# Face Recognition with Support Vector Machines:
# Global versus Component-based Approach

Bernd Heisele[†]   Purdy Ho[‡]   Tomaso Poggio
Massachusetts Institute of Technology
Center for Biological and Computational Learning
Cambridge, MA 02142
heisele@ai.mit.edu   purdyho@mit.edu   tp@ai.mit.edu

## Abstract

*We present a component-based method and two global methods for face recognition and evaluate them with respect to robustness against pose changes. In the component system we first locate facial components, extract them and combine them into a single feature vector which is classified by a Support Vector Machine (SVM). The two global systems recognize faces by classifying a single feature vector consisting of the gray values of the whole face image. In the first global system we trained a single SVM classifier for each person in the database. The second system consists of sets of viewpoint-specific SVM classifiers and involves clustering during training. We performed extensive tests on a database which included faces rotated up to about $40°$ in depth. The component system clearly outperformed both global systems on all tests.*

## 1. Introduction

Over the past 20 years numerous face recognition papers have been published in the computer vision community; a survey can be found in [4]. The number of real-world applications (e.g. surveillance, secure access, human/computer interface) and the availability of cheap and powerful hardware also lead to the development of commercial face recognition systems. Despite the success of some of these systems in constrained scenarios, the general task of face recognition still poses a number of challenges with respect to changes in illumination, facial expression, and pose.

In the following we give a brief overview on face recognition methods. Focusing on the aspect of pose invariance we divide face recognition techniques into two categories: i) global approach and ii) component-based approach.

i) In this category a single feature vector that represents the whole face image is used as input to a classifier. Several classifiers have been proposed in the literature e.g. minimum distance classification in the eigenspace [18, 20], Fisher's discriminant analysis [1], and neural networks [6]. Global techniques work well for classifying frontal views of faces. However, they are not robust against pose changes since global features are highly sensitive to translation and rotation of the face. To avoid this problem an alignment stage can be added before classifying the face. Aligning an input face image with a reference face image requires computing correspondences between the two face images. The correspondences are usually determined for a small number of prominent points in the face like the center of the eye, the nostrils, or the corners of the mouth. Based on these correspondences the input face image can be warped to a reference face image. In [12] an affine transformation is computed to perform the warping. Active shape models are used in [10] to align input faces with model faces. A semi automatic alignment step in combination with SVM classification was proposed in [9].

ii) An alternative to the global approaches is to classify local facial components. The main idea of component-based recognition is to compensate for pose changes by allowing a flexible geometrical relation between the components in the classification stage. In [3] face recognition was performed by independently matching templates of three facial regions (both eyes, nose and mouth). The configuration of the components during classification was unconstrained since the system did not include a geometrical model of the face. A similar approach with an additional alignment stage was proposed in [2]. In [23] a geometrical model of a face was implemented by a 2-D elastic graph. The recognition was based on wavelet coefficients that were computed on the nodes of the elastic graph. In [14] a window was shifted

---

[†] with Honda Research Laboratory, Boston, MA, from April 2001
[‡] with Hewlett-Packard, Palo Alto, CA, from July 2001

over the face image and the DCT coefficients computed within the window were fed into a 2-D Hidden Markov Model.

We present two global approaches and a component-based approach to face recognition and evaluate their robustness against pose changes. The first global method consists of a face detector which extracts the face part from an image and propagates it to a set of SVM classifiers that perform the face recognition. By using a face detector we achieve translation and scale invariance. In the second global method we split the images of each person into viewpoint-specific clusters. We then train SVM classifiers on each single cluster. In contrast to the global methods, the component system uses a face detector that detects and extracts local components of the face. The detector consists of a set of SVM classifiers that locate facial components and a single geometrical classifier that checks if the configuration of the components matches a learned geometrical face model. The detected components are extracted from the image, normalized in size and fed into a set of SVM classifiers.

The outline of the paper is as follows: Chapter 2 gives a brief overview on SVM learning and on strategies for multi-class classification with SVMs. In Chapter 3 we describe the two global methods for face recognition. Chapter 4 is about the component-based system. Chapter 5 contains experimental results and a comparison between the global and component systems. Chapter 6 concludes the paper.

## 2. Support Vector Machine Classification

We first explain the basics of SVMs for binary classification [21]. Then we discuss how this technique can be extended to deal with general multi-class classification problems.

### 2.1. Binary Classification

SVMs belong to the class of maximum margin classifiers. They perform pattern recognition between two classes by finding a decision surface that has maximum distance to the closest points in the training set which are termed support vectors. We start with a training set of points $\mathbf{x}_i \in \mathbb{R}^n$, $i = 1, 2, \ldots, N$ where each point $\mathbf{x}_i$ belongs to one of two classes identified by the label $y_i \in \{-1, 1\}$. Assuming linearly separable data[1], the goal of maximum margin classification is to separate the two classes by a hyperplane such that the distance to the support vectors is maximized. This hyperplane is called the optimal separating hyperplane (OSH). The OSH has the form:

$$f(\mathbf{x}) = \sum_{i=1}^{\ell} \alpha_i y_i \mathbf{x}_i \cdot \mathbf{x} + b, \qquad (1)$$

[1]For the non-separable case the reader is referred to [21].

The coefficients $\alpha_i$ and the $b$ in Eq. (1) are the solutions of a quadratic programming problem [21]. Classification of a new data point $\mathbf{x}$ is performed by computing the sign of the right side of Eq. (1). In the following we will use

$$d(\mathbf{x}) = \frac{\sum_{i=1}^{\ell} \alpha_i y_i \mathbf{x}_i \cdot \mathbf{x} + b}{\| \sum_{i=1}^{\ell} \alpha_i y_i \mathbf{x}_i \|} \qquad (2)$$

to perform multi-class classification. The sign of $d$ is the classification result for $\mathbf{x}$, and $|d|$ is the distance from $\mathbf{x}$ to the hyperplane. Intuitively, the farther away a point is from the decision surface, i.e. the larger $|d|$, the more reliable the classification result.

The entire construction can be extended to the case of nonlinear separating surfaces. Each point $\mathbf{x}$ in the input space is mapped to a point $\mathbf{z} = \Phi(\mathbf{x})$ of a higher dimensional space, called the feature space, where the data are separated by a hyperplane. The key property in this construction is that the mapping $\Phi(\cdot)$ is subject to the condition that the dot product of two points in the feature space $\Phi(\mathbf{x}) \cdot \Phi(\mathbf{y})$ can be rewritten as a kernel function $K(\mathbf{x}, \mathbf{y})$. The decision surface has the equation:

$$f(\mathbf{x}) = \sum_{i=1}^{\ell} y_i \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b,$$

again, the coefficients $\alpha_i$ and $b$ are the solutions of a quadratic programming problem. Note that $f(\mathbf{x})$ does not depend on the dimensionality of the feature space.

An important family of kernel functions is the polynomial kernel:

$$K(\mathbf{x}, \mathbf{y}) = (1 + \mathbf{x} \cdot \mathbf{y})^d,$$

where $d$ is the degree of the polynomial. In this case the components of the mapping $\Phi(\mathbf{x})$ are all the possible monomials of input components up to the degree $d$.

### 2.2. Multi-class classification

There are two basic strategies for solving $q$-class problems with SVMs:

i) In the one-vs-all approach $q$ SVMs are trained. Each of the SVMs separates a single class from all remaining classes [5, 17].

ii) In the pairwise approach $q(q-1)/2$ machines are trained. Each SVM separates a pair of classes. The pairwise classifiers are arranged in trees, where each tree node represents an SVM. A bottom-up tree similar to the elimination tree used in tennis tournaments was originally proposed in [16] for recognition of 3-D objects and was applied to face recognition in [7]. A top-down tree structure has been recently published in [15].

There is no theoretical analysis of the two strategies with respect to classification performance. Regarding the training effort, the one-vs-all approach is preferable since only

$q$ SVMs have to be trained compared to $q(q-1)/2$ SVMs in the pairwise approach. The run-time complexity of the two strategies is similar: The one-vs-all approach requires the evaluation of $q$, the pairwise approach the evaluation of $q-1$ SVMs. Recent experiments on person recognition show similar classification performances for the two strategies [13]. Since the number of classes in face recognition can be rather large we opted for the one-vs-all strategy where the number of SVMs is linear with the number of classes.

## 3. Global Approach

Both global systems described in this paper consist of a face detection stage where the face is detected and extracted from an input image and a recognition stage where the person's identity is established.

### 3.1. Face detection

We developed a face detector similar to the one described in [8]. In order to detect faces at different scales we first computed a resolution pyramid for the input image and then shifted a $58 \times 58$ window over each image in the pyramid. We applied two preprocessing steps to the gray images to compensate for certain sources of image variations [19]. A best-fit intensity plane was subtracted from the gray values to compensate for cast shadows. Then histogram equalization was applied to remove variations in the image brightness and contrast. The resulting gray values were normalized to be in a range between 0 and 1 and were used as input features to a linear SVM classifier. Some detection results are shown in Fig. 1.

The training data for the face detector were generated by rendering seven textured 3-D head models [22]. The heads were rotated between $-30°$ and $30°$ in depth and illuminated by ambient light and a single directional light pointing towards the center of the face. We generated 3,590 face images of size $58 \times 58$ pixels. The negative training set initially consisted of 10,209 $58 \times 58$ non-face patterns randomly extracted from 502 non-face images. We expanded the training set by bootstrapping [19] to 13,655 non-face patterns.

### 3.2. Recognition

We implemented two global recognition systems. Both systems were based on the one-vs-all strategy for SVM multi-class classification described in the previous Chapter.

The first system had a linear SVM for every person in the database. Each SVM was trained to distinguish between all images of a single person (labeled $+1$) and all other images



**Figure 1. The upper two rows are example images from our training set. The lower two rows show the image parts extracted by the SVM face detector.**

in the training set (labeled $-1$). For both training and testing we ran the face detector on the input image to extract the face. We re-saled the face image to $40 \times 40$ pixels and converted the gray values into a feature vector[2]. Given a set of $q$ people and a set of $q$ SVMs, each one associated to one person, the class label $y$ of a face pattern $\mathbf{x}$ is computed as follows:

$$y = \begin{cases} n & \text{if} \quad d_n(\mathbf{x}) + t > 0 \\ 0 & \text{if} \quad d_n(\mathbf{x}) + t \leq 0 \end{cases} \quad (3)$$

with $d_n(\mathbf{x}) = \max \{d_i(\mathbf{x})\}_{i=1}^{q}$.

where $d_i(\mathbf{x})$ is computed according to Eq. (2) for the SVM trained to recognize person $i$. The classification threshold is denoted as $t$. The class label $0$ stands for rejection.

Changes in the head pose lead to strong variations in the images of a person's face. These in-class variations complicate the recognition task. That is why we developed a second method in which we split the training images of each person into clusters by a divisive cluster technique [11]. The algorithm started with an initial cluster including all face images of a person after preprocessing. The cluster with the highest variance is split into two by a hyperplane. The variance of a cluster is calculated as:

$$\sigma^2 = \min \left\{ \frac{1}{N} \cdot \sum_{m=1}^{N} \|\mathbf{x}_n - \mathbf{x}_m\|^2 \right\}_{n=1}^{N}$$

---

[2]We applied the same preprocessing steps to generate the features as for the face detector described.

where $N$ is the number of faces in the cluster. After the partitioning has been performed, the face with the minimum distance to all other faces in the same cluster is chosen to be the average face of the cluster. Iterative clustering stops when a maximum number of clusters is reached[3]. The average faces can be arranged in a binary tree. Fig. 2 shows the result of clustering applied to the training images of a person in our database. The nodes represent the average faces; the leaves of the tree are some example faces of the final clusters. As expected divisive clustering performs a viewpoint-specific grouping of faces.

We trained a linear SVM to distinguish between all images in one cluster (labeled $+1$) and all images of other people in the training set (labeled $-1$)[4]. Classification was done according to Eq. (3) with $q$ now being the number of clusters of all people in the training set.
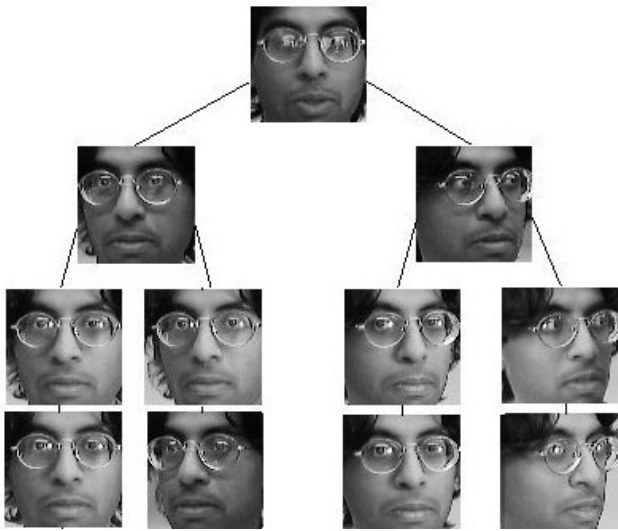


**Figure 2. Binary tree of face images generated by divisive clustering.**

## 4. Component-based Approach

The global approach is highly sensitive to image variations caused by changes in the pose of the face. The component-based approach avoids this problem by independently detecting parts of the face. For small rotations, the changes in the components are relatively small compared to the changes in the whole face pattern. Changes in the 2-D locations of the components due to pose changes are accounted for by a learned, flexible face model.

[3]In our experiments we divided the face images of a person into four clusters.

[4]This is not exactly a one-vs-all classifier since images of the same person but from different clusters were omitted.

### 4.1. Detection

We implemented a two-level component-based face detector which is described in detail in [8]. The principles of the system are illustrated in Fig. 3. On the first level, component classifiers independently detected facial components. On the second level, a geometrical configuration classifier performed the final face detection by combining the results of the component classifiers. Given a $58 \times 58$ window, the maximum continuous outputs of the component classifiers within rectangular search regions around the expected positions of the components were used as inputs to the geometrical configuration classifier. The search regions have been calculated from the mean and standard deviation of the components' locations in the training images. We also provided the geometrical classifier with the precise positions of the detected components relative to the upper left corner of the $58 \times 58$ window. The 14 facial components used in the detection system are shown in Fig. 4 (a). The shapes and positions of the components have been automatically determined from the training data in order to provide maximum discrimination between face and non-face images; see [8] for details about the algorithm. The training set was the same as for the whole face detector described in the previous Chapter.
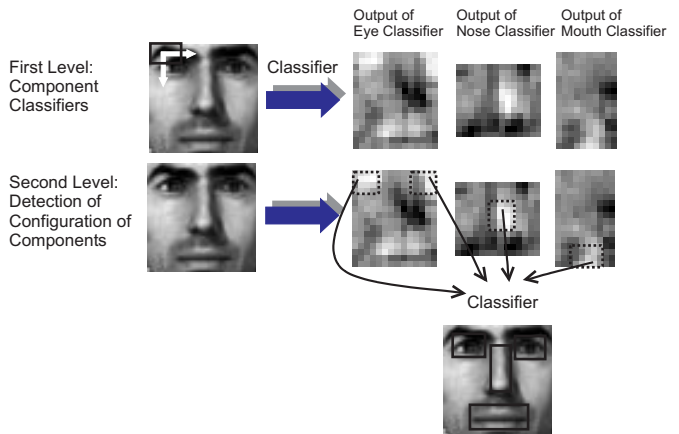


**Figure 3. System overview of the component-based face detector using four components. On the first level, windows of the size of the components (solid lined boxes) are shifted over the face image and classified by the component classifiers. On the second level, the maximum outputs of the component classifiers within predefined search regions (dotted lined boxes) and the positions of the detected components are fed into the geometrical configuration classifier.**
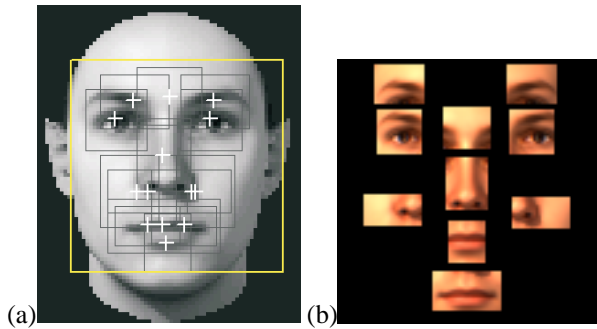
**Figure 4. (a) shows the 14 components of our face detector. The centers of the components are marked by a white cross. The 10 components that were used for face recognition are shown in (b).**

## 4.2. Recognition

To train the face recognizer we first ran the component-based detector over each image in the training set and extracted the components. From the 14 original we kept 10 for face recognition, removing those that either contained few gray value structures (e.g. cheeks) or strongly overlapped with other components. The 10 selected components are shown in Fig. 4 (b). Examples of the component-based face detector applied to images of the training set are shown in Fig. 5. To generate the input to our face recognition classifier we normalized each of the components in size and combined their gray values into a single feature vector [5]. As for the first global system we used a one-vs-all approach with a linear SVM for every person in the database. The classification result was determined according to Eq. (3).
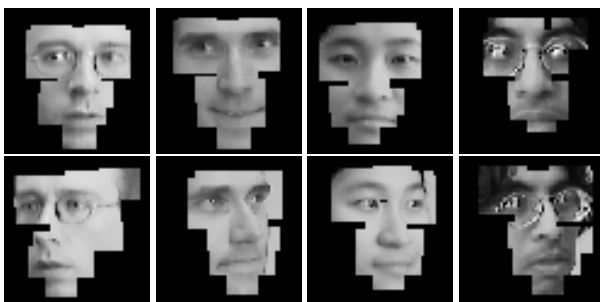


**Figure 5. Examples of component-based face detection. Shown are face parts covered by the 10 components that were used for face recognition.**

---

[5]Before extracting the components we applied the same preprocessing steps to the detected $40 \times 40$ face image as in the global systems.

## 5. Experiments

The training data for the face recognition system were recorded with a digital video camera at a frame rate of about 5 Hz. The training set consisted of 8,593 gray face images of five subjects from which 1,383 were frontal views. The resolution of the face images ranged between $80 \times 80$ and $130 \times 130$ pixels with rotations in azimuth up to about $\pm 40^\circ$. The test set was recorded with the same camera but on a separate day and under different illumination and with different background. The set included 974 images of all five subjects in our database. The rotations in depth was again up to about $\pm 40^\circ$.

Two experiments were carried out. In the first experiment we trained on all 8,593 rotated and frontal face images in the training set and tested on the whole test set. This experiment contained four different tests: Global approach using one linear SVM classifier for each person, using one linear SVM classifier for each cluster, using one second degree polynomial SVM classifier for each person, and component-based approach using one linear SVM classifier for each person.
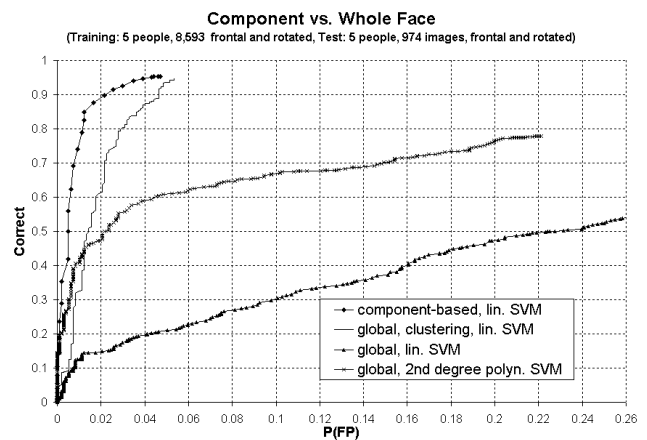


**Figure 6. ROC curves when trained and tested on frontal and rotated faces.**

In the second experiment we trained only on the 1,383 frontal face images in the training set but tested on the whole test set. This experiment contained three different tests: Global approach using one linear SVM classifier for each person, using one linear SVM classifier for each cluster, and component-based approach using one linear SVM classifier for each person.

The ROC curves of these two experiments are shown in Fig. 6 and Fig. 7, respectively. Each point on the ROC curve corresponds to a different value of the classification threshold $t$ from Eq. (3). At the end points of the ROC curves
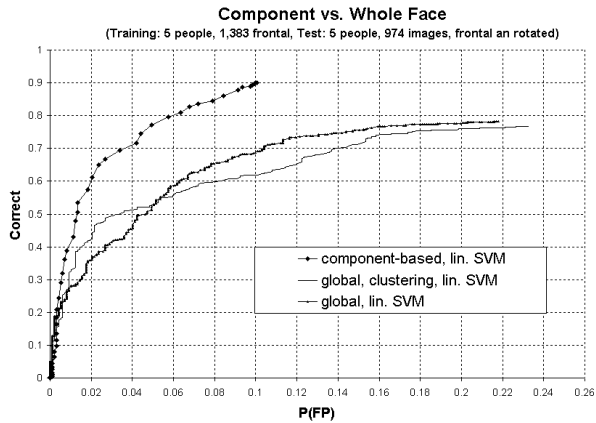
**Figure 7. ROC curves when trained on frontal faces and tested on frontal and rotated faces.**



**Figure 8. Examples of component-based face recognition. The first 3 rows and the first image in the last row show correct identification. The last two images in the bottom row show misclassifications due to strong rotation and facial expression.**

the rejection rate is 0. Some results of the component-based recognition system are shown in Fig 8.

There are three interesting observations:

- In both experiments the component system clearly outperformed the global systems. This although the face classifier itself (5 linear SVMs) was less powerful than the classifiers used in the global methods (5 non-linear SVMs in the global method without clustering, and 20 linear SVMs in the method with clustering).

- Involving clustering lead to a significant improvement of the global method when the training set included rotated faces. This is because clustering generates viewpoint-specific clusters that have smaller in-class variations than the whole set of images of a person. The global method with clustering and linear SVMs was also superior to the global system without clustering and a non-linear SVM (see Fig. 6). This shows that a combination of weak classifiers trained on properly chosen subsets of the data can outperform a single, more powerful classifier trained on the whole data.

- Adding rotated faces to the training set improves the results of the global method with clustering and the component method. Surprisingly, the results for the global method without clustering got worse. This indicates that the problem of classifying faces of one person over a large range of views is too complex for a linear classifier. Indeed, the performance significantly improved when using non-linear SVMs with second-degree polynomial kernel.

## 6. Conclusion

We presented a component-based technique and two global techniques for face recognition and evaluated their performance with respect to robustness against pose changes. The component-based system detected and extracted a set of 10 facial components and arranged them in a single feature vector that was classified by linear SVMs. In both global systems we detected the whole face, extracted it from the image and used it as input to the classifiers. The first global system consisted of a single SVM for each person in the database. In the second system we clustered the database of each person and trained a set of view-specific SVM classifiers.

We tested the systems on a database which included faces rotated in depth up to about $40°$. In all experiments the component-based system outperformed the global systems even though we used more powerful classifiers (i.e. non-linear instead of linear SVMs) for the global system. This shows that using facial components instead of the whole face pattern as input features significantly simplifies the task of face recognition.

## Acknowledgements

## References

[1] P. Belhumeur, P. Hespanha, and D. Kriegman. Eigenfaces vs fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

[2] D. J. Beymer. Face recognition under varying pose. A.I. Memo 1461, Center for Biological and Computational Learning, M.I.T., Cambridge, MA, 1993.

[3] R. Brunelli and T. Poggio. Face recognition: Features versus templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):1042–1052, 1993.

[4] R. Chellapa, C. Wilson, and S. Sirohey. Human and machine recognition of faces: a survey. *Proceedings of the IEEE*, 83(5):705–741, 1995.

[5] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20:1–25, 1995.

[6] M. Fleming and G. Cottrell. Categorization of faces using unsupervised feature extraction. In *Proc. IEEE IJCNN International Joint Conference on Neural Networks*, pages 65–70, 90.

[7] G. Guodong, S. Li, and C. Kapluk. Face recognition by support vector machines. In *Proc. IEEE International Conference on Automatic Face and Gesture Recognition*, pages 196–201, 2000.

[8] B. Heisele, T. Poggio, and M. Pontil. Face detection in still gray images. AI Memo 1687, Center for Biological and Computational Learning, MIT, Cambridge, MA, 2000.

[9] K. Jonsson, J. Matas, J. Kittler, and Y. Li. Learning support vectors for face verification and recognition. In *Proc. IEEE International Conference on Automatic Face and Gesture Recognition*, pages 208–213, 2000.

[10] A. Lanitis, C. Taylor, and T. Cootes. Automatic interpretation and coding of face images using flexible models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):743–756, 1997.

[11] Y. Linde, A. Buzo, and R. Gray. An algorithm for vector quantizer design. *IEEE Transactions on Communications*, 28(1):84–95, 1980.

[12] B. Moghaddam, W. Wahid, and A. Pentland. Beyond eigenfaces: probabilistic matching for face recognition. In *Proc. IEEE International Conference on Automatic Face and Gesture Recognition*, pages 30–35, 1998.

[13] C. Nakajima, M. Pontil, B. Heisele, and T. Poggio. Person recognition in image sequences: The mit espresso machine system. *submitted to IEEE Transactions On Neural Networks*, 2000.

[14] A. Nefian and M. Hayes. An embedded hmm-based approach for face detection and recognition. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 6, pages 3553–3556, 1999.

[15] J. Platt, N. Cristianini, and J. Shawe-Taylor. Large margin dags for multiclass classification. *Advances in Neural Information Processing Systems*, 2000.

[16] M. Pontil and A. Verri. Support vector machines for 3-d object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 637–646, 1998.

[17] B. Schölkopf, C. Burges, and V. Vapnik. Extracting support data for a given task. In U. Fayyad and R. Uthurusamy, editors, *Proceedings of the First International Conference on Knowledge Discovery and Data Mining*, Menlo Park, CA, 1995. AAAI Press.

[18] L. Sirovitch and M. Kirby. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A*, 2:519–524, 1987.

[19] K.-K. Sung. *Learning and Example Selection for Object and Pattern Recognition*. PhD thesis, MIT, Artificial Intelligence Laboratory and Center for Biological and Computational Learning, Cambridge, MA, 1996.

[20] M. Turk and A. Pentland. Face recognition using eigenfaces. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–591, 1991.

[21] V. Vapnik. *Statistical learning theory*. John Wiley and Sons, New York, 1998.

[22] T. Vetter. Synthesis of novel views from a single face. *International Journal of Computer Vision*, 28(2):103–116, 1998.

[23] L. Wiskott, J.-M. Fellous, N. Krüger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):775–779, 1997.