Face Recognition Under Varying Pose[*]

David J. Beymer
MIT Artificial Intelligence Laboratory
Cambridge, MA 02139
email: beymer@ai.mit.edu

Abstract

Researchers in computer vision and pattern recognition have worked on automatic techniques for recognizing human faces for the last 20 years. While some systems, especially template-based ones, have been quite successful on expressionless, frontal views of faces with controlled lighting, not much work has taken face recognizers beyond these narrow imaging conditions. Our goal is to build a face recognizer that works under varying pose, the difficult part of which is to handle face rotations in depth. Building on successful template-based systems, our basic approach is to represent faces with templates from multiple model views that cover different poses from the viewing sphere. To recognize a novel view, the recognizer locates the eyes and nose features, uses these locations to geometrically register the input with model views, and then uses correlation on model templates to find the best match in the data base of people. Our system has achieved a recognition rate of 98% on a data base of 62 people containing 10 testing and 15 modeling views per person.

1 Introduction

Researchers in computer vision and pattern recognition have worked on automatic techniques for recognizing human faces for the last 20 years. The basic task, given as input the visual image of a face, is to compare the input face against models of faces stored in a library and report a match if one is found. The problem of locating the face – distinguishing it from a cluttered background – is usually avoided by imaging faces against a uniform background. The problem of face recognition has attracted researchers not only because faces represent a challenging class of naturally textured 3D objects, but because of the many applications of automatic face recognition, such as enhancing security systems or adding a recognition ability to HCI systems.

Face recognition is difficult for two major reasons. First, faces form a class of fairly similar objects; all faces consist of the same facial features in roughly the same geometrical configuration, which makes the recognition problem a fine discrimination task. The second source of difficulty lies in the wide variation in the appearance of a particular face due to changes in pose, lighting, and facial expression.

There is an abundance of existing work in face recognition, and the topic has seen renewed interest in the last few years. Most face recognition systems follow the same basic recognition technique. The recognizer scans through a library of known faces, comparing the input to each model face. This comparison uses a distance metric, such as a weighted Euclidean distance or correlation, in the space used for representing faces. The model yielding the smallest distance is reported as the identified person. In addition, some systems reject the input if the best match is not good enough.

As existing face recognition systems compare model and input faces using fairly standard distance metrics, the main factor that distinguishes different approaches is input representation. There are two main approaches to input representation, a geometrical approach that uses the spatial configuration of facial features, and a pictorial approach that uses an image-based representation. Feature-based systems ([14], [9], [6], and [8]) locate a set of facial features (e.g. corners of the eyes and mouth, sides of the face and nose, nostrils) and then capture the spatial configuration in feature vector whose dimensions typically include measurements like distances, angles, and curvatures. Pictorial approaches, representing faces by using filtered images of model faces, include template-based systems ([2], [6], [13], [7], and [5]), systems using principal components analysis to derive a pictorial "face space" ([15], [20], [1], [9]), and connectionist approaches ([16], [11], [10], [21], and [12]). [18] explores an interesting hybrid representation that combines the geometrical and pictorial approaches, representing faces as elastic graphs of local textural features.

The wide variation in face appearance under changes in pose, lighting, and expression makes face recognition a difficult task. While existing systems do not allow much flexibility in pose, lighting, and expression, most systems do provide some flexibility by using invariant representations or performing an explicit geometrical normalization step. As example invariant representations, filtering the face image with a bandpass filter like the Laplacian provides some invariance to lighting, and shift invariance

[*] This report describes research done at the Artificial Intelligence Laboratory and the Center for Biological and Computational Learning. Support for the A.I. Laboratory's research is provided in part by ONR contract N00014-91-J-4038. The author is supported by a Howard Hughes Doctoral Fellowship from the Hughes Aircraft Company. Thanks to Tomaso Poggio for his guidance and support and Amnon Shashua for useful discussions.

can be provided by using the Fourier transform magnitude [1] or autocorrelation [17]. The face can be normalized for translation, scale, and image-plane rotation by finding at least two facial features – usually the eyes in existing systems – and using these features to register model and input representations.

Most face recognition systems are not designed to handle changes in facial expression or rotations out of the image plane. By tackling changes in pose and lighting with the invariant representations and normalization techniques described above, current systems treat face recognition mostly as a rigid, 2D problem. There are exceptions, however, as some systems have used multiple views ([1], [17]) and flexible matching strategies [18] to handle some degree of expression and out-of-plane rotations. What distinguishes our approach from these techniques will be a wider allowed variation in viewpoint.

Overall, while face recognition systems have been successful (the template-based systems in [2] and [6] achieved 100% recognition on a data base of over 40 people), most recognition systems work with frontal views, no expressions, and controlled lighting. Our goal is to build a face recognizer that works under varying pose, the difficult part of which is to handle face rotations in depth. Building on successful template-based systems, our basic approach is to represent faces with templates from multiple model views that cover different poses from the viewing sphere.

Our face recognizer deals with the problem of arbitrary pose by applying a feature finder/pose estimation module before recognition. As mentioned previously, one can normalize the input image for translation, scale, and image-plane rotation by detecting the eyes and then applying a similarity transform to place the eyes at known locations. The remaining pose parameters, rotations in depth, can be estimated by a pose module and then used to select model views similar in pose to the input.

Our feature finder/pose estimation module finds the two eyes and a nose lobe feature and estimates the pose rotation parameters out of the image plane. The method is template-based, with tens of facial feature templates covering different poses and different people. To geometrically align the input face with a model view, the recognizer applies an affine transform to the input to bring the three feature points into correspondence with the same points on the model.

The template-based recognizer uses templates of the eyes, nose, and mouth to represent faces. These templates, as well as the input image, are preprocessed with a differential operator such as the gradient or Laplacian to provide some invariance to lighting. After the geometrical alignment step, the templates are matched against the input using normalized correlation as a metric.

Before describing the template-based recognizer in detail, we quickly review the experimental setup and the feature finder.

2 Experimental setup

In our view-based approach for face recognition under varying pose, faces are represented using many images that cover the viewing sphere. Currently we use 15 views per person, sampling 5 left/right rotations and 3 up/down rotations, as shown in figure 1. When a subject is added to the library of faces, the subject is asked to point their head at each of 15 dots – one for each view – on a piece of foam core fit around the camera. This field of dots sample the 5 left/right rotations at approximately -30, -15, 0, 15, and 30 degrees and the 3 up/down rotations at approximately -20, 0, and 20 degrees. The two rotation parameters are restricted so that the two eyes are always visible.

In addition to the 15 modeling views, 10 test views are taken per person. For these test views, the subject is instructed to choose 10 points at random (not necessarily at a model dot) within the rectangle defined by the outer border of dots. The 10 views are divided into two groups of 5: the first group allows variation in the left/right and up/down rotational parameters and the second group allows the subject to *also* include an image-plane rotation.

We currently have 62 people in the data base for a total of 930 modeling and 620 testing views. The data base includes 44 males and 18 females, people from different races, and an age range from the 20s to the 40s. We plan to expand the data base to around 100 people.

For both the modeling and testing views, the lighting conditions are fixed and consist of a 60 watt lamp near the camera supplemented by background lighting from windows and overhead lights. Facial expressions are also fixed at a neutral expression.

After taking the modeling and testing images, we manually specify the locations of the two irises, nose lobes, and corners of the mouth. In the feature finder, these manual locations are used as ground truth data for validating the locations returned by the feature finder and as the "interest points" – irises, lobes of the nose – within the model templates used by the feature finder. In the template-based recognizer, the manual locations are are used to automatically define the bounding boxes of facial feature templates in the model images and as anchor points in the model views during the geometrical alignment step between input and model images.

3 Feature detection and pose estimation

The first stage of processing in the proposed face recognition architecture is a person-independent feature finding and pose estimation module. As mentioned in the introduction, the kind of facial features sought by the feature finder are the two eyes and at least one nose feature. The locations of these features are used to bring input faces into rough geometrical alignment with model faces. Pose estimation is used as a filter on the library models, selecting only those models whose pose is similar to the input's pose. By pose estimation we really mean an estimate of the rotation angles out of the image plane since feature locations have already been used to normalize for position, scale, and image-plane rotation. Pose estimation is really an optimization step, for even in the absence of a robust pose estimator, the system could still test the input against all model poses of all people.

While techniques already exist for finding facial fea-

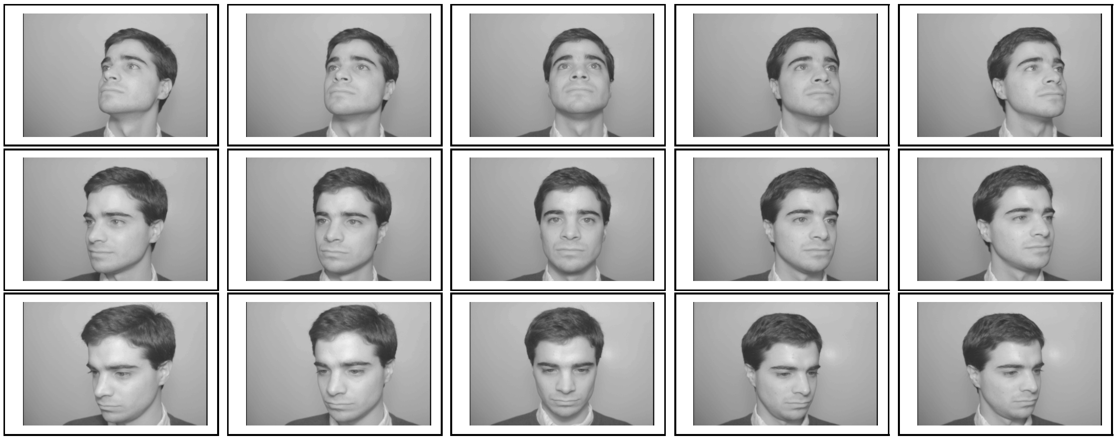

Figure 1: The view-based face recognizer uses 15 views to model a person's face.

tures, no current system can deal with large face rotations out of the image plane, so we needed to build our own feature finder. Because of the variety of views the system would have to work under, we were attracted by the simplicity of a template-based approach.

To serve as the front end of a pose independent face recognizer, the feature finder must handle varying pose and be person independent. Our system addresses these requirements by using a large number of templates taken from multiple poses and from different people. To handle rotations out of the image plane, templates from different views on the viewing sphere are used. Templates from different scales and image-plane rotations can be generated by using standard 2D rotation and scaling operations. To make the feature finder person independent, the templates must cover identity-related variability in feature appearance (e.g. tip of nose slanted up versus down, feature types specific to certain races). We use templates from a variety of exemplar faces that sample these basic feature appearances. The choice of exemplars was guided by a simple clustering algorithm that measures face similarity though correlation.

Our feature finder, then, entails correlation with a large number of templates sampling different poses and exemplars. To keep this search under control, we use a hierarchical coarse-to-fine strategy on a multi-level pyramid representation of the image. The search begins by generating face location hypotheses at the coarsest level, where the pose parameters are very coarsely sampled and only one exemplar is used. Exploring an hypothesis is organized as a tree search through the finer pyramid levels. As processing proceeds to finer levels, the pose parameters are sampled at a higher resolution and the different exemplars are used. A branch at any level in the search tree is pruned if the template correlation values are not above a level-dependent threshold. Space limitations in these proceedings prevent a more detailed presentation; for details, see [4].

To evaluate the feature finder, the system was run on all 1550 images in the data base, the 15 modeling and 10 testing images of each of the 62 people. Using the manual locations as ground truth, in 99.6% of the images all of the features were located to within an average distance of $.021d$ and a maximum distance of $.2d$, where d is the interocular distance of a frontal view. Figure 2 shows some of the features returned by the system.

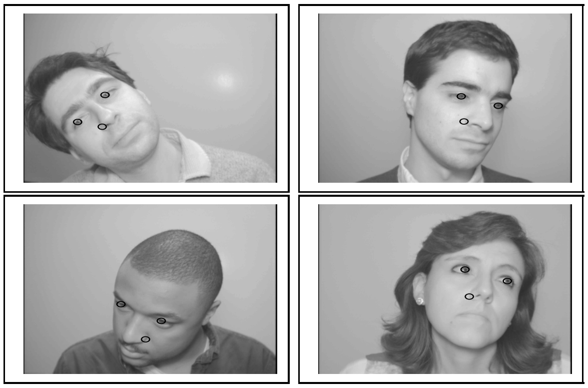

Figure 2: Iris and nose lobe features located by the feature finder in some example test images.

Because of the large number of templates, the computation takes around 10-15 minutes on a Sun Sparc 2. Using fewer exemplars decreases the running time but also reduces system flexibility and recognition performance.

4 Face recognition using multiple views

As mentioned in the introduction, template-based face recognizers have been quite successful on frontal views of the face ([2], [6]). Our goal is to extend template-based systems to handle varying pose, notably facial rotations in depth. Our approach is view-based, representing faces with templates from many images that cover the viewing sphere. In this section we describe the view-based recognizer and experimental results on our data base of face images.

4.1 Input representation: templates

In order to build face models for the recognizer, templates from the eyes, nose, and mouth are extracted from

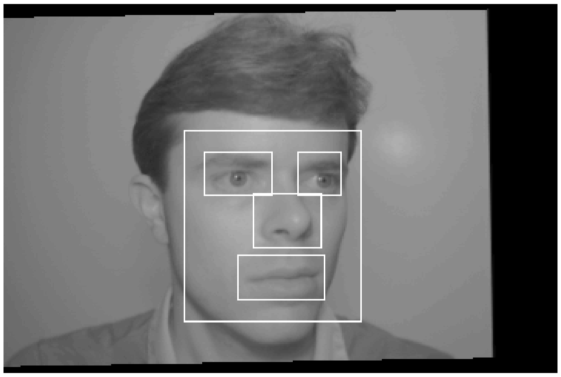

Figure 3: Templates of the eyes, nose, and mouth are used to represent faces.

the modeling images, as shown in figure 3. Before extracting the templates, scale and image-plane rotation are normalized in the model images to fix the interocular distance and eliminate any head tilt. This is done by placing the eyes, as located manually, at fixed locations in the image. Next, the bounding boxes of the templates are automatically computed using the manually specified feature locations.

We have done experiments to explore two aspects of template design, model image preprocessing and template scale. As discussed previously, it is common in face recognition to preprocess the templates to introduce some invariance to lighting conditions. So far we have tested preprocessing with the gradient magnitude, Laplacian, and x and y components of the gradient. The overall scale of the templates, as measured by the interocular distance, is another design parameter we examined. These experiments on preprocessing and scale will be described in the experimental results section.

4.2 Recognition algorithm

Our template-based recognizer takes as input a view of an unidentified person, compares it against all the people in the library, and returns the best match. Pseudocode sketching the steps of our recognizer is given in figure 4. First, in step (1), the pose calculated by the feature finder/pose estimation module acts as a filter on the model poses: only those model poses that are similar to the input pose will be selected. Since our current implementation of the pose estimator can only distinguish between looking left and looking right (see [4]), the poses selected by the recognizer for comparison are either the left three columns or right three columns of figure 1. In the future a more refined pose estimate will allow the recognizer to further winnow down the number of model poses it needs to test for each person.

Next, in steps (2) and (3) the recognizer loops over the selected poses of all model people, recording template correlation scores from each model view in the **cor** array. The main part of the recognizer, steps (4)-(6), compares the input image against a particular model view. This comparison consists of a geometrical alignment step (step (4)) followed by correlation (steps (5)-(6)). The geometrical alignment step brings the input and model images into close spatial correspondence in preparation for the correlation step. To geometrically align the input image against the model image, first an affine transform is applied to the input to align three feature points, currently the two eyes and a nose lobe feature. In the input image these features are automatically located using the feature finder described in the previous section. For the models, manual feature locations are used.

The second part of the geometrical alignment step attempts to compensate for any small remaining geometrical differences due to rotation, scale, or expression. A dense set of pixelwise correspondence between the affine transformed input and the model is computed using optical flow [3]. Given this dense set of correspondences, the affine transformed input can be brought into pixel-level correspondence with the model by applying a 2D warp operation driven by the optical flow. Basically, pixels in the affine transformed input are “pushed” along the flow vectors to their corresponding pixels in the model.

Now that the input and model image have been geometrically registered, in steps (5) and (6) the eye, nose, and mouth model templates are correlated against the input. Each model template is correlated over a small region (e.g. 5x5) centered around its expected location in the input. We use normalized correlation as the matching metric, primarily because it factors out differences in template mean and standard deviation, which might be caused by differences in lighting.

When scoring a person in step (7), the system takes the sum of correlations from the best matching eye, nose, and mouth templates. Note that we maximize over the poses separately for each template, so the best matching left eye could be from pose 1 and the best matching nose from pose 2, and so on. We found that switching the order of the sum and max operations – first summing template scores and then maximizing over poses – gives slightly worse performance, probably because the original sum/max ordering is more flexible.

After comparing the input against all people in the library, the recognizer returns the person with the highest correlation score – we have not yet developed a criterion on how good a match has to be to be believable. Considering a task like face verification, however, having the ability to reject inputs is important and is something we plan under future work.

4.3 Experimental results

We have tested our face recognizer under different template resolutions and methods of preprocessing. For each recognition experiment, we ran the recognizer on our data base of 620 test images, 10 images each of 62 people. The recognition experiments use the eyes and nose features found by our feature finder to drive the geometrical alignment stage. The feature finder fails to return any features for two images – these are listed in the rightmost column of tables 1 and 2.

Table 1 summarizes our recognition results for the preprocessing experiments. The types of preprocessing we tested include the gradient magnitude (mag), Laplacian (lap), sum of separate correlations on x and y compo-

```
selected poses ← left or right group of poses, from pose estimator
for person ← 1 to NUM_PEOPLE                              /* for all people in data base */
    forall pose ∈ selected poses                          /* for all poses to search */
        align input to model pose: affine transform & optical flow
        for template ← 1 to NUM_TEMPLATES                 /* loop over eyes, nose, mouth */
            cor[person][pose][template] ← correlation value
    score[person] ← $\sum_{\mathbf{template}=1}^{NUM\_TEMPLATES} ( \max_{\mathbf{pose} \in \mathbf{selected\ poses}} (\mathbf{cor}[\mathbf{person}][\mathbf{pose}][\mathbf{template}]))$
```

Figure 4: Pseudocode for our template-based recognizer.

nents of the gradient (dx+dy), and the original grey levels (grey). For these preprocessing experiments we used an intermediate template scale, an interocular distance of 30. In table 1, we list the number of correct recognitions and the number of times the correct person came in second, third, or past third place. Best performance was had from dx+dy, mag, and lap, with dx+dy yielding the best recognition rate at 98.7%. Preprocessing with the gradient magnitude performs nearly as well, a result in agreement with the preprocessing experiments of [6]. Given that the original grey levels lead to the lower rate of 94.5%, our results indicate that preprocessing the image with a differential operator gives the system a performance advantage. We think the performance differences between dx+dy, mag, and lap are too small to say that one preprocessing type stands out over the others.

Table 2 summarizes our recognition results for the template scale experiments, where scale is measured by the interocular distance of a frontal view. The preprocessing was fixed at dx+dy. The intermediate and fine scales perform the best, indicating that at least for our input representation, the coarsest scale may be losing detail needed to distinguish between people. Since the intermediate scale has a computational advantage over the finer scale, we would recommend operating a face recognizer at the intermediate scale.

Having examined the error cases, we have noticed that in the system's false positive matches, using optical flow to warp the input to the model may be contributing to the problem. If two people are similar enough, the optical flow can effectively "morph" one person into the other, making the matcher a bit *too* flexible at times. This problem with optical flow suggests some extensions to the recognizer. Since we only want to compensate for rotational, scale, or expression changes and not allow "identity-changing" transforms, perhaps the optical flow can be interpreted and the match discarded if the optical flow is not from the allowed class of transformations. Another approach would be to penalize a match using some smoothness measure of optical flow. The new matching metric would have a regularized flavor, being the sum of correlation and smoothness terms

$$||I(x + \Delta x) - T||^2 + \lambda\phi(\Delta x),$$

where $I(x + \Delta x)$ is the input warped by the flow Δx, T is the template, ϕ is a smoothness functional including derivatives, and λ is a parameter controlling the trade off between correlation and smoothness. This functional has an interpretation as the combination of a noise model on the intensity image and priors on the flow.

In terms of execution time, our current system takes about 1 second to do each input/model comparison on a Sun Sparc 1. The computation time is dominated by resampling the image during the affine transform, optical flow, and correlation. In our unoptimized CM-5 implementation, it takes about 10 seconds for the recognizer to run since we can distribute the data base so that each processor compares the input against one person. Specialized hardware, for example correlation chips[13], can be used to further speed up the computation.

5 Conclusion

In this paper we presented a view-based approach for recognizing faces under varying pose. Motivated by the success of recent template-based approaches for frontal views, our approach models faces with templates from 15 views that sample different poses from the viewing sphere. The recognizer consists of two main stages, a geometrical alignment stage where the input is registered with the model views and a correlation stage for matching. Our recognizer has achieved a recognition rate of 98% on a data base 62 people. The data base consists of 930 modeling views and 620 testing views covering a variety of poses, including rotations in depth and rotations in the image plane.

We have also developed a facial feature finder to provide feature locations for the geometrical alignment stage in the recognizer. Like the recognizer, our feature finder is template-based, employing a bank of templates of the eyes and nose regions to locate the two irises and one nose lobe feature. While the features are currently used to register input and model views, the feature finder has other applications. For instance, it could be used to initialize a facial feature tracker, finding the feature locations in the first frame. This would be useful for virtual reality, HCI, and low bandwidth teleconferencing.

In the future, we plan on adding more people to the data base and adding a rejection criterion to the recognizer. We would also like to improve the estimate of pose returned by the feature finder. A better pose estimate will enable the recognizer to search over a smaller set of model poses.

preprocessing	performance – 620 test images								bad features	
	correct		2nd place		3rd place		>3rd place			
dx+dy	98.71%	(612)	0.32%	(2)	0.48%	(3)	0.16%	(1)	0.32%	(2)
mag	98.23%	(609)	0.81%	(5)	0.32%	(2)	0.32%	(2)	0.32%	(2)
lap	98.07%	(608)	0.81%	(5)	0.32%	(2)	0.48%	(3)	0.32%	(2)
grey	94.52%	(586)	1.94%	(12)	0.48%	(3)	2.74%	(17)	0.32%	(2)

Table 1: Face recognition performance versus preprocessing. Best performance is from using the gradient magnitude (mag), Laplacian (lap), or the sum of separate correlations on the x and y gradient components (dx+dy). An intermediate scale was used, with an interocular distance of 30.

interocular distance	performance – 620 test images								bad features	
	correct		2nd place		3rd place		>3rd place			
15	96.13%	(596)	2.26%	(14)	0.32%	(2)	0.97%	(6)	0.32%	(2)
30	98.71%	(612)	0.32%	(2)	0.48%	(3)	0.16%	(1)	0.32%	(2)
60	98.39%	(610)	0.81%	(5)	0.16%	(1)	0.32%	(2)	0.32%	(2)

Table 2: Face recognition performance versus scale, as measured by interocular distance (in pixels). The intermediate scale performs the best, a result in agreement with Brunelli and Poggio[6]. For preprocessing, separate correlations on the x and y components of the gradient were computed and then summed (dx+dy).

In a related line of research, we plan to address the problem of recognizing a person's face under varying pose when only *one* view of the person is available. The key new component will be an example-based learning system that uses many images of prototype faces undergoing changes in pose to "learn" what it means to rotate a face (see [19]). The system will apply this knowledge to synthesize new "virtual" views of the person's face.

Overall, we have demonstrated in this paper that template-based face recognition systems can be extended in a straightforward way to handle the problem of varying pose. However, to make a truly general face recognition system, more work needs to be done, especially to handle variability in expression and lighting conditions.

References

[1] Shigeru Akamatsu, Tsutomu Sasaki, Hideo Fukamachi, Nobuhiko Masui, and Yasuhito Suenaga. An accurate and robust face identification scheme. In *Proceedings Int. Conf. on Pattern Recognition*, volume 2, pages 217–220, The Hague, The Netherlands, 1992.

[2] Robert J. Baron. Mechanisms of human facial recognition. *International Journal of Man Machine Studies*, 15:137–178, 1981.

[3] J.R. Bergen and R. Hingorani. Hierarchical motion-based frame rate conversion. Technical report, David Sarnoff Research Center, Princeton, New Jersey, April 1990.

[4] David J. Beymer. Face recognition under varying pose. A.I. Memo No. 1461, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1993.

[5] Martin Bichsel. *Strategies of Robust Object Recognition for the Automatic Identification of Human Faces*. PhD thesis, ETH, Zurich, 1991.

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

[7] Peter J. Burt. Multiresolution techniques for image representation, analysis, and 'smart' transmission. In *SPIE Vol. 1199, Visual Communications and Image Processing IV*, pages 2–15, 1989.

[8] Chin-Wen Chen and Chung-Lin Huang. Human face recognition from a single front view. *International Journal of Pattern Recognition and Artificial Intelligence*, 6(4):571–593, 1992.

[9] Ian Craw and Peter Cameron. Face recognition by computer. In David Hogg and Roger Boyle, editors, *Proc. British Machine Vision Conference*, pages 498–507. Springer Verlag, 1992.

[10] Shimon Edelman, Daniel Reisfeld, and Yechezkel Yeshurun. Learning to recognize faces from examples. In *Proceedings of the European Conference on Computer Vision*, pages 787–791, 1992.

[11] Michael K. Fleming and Garrison W. Cottrell. Categorization of faces using unsupervised feature extraction. In *Proceedings of the International Joint Conference on Neural Networks*, volume 2, pages 65–70, 1990.

[12] A. Fuchs and H. Haken. Pattern recognition and associative memory as dynamical processes in a synergetic system; I. translational invariance, selective attention, and decomposition of scenes. *Biological Cybernetics*, 60:17–22, 1988.

[13] Jeffrey M. Gilbert and Woody Yang. A real-time face recognition system using custom VLSI hardware. In *IEEE Workshop on Computer Architectures for Machine Perception*, pages 58–66, December 1993.

[14] Takeo Kanade. Picture processing by computer complex and recognition of human faces. Technical report, Kyoto University, Dept. of Information Science, 1973.

[15] M. Kirby and L. Sirovich. Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):103–108, 1990.

[16] T. Kohonen. *Self-organization and Associative Memory*. Springer-Verlag, Berlin, 1989.

[17] T. Kurita, N. Otsu, and T. Sato. A face recognition method using higher order local autocorrelation and multivariate analysis. In *Proceedings Int. Conf. on Pattern Recognition*, volume 2, pages 213–216, The Hague, The Netherlands, 1992.

[18] B.S. Manjunath, R. Chellappa, and C. von der Malsburg. A feature based approach to face recognition. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition*, pages 373–378, 1992.

[19] Tomaso Poggio and Thomas Vetter. Recognition and structure from one 2D model view: Observations on prototypes, object classes, and symmetries. A.I. Memo No. 1347, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1992.

[20] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[21] John J. Weng, N. Ahuja, and T.S. Huang. Learning recognition and segmentation of 3-D objects from 2-D images. In *Proceedings of the International Conference on Computer Vision*, pages 121–128, Berlin, May 1993.