-action, D, which itself generates the observed feature vectors Z. That is, we assume that $P(\mathbf{ZDC}) = P(\mathbf{Z}|\mathbf{D})P(\mathbf{D}|\mathbf{C})$: the input sequence $\mathbf{Z}$ is independent of the context sequence, $\mathbf{C}$, given the facial display sequence $\mathbf{D}$, as shown in the Bayesian network in Figure 2.

Although there may be complex temporal dependencies in the $\mathbf{C}$ and $\mathbf{D}$ sequences in general, we make a first order Markov assumption, such that $P(C_\tau|C_0...C_\mathcal{T}) = P(C_\tau|C_{\tau-1})$, and $P(D_{\tau'}|D_0...D_{\mathcal{T}'}) = P(D_{\tau'}|D_{\tau'-1})$. Stochastic context free grammars [4], or variable length Markov models [16], are other options. We now assume that each facial display, D, *generates* one subsequence $\mathbf{Z} = Z_i...Z_j$ of the the complete input sequence, $\mathbf{Z} = Z_0...Z_T$, and that each subsequence is generated by only one display state, D. This is a simple independence assumption, but will not model any effects similar to co-articulation in speech. It also implies some method for temporally segmenting the input sequence. Previous authors have approached this temporal segmentation problem by exhaustive search for the most likely time scale [29, 11, 28], or by searching for discontinuities in the temporal trajectories [27, 24] over a range of scales. The complete Bayesian solution involves maximizing over temporal segmentations [13]. Since we are searching for patterns which correspond with context states, we use only the occurrence of these contextual changes to segment the input video. That is, if the context states $\mathbf{C} = C_0...C_\mathcal{T}$ occur at frame times $\mathbf{g} = g_0...g_\mathcal{T}$, then

$$P(Z_0...Z_T|D_0...D_{\mathcal{T}'}) = \prod_{\tau=0}^{\mathcal{T}'} P(Z_{g_\tau}...Z_{g_{\tau+1}-1}|D_{\tau'})$$

This assumption implies that the sequences of contexts, C, and displays, D, are synchronous, which will not be true in general, but is sufficient for our purposes here. Finally, we assume that each facial display, D_τ, only depends on the current context, C_τ. The interaction of the C and D chains can be modeled in many ways [5], including dependencies of C on D (the agent reacts to the user). The independence assumptions just discussed are summarized in Figure 2. In the work we present here, we further assume temporal independence in the $\mathbf{C}$ and $\mathbf{D}$ chains. This simplifying assumption will allow us to focus only on the dependence $P(D|C)$ and is valid for the experiments discussed in Section 4. We have investigated the temporal dependencies in the facial display sequence in a related paper [17], which we plan to integrate with the methods presented here.

The learned distribution $P(D|C)$ (or $P(D_\tau|D_{\tau-1}C_\tau)$ in general) can represent a number of things. Most notably, if the context C represents the actions of a computational agent, then it indicates the effects of the agents actions on the user, and can be used to derive a policy of action to achieve some goal.

Our task is now to cluster the sequences of feature vectors, $\mathbf{S} = S_0...S_\mathcal{T}$, where $S_\tau = Z_{g_\tau}...Z_{g_{\tau+1}-1}$. That is, we wish to uncover K sets of these sequences, $D_j = S_{\tau_0}...S_{\tau_{N_j}} j = 1...K$. We use a mixture of hidden Markov models conditioned on the context variable C. The following describes this model, show how to learn the parameters using the expectation-maximization (EM) algorithm, and describes the initialization procedure we have used. We assume throughout that the number of classes, K, is known.

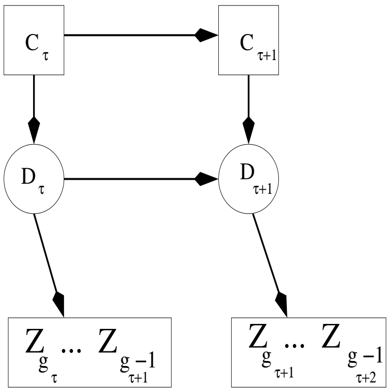


Figure 2: Independence assumptions between context, C, facial displays, D, and feature vectors, Z. Variables in squares are observed, while those in circles are hidden.

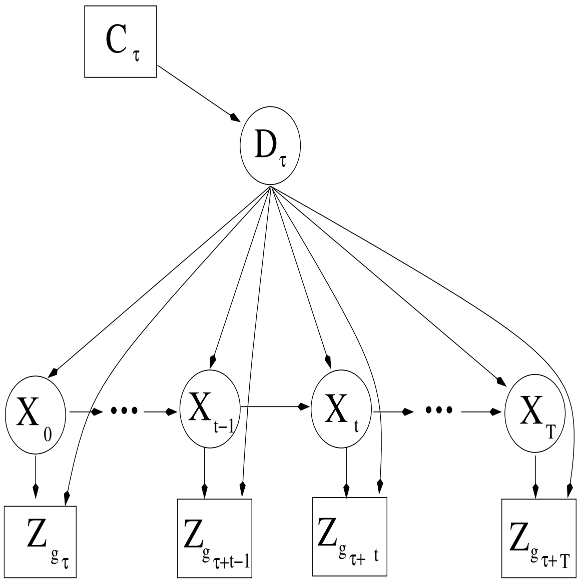


Figure 3: Mixture of hidden Markov models as a dynamic Bayesian network, including context variable, C.

3.1 Mixtures of HMMs

Figure 3 shows a time slice of the context dependent mixture of hidden Markov models (cmHMM) as a dynamic Bayesian network. The description of the entire video sequence (as given by $\mathbf{Z} = Z_0 ... Z_T$) would be a series of such models, each conditioned on an (observable) context variable $\mathbf{C} = C_0 ... C_\tau$. As a generative model of the computer user, we can describe Figure 3 as follows. The context C prompts the computer user to perform facial display D. The state of D thus generated is a high level description of the facial display to be performed (such as smile or raise eyebrows, but not labeled as such in the model), which itself generates a sequence of $T = g_{\tau+1} - g_\tau$ discrete states $\mathbf{X} = X_0 ... X_T$ occurring at frame time. A state X_t generates a specific optical flow description Z_t from a Gaussian distribution over the space of such descriptions. The model is described by three conditional probability distributions:

1. Initialization: $P(X_0|D)$
2. Transition: $P(X_t|X_{t-1}D)$ fully connected.
3. Observation: $P(Z_t|X_tD)$ is parametrized by full covariance Gaussian distributions.
4. Context: $P(D_\tau|C_\tau)$.

If we remove the context, C, and the conditioning link, $P(D|C)$, we recover a mixture of hidden Markov models as described in [25]. The context variable, C, can be seen as an input to the generative model, which will bias clustering of the input sequences according to the context states, C [2]. This distribution, $P(D|C)$, will be learned simultaneously with the other parameters in the model.

Given a set of (temporally segmented) feature vectors and a set of context variables, $\mathbf{C} = C_0...C_\tau$, we wish to learn the maximum likelihood parameters of the cmHMM. That is, we want to find the model parameters, Θ, which maximize the probability

$$P(\mathbf{ZC}\Theta) = \int_{\mathbf{XD}} P(\mathbf{ZXDC}\Theta).$$

We can use the expectation maximization algorithm [12], which lower bounds this probability distribution with a function $q(\mathbf{XD})$, maximizes this bound at the current estimate of the model parameters, Θ', (the 'E' step) giving $q(\mathbf{XD}) = P(\mathbf{XD}|\mathbf{ZC})$ [3], and then maximizes the bound

$$\int_{\mathbf{XD}} P(\mathbf{XD}|\mathbf{ZC}) \log P(\mathbf{ZXDC}) \tag{5}$$

over the model parameters, Θ, (the 'M' step). We factor $P(\mathbf{ZXDC}) = P(\mathbf{ZXD}|\mathbf{C})P(\mathbf{C})$ and the integral in Equation 5 becomes:

$$\int_{\mathbf{XD}} P(\mathbf{XD}|\mathbf{ZC}) \log P(\mathbf{ZXD}|\mathbf{C}) + \int_{\mathbf{D}} P(\mathbf{D}|\mathbf{ZC}) \log P(\mathbf{C}). \tag{6}$$

The first term in Equation 6 is the term that is maximized for a mixture of hidden Markov models, conditioned on the (observed) variable C. Smyth [25] has pointed out that this can be achieved by clustering the variables along each link from D to X variables. The result is a simple hidden Markov model, with hidden states given by the joint variable $\{X, D\}$. The constraint that the variable D does not change over the course of the sequence can be enforced by initializing the the transition matrix for the mixture model, A, as a block diagonal matrix where the blocks are the transition matrices, $A_i = P(X_t|X_{t-1}D_i)$, for each state, i, of the cluster variable, D.

$$A = \begin{pmatrix} A1 & 0 & ... & 0 \\ 0 & A2 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & A_K \end{pmatrix} \tag{7}$$

The initial state probabilities, $P(X_0|D)$ are then chosen to reflect the weights of the mixture components as given by the conditional distribution $P(D|C)$.

Maximizing the second term in Equation 6 updates the parametrized probability distribution $\Theta_{Dij} = P(D = i|C = j)$ by counting the expected number of times the cluster variable $D = i$ when the context variable $C = j$, N_{Dij}. That is,

$$\Theta_{Dij} = \frac{E_{P(D|\mathbf{ZC})} N_{Dij}}{\sum_i E_{P(D|\mathbf{ZC})} N_{Dij}}$$

where

$$E_{P(D|ZC)} N_{Dij} = \sum_\tau P(D_\tau = i|\mathbf{ZC}) \delta(C_\tau = j).$$

The probability distribution over D given $\mathbf{Z}$ and $\mathbf{C}$ is obtained from the expectation step described above.

3.2 Initialization

The EM algorithm performs hill climbing on the likelihood surface, and therefore is dependent on the initial choice of the parameters. We use the clustering in log-likelihood space technique described in [25]. It involves fitting a simple HMM to each individual sequence, evaluating the log-likelihood of each sequence given every simple HMM, and then clustering the sequences into K groups using the log-likelihood distance matrix. We use agglomerative clustering with complete linkage (furthest neighbors merging). Simple HMMs are then fit to each of these K clusters, and the results are used to initialize the matrix for the cmHMM. The conditional probabilities of cluster membership, D, given the context variables, C, are initialized by counting the number of observed states C in each cluster. Since all the HMMs we are training have output distributions characterized by Gaussians in the Zernike feature vector space, they require sufficient data to avoid singularities in the covariance matrices. Fitting a HMM to an individual sequence sometimes does not work, especially if the Zernike vectors have too little variation along some dimension. We took two approaches to solve this problem. If there were few

[2] Addition of an *entropic* prior [6] on the distribution $P(D|C)$ would favor clustering of sequences towards correspondences with the context states C through $P(D|C)$.

[3] We omit explicit representation of Θ, for notational ease

singular sequences, they were removed from the clustering procedure, and then inserted into the cluster to which they were closest. If there were too many singularities, the dimension of the feature vectors was reduced for the initial clustering criterion. Once the clusters were formed, the feature vectors were re-set to their original dimensionality.

4 Results

To evaluate the model presented in the last section, we asked volunteers to perform a simple facial expression imitation task. They were seated in front of a computer terminal on which an animated cartoon shows facial displays. While many face generation systems use complex 3D graphics, this face is a simple cartoon. This allows for fast rendering, and does not detract from interaction quality, since humans will interact with even the simplest of generated faces as a real human face [23]. Cartoon displays start from a neutral face, as shown in Figure 4(a), then warp to one of the 4 poses shown in Figures 4 for values of $C = 1...4$. These cartoon displays will be referred to as $C_1...C_4$ in the following. The pose is held for roughly a second, and the face then warps back to the neutral pose where it remains for an additional second. Although these displays may be reminiscent of so-called *prototypical facial expressions*, the displays they elicited were clearly *not* expressions of emotion, but only reactions D to contexts C. That they may have been *interpreted* as emotional displays by the subjects is not relevant here. The subjects were told that their task is to imitate these displays, and were shown each of displays initially


(a) C_1 C_2 C_3 C_4

Figure 4: (a) neutral face (C=1...4) Faces which subjects were told to imitate

and told to practice imitating them. Once they were satisfied with their imitations, they pressed a key, and the system began recording a video sequence through a Sony EVI-D30 color camera mounted above the computer screen. While the subjects were being recorded, the cartoon face performed a series of 40 randomly selected facial displays over a period of 2 minutes.

Zernike feature vectors were recovered for the recorded videos (3600 frames), using a selected basis (specifically ZP coefficients ${}^{u}A_1^1, {}^{v}B_1^1, {}^{u}A_2^2, {}^{v}B_2^2$). The videos were temporally segmented using the onset times of the cartoon displays and the resulting sequences were input to the HMM clustering and training algorithm described in Section 3, using 3 X states (facial displays are tri-phasic) and 4 clusters (the number of displays the subjects were trying to imitate). The Viterbi algorithm was used to assign cluster membership, D, to each sequence, which were then compared to the known classes of displays the subjects were trying to imitate. The exact model recovered is partially dependent on the randomness in the initialization procedure. However, we found the recovered cluster membership to be fairly consistent, an indication that our initialization procedure is robust to the random starting points. The results we present in the following section are usually the results we obtained on the first trial. Some, however, were given multiple trials and the most often re-occuring results are presented.

In a cross-validation study, we found that the modeling of the context information did not significantly improve the likelihoods of test data. The sequences were randomly split into 35 training and 5 test sequences 20 times, and cmHMMs and mHMMs were trained on the training data. The likelihoods of the test data were then evaluated. The results, averaged over the 20 trials and over the 4 subjects were 21.9 ± 2.0 and 22.1 ± 2.0 for the cmHMM and the mHMM, respectively. However, we do not expect any increases in likelihood, but we do expect that the recovered models will be structured so as to be more easily *interpretable*. That is, they will be correlated with the context states, C. Such correlations could be encouraged further with the inclusion of an entropic prior [6].

The following sections evaluate the results from all four subjects who performed the experiment. For each subject, we present the confusion matrix of context states, C, and recovered clusters, D. Then, we evaluate the clusters with respect to the original images and flow fields, and present a selection of sequences which are representative of the major clustering effects observed. After the experiment, most subjects reported either that they did not notice a significant difference between cartoon displays C_1 and C_2, or that they could not find a way to imitate the second one, C_2, due to the extremely down-turned mouth.

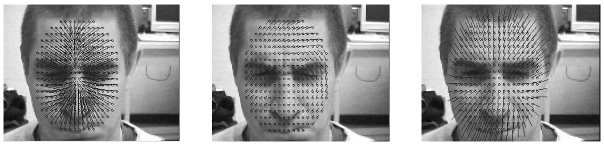
4.1 Subject $\mathcal{A}$

Table 1 shows the confusion matrix for subject $\mathcal{A}$, in which each row is one C state (facial display on screen) and each column is one recovered cluster, $D_1...D_4$.

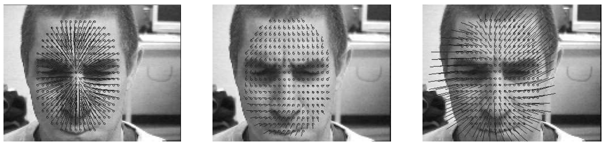
The cmHMM clustered most imitations of C_1 and C_2 together in cluster D_2. Key frames from two sequences in this cluster are shown in Figure 5. Scaled flow fields reconstructed from the feature vectors are shown superimposed. The top

	cluster			
	D_1	D_2	D_3	D_4
C_1	2	6	0	0
C_2	0	4	0	2
C_3	0	0	0	12
C_4	7	0	7	0

Table 1: Confusion matrix for subject $\mathcal{A}$



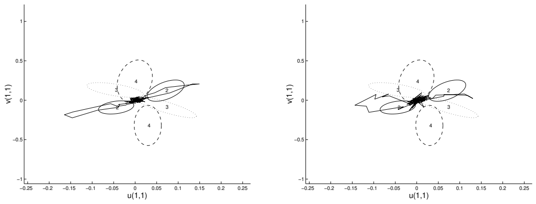
frame= 2807 2819 2862



frame= 2993 3006 3041

Figure 5: Example sequences for subject $\mathcal{A}$.

sequence was in response to a C_1 cartoon display, while the bottom one was in response to a C_2 cartoon display. Figure 6 shows the feature vector trajectories in 2 of the feature vector dimensions for these two sequences. It also shows three of the Gaussian covariances as level curves for the mixture models in the learned cmHMM. The Gaussians are labeled with their corresponding D values. The $D = 1$ model (not shown) has a large Gaussian which encompasses most of the others, and seems to be modeling all the sequences which do not fit well into any of the other three models. The remaining three models clearly partition the space evenly, describing vertical expansion and contraction (D=4), horizontal expansion and contraction (D=3), and a combination of both (D=2).



(a) (b)

Figure 6: Feature vector components ${}^uA_1^1$ and ${}^vB_1^1$ (a), (b) correspond to top and bottom rows in Figure 5

Imitations of C_3 were clustered together, with two imitations of C_2 in the same group These C_2 imitation sequences did not start in a neutral expression (the temporal segmentation procedure did not correspond). The C_4 imitations were split into two distinct groups (clusters D_1 and D_3), which differed in the amount of eyebrow motion present. Figure 7 shows two sequences which were responses to C_4, but which were clustered into separate groups. The top row sequence contains a significant eyebrow raise (at frame 108), while the bottom one does not. Figure 8 shows the feature vector trajectories for these two sequences. The difference is the additional trajectory in Figure 8(a) which extends into the positive u and v quadrant, and corresponds to the eyebrow raising. Clearly the model has uncovered that this is a more important difference in terms of flow fields than any differences between imitations of C_1 and C_2.

4.2 Subject $\mathcal{B}$

Consider now subject $\mathcal{B}$, whose confusion matrix is shown in Table 2. This subject attempted to differentiate between C_1 and C_2, while consistently performing displays C_3 and C_4. The C_2 imitations were split into two clusters: those that were similar to the C_1 imitations, and those that were not. The C_2 imitations grouped together in cluster D_3 contain almost no motion at all. Figure 9 shows key frames from three example sequences. The top row is a C_1 imitation in cluster D_1, in which the face contracts and then expands. The second row is a C_3 imitation in cluster D_2. The bottom row shows what was supposed to be a C_2 imitation which clustered with the C_3 imitations in cluster D_2. The reason for the apparent mis-classification was a

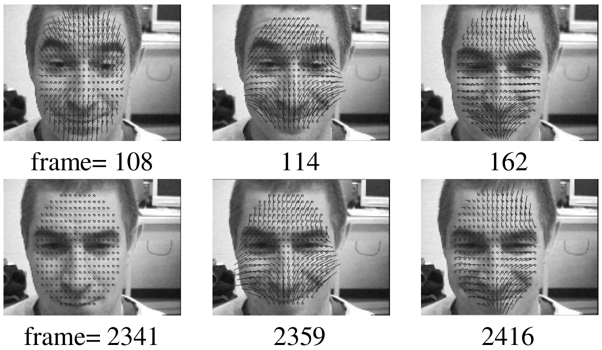


Figure 7: Example sequences for subject $\mathcal{A}$.

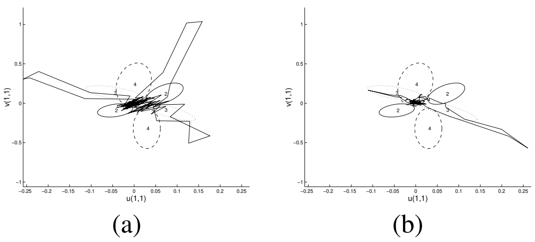


Figure 8: Feature vectors components ${}^{u}A_1^1$ and ${}^{v}B_1^1$ (a), (b) correspond to top and bottom rows in Figure 7

	cluster D_1	D_2	D_3	D_4
C_1	4	1	1	0
C_2	5	2	9	0
C_3	0	8	0	0
C_4	0	0	0	10

Table 2: Confusion matrix for subject $\mathcal{B}$

	cluster			
	D_1	D_2	D_3	D_4
C_1	10	0	0	1
C_2	0	0	1	5
C_3	0	15	0	0
C_4	0	0	8	0

Table 3: Confusion matrix for subject $\mathcal{C}$

natural *adaptor* which occurred (the subject bit her lip). The subject also closed her eyes (see frame 3405), and seems to have missed the C_2 cartoon display.

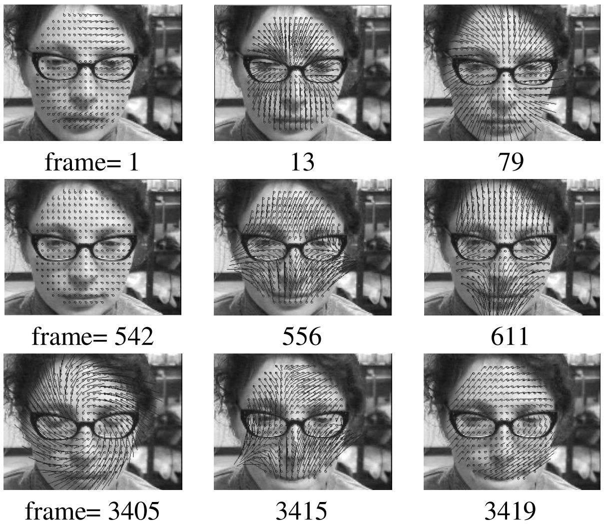


Figure 9: Example sequences for subject $\mathcal{B}$.

4.3 Subject $\mathcal{C}$

In contrast to other subjects, subject $\mathcal{C}$'s model was trained using only the ZP coefficients ${}^{u}A_1^1$ and ${}^{v}B_1^1$, which were found to be sufficient to separate the four clusters. Table 3 shows the confusion matrix for subject $\mathcal{C}$. This subject's imitiations of C_2 were clustered together in D_4. Only two sequences did not fit into the overall clustering, for reasons we will show below. First, consider representative sequences for each of the four clusters, shown in Figure 10. The corresponding feature vector trajectories are shown in Figure 11, along with Gaussian level curves for the four learned models. These Gaussians divide up the plane according to the significant motions in each display. The numbers in the Gaussian level curves are the cluster numbers.

The C_4 displays were all modeled best by cluster D_3. Frames at peak motions are indicated in Figure 11(a), which correspond to the frames shown in the top row in Figure 10. The motions at the peaks have u contracting while v is expanding (frame 2987), and vice-versa (frame 3042). The C_3 displays were modeled best by cluster D_2, as shown in Figure 11(b) and the second row in Figure 10. Here we see motion primarily in the vertical direction: expansion at frame 109 and contraction at frame 150. The C_1 displays were modeled by cluster D_1, as shown in Figure 11(c) and the third row in Figure 10. This model differs from the D_3 model in that u and v both contract together (frame 561) and then expand together (frame 601). The C_2 displays were modeled by cluster D_4, as shown in Figure 11(d) and the bottom row in Figure 10. These displays happened in four phases, rather than the usual three. An initial contraction in u and v (frame 1819) was immediately followed by a larger contraction in v (frame 1827), bringing the face to its apical pose. The ensuing expansion at frame 1871 is very much like the expansions seen in model D_3 (see Figure 11(a), and the top row in Figure 10).

Key frames from the two sequences which were not clustered with the other groups are shown in Figure 12. Corresponding feature vector trajectories are shown in Figure 13. The top row in Figure 12 shows an imitation of a C_2 display which was clustered with the C_4 imitations in cluster D_3. This display was not consistent with any other displays. It appears as though the user initially committed to a C_4 imitation, then changed his mind (somewhere around frame 3442), attempting to shift to a C_2 imitation. Comparison of the feature vector trajectory in Figure 13(a) with the trajectories in Figure 11(d) and (a) reveals

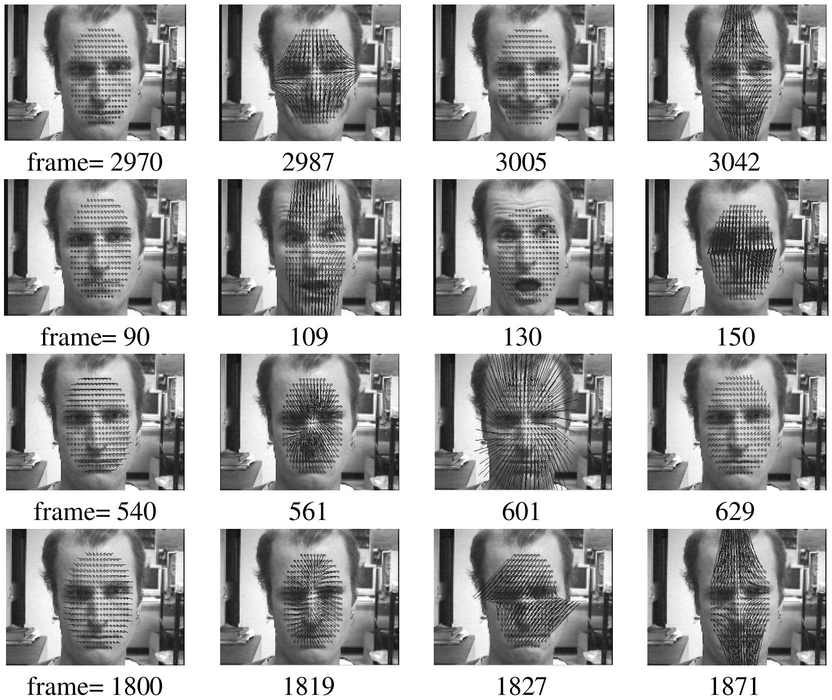


Figure 10: Top to bottom rows are representative sequences from clusters 3,2,1 and 4 respectively, for subject $\mathcal{C}$.

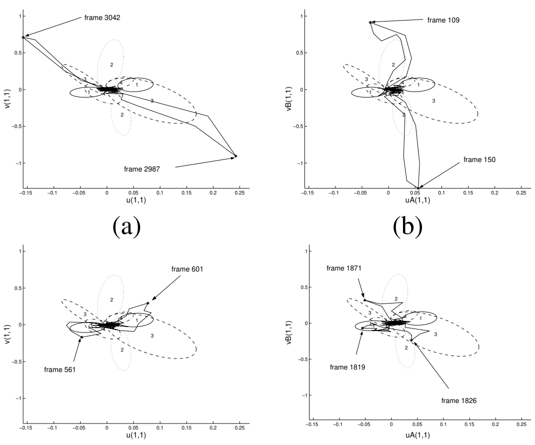


Figure 11: Feature vector components ${}^{u}A_1^1$ and ${}^{v}B_1^1$ (a), (b), (c), (d) correspond to rows in Figure 10 from top to bottom. Gaussian level curves are shown for the four learned models, and are numbered by D values.

	cluster			
	D_1	D_2	D_3	D_4
C_1	0	11	2	0
C_2	0	4	1	0
C_3	10	0	1	0
C_4	0	0	0	11

Table 4: Confusion matrix for subject $\mathcal{D}$

the same thing. Although the trajectory in Figure 13(a) is similar to the one in Figure 11(d) (same C value), it has a much larger contraction motion at frame 3442, which effectively makes it look more similar to the trajectories in Figure 11(a). The bottom row in Figure 12 shows an imitation of a C_1 display which was clustered with the C_2 imitations in cluster D_4. The data here is clearly in support of an imitation which is similar to all other C_2 imitations for this subject, and hence we can attribute the mis-classification to the subject wrongly interpreting the display C_1 as a C_2.

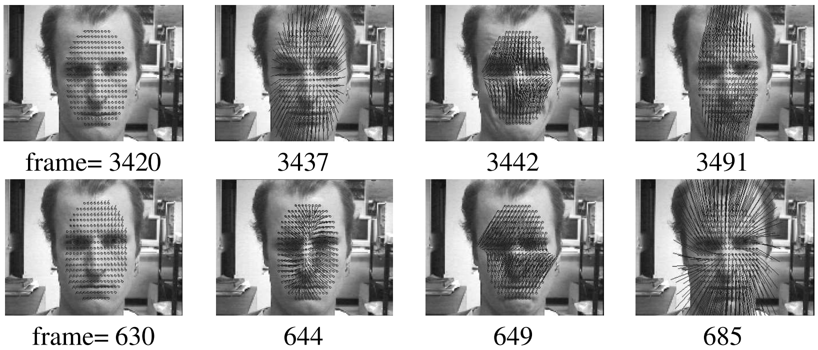
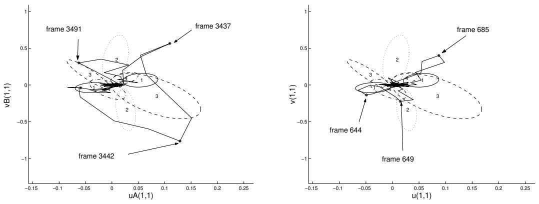


Figure 12: Example sequences for subject $\mathcal{C}$.



(a) (b)

Figure 13: Feature vectors components ${}^{u}A_1^1$ and ${}^{v}B_1^1$ (a), (b) correspond to top and bottom rows in Figure 12

4.4 Subject $\mathcal{D}$

Subject $\mathcal{D}$ could not imitate a different response to C_1 and C_2. Indeed, the clustering procedure found essentially three significant clusters, as can be seen in the confusion matrix (Table 4). Imitations of C_3 and C_4 were consistently classified as D_1 and D_4, respectively. The top row in Figure 14 shows an example of a C_3 imitation clustered in D_1. Figure 15(a) shows the trajectory for this sequence, as well as the Gaussian level curves in the $D = 1$ model, which can be seen to be modeling this trajectory. The bottom row in Figure 14 shows the C_3 imitation which was clustered in D_3. The trajectory (Figure 15) shows that this imitation did not have the strength of most of the others (the expansion was not as fast). Furthermore, the display began with a different eyebrow raising than in the other C_3 imitations. While the D_2 cluster modeled the C_1 and C_2 imitations accurately, there were some which were better modeled with cluster D_3. As with subject $\mathcal{A}$'s model $D = 1$, this subject's model $D = 3$ was a kind of garbage collector, which modeled all the sequences (4 of them) which were not well modeled by any other. This is clearly an indication of overfitting in the number of classes. This subject, as with subject $\mathcal{A}$, would be better modeled with only three classes.

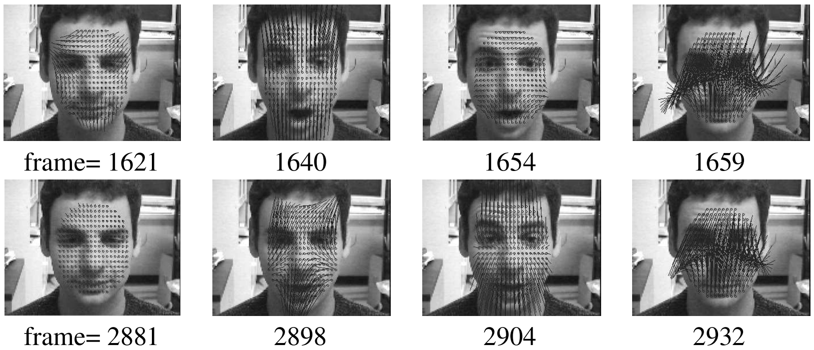
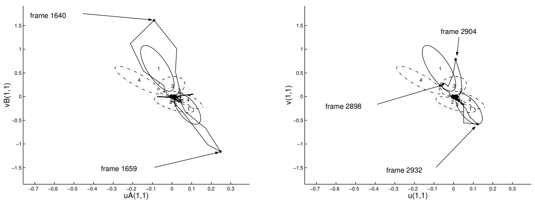


Figure 14: Example sequences for subject $\mathcal{D}$.



(a) (b)

Figure 15: Feature vector components ${}^{u}A_1^1$ and ${}^{v}B_1^1$ (a), (b) correspond to top and bottom rows in Figure 14

5 Conclusions

We have motivated and presented an approach to adaptive context dependent facial display recognition. We use a holistic optical flow representation projected to a complete orthogonal basis of Zernike polynomials, which has been shown to be an effective and *a priori* representation of facial motion. The clustering method uses a mixture of hidden Markov models conditioned on a context variable. We have discussed the application of this method to a simple imitation experiment, and have shown how it can uncover classes of facial displays. It is worth noting that the emphasis on context also implies that the recognition of facial expression during speech can not be attempted without an integrated approach [7, 8]. The probabilistic formulation of the models we propose allow them to be integrated with speech recognition, and thus our methods are scalable in this important direction.

References

[1] T. Bickmore and J. Cassell. "how about this weather?" social dialogue with embodied conversational agents. In *Proc. AAAI Fall Symposium on Socially Intelligent Agents*, 2000.

[2] M. Black and P. Anandan. The robust estimation of multiple motions: parametric and piecewise smooth flow fields. *Computer Vision and Image Understanding*, 63(1):75–104, January 1996.

[3] M. Black and Y. Yacoob. Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motions. *International Journal of Computer Vision*, 25(1):23–48, 1997.

[4] A. Bobick and Y. Ivanov. Action recognition using probabilistic parsing. In *IEEE Intl. Conference on Computer Vision*, pages 196–202, Mumbai, India, 1998.

[5] M. Brand. Coupled hidden Markov models for modeling interacting processes. Technical Report 405, MIT Media Lab Perceptual Computing, June 1997.

[6] M. Brand. Structure learning in conditional probability models via an entropic prior and parameter extinction. *Neural Computation*, 11:1155–1182, 1999.

[7] J. Cassell. Embodied converstation: Integrating face and gesture into automatic spoken dialogue systems. In S. Luperfoy, editor, *Spoken Dialogue Systems*. MIT Press, in press.

[8] N. Chovil. Social determinants of facial displays. *Journal of Nonverbal Behavior*, 15(3):141–154, Fall 1991.

[9] N. Chovil. Facing others: A social communicative perspective on facial displays. In J. A. Russell and J. M. Ferna'ndez-Dols, editors, *The Psychology of Facial Expression*, chapter 14, pages 321–333. Cambridge University Press, Cambridge, UK, 1997.

[10] H. H. Clark. *Using Language*. Cambridge University Press, Cambridge, UK, 1996.

[11] B. Clarkson and A. Pentland. Unsupervised clustering of ambulatory audio and video. In *Proc. ICASSP*, 1999.

[12] A. Dempster, N.M.Laird, and D. Rubin. Maximum likelihood from incomplete data using the EM algorithm. *Journal of the Royal Statistical Society*, 39(B), 1977.

[13] S. Fine, Y. Singer, and N. Tishby. The hierarchical Hidden Markov Model: Analysis and applications. *Machine Learning*, 32:41, 1998.

[14] A. J. Fridlund. *Human facial expression: an evolutionary view*. Academic Press, San Diego, CA, 1994.
[15] A. J. Fridlund. The new ethology of human facial expressions. In J. A. Russell and J. M. Ferna'ndez-Dols, editors, *The Psychology of Facial Expression*, chapter 5, pages 103–129. Cambridge University Press, Cambridge, UK, 1997.
[16] A. Galata, N. Johnson, and D. Hogg. Learning structured behaviour models using variable length Markov models. In *IEEE Workshop on Modelling People*, Corfu, Greece, September 1999.
[17] J. Hoey. Hierarchical unsupervised learning of facial expression categories. In *Proc. ICCV Workshop on Detection and Recognition of Events in Video*, Vancouver, Canada, July 2001. to appear.
[18] J. Hoey and J. J. Little. Representation and recognition of complex human motion. In *Proc. IEEE CVPR*, Hilton Head, SC, June 2000.
[19] Y. li Tian, T. Kanade, and J. F. Cohn. Recognizing action units for facial expression analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2), February 2001.
[20] M.Turk and A.Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.
[21] M. R. Naphade and T. S. Huang. A probabilistic framework for semantic video indexing, filtering, and retrieval. *IEEE Transactions on Multimedia*, 3(1):141–151, Marcho 2001.
[22] A. Prata and W. Rusch. Algorithm for computation of Zernike polynomials expansion coefficients. *Applied Optics*, 28(4):749–754, February 1989.
[23] B. Reeves and C. Nass. *The media equation*. Cambridge University Press, 1996.
[24] M. Slaney and D. Ponceleon. Hierarchical segmentation using latent semantic indexing in scale space. In *Proc. ICASSP*, Salt Lake City, UT, May 2001.
[25] P. Smyth. Clustering sequences with hidden Markov models. In *NIPS 10*, 1997.
[26] K. R. Thorisson. Layered, modular action control for communicative humanoids. In *Computer Animation '97*, pages 134–143, Geneva, Switzerland, June 5-6 1997.
[27] M. Walter, A. Psarrou, and S. Gong. Data driven gesture model acquisition using minimum description length. In *Proc. BMVC*, Manchester, UK, September 2001.
[28] A. D. Wilson, A. F. Bobick, and J. Cassell. Temporal classification of natural gesture and application to video coding. In *Proc. CVPR*, Puerto Rico, June 1997.
[29] C. R. Wren, B. P. Clarkson, and A. P. Pentland. Understanding purposeful human motion. In *Proc. Face and Gesture Recognition*, Grenoble, France, 2000.