Face Recognition using a Hybrid Supervised/Unsupervised Neural Network

Nathan Intrator Daniel Reisfeld Yehezkel Yeshurun
Department of Computer Science
Tel-Aviv University
Ramat Aviv 69978, Israel

Revised: June 22, 1995

Abstract

A system for automatic face recognition is presented. It consists of several steps; Automatic detection of the eyes and mouth is followed by a spatial normalization of the images. The classification of the normalized images is carried out by a hybrid (supervised and unsupervised) Neural Network. Two methods for reducing the overfitting – a common problem in high dimensional classification schemes – are presented, and the superiority of their combination is demonstrated.

Key words: Face recognition, Neural Networks, Interest points, Symmetry operator.

1 Introduction

Automatic face recognition has gained much attention in recent years, due to the variety of potential applications, and the increase in computational power which enables effective implementation of algorithms.

Traditionally, face recognition was based on extracting certain features (e.g. spatial location of facial features and their geometrical relations) [4, 20]. These features are detected either manually, or by automatic algorithms [32, 8, 7, 27]. Another approach [21, 30], is based on direct processing of the grey level images. A review of face processing systems could be found in [6, 29].

The task of recognizing faces is inherently a classification problem in high dimensional feature space, and thus subject to the "curse of dimensionality" [1] which essentially says that the number of training patterns needed for robust classification, should be restrictively high.

Regarding an image merely as a matrix and looking for algebraic invariants [13] reduces the dimensionality. However, such algebraic constraints can be designed to be invariant to practically any transformation but they are too general. For instance they are not affected by upside down inversion, while biological systems are [24]. A recent approach to this problem [3] is based on the finding that facial images are projected to connected domains (an extension of clusters) and thus could be used to reduce dimensionality. An alternative approach for the reduction of the dimensionality is to use a limited set of biologically motivated receptive fields [23, 10]. Yet another way to overcome the "curse" is to base the recognition on a small number of linear combinations (projections) of the high dimensional space. This approach is at the heart of projection pursuit methods [15] and neural network methods. Taking this approach, one is then confronted with the task of finding such an optimal projection. A commonly used approach is based on second order statistics of the data where one extracts the directions in which the variance is maximized - also called the principal components of the data [21, 30].

In this paper we adopt a different approach to dimensionality reduction and classification, based on a combination of supervised and unsupervised learning [16]. We first automatically detect the eyes and the mouth in face image by using the *Generalized Symmetry Transform*, and use this information to normalize the images by affine transformation. We then proceed to classification. The supervised learning seeks projections that minimize mean squared error between the output of a feed-forward network and the class label of the image. The unsupervised learning seeks projections which demonstrate some interesting structure in the data essentially by measuring deviation from Gaussian distribution in the form of multi-modality. We conclude by comparing our method with principal components based recognition, and by discussing the interpretability of our results.

2 Facial preprocessing and normalization

The main goal of the preprocessing step in our method is the reduction of the dimensionality problem by spatial normalization of the face image. We detect the eyes and mouth in face images using the Generalized Symmetry Transform, and then we warp the face image to a "standard" location using these points.

The Generalized Symmetry Transform is described in [26, 25]. The main idea behind it is the following: starting with an adge map, every pixel is assigned a magnitude M that estimates the probability that there is a symmetric spatial configuration of edges around it, and an orientation α, that points in the direction of the main axis of symmetry around the pixel. Thus, for example, the pixel (or pixels) in the center of a circular, elliptic or rectangular area surrounded by edges, will be assigned a high value of M. This results in a Symmetry Map, where every pixel has a value, and the highest peaks of symmetry could be detected. The main difference between our method and other symmetry estimation methods (or even straight forward detection of centers of gravity), stems from the fact that the symmetry map is computed prior to the segmentation stage, while most other methods are performed only when the contours of specific object are already available.

The Generalized Symmetry Transform is context-free, in the sense that it operates directly on pixels and not on known objects. However, it is possible to incorporate application specific information to enhance its performance. In this paper we have used the following operations on face images in order to detect the location of the eyes and mouth:

- Computation of the symmetry magnitude and orientation. This is the standard Symmetry operation described before.
- Computation of the Radial Symmetry (RS) [26]. While the regular symmetry definition does not depend on the specific spatial organization of the edge that contribute to the symmetry measure, this measure assigns high value to pixels that are surrounded by circular contours.
- Detection of the highest peaks of the regular and radial symmetry in the image.
- Detection of the midline of the face image by finding the peak in the autocorrelation function of the edge image.
- Detection of the eyes and mouth by including geometric considerations. This is carried out by finding the location of the highest peaks of the symmetry values, with the assumption that the eyes should be on both sides of the midline, and the mouth should intersect it.

Once we have detected the location of the eyes and mouth, we warp the image by using affine transformation based on 3 anchor points: the centers of the eyes and the mouth. The images are warped such that the eyes and center of mouth are translocated to predefined locations, thus forming a normalized grey level image of the face. We further try to reduce the variability by considering the gradient of these images, to compensate for variable lighting conditions.

3 The Feature Extraction/Classification

We have employed several variations of the frequently used feed-forward artificial neural network for classification. We have chosen to use feed-forward artificial neural networks due to their ability to cope with very high dimensional data, thus making them excellent candidates to perform recognition from pixel values. The class of functions that can be approximated by a back-propagation type network is very large; This architecture (with an unlimited number of projections) can uniformly approximate arbitrary continuous functions on compact sets, as well as their derivatives [14]. The ability to approximate a function and its derivatives will be used below for model interpretability.

The error is propagated backwards to the previous (hidden) layer for modification of its synaptic weights (projections). The single hidden layer architecture is of the form

$$f(x) = \sum_j \beta_j \sigma(\sum_{k=1}^{d} \omega_{jk} x_k + w_{j0}),$$

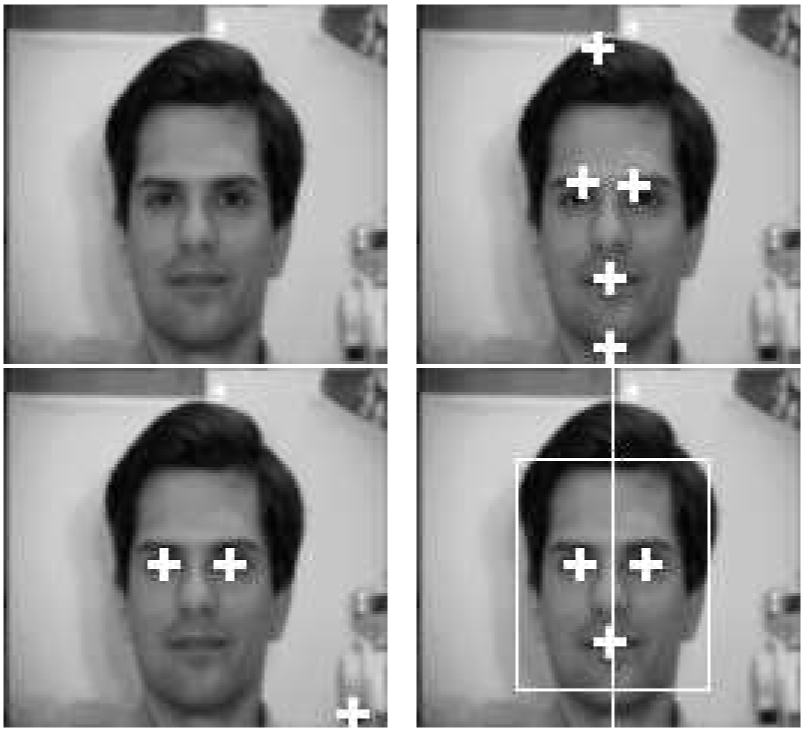

Figure 1: Locating facial features. Top left to bottom right: An image, the highest peaks of the symmetry magnitude, the highest peaks of RS, the midline and the features after applying geometric constraint.

where σ is an arbitrary (fixed) bounded monotone function. The form

$$f(x) = \sigma(\sum_j \beta_j \sigma(\sum_{k=1}^{d} \omega_{jk} x_k + w_{j0})),$$

is more suitable for classification tasks. Since this method can approximate any continuous function, great care should be taken so that the variance of the estimated weights is small, and the model does not "overfit" the training data [12, for discussion]. This is often done using some form of complexity regularization such as weight decay [31, for review].

The performance of a single back-propagation network can be easily enhanced by training several different networks and averaging their result [22]. On this network ensemble, we have used a hybrid training method [16]. This method is based on a formulation that combines unsupervised (exploratory) methods for finding structure (extracting features) and supervised methods for reducing classification error. The unsupervised training portion is aimed at finding features such as clusters. The supervised portion is aimed at finding features that minimize classification error on the training set. The combination of both methods may give better generalization performance (under "good" a-priori assumptions about the structure of the data). The application of the hybrid training in a feed-forward neural network is done by modifying the learning rule of the hidden units to reflect the additional constraints (Figure 3).

Figure 2: Normalization transformations. Left to right: Original image; its warping; gradient of the warped image.

The unsupervised feature extraction which we used, is based on the biologically motived BCM neuron [2]. This method essentially seeks clusters in the data distribution by seeking multimodality in the projected distribution via a robust measure that is based on the third and second order statistics of the data.

3.1 The unsupervised constraint

The network implementation described below can find several projections in parallel while retaining its computational efficiency. It was found to be applicable for extracting features from very high dimensional vector spaces [18].

Below is a brief description of the unsupervised portion of the network (see [17] for details.) The activity of neuron k in the network is $c_k = \sum_i x_i w_{ik} + w_{0k}$. The *inhibited* activity and threshold of the k'th neuron is given by

$$\tilde{c}_k = \sigma(c_k - \eta \sum_{j \neq k} c_j), \qquad \tilde{\Theta}_M^k = E[\tilde{c}_k^2].$$

The threshold $\tilde{\Theta}_M^k$ is the point at which the modification function ϕ changes sign. The function ϕ is given by

$$\phi(c, \Theta_M) = c(c - \Theta_M).$$

The risk (projection index) for a single neuron is given by

$$R(w_k) = -\{\frac{1}{3} E[\tilde{c}_k^3] - \frac{1}{4} E^2[\tilde{c}_k^2]\}.$$

The total risk is the sum of each local risk. The negative gradient of the risk that leads to the synaptic modification equations is given by

$$\frac{\partial w_{ij}}{\partial t} = E[\phi(\tilde{c}_j, {\Theta_M}^j)\sigma'(\tilde{c}_j)x_i - \eta \sum_{k \neq j} \phi(\tilde{c}_k, \tilde{\Theta}_m^k)\sigma'(\tilde{c}_k)x_i].$$

This last equation is an additional penalty to the energy minimization of the supervised network.

Note that there is an interaction between adjacent neurons in the hidden layer. In practice, the stochastic version of the differential equation can be used as the learning rule. In the results

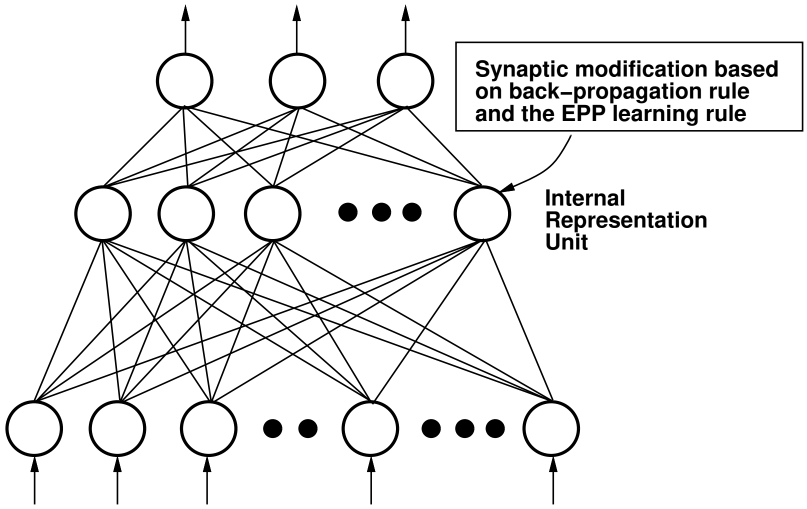

Figure 3: A hybrid EPP/PPR neural network (EPPNN). The unsupervised learning rule is applied to the hidden layer units only.

reported here, a feed-forward architecture with a single hidden layer of 12 units was used in all the experiments. Training was done using the back-propagation algorithm [28] for the supervised part and using the projection pursuit learning [17] for the unsupervised part.

For comparison, we also report classification results based on other classification techniques.

The calculation of significance of the object features for recognition was done via a newly introduced method for interpreting neural networks which is described elsewhere [19]. This method extends the interpretability associated with linear or logistic regression to feed-forward neural networks.

4 Experimental Methodology

We have used a subset of the MIT Media Lab database of face images, courtesy of Turk and Pentland [30]. Previous results using the same preprocessing and dimensionality reduction using receptive fields and radial basis function networks have been described in [10].

The database we used contained 27 instances of each of 16 different persons. The images were taken under varying illumination and camera location. Of the 27 images, 17 were randomly chosen for each person to be used in training, while the remaining 10 were used for testing. The images were preprocessed as described in section 2 namely, the eyes and mouth were transformed to standard locations via the symmetry transformation. The size of the warped image was 40×60 pixels, the eyes locations at $(13, 27)$ and $(27, 27)$, and the mouth location at $(20, 44)$. The processed images are shown in Figure 4; on the left, a representation of each of the 16 faces is shown, and on the right 16 instances of a single face are presented to demonstrate the variability between instances of a single image.

Figure 4: One normalized image of each class (left) and the variability within normalized subjects for a single face (left)

5 Results and Discussion

We have performed many experiments aiming at analyzing different components of the face recognition scheme. First, we describe the main building blocks of our recognition scheme, and then we proceed with the more refined and unique methods which further improved the results. We start with a discussion on the image background of the Turk and Pentland database [30].

5.1 Preprocessing

Contribution of the facial background

The faces in the Turk & Pentland data-set are part of a larger picture that contains some background scenery usually a laboratory room of some sort. Generally, the facial background should have a negative effect on recognition performance due to its high variability. However, in this particular data-set, it appears that different subjects were photographed at different places, thus the background such as a corner of a blackboard, picture on the wall etc, was common to the same subject but not to all of them. The positive effect of the image background on recognition results is best demonstrated in Table 1 which includes classification results using one nearest neighbor classifier. Performance on the full images without any preprocessing is already very good. However, this performance drops sharply when the image background is removed, and only slightly when the faces are removed.

Preprocessing type	*success ratio*	
Full image	156/160	97.50%
No background	115/160	71.88%
No face	154/160	96.25%

Table 1: Effect of the background on nearest neighbor classification. Success ratio is given using the number of images correctly classified and their percentage.

Warp	*Gradient*	*success ratio*	
no	no	115/160	71.88%
no	yes	118/160	73.75%
yes	no	131/160	81.88%
yes	yes	147/160	91.88%

Table 2: Nearest neighbor performance on the warped faces (with no background.)

Classifier	*success ratio*	
1 NN	147/160	91.88%
3 NN	83/160	51.88%
RBF	147/160	91.88%

Table 3: The effect of classification method on the removed-background images with warping and gradient normalization.

Contribution of the image warp

Once the image background is removed we are left with the facial image only of a fixed size. The image warp, i.e., the affine transformation which takes the eyes and tip of mouth to a fixed location, has now a strong effect on recognition performance. To account for varying light source locations, we also normalized the images so that the mean *gradient* of pixel intensities is zero in all directions. The effect of image-warp transformation and gradient normalization on nearest neighbor classification results is exhibited in Table 2. Classification results were very sensitive to the classification scheme. For example, the gradient normalization was helpful in recognition via nearest neighbor classification, but not needed for the neural network classification schemes. It was surprising to find that three nearest neighbors performed much worse that one nearest neighbor (Table 3,5), suggesting that even after the removal of background and image normalization, the variability between images of the same class was still large enough, so that under simple Euclidian metric, there were closer neighbors from other classes. The implication of this finding is that another transformation was needed to reduce the dimensionality to a more invariant image representation. Earlier work with Radial Basis Function (RBF) classification [10] produced similar results to one nearest neighbor scheme on the warped images (Table 3,5).

5.2 Principal component extraction

Due to the success of principal components for face recognition [21, 30], we have studied the classification performance based on projections onto a varying number of principal components extracted from the data (PC fetures). In all cases, the principal components were the eigen vectors of the co-

No. of eigenvectors	*success ratio*	
1	29/160	18.13%
5	115/160	71.88%
20	142/160	88.75%
40	150/160	93.75%
50	152/160	95.00%
62	151/160	94.38%

Table 4: One nearest neighbor classification using varying number of principle components extracted from 17 images for each person, without background, with warping and gradient.

Classifier	*success ratio*	
1 NN	152/160	95.00%
3 NN	99/160	61.88%
RBF	152/160	95.00%

Table 5: Various classification techniques using 17 learned learned images for a person, without background, with warping and gradient for 44 eigenvectors.

variance matrix of the pixel correlations with highest eigen values. Nearest neighbor performance under the various preprocessing described above, is given in Table 4 for a varying number of PC features. The extraction of features via principal components did not eliminate the strong dependency on the number of nearest neighbors (Table 5) however improved the results compared with the original images (Table 3), suggesting that the image data representation is redundant, and that dimensionality reduction, such as the one done by projecting onto the principal components, can reduce the error to a half, while reducing the representation from 40×60 dimensions to 44. In the next section we study the classification performance achieved by a much stronger dimensionality reduction based on neural network approaches.

5.3 Neural network classification

Training neural networks was described in section 3. Classification results are summarized in Table 6. We first give results of the images obtained after eliminating the background and leaving an image of 60×40 pixels. The last two lines in the table correspond to results obtained by averaging over the outputs of 5 networks before producing the classification results. This network ensemble average method can simply be considered as reducing the variance of the network outputs (considered as random variables) by summing over an ensemble of networks [22]. This technique has been shown to be a good stabilizer for neural network results [5].

Often the cost of making a mistake (substitution error) is larger than the cost of no decision (rejection). In digit recognition, a frequent classification measure suitable in such cases is the *figure of merit*, in which the cost of substitution is 10 times the cost of rejection. The Figure of Merit, s, is calculated as $s = 100 - \text{Rejections} - 10 \times \text{substitutions}$.

Two points are worth mentioning in the results. First, as is often found, network ensemble reduces classification error. The results of networks trained with additional bias constraints in the form of BCM [17, 2] are intriguing; While the mean performance of networks trained with additional (bias) constraints, which are supposed to seek structure in the form of multi-modality, is slightly

Figure 5: Features extracted using Back-Propagation Network (left) and features extracted using a hybrid BCM/Back-Propagation Network (right)

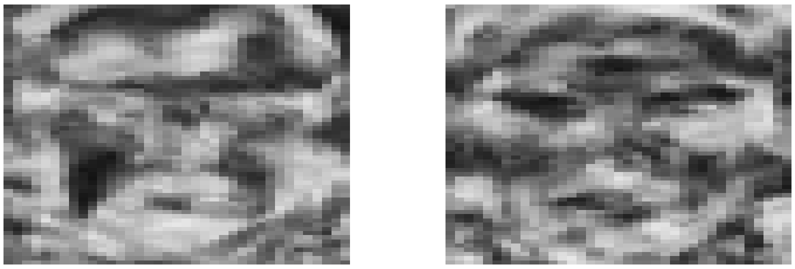

Figure 6: The significance of the features extracted by using a Back-Propagation Network (left) and by using a Hybrid BCM/Back-Propagation Network (right)

worse compared to networks that were not trained with such constraints, the ensemble performance of such networks yields better performance. These results are best explained by the bias/variance tradeoff [12, for review]; the effort to control the bias via bias constraints, increases the variance in single networks, however, the ensemble network averaging does not affect the bias, but reduces the variance leading to an overall improvement in classification results. An indication of the increased variance can be seen in Table 6 by the increased standard deviation of the results for the hybrid method. These results complement a different set of experiments which tried to study the effect of variance constraints on feed-forward neural networks. In that work, variance constraints in the form of weight decay [31] were used in a real-world character recognition problem. While performance of single networks improved on average, the performance of the network ensemble was worse than the performance of an ensemble of networks that were not trained using variance constraints. This is because the variance control via weight decay introduced bias which could not be removed by the network ensemble.

Method	% Success	% Figure of Merit
Back-Propagation	96.72$\pm$ .31	73.36$\pm$ 13.43
Hybrid BCM/BP	96.04$\pm$ .96	71.14$\pm$ 17.33
Averaged Back-Propagation	98.75	96.3
Averaged Hybrid BCM/BP	99.38	98.1

Table 6: Classification error on a test set from the Turk/Pentland database. Average is done on 5 networks. Figure of Merit is calculated as 100 - Rejections - 10 $\times$ substitutions.

Interpretability of the networks

Figure 5 shows a hidden unit representations of a plain back-propagation network (left) and of a hybrid BCM/Back-propagation (right) each taken from one of the corresponding networks. The different networks had various initial conditions and various relative strength of the unsupervised contribution. Although a total of 12 features were extracted, only 7 of the projections are different, which gives the surprising result that an efficient dimensionality reduction can give good classification performance of 16 different faces using only 7 features.

Figure 6 presents another way to interpret the results of either network. The mean derivative with respect to the inputs for each of the 16 persons is shown. This form of interpretation is very useful when considering the network architecture as a non-linear regression function approximation. In this case it indicates which parts of the image are most useful in improving the classification results, (the white areas) and which parts are mostly contributing to classification errors (the dark areas). There are various robustification issues related to the fact that the models which a network converges to are not unique. The full details of the method are described in [19]. The images presented in Figure 6 give the relative importance of parts of the images for the recognition of that specific prototype. The extremum parts of the image (both negative – dark, and positive – bright) indicate the important features. Notice that the head outline, eyes and mouth are more salient on the Hybrid BCM/BP method (right) than on the BP method (left). This is more consistent with psychophysical experiments [9, 11] that show that more attention is devoted to prominent facial features such as eyes and mouth. Such interpretability method may be useful for human psychophysics studies, and for possible comparison between human and machine recognition, and for the study of object features.

Summary

We have presented a system for face recognition that addresses several of the important issues in robust recognition:

- Location variability is addressed by the ability of the generalized symmetry transform to locate anchor points in the image and thus shift the image to a fixed location.
- The warping of the image using affine transformation such that the eyes and mouth are mapped to standard locations reduces variability between images, thus reducing the number of prototypes needed for training, and helps to overcome viewpoint variability.
- The use of neural network classification reduces dimensionality of image representation and improves recognition performance.

- The use of ensemble of networks improves recognition performance and reduces substitution errors.
- The use of BCM feature extraction, further improves recognition and reduces rejections for zero substitution errors.

Further work remains in studying the scaling properties of artificial neural networks to large datasets of faces.

References

[1] R. E. Bellman. *Adaptive Control Processes.* Princeton University Press, Princeton, NJ, 1961.

[2] E. L. Bienenstock, L. N. Cooper, and P. W. Munro. Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *Journal Neuroscience*, 2:32–48, 1982.

[3] M. Bischel and A. Pentland. Human face recognition and the face image set's topology. *Computer Vision, Graphics, and Image Processing*, 59(2):254–261, 1994.

[4] W.W. Bledsoe. Man-machine facial recognition. Technical Report Rep. PRI:22, Panoramic Research Inc., Palo Alto, CA, August 1966.

[5] L. Breiman. Stacked regression. Technical Report TR-367, Department of Statistics, University of California, Berkeley, August 1992.

[6] V. Bruce and M. Burton. Computer recognition of faces. In A.W. Young and H.D. Ellis, editors, *Handbook of face processing*, pages 487–506. Elsevier Science publisher, B.V. (North-Holland), 1989.

[7] R. Brunelli and T. Poggio. Face recognition through geometrical features. In *Proceedings of the 2nd European Conference on Computer Vision*, pages 972–800, Santa Margherita Ligure, Italy, May 1992.

[8] I. Craw, D. Tock, and A. Bennet. Finding face features. In *Proceedings of the 2nd European Conference on Computer Vision*, pages 92–96, Santa Margherita Ligure, Italy, May 1992.

[9] G.M. Davis, H.D. Ellis, and J.W. Shepherd. Face recognition accuracy as a function of mode of representation. *Journal of Applied Psychology*, 63:180–187, 1978.

[10] S. Edelman, D. Reisfeld, and Y. Yeshurun. Learning to recognize faces from examples. In *Proceedings of the 2nd European Conference on Computer Vision*, pages 787–791, Santa Margherita Ligure, Italy, May 1992.

[11] I. Fraser and D. Parker. Reaction time measures of feature saliency in perceptual integration task. In H. Ellis, M. Jeeves, F. Newcombe, and A.W. Young, editors, *Aspects of Face Processing.* Martinus Nijhoff, Dordrecht, 1986.

[12] S. Geman, E. Bienenstock, and R. Doursat. Neural networks and the bias-variance dilemma. *Neural Computation*, 4:1–58, 1992.

[13] Zi-Quan Hong. Algebraic feature extraction of image for recognition. *Pattern Recognition*, 24(3):211–219, 1991.

[14] K. Hornik, M. Stinchcombe, and H. White. Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks. *Neural Networks*, 3:551–560, 1990.

[15] P. J. Huber. Projection pursuit. (with discussion). *The Annals of Statistics*, 13:435–475, 1985.

[16] N. Intrator. Combining exploratory projection pursuit and projection pursuit regression with application to neural networks. *Neural Computation*, 5(3):443–455, 1993.

[17] N. Intrator and L. N. Cooper. Objective function formulation of the BCM theory of visual cortical plasticity: Statistical connections, stability conditions. *Neural Networks*, 5:3–17, 1992.

[18] N. Intrator and J. I. Gold. Three-dimensional object recognition of gray level images: The usefulness of distinguishing features. *Neural Computation*, 5:61–74, 1993.

[19] O. Intrator and N. Intrator. Using neural networks for interpretation of nonlinear models. In *American Statistical Society: 1993 Proceedings of the Statistical Computing Section*, pages 244–249. American Statistical Association, August 1993.

[20] T. Kanade. *Picture processing system by computer complex and recognition of human faces.* PhD thesis, Dept. of Information Science, Kyoto University, November 1973.

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

[22] W. P. Lincoln and J. Skrzypek. Synergy of clustering multiple back-propagation networks. In D. S. Touretzky and R. P. Lippmann, editors, *Advances in Neural Information Processing Systems*, volume 2, pages 650–657. Morgan Kaufmann, San Mateo, CA, 1990.

[23] B.S. Manjunath, R. Chellappa, and C. von der Malsburg. A feature based approach to face recognition. Technical Report CS-TR-2834, Center for Automated Research. University of Maryland, 1992.

[24] Y. Moses, S. Ullman, and S. Edelman. Selective attention gates visual processing in the extrastriate cortex. Technical report, The Weizmann Institute of Science, 1993.

[25] D. Reisfeld, H. Wolfson, and Y. Yeshurun. Detection of interest points using symmetry. In *Proceedings of the 3rd International Conference on Computer Vision*, pages 62–65, Osaka, Japan, December 1990.

[26] D. Reisfeld, H. Wolfson, and Y. Yeshurun. Context free attentional operators: the generalized symmetry transform. *Int. J. of Computer Vision, Special Issue on Qualitative Vision*, 14:119–130, 1995.

[27] D. Reisfeld and Y. Yeshurun. Facial normalization using few anchor points. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition*, Jerusalem, Israel, 1994.

[28] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. In D. E. Rumelhart and J. L. McClelland, editors, *Parallel Distributed Processing*, volume 1, pages 318–362. MIT Press, Cambridge, MA, 1986.

[29] A. Samal and P.A. Iyengar. Automatic recognition and analysis of human faces and facial expressions: A survey. *Pattern Recognition*, 25(1):65–77, 1992.

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

[31] A. S. Weigend, D. E. Rumelhart, and B. A. Huberman. Generalization by weight-elimination with application to forecasting. In R. P. Lippmann, J. E. Moody, and D. S. Touretzky, editors, *Advances in Neural Information Processing Systems*, volume 3, pages 875–882. Morgan Kaufmann, San Mateo, CA, 1991.

[32] A.L. Yuille, P.W. Halilinman, and D.S. Cohen. Feature extraction from faces using deformable templates. *International Journal of Computer Vision*, 8(2):99–111, 1992.