Facial Expression Recognition from Video Sequences: Temporal and Static Modelling

Ira Cohen[1], Nicu Sebe[2], Larry Chen[3], Ashutosh Garg[1], Thomas S. Huang[1]

[1]Beckman Institute for Advanced Science and Technology, University of Illinois at Urbana-Champaign
iracohen@ifp.uiuc.edu, ashutosh@ifp.uiuc.edu, huang@ifp.uiuc.edu

[2]Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands, nicu@liacs.nl

[3]Imaging Science and Technology Lab, Eastman Kodak Company, lawrence.chen@kodak.com

Abstract

Human-computer intelligent interaction (HCII) is an emerging field of science aimed at providing natural ways for humans to use computers as aids. It is argued that for the computer to be able to interact with humans, it needs to have the communication skills of humans. One of these skills is the ability to understand the emotional state of the person. The most expressive way humans display emotions is through facial expressions. In this work we report on several advances we have made in building a system for classification of facial expressions from continuous video input. We introduce and test different architectures, focusing on changes in distribution assumptions and feature dependency structures. We also introduce a facial expression recognition from live video input using temporal cues. Methods for using temporal information have been extensively explored for speech recognition applications. Among these methods are template matching using dynamic programming methods and hidden Markov models (HMM). This work exploits existing methods and proposes a new architecture of HMMs for automatically segmenting and recognizing human facial expression from video sequences. The architecture performs both segmentation and recognition of the facial expressions automatically using an multi-level architecture composed of an HMM layer and a Markov model layer. We explore both person-dependent and person-independent recognition of expressions and compare the different methods.

1 Introduction

In recent years there has been a growing interest in improving all aspects of the interaction between humans and computers. This emerging field has been a research interest for scientists from several different scholastic tracks, i.e., computer science, engineering, psychology, and neuroscience. These studies focus not only on improving computer interfaces, but also on improving the actions the computer takes based on feedback from the user. Feedback from the user has traditionally been given through the keyboard and mouse. Other devices have also been developed for more application specific interfaces, such as joysticks, trackballs, datagloves, and touch screens. The rapid advance of technology in recent years has made computers cheaper and more powerful, and has made the use of microphones and PC-cameras affordable and easily available. The microphones and cameras enable the computer to "see" and "hear," and to use this information to act. A good example of this is the "Smart-Kiosk" [15]. It is argued that to truly achieve effective human-computer intelligent interaction (HCII), there is a need for the computer to be able to interact naturally with the user, similar to the way human-human interaction takes place. Humans interact with each other mainly through speech, but also through body gestures, to emphasize a certain part of the speech, and display of emotions. Emotions are displayed by visual, vocal, and other physiological means. There is a growing amount of evidence showing that emotional skills are part of what is called "intelligence" [38, 17].

In this work we present several advances we have made towards a facial expression recognition system. We first describe the real time face tracking system used and the features extracted from the face tracking to be used for facial expression recognition. We then describe several different classifiers developed for recognizing the facial expressions. The first class of classifiers use the features extracted for each frame in the video sequence to produce a classification result for that frame. The second type of classifier is a multi-level HMM classifier, combining the temporal information to both automatically segment the video sequence to the different expressions and perform the classification of each segment to the corresponding facial expression.

2 Literature Review

There is little agreement about a definition of emotion. Many theories of emotion have been proposed. Some of these could not be verified until recently when measurement of some physiological signals become available. In general, emotions are short-term, whereas moods are long-term, and temperaments or personalities are very long-term [22]. A particular mood may sustain for several days, and a temperament for months or years. Finally, emotional disorders can be so disabling that people affected are no longer able to lead normal lives.

Darwin [7] held an ethological view of emotional expressions, arguing that the expressions from infancy and lower life forms exist in adult humans. Following the *Origin of Species* he wrote *The Expression of the Emotions in Man and Animals*. According to him, emotional expressions are closely related to survival. Thus in human interactions, these nonverbal expression are as important as the verbal interaction. James [21] viewed emotions not as *causes* but as *effects*. Situations arise around us which cause changes in physiological signals. According to James, "the bodily changes follow directly the perception of the exciting fact, and that our feeling of the same changes as they occur *is* the emotion." Carl Lange proposed a similar theory independently at around the same time. This is often referred to as the "James-Lange" theory of emotion. Cannon [3], contrary to James, believed that emotions are first felt, then exhibited outwardly causing certain behaviors.

2.1 Judgment Studies

Despite these diverse theories, it is evident that people display expressions to various degrees. One frequently studied task is the judgment of emotions—how well can human observers tell the emotional expressions of others, in the voice, on the face, etc? Related questions are: Do these represent their true emotions? Can they be convincingly portrayed? How well can people conceal their emotions? In such tasks, researchers often use two different methods to describe the emotions.

One approach is to label the emotions in discrete categories, i.e., human judges must choose from a prescribed list of word labels, such as *joy, fear, love, surprise, sadness*, etc. One problem with this approach is that the stimuli may contain blended emotions. Also the choice of words may be too restrictive, or culturally dependent.

Another way is to have multiple dimensions or scales to describe emotions. Instead of choosing discrete labels, observers can indicate their impression of each stimulus on several continuous scales, for example, pleasant–unpleasant, attention–rejection, simple–complicated, etc. Two common scales are valence and arousal. Valence describes the pleasantness of the stimuli, with positive (or pleasant) on one end, and negative (or unpleasant) on the other. For example, *happiness* has a positive valence, while *disgust* has a negative valence. The other dimension is arousal or activation. For example, *sadness* has low arousal, whereas *surprise* has high arousal level. The different emotional labels could be plotted at various positions on a two-dimensional plane spanned by these two axes to construct a 2D emotion model [24]. Scholsberg [39] suggested a three-dimensional model in which he had *attention–rejection* in addition to the above two.

Another interesting topic is how the researchers managed to obtain data for observation. Some people used posers, including professional actors and nonactors. Others attempted to induce emotional reactions by some clever means. For example, Ekman showed stress-inducing film of nasal surgery in order to get the disgusted look on the viewers' faces. Some experimenter even dumped water on the subjects or fired blank shots to induce surprise, while others used clumsy technicians who made rude remarks to arouse fear and anger [19]. Obviously, some of these are not practical ways of acquiring data. After studying acted and natural expressions, Ekman concluded that expressions can be convincingly portrayed [10].

2.2 Facial Expression Recognition Studies

Since the early 1970s, Paul Ekman and his colleagues have performed extensive studies of human facial expressions [11]. They found evidence to support universality in facial expressions. These "universal facial expressions" are those representing happiness, sadness, anger, fear, surprise, and disgust. They studied facial expressions in different cultures, including preliterate cultures, and found much commonality in the expression and recognition of emotions on the face. However, they observed differences in expressions as well, and proposed that facial expressions are governed by "display rules" in different social contexts. For example, Japanese subjects and American subjects showed similar facial expressions while viewing the same stimulus film. However, in the presence of authorities, the Japanese viewers were more reluctant to show their real expressions. Matsumoto [29] reported the discovery of a seventh universal facial expression: contempt. Babies seem to exhibit a wide range of facial expressions without being taught, thus suggesting that these expressions are innate [20].

Ekman and Friesen [12] developed the Facial Action Coding System (FACS) to code facial expressions where movements on the face are described by a set of action units (AUs). Each AU has some related muscular basis. This system of coding facial expressions is done manually by following a set of prescribed rules. The inputs are still images of facial expressions, often at the peak of the expression. This process is very time-consuming.

Ekman's work inspired many researchers to analyze facial expressions by means of image and video processing. By tracking facial features and measuring the amount of facial movement, they attempt to categorize different facial expressions. Recent work on facial expression analysis and recognition [28, 42, 25, 1, 34, 13, 31, 37, 9, 27] has used these "basic expressions" or a subset of them.

The work in computer-assisted quantification of facial expressions did not start until the 1990s. Mase [28] used optical flow (OF) to recognize facial expressions. He was one of the first to use image processing techniques to recognize facial expressions. Lanitis et al. [25] used a flexible shape and appearance model for image coding, person identification, pose recovery, gender recognition, and facial expression recognition. Black and Yacoob [1] used local parameterized models of image motion to recover non-rigid motion. Once recovered, these parameters are used as inputs to a rule-based classifier to recognize the six basic facial expressions. Yacoob and Davis [43] computed optical flow and used similar rules to classify the six facial expressions. Rosenblum, Yacoob, and Davis [34] also computed optical flow of regions on the face, then applied a radial basis function network to classify expressions. Essa and Pentland [13] also used an optical flow region-based method to recognize expressions. Donato et al. [9] tested different features for recognizing facial AUs and inferring the facial expression in the frame. Otsuka and Ohya [31] first computed optical flow, then computed their 2D Fourier transform coefficients, which were used as feature vectors for a hidden Markov model (HMM) to classify expressions. The trained system was able to recognize one of the six expressions near real-time (about 10 Hz). Furthermore, they used the tracked motions to control the facial expression of an animated Kabuki system [32]. A similar approach, using different features, was used by Lien [27].

These methods are similar in the general sense that they first extract some features from the images, then these features are used as inputs into a classification system, and the outcome is one of the preselected emotion categories. They differ mainly in the features extracted from the video images and in the processing of video images to classify emotions. The video processing falls into two broad categories. The first is "feature-based," where one tries to detect and track specific features such as the corners of the mouth, eyebrows, etc. The other approach is "region-based" in which facial motions are measured in certain regions on the face such as the eye/eyebrow and mouth regions. People have used different classification algorithms to categorize these emotions.

Ueki et al. [42] extracted AUs and used neural networks (NN) to analyze the emotions. Seventeen AUs were mapped to two dimensions using an identity mapping network, and this showed resemblance of the 2D psychological emotion models. Later on, Morishima [30] proposed a 3D emotion model in order to deal with transitions between emotions, and claimed correlation to the 3D psychological emotion model [39].

Another interesting thing to point out is commonly confused categories in these six basic expressions. As reported by Ekman, *anger* and *disgust* are commonly confused in judgment studies. Also, *fear* and *surprise* are commonly confused. The reason why these confusions occur is because they share many similar facial actions [12]. *Surprise* is sometimes mistaken for *interest*, but not the other way around. In the computer recognition studies, some of these confusions are observed [1, 43].

3 Face Tracking and Feature Extraction

The face tracking we use in our system is based on a system developed by Tao and Huang [41] called the Piecewise Bézier Volume Deformation (PBVD) tracker.

This face tracker uses a model-based approach where an explicit 3D wireframe model of the face is constructed. In the first frame of the image sequence, landmark facial features such as the eye corners and mouth corners are selected interactively. The generic face model is then warped to fit the selected facial features. The face model consists of 16 surface patches embedded in Bézier volumes. The surface patches defined this way are guaranteed to be continuous and smooth. The shape of the mesh can be changed by changing the locations of the control points in the Bézier volume. Before describing the Bézier volume, we begin with the Bézier curve.

The Bézier curve is a parametric curve defined as

$$\mathbf{x}(u) = \sum_{i=0}^{n} \mathbf{b}_i B_i^n(u) = \sum_{i=0}^{n} \mathbf{b}_i \binom{n}{i} u^i (1-u)^{n-i} \qquad (1)$$

where the shape of the curve is controlled by a set of control points $\mathbf{b}_i$. As the control points are moved, a new shape is obtained according to the Bernstein polynomials $B_i^n(u)$ in Equation (1). The displacement of a point on the curve can be described in terms of linear combinations of displacements of the control points.

The Bézier surface is a straight-forward 3D extension of the Bézier curve where the equation becomes

$$\mathbf{v}(u,v,w) = \sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{k=0}^{n} \mathbf{d}_{i,j,k} B_i^n(u) B_j^m(v) B_k^l(w). \qquad (2)$$

This can also be written in matrix notation as

$$\mathbf{V} = \mathbf{B}\mathbf{D} \qquad (3)$$

where $\mathbf{V}$ is the displacement of the mesh nodes, $\mathbf{D}$ is the matrix whose columns are the control point displacement vectors of the Bézier volume, and $\mathbf{B}$ is the mapping in terms of Bernstein polynomials. In other words, the change in the shape of the face model can be described in terms of the deformations in $\mathbf{D}$.

Once the model is constructed and fitted, head motion and local deformations of the facial features such as the eyebrows, eyelids, and mouth can be tracked. First the 2D image motions are measured using template matching between frames at different resolutions. Image templates from the previous frame and from the very first frame are both used for more robust tracking. The measured 2D image motions are modeled as projections of the true 3D motions onto the image plane. From the 2D motions of many points on the mesh, the 3D motion can be estimated by solving an overdetermined system of equations of the projective motions in the least squared sense. Figure 1(a) shows an example of the face tracker interface for one frame.

The recovered motions are represented in terms of magnitudes of some predefined motion of various facial features. Each feature motion corresponds to a simple deformation on the face, defined in terms of the Bézier volume control parameters. With some abuse of notation, we refer to these motions vectors as Action-Units (AU's), but note that they are not equivalent to Ekman's AU's, and are numeric in nature, representing not only the activation of a facial region, but also the direction and intensity of the motion. Table 1 lists the AU's used in the face tracker which are also shown in Figure 1(b). Each facial expression is modeled as a linear combination of the AUs

$$\mathbf{V} = \mathbf{B}\left[\mathbf{D}_0\mathbf{D}_1 \ldots \mathbf{D}_m\right] \begin{bmatrix} p_0 \\ p_1 \\ \vdots \\ p_m \end{bmatrix} = \mathbf{BDP} \qquad (4)$$

where each of the $\mathbf{D}_i$ corresponds to an AU, and the p_i are the corresponding magnitudes (or coefficients) of each deformation. The overall motion of the head and face is

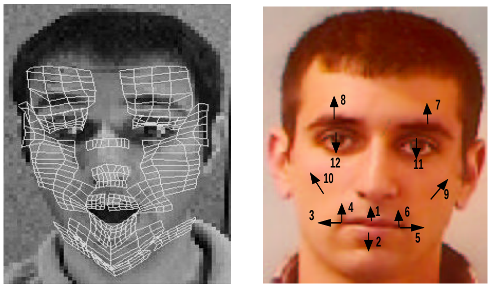
$$\mathbf{R}(\mathbf{V}_0 + \mathbf{BDP}) + \mathbf{T} \qquad (5)$$

where $\mathbf{R}$ is the 3D rotation matrix, $\mathbf{T}$ is the 3D translation matrix, and $\mathbf{V}_0$ is the initial face model.

4 SNow-Naive-Bayes and SNoW Classifiers Using Discrete Features

In this section we describe the use of SNoW (Sparse Network of Winnows) and SNow-Naive-Bayes (NB) classifiers for our problem, using discretized versions of the features discussed in the previous section.

The advantage of discretizing the features is that no assumptions on the distribution of the features has to be made (see next section) and the true distribution is approximated well. The disadvantage is that the complexity of the classifier increases exponentially with the number of values the features take, requiring larger training sets for reliable estimation.



(a) (b)

Figure 1. (a) The wireframe model, (b) the facial motion measurements

The SNoW classifier was developed by Roth and has been successfully applied to natural language processing for context-sensitive spelling correction [35], and more recently for face detection in images [36]. Here we use it for image sequence analysis. First the original features (i.e., the Action Unit measurements) are transformed into a higher dimensional feature space of features, after which the connections from (transformed) feature nodes to each of the output target nodes will be sparse. The idea is to transform from the original feature space where the classes are not linearly separable to a higher dimensional space where the classes become more linearly separable.

Figure 2 shows the SNoW classifier. In the input layer, raw measurements $\{r_1, \ldots, r_n\}$ are transformed to a larger space of binary features $\{f_1, \ldots, f_m\}$, where $m \gg n$. A number of transformations can be used. In this work, we discretize the raw input into bins, activating a certain feature if input falls into the associated bin. We used both uniform-sized bins and nonuniform-sized bins. We also threshold the raw inputs and combinations of raw inputs for additional features. This classifier is data-driven, i.e., features are only allocated and activated when input data contribute to them. In the output layer, each class has a target node T_1 to T_k, where the output of each node equals the weighted sum of the features $\Sigma w_i f_i$. During training, all input data are labelled with the correct output class. Each connection from the feature layer $\{f_i\}$ to the output layer has an initial weight, which is updated in training (promoted or demoted) according to the type of error made for each training sample.

Each output node has two updating parameters: promotion parameter α and demotion parameter β. If the correct output target node fails to win the competition using the current features, weights from the feature layer to this output node are promoted using a multiplicative rule $W \leftarrow \alpha \times W$. On the other hand, if the output is turned on where it should be off, all its weights are demoted according to the demo-

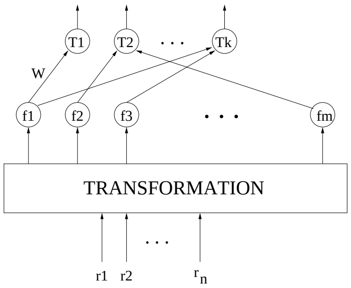


Figure 2. The SNoW architecture.

tion parameter $W \leftarrow \beta \times W$. This multiplicative update is very fast. For testing, a competition is carried out in the output layer in the winner-takes-all fashion. Another advantage is that, unlike in fully connected networks such as multilayered perceptrons, SNoW does not require many training examples to train the network. Other advantages include the sparseness of the connections established in a data-drive fashion and the incorporation of prior knowledge. A variant of SNoW, which will be referred to as SNoW-NB, is also used where output targets are replaced with naive Bayes nodes.

The video features (AUs) described in the previous section are the "raw inputs" here for SNoW, so we need to apply some feature extraction (transformation) to produce the binary features.

The raw features $\{r_1, \ldots, r_n\}$ are discretized into bins, where each bin corresponds to one binary input in $\{f_1, \ldots, f_m\}$ for SNoW. The discretization can be of uniform or nonuniform sized bins, depending on the distribution of the feature values. The number of the bins also depends on the application.

In addition to discretization, thresholding can also be used to produce additional inputs. For example, binary feature f_j can be activated if a certain raw input is below some threshold value. Here we tried different thresholds in an overlapping fashion.

SNoW-NB (with naive Bayes output nodes) classifier is a probabilistic classifier in which the features are assumed independent given the class. The SNoW-NB can be understood as a Naive-Bayes Bayesian network, where all the features are discrete. NB classifiers have a very good record in many classification problems, although the independence assumption is usually violated in practice. The reason for the NB success as a classifier is attributed to the small number of parameters needed to be estimated. Recently, Garg and Roth [16] showed using information theoretic arguments additional reasons for the success of NB classifiers.

5 Continuous Naive-Bayes and Changing the Distribution: Cauchy Naive Bayes Classifier

Consider a binary classification problem with $y \in \{0, 1\}$ (class label) and $X \in R^n$ (feature vector) the observed data. The classification problem under the maximum likelihood framework (ML) can be formulated as:

$$\hat{y} = argmax\ P(X|y) \tag{6}$$

If the features in X are assumed to be independent of each other conditioned upon the class label (the Naive Bayes framework), Equation (6) reduces to:

$$\hat{y} = argmax \prod_{i=1}^{N} P(x_i|y) \tag{7}$$

Now the problem is how to model the probability of features given the class label $P(x_i|y)$. In practice, the common assumption is that we have a Gaussian distribution and the ML can be used to obtain the estimate of the parameters (mean and variance). However, Sebe, et al. [40] have shown that the Gaussian assumption is often invalid and proposed the Cauchy distribution as an alternative model. Intuitively, this distribution can be thought of as being able to model the heavy tails observed in the empirical distribution. This model is referred to as *Cauchy Naive Bayes*.

The difficulty of this model is in estimating the parameters of the Cauchy distribution. For a sample of size n sampled from the Cauchy distribution the likelihood is given by:

$$L(X; a, b) = \prod_{i=1}^{n} \left[\frac{b}{\pi(b^2 + (x_i - a)^2)} \right] \tag{8}$$

where a is the location parameter and b is the scale parameter. Note that similar with the Gaussian case we have to estimate only two parameters.

Let $\hat{a}$ and $\hat{b}$ be the maximum likelihood estimators for a and b. The maximum likelihood equations are

$$\sum_{i=1}^{n} \frac{x_i - \hat{a}}{\hat{b}^2 + (x_i - \hat{a})^2} = 0 \tag{9}$$

$$\sum_{i=1}^{n} \frac{\hat{b}^2}{\hat{b}^2 + (x_i - \hat{a})^2} = \frac{n}{2} \tag{10}$$

The Equations (9) and (10) are high order polynomials and therefore a numerical procedure must be used in order to solve them for $\hat{a}$ and $\hat{b}$. For solving these equations we used a Newton-Raphson iterative method with the starting points given by the mean and the variance of the data. We were always able to find unique positive solutions for $\hat{a}$ and $\hat{b}$ which is in accordance with the conjecture stated by Hass, et al. [18]. In certain cases, however, the Newton-Raphson iteration diverged, in which cases we selected new starting points.

5.1 Choosing the Distribution

We consider that representative ground truth is provided. We split the ground truth in two nonoverlapping sets: the training set and the test set. The estimation of the parameters is done using only the training set. The classification is performed using only the test set.

An interesting problem is determining when to use the Cauchy assumption versus the Gaussian assumption. One solution is to compute the distribution for the data and to match this distribution using a Chi-square or a Kolmogorov-Smirnov test with the model distributions (Cauchy or Gaussian) estimated using the ML approach described above. Another solution (considered here) is to extract a random subsample from the training set and to perform an initial classification. The model distribution which provides better results would be used further in the classification of the test set. The assumption behind this solution is that the training set and the test set have similar characteristics.

In summary, our algorithm can be described as follows:

Step 1. For each class consider the corresponding training set and estimate the parameters of the model (Gaussian and Cauchy) using the ML framework.

Step 2. Extract a random sample from the training set and perform classification. The model which provides the best results will be assigned for each individual class in the classification step.

Step 3. Perform classification using only the test set.

6 Beyond the NB Assumption: Finding Dependencies among Features Using a Hybrid TAN and Gaussian Classifier

As mentioned before, the NB classifier was successful in many applications. However, the strong independence assumptions seem to be very unreasonable in our case. It could be beneficial to search for a better structure that captures better the dependencies among the features. Of course, to attempt to find all the dependencies is an NP-complete problem. So, we restrict ourselves to a smaller class of structures called the Tree-Augmented-Naive Bayes (TAN) classifiers. TAN classifiers have been introduced by Friedman et al. [14] and are represented as Bayesian networks. Bayesian networks are acyclic graphical models, with the class and features as the nodes, and dependencies are represented by the directed edges in the graph between the nodes. The joint probability distribution is factored to a collection of conditional probability distributions of each node in the graph.

In the TAN classifier structure the class node has no parents and each feature has the class node as a parent and at most one other feature, such that the result is a tree structure for the features. An example of a TAN classifier is given in Figure 3. Friedman et al. [14] proposed using the TAN model as a classifier, to enhance the performance over the simple Naive-Bayes classifier. TAN models are more complicated then the Naive-Bayes, but are not fully connected graphs. The existence of an efficient algorithm to compute the best TAN model makes it a good candidate in the search for a better structure over the simple NB. Learning the TAN classifier is more complicated. In this case, we do not fix the structure of the Bayesian network, but try to find the TAN structure that maximizes the likelihood function given the training data out of all possible TAN structures.

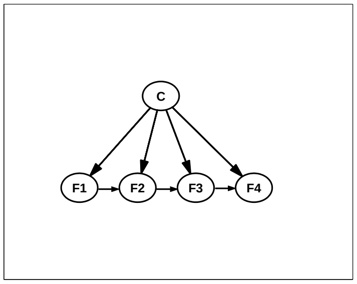


Figure 3. An example of a TAN classifier.

In general, searching for the best structure has no efficient solution, however, searching for the best TAN structure does have one. The method is using the modified Chow-Liu algorithm [5] for constructing tree augmented Bayesian networks [14]. This is done as follows:

1. Compute the class conditional pair-wise mutual information between each pair of features, $I_P(X_i, X_j|C) = \sum_{X_i,X_j,C} P(x_i,x_j,c) log \frac{P(x_i,x_j|c)}{P(x_i|c)P(x_j|c)}, i \neq j.$

2. Build a complete undirected graph in which each vertex is a variable, and the weight of each edge is the mutual information computed in 1.

3. Build a maximum weighted spanning tree (MWST).

4. Transform the undirected MWST of 3 to a directed graph by choosing a root node and pointing the arrows of all edges away from the root.

5. Make the class node the parent of all the feature nodes in the directed graph of step 4.

This procedure ensures to find the TAN model that maximizes the likelihood of the data we have. The algorithm is computed in polynomial time ($O(n^2 logN)$, with N being the number of instances and n the number of features).

The learning algorithm for the TAN classifier is only feasible in cases where all the features are discrete. In our problem the features are continuous. The number of parameters of the TAN model grows exponentially with respect to the number of discrete values each feature takes. To solve this problem we propose a hybrid TAN and Gaussian classifier. We first discretize the features and use the TAN model learning algorithm to learn the dependency structure among the features. Then, we revert back to the original continuous features and model them as Gaussian, using the TAN graph structure. The added complexity of the Gaussian model is only linear in the number of features, but we are still able to capture dependencies among the features.

The full joint distribution of the Gaussian-TAN model can be written as:

$$p(c, x_1, x_2, ..., x_n) = p(c) \prod_{i=1}^{n} p(x_i | pa_{x_i}, c), \tag{11}$$

where pa_{x_i} is the feature that is the additional parent of feature x_i. pa_{x_i} is empty for the root feature in the directed tree graph of step 4 in the procedure above.

Using the Gaussian assumption, the pdf's of the distribution in the product above are:

$$p(X_i = x_i | pa_{x_i}, C = c) = N_c(\mu_{x_i} + a \cdot pa_{x_i}, \sigma_{x_i}^2 \cdot (1 - \rho^2)), \tag{12}$$

where $N_c(\mu, \sigma^2)$ refers to the Gaussian distribution with mean and variance given that the class is c, $\mu_{x_i}, \sigma_{x_i}^2$ are the mean and variance of the feature x_i, $\rho = \frac{COV(x_i, pa_{x_i})}{\sigma_{x_i} \sigma_{pa_{x_i}}}$ is the correlation coefficient between x_i and pa_{x_i}, and $a = \frac{COV(x_i, pa_{x_i})}{\sigma_{pa_{x_i}}^2}$.

Estimating the Gaussian-TAN model involves estimating all the class conditional means and variances for each feature as in the NB model, then estimate the class conditional covariances between features and their feature parents. In terms of model complexity, there are $|C| \cdot (n-1)$ extra parameters to estimate (the covariances).

Figure 4 shows the learned tree structure of the features learned using a database of subjects displaying different facial expressions. The arrows are from parents to children features. From the tree structure we see that the bottom half of the face is almost disjoint from the top portion, except for a weak link between AU 4 and AU 11.

7 The Temporal Approach: Facial Expression Recognition Using Multi-level HMMs

In this section we suggest another approach for recognizing the emotion through facial expression displayed in live video. In contrast to the methods described in the previous sections, this method uses temporal information displayed in the video also to discriminate different expressions. The logic behind using all of the temporal information is that any emotion being displayed has a unique temporal pattern.

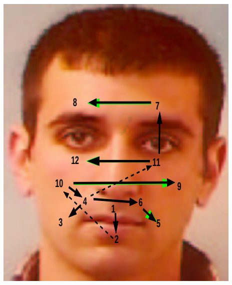


Figure 4. The learned TAN structure for the facial features. Dashed lines represent links that are relatively weaker than the others.

The method we propose automatically segments the video to the different facial expression sequences, using an multi-level HMM structure. The first level of the architecture is comprised of independent HMMs related to the different emotions. This level of HMMs is very similar to the one used in [31] and [27] who use the likelihood of a given sequence in a ML classifier to classify a given video sequence. Instead of classifying using the output of each HMM, we use the state sequence of the HMMs as the input of the higher level Markov model. This is meant to segment the video sequence. Moreover, this also increases the discrimination between the classes since it tries to find not only the probability of each the sequence displaying one emotion, but the probability of the sequence displaying one emotion and not displaying all the other emotions at the same time.

7.1 Hidden Markov Models

Hidden Markov models have been widely used for many classification and modeling problems. Perhaps the most common application of HMM is in speech recognition. One of the main advantages of HMMs is their ability to model nonstationary signals or events. Dynamic programming methods allow one to align the signals so as to account for the non stationarity. However, the main disadvantage of this approach is that it is very time-consuming since all of the stored sequences are used to find the best match. The HMM finds an implicit time warping in a probabilistic parametric fashion. It uses the transition probabilities between the hidden states and learns the conditional probabilities of the observations given the state of the model. In the case of emotion expression, the signal is the measurements of the facial motion. This signal is non stationary in nature, since an expression can be displayed at varying rates, with varying intensities even for the same individual.

An HMM is given by the following set of parameters:

$$\begin{aligned}
\lambda &= (A, B, \pi) \\
a_{ij} &= P(q_{t+1} = S_j | q_t = S_i), 1 \leq i, j \leq N \\
B &= \{b_j(O_t)\} = P(O_t | q_t = S_j), 1 \leq j \leq N \\
\pi_j &= P(q_1 = S_j)
\end{aligned}$$

where A is the state transition probability matrix, B is the observation probability distribution, and π is the initial state distribution. The number of states of the HMM is given by N. It should be noted that the observations (O_t) can be either discrete or continuous, and can be vectors. In the discrete case, B becomes a matrix of probability entries (Conditional Probability Table), and in the continuous case, B will be given by the parameters of the probability distribution function of the observations (normally chosen to be the Gaussian distribution or a mixture of Gaussians). Given an HMM there are three basic problems that are of interest. The first is how to efficiently compute the probability of the observations given the model. This problem is related to classification in the sense that it gives a measure of how well a certain model describes an observation sequence. The second is how to find the corresponding state sequence in some optimal way, given a set of observations and the model. This will become an important part of the algorithm to recognize the expressions from live input and will be described later in this paper. The third is how to learn the parameters of the model λ given the set of observations so as to maximize the probability of observations given the model. This problem relates to the learning phase of the HMMs which describe each facial expression sequence. A comprehensive tutorial on HMMs is given by Rabiner [33].

7.2 Expression Recognition Using Emotion-Specific HMMs

Since the display of a certain facial expression in video is represented by a temporal sequence of facial motions it is natural to model each expression using an HMM trained for that particular type of expression. There will be six such HMMs, one for each expression: {*happy(1), angry(2), surprise(3), disgust(4), fear(5), sad(6)*}. There are several choices of model structure that can be used. The two main models are the left-to-right model and the ergodic model. In the left-to-right model, the probability of going back to the previous state is set to zero, and therefore the model will always start from a certain state and end up in an 'exiting' state. In the ergodic model every state can be reached from any other state in a finite number of time steps. In [31], Otsuka and Ohya used left-to-right models with three states to model each type of facial expression. The advantage of using this model lies in the fact that it seems natural to model a sequential event with a model that also starts from a fixed starting state and always reaches an end state. It also involves fewer parameters and therefore is easier to train. However, it reduces the degrees of freedom the model has to try to account for the observation sequence. There has been no study to indicate that the facial expression sequence is indeed modeled well by the left-to-right model. On the other hand, using the ergodic HMM allows more freedom for the model to account for the observation sequences, and in fact, for an infinite amount of training data it can be shown that the ergodic model will reduce to the left-to-right model, if that is indeed the true model. In this work both types of models were tested with various numbers of states in an attempt to study the best structure that can model facial expressions.

The observation vector O_t for the HMM represents continuous motion of the facial action units. Therefore, B is represented by the probability density functions (pdf) of the observation vector at time t given the state of the model. The Gaussian distribution is chosen to represent these pdf's, i.e.,

$$B = \{b_i(O_t)\} \sim N(\mu_j, \Sigma_j), 1 \leq j \leq N \qquad (13)$$

where μ_j and Σ_j are the mean vector and full covariance matrix, respectively.

The parameters of the model of emotion-expression specific HMM are learned using the well-known Baum-Welch reestimation formulas. See [26] for details of the algorithm. For learning, hand labeled sequences of each of the facial expressions are used as ground truth sequences, and the Baum algorithm is used to derive the maximum likelihood (ML) estimation of the model parameters (λ).

Parameter learning is followed by the construction of a ML classifier. Given an observation sequence O_t, where $t \in (1, T)$, the probability of the observation given each of the six models $P(O_t|\lambda_j)$ is computed using the forward-backward procedure [33]. The sequence is classified as the emotion corresponding to the model that yielded the highest probability, i.e.,

$$c^* = argmax_{1 \leq c \leq 6}[P(O|\lambda_c)] \qquad (14)$$

7.3 Automatic Segmentation and Recognition of Emotions Using Multi-level HMM.

The main problem with the approach taken in the previous section is that it works on isolated facial expression sequences or on pre-segmented sequences of the expressions from the video. In reality, this segmentation is not available, and therefore there is a need to find an automatic way of segmenting the sequences. Concatenation of the HMMs representing phonemes in conjunction with the use of grammar has been used in many systems for continuous speech recognition. Dynamic programming for continuous speech has also been proposed in different researches. It is not very straightforward to try and apply these methods to the emotion recognition problem since there is no clear notion of language in displaying emotions. Otsuka and Ohya [31] used a heuristic method based on changes in the motion of several regions of the face to decide that an expression sequence is beginning and ending. After detecting the boundaries, the sequence is classified to one of the emotions using the emotion-specific HMM. This method is prone to errors because of the sensitivity of the classifier to the segmentation result. Although the result of the HMMs are independent of each other, if we assume that they model realistically the motion of the facial features related to each emotion, the combination of the state sequence of the six HMMs together can provide very useful information and enhance the discrimination between the different classes. Since we will use a left-to-right model (with return), the changing of the state sequence can have a physical attribute attached to it (such as opening and closing of mouth when smiling), and therefore there we can gain useful information from looking at the state sequence and using it to discriminate between the emotions at each point in time.

To solve the segmentation problem and enhance the discrimination between the classes, a different kind of architecture is needed. Figure 5 shows the proposed architecture for automatic segmentation and recognition of the displayed expression at each time instance. The motion features are continuously used as input to the six emotion-specific HMMs. The state sequence of each of the HMMs is decoded and used as the observation vector for the high level Markov model. The high-level Markov model consists of seven states, one for each of the six emotions and one for *neutral*. The *neutral* state is necessary as for the large portion of time, there is no display of emotion on a person's face. In this implementation of the system, the transitions between emotions are imposed to pass through the *neutral* state since our training data consists of facial expression sequences that always go through the The *neutral* state. In unconstraint situation, it is possible (although less likely) for a person to go from one expression to another without passing through a neutral expression. In this case, the higher level Markov model will have non-zero transition probabilities of passing from all states to all states (which appear as arcs between the different states). The recognition of the expression is done by decoding the state that the high-level Markov model is in at each point in time since the state represents the displayed emotion.

The training procedure of the system is as follows:

- Train the emotion-specific HMMs using a hand segmented sequence as described in the previous section.

- Feed all six HMMs with the continuous (labeled) facial expression sequence. Each expression sequence contains several instances of each facial expression with *neutral* instances separating the emotions.

- Obtain the state sequence of each HMM to form the six-dimensional observation vector of the higher-level Markov model, i.e., $O_t^h = [q_t^{(1)},...,q_t^{(6)}]^T$, where $q_t^{(i)}$ is the state of the i^{th} emotion-specific HMM. The decoding of the state sequence is done using the Vitterbi algorithm [33].

- Learn the probability observation matrix for each state of the high-level Markov model using $P(q_j^{(i)}|S_k) = \{$expected frequency of model i being in state j given that the true state was $k\}$, and

$$B^{(h)} = \{b_k(O_t^h)\} = \left\{\prod_{i=1}^{6}(P(q_j^{(i)}|S_k)\right\} \qquad (15)$$

  where $j \in (1$,*Number of States for Lower Level HMM)*.

- Compute the transition probability $A = \{a_{kl}\}$ of the high-level HMM using the frequency of transiting from each of the six emotion classes to the *neutral* state in the training sequences and from the *neutral* state to the other emotion states. For notation, the *neutral* state is numbered 7 and the other states are numbered as in the previous section. All the transition probabilities could also be set using expert knowledge, and not necessarily from training data.

- Set the initial probability of the high level Markov model to be 1 for the *neutral* state and 0 for all other states. This forces the model to always start at the *neutral* state and assumes that a person will display a *neutral* expression in the beginning of any video sequence. This assumption is made just for simplicity of the testing.

The steps followed during the testing phase are very similar to the ones followed during training. The face tracking sequence is used as input into the lower-level HMMs and a decoded state sequence is obtained using the Vitterbi algorithm. The decoded lower-level state sequence O_t^h is used as input to the higher-level HMM and the observation probabilities are computed using Equation (15). Note that in this way of computing the probability, it is assumed that the state sequences of the lower-level HMMs are independent given the true labeling of the sequence. This assumption is reasonable since the HMMs are trained independently and on different training sequences. In addition, without this assumption, the size of B will be enormous, since it will have to account for all possible combinations of states of the six lower-level HMMs, and it would require a huge amount of training data.

Using the Vitterbi algorithm again for the high level Markov model, a most likely state sequence is produced. The state that the HMM was in at time t corresponds to the expressed emotion in the video sequence at time t. To make

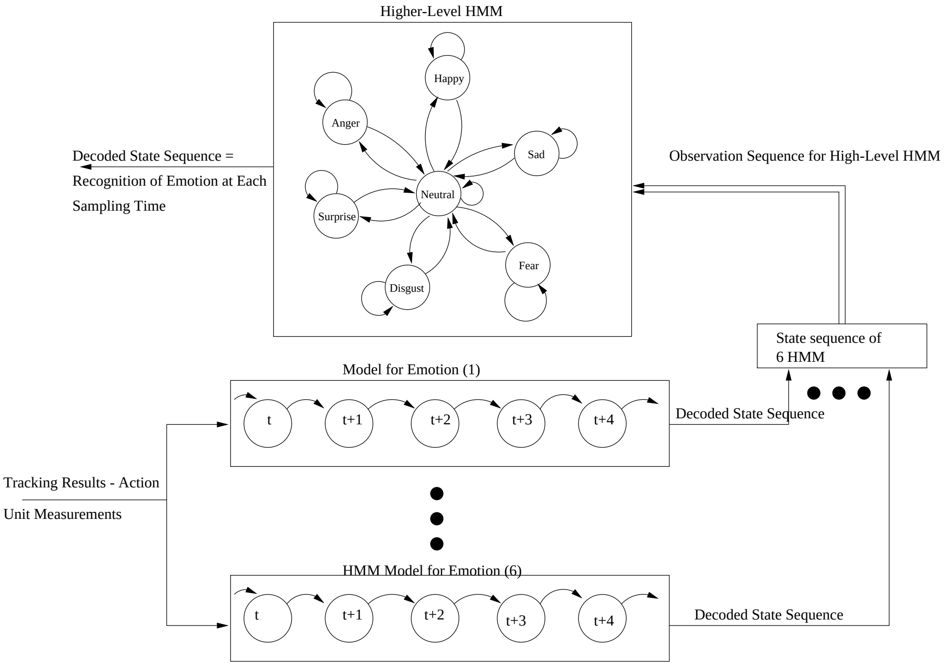


Figure 5. Multilevel HMM architecture for automatic segmentation and recognition of emotion.

the classification result robust to undesired fast changes, a smoothing of the state sequence is done by preserving the actual classification result if the HMM did not stay in a particular state for more than T times, where T can vary between 1 and 15 samples (assuming a 30-Hz sampling rate). The introduction of the smoothing factor T will cause a delay in the decision of the system, but of no more than T sample times.

8 Experiments

In order to test the algorithms described in the previous sections, we collected data of people that are instructed to display facial expressions corresponding to the six types of emotions. The data collection method is described in detail in [4]. All the tests of the algorithms are performed on a set of five people, each one displaying six sequences of each one of the six emotions, and always coming back to a neutral state between each emotion sequence. The video was used as the input to the face tracking algorithm described in Section 3. The sampling rate was 30 Hz, and a typical emotion sequence is about 70 samples long ($\sim$ 2 s). Figure 6 shows one frame of each emotion for three subjects.

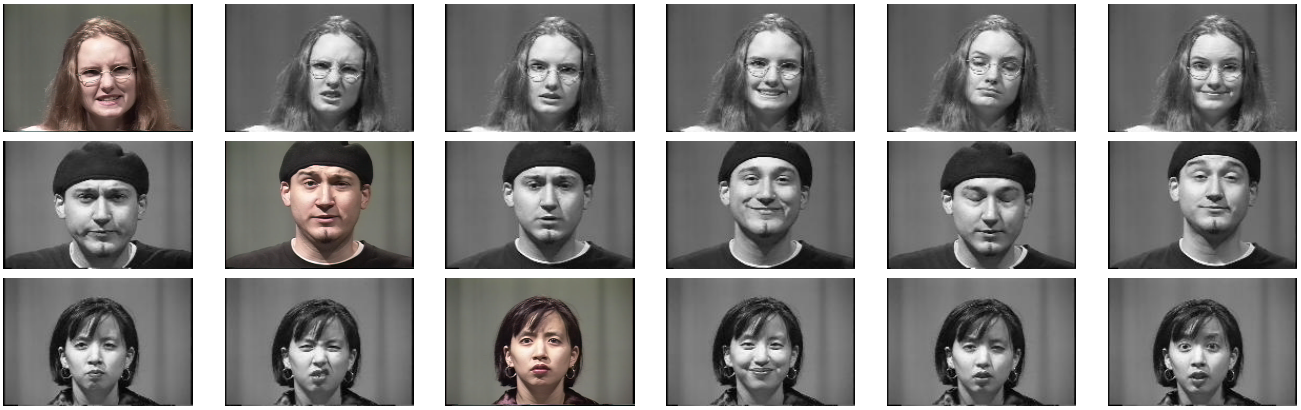
The data was collected in an open recording scenario, where the person was asked to display the expression corresponding to the emotion being induced. This is of course not the ideal way of collecting emotion data. The ideal way would be using a hidden recording, inducing the emotion through events in the normal environment of the subject, not in a studio. The main problem with collecting the data this way is the impracticality of it and the ethical issue of hidden recording.

We used the database described above to test our algorithms. We performed two types of experiments. First we performed person dependent experiments, in which part of the data for each subject was used as training data, and another part as test data. Second, we performed person independent experiments, in which we used the data of all but one person as training data, and tested on the person that was left out.

For the SNoW and SNoW-NB classifiers we report the results obtained after discretizing the features to 100 bins using uniform discretization. Results using other parameters are reported in [4]. For the TAN classifiers we used the dependencies shown in Figure 4, learned using the algorithm described in Section 6. For the HMM-based models, several states were tried (3-12) and both the ergodic and left-to-right with return were tested. The results presented below are of the best configuration (an ergodic model using 11 states).

8.1 Person-Dependent Tests

A person-dependent test is first tried. For the frame based methods (SNoW, SNoW-NB, NB-Gaussian, NB-Cauchy, and TAN), we measure the accuracy with respect to the classification result of each frame. The accuracy for the temporal based methods is measured with respect to the misclassification rate of an expression sequence, not with respect to



(a) Anger (b) Disgust (c) Fear (d) Happiness (e) Sadness (f) Surprise

Figure 6. Examples of images from the video sequences used in the experiment.

each frame. Tables 2 and 3 show the recognition rate of each subject and the average recognition rate of the classifiers.

It can be seen that the discrete SNoW-NB outperforms all of the classifiers. It is also worth noting that the results for subject 5 are consistently worse for all classifiers. The fact that subject 5 was poorly classified can be attributed to the inaccurate tracking result and lack of sufficient variability in displaying the emotions. It can also be seen that the multi-level HMM achieves similar recognition rate (and improves it in some cases) compared to the emotion-specific HMM, even though the input is unsegmented continuous video.

The NB-Cauchy assumption does not give a significant improvement in recognition rate comparing with the NB-Gaussian assumption mainly due to the fact that in this case there are not many outliers in the data (each person was displaying the emotion sequences in the same environment). This may not be the case in a natural setting experiment. It is also important to observe that taking into account the dependencies in the features (the TAN model) gives significantly improved results.

In average the best results are obtained by using the SNoW-NB classifier, followed by the TAN, NB-Cauchy, NB-Gaussian, and SNoW classifiers.

The confusion matrices for the NB-Cauchy and the TAN classifiers are presented in Table 4 and Table 5. The analysis of the confusion between different emotions shows that Happy and Surprise are well recognized. The other more subtle emotions are confused with each other more frequently, with Sad being the most confused emotion. The confusion matrices for the HMM based classifiers (described in details in [6]) show similar results, with *happiness* achieving near 100%, and *surprise* approximately 90%. These observations suggest that we can see the facial expression recognition problem from a slightly different perspective. Suppose that now we only want to detect whether the person is in a good mood, bad mood, or is just surprised (this is separated since it can belong to both positive and negative facial expressions). This means that we consider now only 4 classes in the classification: Neutral, Positive, Negative, and Surprise. Anger, Disgust, Fear, and Sad will count for the Negative class while Happy will count for the Positive class.

The confusion matrix obtained in this case for the NB-Cauchy classifier is presented in Table 6. The system can tell now with 88-89% accuracy if a person displays a negative or a positive facial expression.

8.2 Person-Independent Tests

In the previous section it was seen that a good recognition rate was achieved when the training sequences were taken from the same subject as the test sequences. A more challenging application is to create a system which is person-independent. In this case the variation of the data is more significant and we expect that using a Cauchy-based classifier we will obtain significantly better results.

For this test all of the sequences of one subject are used as the test sequences and the sequences of the remaining four subjects are used as training sequences. This test is repeated five times, each time leaving a different person out (leave one out cross validation). Table 7 shows the recognition rate of the test for all classifiers. In this case the recognition rates are lower compared with the person-dependent results. This means that the confusions between subjects are larger than those within the same subject.

In this case the TAN classifier provides the best results. It is important to observe that the Cauchy assumption also yields an improvement compared to the other classifiers, due to the capability of the Cauchy distribution to handle outliers. One of the reasons for the misclassifications is the fact that the subjects are very different from each other (three females, two males, and different ethnic backgrounds); hence, they display their emotion differently. Although it appears to contradict the universality of the facial expressions as studied by Ekman and Friesen [12], the results show that for practical automatic emotion recognition, consideration of gender and race play a role in the training of the system.

Table 8 and Table 9 show the confusion matrices for the NB-Cauchy and the TAN classifiers. Again we see that surprise and happy are detected with high accuracy, and other expressions are greatly confused.

If we now consider the problem where only the person mood is important, the classification rates are significantly higher. The confusion matrix obtained for the NB-Cauchy classifier is presented in Table 10.

Now the recognition rates are much higher. The system can tell now with about 80% accuracy if a person displays a negative or a positive facial expression.

9 Discussion

In this work we presented several methods for expression recognition from video.

We showed frame by frame based classifiers, the SNoW classifer, several Naive Bayes classifiers, different by the distribution assumptions on the features, and a TAN classifier that takes into account the dependencies between the features. For continuous features we successfully used the Cauchy distribution assumption and provided an algorithm to test whether the Cauchy assumption is better than the Gaussian assumption. We performed person-dependent and person-independent experiments and we showed that the Cauchy distribution assumption provides better results than the Gaussian distribution assumption. We also showed that incorporating the dependencies between the features provides significantly improved results. Moreover, we showed that when the facial expression recognition problem is reduced to a mood recognition problem the classification results are significantly higher.

We introduced the multi-level HMM architecture and compared it to the straight forward Emotion-specific HMM. We showed that comparable results can be achieved with this architecture, although it does not rely on any pre-segmentation of the video stream.

One of the main drawbacks in all of the works done on emotion recognition from facial expression videos is the lack of a benchmark database to test different algorithms. This work relied on a database collected by Chen [4], but it is difficult to compare the results to other works using different databases. The recently constructed database by Kanade et al [23] will be a useful tool for testing these algorithms.

Are the recognition rates sufficient for real world use? We think that it depends upon the particular application. In the case of image and video retrieval from large databases, the current recognition rates could aid in finding the right image or video by giving additional options for the queries. For future research, the integration of multiple modalities such as voice analysis and context would be expected to improve the recognition rates and eventually improve the computer's understanding of human emotional states. Voice and gestures are widely believed to play an important role as well [4, 8], and physiological states such as heart beat and skin conductivity are being suggested [2]. People also use context as an indicator of the emotional state of a person. This work is just another step on the way toward achieving the goal of building more effective computers that can serve us better.

References

[1] M. J. Black and Y. Yacoob. Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion. In *Proc. International Conf. Computer Vision*, pages 374–381, Cambridge, USA, 1995.

[2] J. T. Cacioppo and L.G. Tassinary. Inferring psychological significance from physiological signals. *American Psychologist*, 45:16–28, January 1990.

[3] W. B. Cannon. The James-Lange theory of emotion: A critical examination and an alternative theory. *American Journal of Psychology*, 39:106–124, 1927.

[4] L. S. Chen. *Joint processing of audio-visual information for the recognition of emotional expressions in human-computer interaction*. PhD thesis, University of Illinois at Urbana-Champaign, Dept. of Electrical Engineering, 2000.

[5] C.K. Chow and C.N. Liu. Approximating discrete probability distribution with dependence trees. *IEEE Transactions on Information Theory*, 14:462–467, 1968.

[6] I. Cohen. Automatic facial expression recognition from video sequences using temporal information. In *MS Thesis*, University of Illinois at Urbana-Champaign, Dept. of Electrical Engineering, 2000.

[7] C. Darwin. *The Expression of the Emotions in Man and Animals*. John Murray, London, 2nd edition, 1890.

[8] L. C. De Silva, T. Miyasato, and R. Natatsu. Facial emotion recognition using multimodal information. In *Proc. IEEE Int. Conf. on Information, Communications and Signal Processing (ICICS'97)*, pages 397–401, Singapore, Sept. 1997.

[9] G. Donato, M.S. Bartlett, J.C. Hager, P. Ekman, and T.J. Sejnowski. Classifying facial actions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(10):974–989, 1999.

[10] P. Ekman, editor. *Emotion In the Human Face*. Cambridge University Press, New York, NY, 2nd edition, 1982.

[11] P. Ekman. Strong evidence for universals in facial expressions: A reply to Russell's mistaken critique. *Psychological Bulletin*, 115(2):268–287, 1994.

[12] P. Ekman and W. V. Friesen. *Facial Action Coding System: Investigator's Guide*. Consulting Psychologists Press, Palo Alto, CA, 1978.

[13] I. A. Essa and A. P. Pentland. Coding, analysis, interpretation, and recognition of facial expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):757–763, July 1997.

[14] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29(2):131–163, 1997.

[15] A. Garg, V. Pavlovic, J. Rehg, and T. S. Huang. Audio–visual speaker detection using dynamic Bayesian networks. In *Proc. of 4rd Intl Conf. Automatic Face and Gesture Rec.*, pages 374–471, 2000.

[16] A. Garg and D. Roth. Understanding probabilistic classifiers. In *European Conference on Machine Learning*, Sep 2001.

[17] D. Goleman. *Emotional Intelligence*. Bantam Books, New York, 1995.

[18] G. Haas, L. Bain, and C. Antle. Inferences for the cauchy distribution based on maximum likelihood estimators. *Biometrika*, 57(2):403–408, 1970.

[19] E. Hilgard, R. C. Atkinson, and R. L. Hilgard. *Introduction to Psychology*. Harcourt Brace Jovanovich, New York, NY, 5th edition, 1971.

[20] C. E. Izard. Innate and universal facial expressions: Evidence from developmental and cross-cultural research. *Psychological Bulletin*, 115(2):288–299, 1994.

[21] W. James. *The Principles of Psychology*. Henry Holt, New York, NY, 1890.

[22] J. M. Jenkins, K. Oatley, and N. L. Stein, editors. *Human Emotions: A Reader*. Blackwell Publishers, Malden, MA, 1998.

[23] T. Kanade, J.F. Cohn, and Y. Tian. Comprehesive database for facial expression analysis. In *Proc. of 4rd Intl Conf. Automatic Face and Gesture Rec.*, pages 46–53, 2000.

[24] P. Lang. The emotion probe: Studies of motivation and attention. *American Psychologist*, 50(5):372–385, May 1995.

[25] A. Lanitis, C. J. Taylor, and T. F. Cootes. A unified approach to coding and interpreting face images. In *Proc. 5th International Conference on Computer Vision (ICCV)*, pages 368–373, Cambridge, MA, USA, 1995.

[26] S.E. Levinson, L.R. Rabiner, and M.M. Sondhi. An introduction to the application of the theory of probabilitic functions of a markov process to automatic speech recognition. *The Bell Lab System Technical Journal*, 62(4):1035–1072, apr 1983.

[27] J. Lien. *Automatic recognition of facial expressions using hidden Markov models and estimation of expression intensity*. PhD thesis, Carnegie Mellon University, 1998.

[28] K. Mase. Recognition of facial expression from optical flow. *IEICE Transactions*, E74(10):3474–3483, October 1991.

[29] D. Matsumoto. Cultural influences on judgments of facial expressions of emotion. In *Proc. 5th ATR Symposium on Face and Object Recognition*, pages 13–15, Kyoto, Japan, April 1998.

[30] S. Morishima. Emotion model–a criterion for recognition, synthesis and compression of face and emotion. In *Proc. Automatic Face and Gesture Recognition*, pages 284–289, 1995.

[31] T. Otsuka and J. Ohya. Recognizing multiple persons' facial expressions using HMM based on automatic extraction of significant frames from image sequences. In *Proc. Int. Conf. on Image Processing (ICIP-97)*, pages 546–549, Santa Barbara, CA, USA, Oct. 26-29, 1997.

[32] T. Otsuka and J. Ohya. A study of transformation of facial expressions based on expression recognition from temproal image sequences. Technical report, Institute of Electronic, Information, and Communications Engineers (IEICE), 1997.

[33] L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech processing. *Proceedings of IEEE*, 77(2):257–286, 1989.

[34] M. Rosenblum, Y. Yacoob, and L.S. Davis. Human expression recognition from motion using a radial basis function network architecture. *IEEE Transactions on Neural Network*, 7(5):1121–1138, September 1996.

[35] D. Roth. Learning to resolve natural language ambiguities: A unified approach. In *National Conference on Artifical Intelligence*, pages 806–813, Madison, WI, USA, 1998.

[36] D. Roth, M.-H. Yang, and N. Ahuja. A SNoW-based face detector. In *Advances in Neural Information Processing Systems 12 (NIPS 12)*, Cambridge, MA, 2000. MIT Press.

[37] T. Sakaguchi. Facial feature extraction based on the wavelet transform for dynamic expression recognition. submitted to *IEEE Trans. PAMI*.

[38] P. Salovey and J.D. Mayer. Emotional intelligence. *Imagination, Cognition and Personality*, 9(3):185–211, 1990.

[39] H. Schlosberg. Three dimensions of emotion. *Psychological Review*, 61:81–88, 1954.

[40] N. Sebe, M.S. Lew, and D.P. Huijsmans. Toward improved ranking metrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1132–1143, 2000.

[41] H. Tao and T. S. Huang. Connected vibrations: A modal analysis approach to non-rigid motion tracking. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition 1998 (CVPR'98)*, Santa Barbara, CA, USA, June 23-25, 1998.

[42] N. Ueki, S. Morishima, H. Yamada, and H. Harashima. Expression analysis/synthesis system based on emotion space constructed by multilayered neural network. *Systems and Computers in Japan*, 25(13):95–103, Nov. 1994.

[43] Y. Yacoob and L.S. Davis. Recognizing human facial expressions from long image sequences using optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(6):636–642, June 1996.

AU	Description
1	vertical movement of the center of upper lip
2	vertical movement of the center of lower lip
3	horizontal movement of left mouth corner
4	vertical movement of left mouth corner
5	horizontal movement of right mouth corner
6	vertical movement of right mouth corner
7	vertical movement of right brow
8	vertical movement of left brow
9	lifting of right cheek
10	lifting of left cheek
11	blinking of right eye
12	blinking of left eye

Table 1. Action units used in our face tracker.

Subject	SNoW	SNoW-NB	NB-Gaussian	NB-Cauchy	TAN
1	83.43%	88.15%	80.97%	81.69%	85.94%
2	77.11%	85.98%	87.09%	84.54%	89.39%
3	82.76%	87.96%	82.5%	83.05%	86.58%
4	76.63%	87.91%	77.18%	79.25%	82.84%
5	71.74%	82.29%	69.06%	71.74%	71.78%
Average	78.53%	86.45%	79.36%	80.05%	83.31%

Table 2. Person-dependent facial expression recognition accuracies using frame based methods.

Subject	Single HMM	Multilevel HMM
1	82.86%	80%
2	91.43%	85.71%
3	80.56%	80.56%
4	83.33%	88.89%
5	54.29%	77.14%
Average	78.49%	82.46%

Table 3. Person-dependent facial expression recognition rates using the emotion-specific HMM and multilevel HMM.

Emotion	Neutral	Happy	Anger	Disgust	Fear	Sad	Surprise
Neutral	74.52	0.48	5.04	3.11	6.19	6.44	4.18
Happy	2.77	87.16	0.83	1.87	1.06	2.19	4.08
Anger	11.3	2.27	74.81	6.03	2.48	2.05	1.02
Disgust	0.92	0	2.73	86.39	2.66	4.03	3.23
Fear	5.51	0	2.96	8.36	77.09	2.43	3.61
Sad	13.59	0.19	2.18	5.61	2.10	74.45	1.84
Surprise	4.39	0	0	0.47	5.14	2.92	87.06

Table 4. Person-dependent confusion matrix using the NB-Cauchy classifier

Emotion	Neutral	Happy	Anger	Disgust	Fear	Sad	Surprise
Neutral	79.58	1.21	3.88	2.71	3.68	5.61	3.29
Happy	1.06	87.55	0.71	3.99	2.21	1.71	2.74
Anger	5.18	0	85.92	4.14	3.27	1.17	0.30
Disgust	2.48	0.19	1.50	83.23	3.68	7.13	1.77
Fear	4.66	0	4.21	2.28	83.68	2.13	3.00
Sad	13.61	0.23	1.85	2.61	0.70	80.97	0
Surprise	5.17	0.80	0.52	2.45	7.73	1.08	82.22

Table 5. Person-dependent confusion matrix using the TAN classifier

Emotion	Neutral	Positive	Negative	Surprise
Neutral	74.42	0.48	20.89	4.18
Positive	2.77	88.16	4.97	4.08
Negative	7.83	0.61	89.11	2.43
Surprise	5.39	0	8.54	86.06

Table 6. Person-dependent average confusion matrix using the Cauchy assumption

	SNoW	SNow-NB	NB-Gaussian	NB-Cauchy	TAN	Single HMM	Multilevel HMM
Recognition rate	57.69%	61.31%	58.94%	63.58%	65.11%	55%	58%

Table 7. Recognition rate for person-independent test.

Emotion	Neutral	Happy	Anger	Disgust	Fear	Sad	Surprise
Neutral	71.30	0.64	3.75	4.06	8.29	10.62	1.31
Happy	5.45	81.16	1.41	8.13	0.15	2.27	1.40
Anger	11.19	2.64	59.27	14.87	0.86	11.14	0
Disgust	5.67	9.94	2.73	50.2	6.48	8.88	6.03
Fear	8.99	0	2.34	1.36	75.53	2.40	9.35
Sad	10.00	10.39	5.14	8.25	17.37	39.41	9.41
Surprise	10.81	8.79	0.98	2.35	4.49	4.40	68.15

Table 8. Person-independent average confusion matrix using the NB-Cauchy classifier

Emotion	Neutral	Happy	Anger	Disgust	Fear	Sad	Surprise
Neutral	76.95	0.46	3.39	3.78	7.35	6.53	1.50
Happy	3.21	77.34	2.77	9.94	0	2.75	3.97
Anger	14.33	0.89	62.98	10.60	1.51	9.51	0.14
Disgust	6.63	8.99	7.44	52.48	2.20	10.90	11.32
Fear	10.06	0	3.53	0.52	73.67	3.41	8.77
Sad	13.98	7.93	5.47	10.66	13.98	41.26	6.69
Surprise	4.97	6.83	0.32	7.41	3.95	5.38	71.11

Table 9. Person-independent average confusion matrix using the TAN classifier

Emotion	Neutral	Positive	Negative	Surprise
Neutral	71.30	0.64	26.73	1.31
Positive	5.45	81.16	11.97	1.40
Negative	8.96	5.74	79.08	6.2
Surprise	10.81	4.4	12.23	68.15

Table 10. Person-independent average confusion matrix using the NB-Cauchy classifier.