A MIXTURE OF MULTILAYER PERCEPTRON EXPERTS NETWORK FOR MODELING FACE/NONFACE RECOGNITION IN CORTICAL FACE PROCESSING REGIONS

REZA EBRAHIMPOURa,c,*, EHSANOLLAH KABIRb, HOSSEIN ESTEKYa, MOHAMMAD REZA YOUSEFIc

aSchool of Cognitive Sciences
Institute for Studies on Theoretical Physics and Mathematics
Niavaran, Tehran, P.O.Box 19395–5746, Iran

bDepartment of Electrical Engineering
Tarbiat Modarres University
Tehran, P.O. Box 14115–143, Iran

cDepartment of Electrical Engineering
Shahid Rajaee University
Tehran, Iran

ABSTRACT—Recent studies in neurobiology and especially in neuroimaging report that a gating mechanism prior to face processing levels of human visual system, facilitates the face/nonface recognition task. In accordance to these biological evidences, we propose a face/nonface recognition model which makes use of mixture of experts network. In order to improve the face/nonface recognition accuracy, the outputs of the expert networks are combined using a gating network. A novel structure, which is the use of multilayer perceptrons (MLPs) in forming the expert networks, is introduced. The learning algorithm is modified to be adapted with the MLP networks. The results reveal that using a mixture of simple MLPs is much more beneficial, in many respects, as it shows more certainty at its output and is also easier to train than a single, but complex, MLP.

Key Words: Mixture of Experts; Face/Nonface Recognition; Eigenfaces; Inferotemporal cortex; Fusiform Face Area

1. INTRODUCTION

The human visual system consists of a hierarchy of multiple cortical areas performing feedforward neural computation on the incoming visual signals. At the early steps, the visual cortical areas V1 and V2 perform edge and line detection. In a higher stage of processing, area V4 represent partially complex shapes with information about the structural description of the represented features [1]. In the final stage of visual processing lies the inferotemporal cortex (IT) which is thought to execute visual object recognition. Near 25% of cells in IT have been shown to respond selectively to face images, making IT the ultimate cortical machinery for performing face/nonface recognition tasks [2–4].

Recognition of face/nonface is a difficult problem, which confronts all the major challenges in computer vision and pattern recognition [5]. Earlier efforts of face detection research have been focused on correlation or template matching, matched filtering, subspace methods, deformable templates, etc. [6,7].
For comprehensive surveys of these early methods, see Refs. [8–10]. Recent face/nonface recognition approaches, however, emphasize on statistical modeling and machine learning techniques [5,11]. Some representative methods are the probabilistic visual learning method [12], the example–based learning method [13], the neural network–based learning method [14,15], the probabilistic modeling method [16,17], the mixture of linear subspaces method [18], the machine learning approach using a boosted cascade of simple features [19], statistical learning theory and SVM–based methods [20–22], the Markov random field–based methods [23,24], the color–based face detection method [25], and the Bayesian discriminating feature (BDF) method [26].

Sung and Poggio [13] presented an example–based learning method by means of modeling the distributions of face and nonface patterns. To cope with the variability of face images, they empirically chose six Gaussian clusters to model the distributions for face and nonface patterns, respectively. The density functions of the distributions are then fed to a multiple layer perceptron for face detection. Rowley et al. [14] developed a neural network–based upright, frontal face recognition system, which applies a retinally connected neural network to examine small windows of an image and decide whether each window contains a face. The face/nonface recognizer, which was trained using a large number of face and nonface examples, contains a set of neural network–based filters and an arbitrator which merges detections from individual filters and eliminates overlapping detections. In order to recognize faces at any degree of rotation in the image plane, the system was extended to incorporate a separate router network, which determines the orientation of the face pattern. The pattern is then derotated back to the upright position, which can be processed by the early developed system [15]. Schneiderman and Kanade [17] proposed a face/nonface recognizer based on the estimation of the posterior probability function, which captures the joint statistics of local appearance and position as well as the statistics of local appearance in the visual world. To detect side views of a face, profile images were added to the training set to incorporate such statistics [16]. Viola and Jones [19] presented a machine learning approach for face/nonface recognition. The novelty of their approach comes from the integration of a new image representation (integral image), a learning algorithm (based on AdaBoost), and a method for combining classifiers (cascade).

In the late 1980s, Sirovich and Kirby [27] developed a technique using PCA to efficiently represent human faces. Given a set of different face images, the technique first finds the principal components of the distribution of faces, expressed in terms of eigenvectors of the covariance matrix of the distribution. Each individual face in the face set can then be approximated by a linear combination of the largest eigenvectors, more commonly referred to as eigenfaces, using appropriate weights. Turk and Pentland [28] later developed this technique for face recognition. Their method exploits the distinct nature of the weights of eigenfaces in individual face representation. Since the face reconstruction by its principal components is an approximation, a residual error is defined in the algorithm as a preliminary measure of “faceness”. This residual error which they termed “distance–from–face–space” gives a good indication of face existence through the observation of global minima in a distance map.

Consistent with prior patient research on prosopagnosia [29,30], in functional magnetic resonance imaging (fMRI) studies a cortical region in the fusiform gyrus called the fusiform face area (FFA) has been shown to respond much more strongly to faces than to any other class of stimulus [31–33]. These findings have led to the hypothesis that the FFA contains specialized mechanisms for face processing.

The present paper illustrates a model for the gateway into the FFA. According to neuroimaging studies, a special gating mechanism exists that discriminates between faces and nonfaces and allows only the face into the FFA. Following up on this parallel and in agreement with these recent findings on face/nonface recognition mechanisms, we propose a model on the basis of mixture of experts architecture. To enhance the performance of the mixture network, a new structure for the expert networks is applied, which employs multilayer perceptrons (MLPs) to form the building blocks of the mixture architecture. But this application of MLPs in the expert networks, calls for a revised learning algorithm to best train this new mixture of experts networks.

The rest of this paper is organized as follows: Section 2 describes computational modeling of face/nonface recognition. Section 3 presents experimental results on face/nonface recognition and some discussions; and finally Section 4 concludes and summarizes the paper.
2. COMPUTATIONAL MODELING OF FACE/NONFACE RECOGNITION

In the present paper, we introduce a new and biologically plausible computational model for face/nonface recognition. The model consists of three processing layers: perceptual analysis, object representation, and recognition. The recognition layer, which is of vital importance, consists of two expert networks, each of which is dealing with a part of face/nonface space; in the distribution of this space, gating network learns to weight the expert network which is doing the best at solving the problem. The next section describes the model at length.

3. THE MODEL

We propose a new model based on biological pathways as shown in Figure 1. The first layer of the model is a set of neurons whose response properties are similar to those of complex cells in the visual cortex. A standard way to model these cells in other visual recognition networks is to use so-called “Gabor filters” [34]. As a matter of fact, these units do nonlinear edge detection at five different scales and eight different orientations. This means that the first layer of the model gives responses that are robust to small changes in the image. We term this level the “Perceptual” layer.

In order to avoid a high dimensional and redundant input space and optimally design and train the expert networks, the need to reduce the input space dimensions seems to be vigorous. To have an easier decision making we also need to completely separate the input classes by extracting orthogonal dimensions from the data. To solve the problem of high dimensional input space and the extraction of orthogonal dimensions from the data, we use PCA which is a biologically plausible means of dimensionality reduction in the sense that it is unsupervised, and can be learned by simple networks employing Hebbian learning rules [35]. The resulting low-dimensional object-level representation is specific to face processing, as is the population of so-called “face cells” in inferior temporal cortex [36]. We term this layer the “Representation” layer.

![Figure 1. The proposed model consists of three layers: perceptual analysis, object representation, and recognition.](image)

Although PCA is the simplest and most efficient methods for coding a set of stimuli [28], other methods such as linear discriminant analysis ‘LDA’ and independent-component analysis ‘ICA’ [37,38], would probably also work. For the current model, it is only important that the code be low dimensional, to facilitate generalization to new faces.

On the other hand, there are biological evidences indicating the existence of IT neurons which completely separate the input classes by extracting orthogonal dimensions from the data [39].

The implementation of PCA method consists of a number of easy steps: first normalizing the data, second calculating the covariance matrix; third calculating the eigenvectors and eigenvalues of the covariance matrix, and then choosing components and forming a feature vector.

The outputs of the representation layer are then recognized as two “faces and nonfaces” classes by a mixture of multilayer perceptron experts network that is a supervised modular neural network. We term this layer the “Recognition” layer.

Inside the recognition layer, we develop an efficient face/nonface recognition method which uses a type of mixture of multilayer perceptron experts architecture shown in Figure 2. Each classifier gives a
decision on which subspace each input space belongs to, and their outputs are combined through the gating network to produce the final decision. In addition to the robustness of the gating mechanisms to handle complex input spaces, recent studies in neurobiology and neuroimaging support this kind of structures: “…Indeed, it is hard to imagine how this could occur without a special gating mechanism that discriminates between faces and nonfaces and allows only the face information into the FFA…” [40].

3.1 Combining Methodology

From a computational viewpoint, according to the principle of divide and conquer, a complex computational task is solved by dividing it into a number of computationally simple tasks and then combining the solutions to those tasks. In supervised learning, computational simplicity is achieved by distributing the learning task among a number of experts, which in turn divides the input space into a set of subspaces. The combination of experts is said to constitute a combination of classifiers.

Combining classifiers are universal approximators. They may be classified into two major categories:

A. Static structures. In this class of combining classifiers, the responses of several predictors (experts) are combined by means of a mechanism that does not involve the input signal, hence the designation “static”. This category includes the following methods:

- Ensemble averaging, where the outputs of different predictors are linearly combined to produce an overall output.
- Boosting, where a weak learning algorithm is converted into one that achieves arbitrarily high accuracy.

B. Dynamic structures. In this second class of combining classifiers, the input signal is directly involved in actuating the mechanism that integrates the outputs of the individual experts into an overall output, hence the designation “dynamic” [41].

In the following we describe classic and our revised version of mixture of experts model. Mixture of experts is the most famous method in the category of dynamic structures of classifier combining.

Mixture of Experts: consider a modular neural network (Figure 2) in which the learning process proceeds by fusing self-organized and supervised forms of learning in a seamless fashion. The experts are technically performing supervised learning in that their individual outputs are combined to model the desired response. There is, however, a sense in which the experts are also performing self-organized learning; that is they self-organize to find a good partitioning of the input space so that each expert does well at modeling its own subspace, and as a whole group they model the input space well. The conventional learning algorithm of the mixture structure is introduced in [42], which is briefly described in Appendix.

Mixture of multilayer perceptron experts architecture: in our revised version of mixture of experts model we use MLPs, instead of linear networks or experts in Figure 2, to improve the performance of the expert networks. But, the application of MLPs in the structure of expert networks calls for a revision in the learning algorithm. In order to match the gating and expert networks, to endow the model the ability to select the expert network best at solving the problem, the learning algorithm is corrected by using an estimation of the posterior probability of the generation of the desired output by each expert. Using this new learning method, the MLP expert networks’ weights are updated on the basis of those estimations and this procedure is repeated for the training data set.

Each expert network is an MLP network with one hidden layer that computes an output vector $O_i$ as a function of the input stimuli vector, $x$, and a set of parameters such as weights of hidden layer and output layer and a sigmoid function as the activation function. We assume that each expert specializes in a different area of the input space. The gating network assigns a weight $g_i$ to each of the experts’ outputs, $O_i$. The gating network determines the $g_i$ as a function of the input vector $x$ and a set of parameters such as weights of the hidden layer, the output layer and a sigmoid function as the activation function. The $g_i$ can be interpreted as estimates of the prior probability that expert $i$ can generate the desired output $y$. The gating network is composed of two layers: the first layer is an MLP network, and the

---

1 The specificity of FFA mechanisms for face processing.
A Mixture of Multilayer Perceptron Experts Network for Face/Nonface Recognition

Figure 2. Sketch of the mixture experts network. The mixture of experts network is composed of expert networks and a Gating network. The expert network competes to learn the training patterns and the gating network mediates the competition.

The second layer is a Softmax nonlinear operator as the gating network’s output. Thus the gating network computes $O_g$, which is the output of the MLP layer of the gating network, then applies the Softmax function to get:

$$g_i = \frac{\exp(O_{g_i})}{\sum_{j=1}^{N} \exp(O_{g_j})} \quad i = 1, \ldots, N$$

where $N$ is the number of expert networks. So the $g_i$ are nonnegative and sum to 1. The final mixed output of the entire network is

$$O_f = \sum_i O_i g_i \quad i = 1, \ldots, N$$

The weights of MLPs are learned using the back-propagation (BP) algorithm, in order to maximize the log likelihood of the training data given the parameters. Assuming the probability density associated with each expert is Gaussian with identity covariance matrix, MLPs obtain the online learning rules:

$$\Delta w_y = \eta \cdot h_i (y - O_i) (O_i (1 - O_i)) Oh_i^T$$

$$\Delta w_h = \eta \cdot h_i \cdot w_y^T (y - O_i) (O_i (1 - O_i)) Oh_i (1 - Oh_i) x_i$$

$$\Delta w_{yg} = \eta \cdot (h - g) (O_g (1 - O_g)) Oh_g^T$$
\[ \Delta w_{sg} = \eta_g w_{sg}^\top (h - g)(O_g (1 - O_g))Oh_g (1 - Oh_g) x_i \]  

(6)

where \( \eta_e \) and \( \eta_g \) are learning rates for the expert networks and the gating network, respectively, \( Oh_i \) the output of the hidden layer of experts network , and \( h_i \) is an estimate of the posterior probability that expert \( i \) can generate the desired output \( y \):

\[ h_i = \frac{g_i \exp(-\frac{1}{2}(y-O_i)^\top(y-O_i))}{\sum_j g_j \exp(-\frac{1}{2}(y-O_j)^\top(y-O_j))} \]  

(7)

This can be thought of as a Softmax function computed on the inverse of the sum squared error of each expert’s output, smoothed by the gating network’s current estimate of the prior probability that the input pattern was drawn from expert \( i \)'s area of specialization. As the network’s learning process progresses, the expert networks “compete” for each input pattern, while the gate network rewards the winner of each competition with stronger error feedback signals. Thus, over time, the gate partitions the input space in response to the expert’s performance.

Figure 3 illustrates the experiments carried out on this work. Each experiment consists of three steps: generation of the feature vector, training the classifier and testing the classifier. In the first step, PCAs are generated inside the face images after the Gabor filtering. In the second step, the classifier is designed and trained. Finally in the third step, the performance of the classification is evaluated. This procedure is repeated for each learning algorithm by randomly choosing different training and test sets.

Figure 3. Schematic Diagram of the implementation steps of the proposed system.

4. EXPERIMENTAL RESULTS

For the training phase, we used a set of 3600 face images collected from Olivetti, UMIST, Harvard, Yale and FERET databases and from the Internet, cropped and resized to 40x40 pixels which have wide variations in pose, facial expression and lighting conditions (Figure 4). We collected nonface samples from images containing landscapes, trees, buildings, cars, rocks, flowers etc. The M_PCA matrix projects face patterns from a 1600–dimensional image space to a 50–dimensional subspace. It should be mentioned that during different experiments, a 50 dimensional subspace turned out to be the optimal case and the images were resized to an optimum size [43–45].
To evaluate the performance of mixture of multilayer perceptron experts network, we compared it with a complex MLP network. We experimented with a three–layer MLP (one hidden layer) with a number of hidden units. The weights of this MLP are learned using the back–propagation (BP) algorithm, which minimizes the Mean Square Error (MSE). During our experiments with MLPs, an optimal MLP structure was found.

To have a reasonable evaluation, we compare this complex structure (MLP) with the mixture model, consisting of two simple MLPs. The results show that two simple MLPs in mixture architecture perform the recognition task much better than a single but complex MLP in many ways (see below). In other words, a mixture structure, with simple MLPs as its building blocks, solves the problem of dealing with a complex MLP with serious problems in training phase and tuning its free parameters.

The MLP was trained and tested on the same images as the mixture of multilayer perceptron experts network. Both of them trained on the 2700 face and 3200 non–face samples. To compare the recognition performance of different networks more fairly, we give the trade off curve between correct recognition rate and false positive rate with variable decision thresholds. On the test image set, which consists of 900 faces and 900 nonfaces, the trade off curves of the MLP and the mixture of multilayer perceptron experts network are plotted in Figure 5. From the results, it is evident that the performance of the mixture of multilayer perceptron experts network is superior to that of the MLP. We repeated training and testing of different MLP topologies such as 50:40:1, 50:65:1, 50:85:1, 50:100:1 of MLPs for 5 times. The MLP with the structure of 50:85:1 was the best one, which is much more complex when compared to MLPs used in expert networks with the structure of 50:20:1. The details of the training parameters and the recognition rates are reported in Table 1. It should be mentioned other training parameters, i.e. $\eta_e$ and $\eta_g$, output node threshold value, and the number of hidden neurons for the gating and expert networks, were selected such that the error on the validation set was minimum.

### Table 1- The details of the training parameters as well as the recognition rates of testing the MLP and mixture of experts networks on the test set.

<table>
<thead>
<tr>
<th>Network</th>
<th>Topology</th>
<th>Output Node Threshold Value</th>
<th>Learning Rate</th>
<th>Training Epochs</th>
<th>Recognition Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLP</td>
<td>50:85:1</td>
<td>0.491</td>
<td>0.05</td>
<td>950</td>
<td>79.8</td>
</tr>
<tr>
<td>Mixture of Experts</td>
<td>Gating: 50:15:2</td>
<td>0.476</td>
<td>$\eta_e$: 0.054</td>
<td>450</td>
<td>97.6</td>
</tr>
<tr>
<td>Experts: 50:20:1</td>
<td>0.014</td>
<td>$\eta_g$: 0.014</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Overall, it is obvious that a mixture of simple MLPs is much more certain at solving a complicated problem, such as face/nonface recognition, than a complex MLP. And this is a remarkable point in the...
application of mixture models, as it seems clear that implementing a mixture of two simple MLPs is much more beneficial than using a complex MLP with a large number of free parameters. It should be taken into consideration that an appropriate training of such an extensive MLP turns to be inaccessible and troublesome; consequently it demonstrates a lower recognition rate and is less certain at its decision (Figure 5).

Figure 5. Recognition performance Curves. Mixture of MLPs exhibits much better performance in comparison with other networks.

5. CONCLUSION

Understanding how biological visual system performs object recognition is one of the ultimate goals in computational neuroscience. Recognition of face/nonface is a difficult problem, which confronts all the major challenges in computer vision and pattern recognition. In this paper we presented a model for the gateway to the right FFA which discriminates between face and nonface stimuli. In the face/nonface recognition task it came in useful to utilize a gating network simultaneously trained for selecting the expert with the best performance at solving the problem. Not only does it automatically decompose a complex problem into subproblems but it can also increase the robustness and the performance of the model to handle a larger face/nonface space. The novel method of using simple MLPs in the expert networks’ structure brings about a modification in the learning algorithm. So, the learning algorithm was adapted to estimate the posterior probability of the generation of the desired output by each expert network; the next step was to use those estimations to update the MLP expert networks’ weights. Comparisons between recognition performance of the mixture of two simple MLPs with a complex MLP reveals that a mixture structure is much more reliable at solving a complex pattern recognition problem such as face/nonface recognition; in addition it has the advantage of easier training, because of simpler expert networks and consequently less free parameters. Finally, we showed that a combination of gating mechanism and simple MLPs, in the mixture architecture, can be much more advantageous than a complex MLP in dealing with difficult problems.

APPENDIX:

Consider the network configuration of Figure 2, referred to as a mixture of experts model. Specifically, it consists of $N$ supervised modules called expert networks or simply experts, and an
integrating unit called a gating network that performs the function of a mediator among the expert networks.

**The learning rules of mixture of experts model**

In this modular neural network, the output layers of an array of linear classifiers is combined by a gating network, as shown in Figure 2. This network is trained with the maximum likelihood gradient ascent learning rules described in [42].

1. **Feed–forward phase**

   In the feed–forward stage, each expert network \( i \) is a single–layer linear network that computes an output vector \( O_i \) as a function of the input vector \( X \) and a set of parameters \( \theta_i \).

   It is assumed that each expert specializes in a different area of the input space. The gating network assigns a weight \( g_i \) to each of the experts' outputs \( O_i \). The gating network determines the \( g_i \) as a function of the input vector \( X \) and a set of parameters \( w \). The \( g_i \) can be interpreted as estimates of the prior probability that expert \( i \) can generate the desired output \( y \), or \( P(y|X,w) \). The gating network is a single–layer linear network with softmax nonlinearity at its output. That is, the linear network computes

\[
O_{gi} = \sum_j x_j w_{ij} \quad i = 1, \ldots, N
\]

where \( N \) is the number of expert networks. Then applies the softmax function to get

\[
g_i = \frac{\exp(O_{gi})}{\sum_{j=1}^{N} \exp(O_{gj})} \quad i = 1, \ldots, N
\]

Thus the \( g_i \) are nonnegative and sum to 1. The final, mixed output of the entire network is

\[
O_T = \sum_i O_i g_i \quad i = 1, \ldots, N
\]

2. **Adaptation by maximum likelihood gradient ascent**

   The network’s estimates of the parameters, \( w \) and \( \theta_i \), are adapted using gradient ascent algorithm for maximizing the log likelihood of the training data given the parameters. Assuming the probability density associated with each expert is Gaussian with identity covariance matrix, they obtain the online learning rules

\[
\Delta \theta_i = \eta_{\theta} h_i (y - o_i) X^T
\]

and

\[
\Delta w_i = \eta_w (h_i - g_i) X^T
\]

where \( \eta_{\theta} \) and \( \eta_w \) are learning rates for the expert networks and the gating network, respectively, and \( h_i \) is an estimate of the posterior probability that expert \( i \) can generate the desired output \( y \):

\[
h_i = \frac{g_i \exp\left(-\frac{1}{2} (y - O_i)^T (y - O_i)\right)}{\sum_j g_j \exp\left(-\frac{1}{2} (y - O_j)^T (y - O_j)\right)}
\]
This can be thought of as a Softmax function computed on the inverse of the sum squared error of each expert’s output, smoothed by the gating network’s current estimate of the prior probability that the input pattern was drawn from expert $i$’s area of specialization. As the network’s learning process progresses, the expert networks “compete” for each input pattern, while the gate network rewards the winner of each competition with stronger error feedback signals. Thus, over time, the gate partitions the input space in response to the expert’s performance.

REFERENCES


ABOUT THE AUTHORS

Reza Ebrahimpour was born in Mahallat, Iran, in July 1977. He received the B.S. degree in Electronics Engineering from Mazandaran University, Mazandaran, Iran and the M.S. degree in Biomedical Engineering from Tarbiat Modarres University, Tehran, Iran, in 1999 and 2001, respectively. He received his Ph.D. degree in July 2007 from the School of Cognitive Science, Institute for Studies on Theoretical Physics and Mathematics, where he worked on view-independent face recognition with mixture of experts. His research interests include Human and Machine Vision, Neural Networks, Pattern Recognition.

E. Kabir is a professor of electrical engineering at Tarbiat Modarres University. He obtained B.Sc. and M.Sc. degrees in electrical and electronics engineering from the University of Tehran. He received his Ph.D. degree in 1990 from the University of Essex, where he worked on the recognition of handwritten postal addresses. His main areas of research are handwriting recognition and document image analysis.

Hossein Esteky received his MD from Tehran University School of Medicine, Tehran, Iran, in 1987, and Ph.D. degree in Neuroscience from University of North Texas, USA, in 1994. From 1996 to 1998 he was a Postdoctoral training, Research fellow at Laboratory for Cognitive Brain Mapping Brain Science Institute, RIKEN, Japan, where he served as the affiliated senior researcher at the Group for Brain and Cognitive Mapping since 1998. He joined the department of Physiology, Shaheed Beheshti School of Medicine, Tehran, Iran as a Professor in 1995, and in 2000 as the Director of the Research Group for Brain and Cognitive Sciences. Between 2000 and 2002 he was the assistant to the director of IPM School of Cognitive Sciences, Tehran, Iran, where he became the Director of the School of Cognitive Sciences in 2002. Professor Esteky has been the member of the Editorial board of journal of progress in neurobiology since 2006. His main research interests include System and Cognitive neuroscience, Neural Basis of Visual Object Recognition and Categorization, Visual Learning and Memory.

Mohammad Reza Yousefi received the B.S. degree in electronics engineering from Shahid Rajaee University in 2007. His research interests are Neural Networks, Pattern Recognition, Multiple Classifier Systems, Computational Vision and Models of Human Vision.