Face Recognition Committee Machine

Tang Ho Man
Department of Computer Science and Engineering
The Chinese University of Hong Kong
Shatin, Hong Kong
email: hmtang@cse.cuhk.edu.hk

November 27, 2002

Abstract

Face recognition has been of interest to a growing number of researchers due to its applications on security. Within past years, there are numerous face recognition algorithms proposed by researchers. However, there is no unified framework for the integration. In this paper, we implement different existing well-known algorithms - Eigenface, Fisherface, Elastic Graph Matching (EGM), Support Vector Machine (SVM) and neural network - to give a comprehensive testing under same face databases. Besides, we present a Face Recognition Committee Machine (FRCM), which is a novel approach for assembling the outputs of various face recognition algorithms to obtain a unified decision with improved accuracy. We have tested our system with ORL face database and Yale face database. A comparative experimental result of different algorithms with the committee machine demonstrates that the proposed system achieves improved accuracy over the individual algorithms. Furthermore, we present a distributed face recognition system to demonstrate how FRCM can be applied on low memory and low CPU power machine for real-time applications.

1 Introduction

In recent years, organizations search for more secure methods for user access, e-commerce, and other security applications. Three main types of authentication are taken into consideration: 1) password authentication, 2) card key authentication and 3) biometric authentication. The study of biometrics is gaining increasing attention in recent years because it is the most secure and convenient tool among the three. It cannot be borrowed, stolen, or forgotten, and forging one is practically impossible. Several biometric signals are suitable for security application; facial patterns, voice patterns, eye retinas, irises, fingerprint and signatures are instances. Among these signals, facial patterns are the most convenient signals for security applications. Users do not need to scan his fingerprint or iris for personal identification but just need to stand in front of a camera to take an image. The system can then check its database to recognize the person from the photo.

1.1 Face Recognition

Face recognition, by definition, is a process of identifies a person from a group of known people by analyzing facial characteristics of that person. Face recognition includes two phases: 1) training phase and 2) recognition phase. In training phase, images of different people (known identity) are taken and form a face database. These images are used to train the face recognition system so that the system is ready to recognize a person in recognition phase. In recognition phase, an image of the person being recognized is taken and passed into the system. The image is then compared with other images in the face database. The image with highest similarity in the face database would be chosen as the recognized image and that person's identity would be the identify of the recognized image.

1.2 Applications

Face recognition can be applied to various kinds of applications. Security system like computerized access control is one of an instance. With face recognition, images of a group of authorized people are taken as the face database and train the system. Whenever a person enters the system, an image of that person is taken. Only recognized people can get access to the system, people do not in the face database would be rejected. With this kind of access control system, important data or place can be securely protected.

Airport security system is another application of face recognition. After September 11 attacks in USA, airport security is highly concerned. Face recognition has advantages over other biometrics because it is difficult to collect fingerprint or iris signals from terrorist groups but easy to take images of them. All passengers in the airport are screened and any possible terrorists are reported. Keflavik International Airport in Iceland is the first airport to use the face recognition technology to prevent terrorist attack.

Apart from the security application, face recognition can be applied in other fields like multimedia search engine. Fast growing on multimedia technology and Internet technology enables searching for multimedia data like video clips possible. However, information retrieval within vast amount of multimedia data is still a challenging task. With face recognition and video segmentation technology, we can find video clips of a particular person easily by simply supply with the search engine a picture of that person. All related video like news clips would be found.

1.3 Objectives

- **Comparison of different face recognition algorithms:** Face recognition has raised extensive attentions since 1990. There are numerous algorithms proposed by researchers which claimed to have satisfactory result. However, the algorithms are tested under different frameworks. There is no comparison of the algorithms under the same framework. In this paper, we give a comprehensive comparison of the five well-known algorithms, namely Eigenface, Fisherface, Elastic Graph Matching (EGM), Support Vector Machine (SVM) and Neural Network on two well-known face databases: ORL and Yale face database.

- **Face Recognition Committee Machine:** We present a novel Face Recognition Committee Machine (FRCM) consisting of five experts mentioned above. It fuses the knowledge acquired by the experts to arrive at a unified decision. As each expert shows various performance on different conditions, we can obtain a final decision with better accuracy over individuals by assembling the results of the experts.

- **Distributed Face Recognition System:** We present a real-time distributed face recognition system which are capable for 1) detect a face and 2) recognize a face on a low memory and low CPU power machine. The system adopts client-server approach on the FRCM which distributes the detected face image to different experts (servers) for recognition. The results of each expert are ensemble in the client for final decision.

1.4 Outline

The remainder of this paper is organized as follows. Section 2 gives a background review of four algorithms used in the committee machine. Section 3 describes the proposed FRCM. Section 4 describes the Distributed Face Recognition System. Section 5 presents and discusses the experimental results. A conclusion and future work are given in Section 6.

2 Background Review

With past years, numerous face recognition methods are proposed by researchers. Among the methods , **Eigenface** [1] is the most popular one due to its effectiveness. It makes use of Principal Component Analysis (PCA) to find a feature space for projection of face images. A similar approach, **Fisherface** [2], is proposed later which makes use of Fisher's Linear Discriminant (FLD) instead of PCA. Apart from template matching approaches, **Elastic Graph Matching** (EGM)[3] is proposed which takes into account the human facial features by extracting the features with Gabor wavelet transform. Recently, **Support Vector Machine** (SVM) [4] is getting importance in face recognition. Based on statistical theory by Vapnik, several SVM algorithms [5][6]are developed by researchers and are proved with impressive result. In this section, we would give a review on four methods mentioned above.

2.1 Eigenface

Eigenface was first proposed by Sirovich and Kirby in 1987 [1] as an application of **Principal Component Analysis (PCA)**. Pentland and Turk refined the method by adding preprocessing and procedures of face detection in 1991 [7]. The main idea of eigenface is to get the features in mathematical sense instead of physical face feature by using mathematical transform for recognition.

There are two phases for face recognition using eigenfaces. The first phase is the training phase. In this phase, a large group of individual faces is used as the training set. The size, orientation and light intensity should be standardized. For example, all images are of size 128 x 128 pixels and all are frontal faces. Each of the images in the training set is represented by a vector of size N by N, with N representing the size of the image. With the training image$s$, a set of eigenvectors is found by PCA.

The basic idea of PCA is to take advantages of the redundancy existing in the training set for representing the set in a more compact way, so that the dimension of the image can be greatly reduced. It works by finding eigenvectors and eigenvalues of covariance matrix C from training set images $\{T_1,T_2,\ldots, T_M\}$

$$C = \frac{1}{M}\sum_{i=1}^{M}(T_i - \psi)(T_i - \psi)^T, \tag{1}$$

where ψ is the average face. With PCA, an image can be represented by E eigenvectors where E is the number of eigenvector used. As E is much smaller than N^2, comparison between vectors would be efficient. The span of the E eigenfaces are called face space.

The second phase is recognition phase. Input image and training images are first subtracted by the average face ψ and are then projected on the eigenfaces. The Euclidean distances of the input image with the training set images are then computed. The training set image with minimum distance from the input image should be the best match.

2.2 Fisherface

Fisherface was suggested by Belhumeur et al in 1997 [2]. It is similar to Eigenface that both methods are template matching method, which makes use of projection of images into a feature space. However, Fisherface uses **Fisher's Linear Discriminant (FLD)** instead of PCA. PCA projection is best for reconstruction of images from a low dimensional basis. However, this method doesn't make use of between-class scatter. The projection may not be optimal from discrimination for different classes. FLD projection maximizes the ratio of between-class scatter to that of within-class scatter. The idea is that it tries to "shape" the scatter in order to make it more reliable for classification. Let the between-class scatter matrix be defined as

$$S_B = \sum_{k=1}^{C} N_i(\psi_k - \psi)(\psi_k - \psi)^T, \tag{2}$$

where C is the number of classes and N_i is the number of samples in class T_i .And the within-class scatter matrix be defined as

$$S_W = \sum_{i=1}^{C} \sum_{T \epsilon C_i} (T_k - \psi_i)(T_k - \psi_i)^T, \tag{3}$$

where ψ_i is the mean image of class T_i. The optimal projection W_{opt} is chosen as the matrix with orthonormal columns, which maximizes the ratio of the determinant of the between-class scatter matrix of the projected samples to the determinant of the within-class scatter matrix of the projected samples as follows:

$$W_{\text{opt}} = \arg \max_W \frac{|W^T S_B W|}{|W^T S_W W|}. \tag{4}$$

2.3 Elastic Graph Matching

Elastic Graph Matching [3][8] was suggested by von der Malsburg et al in 1993, which is based on the dynamic link architecture [9]. Each facial feature is extracted by Gabor wavelet transform on the fiducial points as a jet J. A face is represented by an image graph G consisting of N nodes of jets. Test image graph G^I is compared to all modal graphs G^M by the cost function:

$$C_{\text{total}}(G^I, G^M) = \lambda S_e(G^I, G^M) - S_v(G^I, G^M), \tag{5}$$

where λ is rigidity coefficient, S_e is edge comparison function and S_v is vertex similarity function. S_e is defined as:

$$S_e(G^I, G^M) = \frac{1}{E} \sum_e (\Delta \vec{x}_e^I - \Delta \vec{x}_e^M)^2, \tag{6}$$

where E is the number of edges and $\Delta \vec{x}_e$ are the distance vectors used as labels at edge e. And S_v is defined as:

$$S_v(G^I, G^M) = \frac{1}{N} \sum_n \frac{J_n^I \cdot J_n^M}{J_n^I J_n^M}, \tag{7}$$

where N is the number of nodes and J_n are the jets at nodes n. The training set image with minimum cost would be the best match.

2.4 Support Vector Machine

Support Vector Machine [4] was developed by Vapnik et al. in the late seventies, which is based on Structural Risk Minimization principle from statistical learning theory. It aims to minimize an upper bound on the expected generalization error. Recently, SVM has been applied to face detection and face recognition. Osuna E. used SVM on face detection in 1997[10]. Guo and Kim used SVM on face recognition in 2000 and 2001 respectively[5][11]. Dihua used SVM on facial component extraction and face recognition in 2002 [6].

For classification, SVM takes training sample S_n $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ as input of dimension n from some unknown probability distribution $P(x, y)$. $x_i \epsilon \Re^n$ represents the feature vector and $y_i \epsilon \{-1, +1\}$ represents the class label [12]. For linearly separable data, SVM looks for a separating hyperplane which separates the data with the largest margin. The decision rules is defined as

$$h(x) = \text{sign}(w * x + b) \tag{8}$$

$h(x) = +1$ if $w * x + b > 0$ and $h(x) = -1$ if $w * x + b < 0$ where w is normal to the hyperplane and b is a threshold. As shown in Figure 1, the sample closest to the hyperplane $h$are called Support Vectors (circled). The distance between the support vectors are called margin. According to which sides a feature vector x_i lies on the hyperplane, the sample is classified to either class+1 or -1.

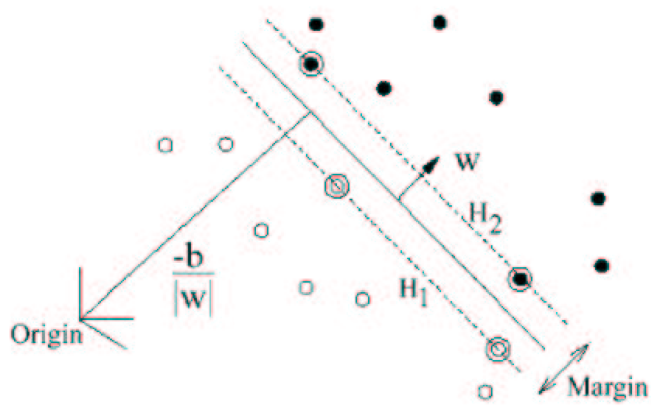

Figure 1: Separating hyperplane for two classes

For linearly non-separable data, it maps the data into a high dimensional space $\boldsymbol{x \epsilon \Re^I} \mapsto \Phi(x) \epsilon \Re^h$ with kernel function $K(a, \text{xb}) = \Phi(a) \cdot \Phi(b)$ to find the hyperplane. There are three popular kernel functions being used

- Polynomial kernel: $K_{\text{poly}}(d_1, d_2) = (d_1 * d_2 + 1)^d$
- Radial basis kernel: $K_{\text{rbf}}(d_1, d_2) = \exp((d_1 - d_2)^2)$
- Hyperbolic tangentl: $K_{\text{sigmoid}}(d_1, d_2) = \tanh(s(d_1 * d_2) + c)$

3 Face Recognition Committee Machine

3.1 Committee Machine

Committee machine has been widely used in neural networks. A number of researchers have applied it to improve the performance of a neural network [13][14]. The basic idea of a committee machine is to ensemble a mixture of experts and to combine the result of each expert. There are mainly two kinds of committee machines:

1. **Static Structure**: This is generally known as an ensemble method. Input data is not involved in combining the committee experts. Examples includes ensemble averaging and boosting.

2. **Dynamic Structure**: Input is directly involved in the combining mechanism that employs an integrating unit to adjust the weight of each expert according to the input.

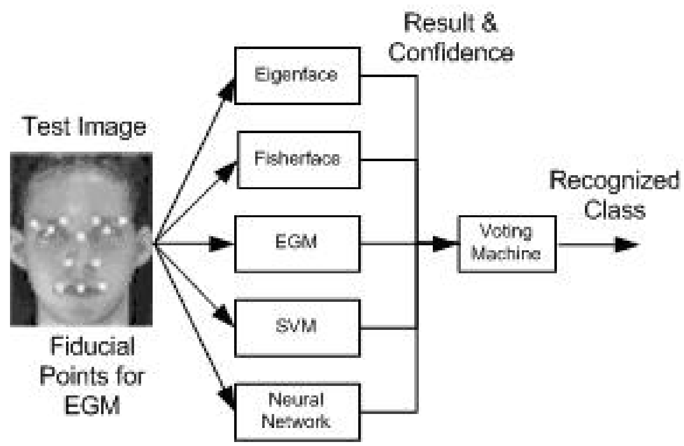

Figure 2: FRCM System Overview and Fiducial points

Figure 2 provides an overview of our FRCM. Our proposed FRCM adopts the static structure with five well-known experts in face recognition. All are proved with good classification ability in the literature [15][5]. As shown in the figure, input image is sent into the five experts for recognition. Each expert gives its **result** r and **confidence** c for the result to the voting machine which are then ensembled to obtain a final decision.

3.2 Confidence and Result

In FRCM, apart from using **result** of each expert, we introduce the use of **confidence** as a weighted vote for the voting machine to avoid low confidence result of individual expert from affecting the final result. Due to different nature of the experts, we adopt different approaches to find the results and confidences.

- **Eigenface**, **Fisherface** and **EGM**: We use K nearest-neighbor classifiers. Five nearest training set projections with the test image projection are chosen for Eigenface and Fisherface, and five training set graphs with the lowest cost are chosen for EGM. The final result for expert i is defined as the class j with the highest votes v in J classes among the five results:
$$r(i) = \arg\max_j(v(j)), \tag{9}$$
where its confidence is defined as the number of votes of the result class divided by K, i.e.,
$$c(i) = \frac{v(r(i))}{K}. \tag{10}$$

- **SVM**: As SVM was originally developed for two-class classification, multi-class classification can be extended by using "one-against-one" approach. To recognize a test image in J different classes, ${}_JC_2$ (i.e., $\frac{J(J-1)}{2}$) SVMs are constructed. The image is tested against each SVM and the class j with the highest votes in all SVMs is selected as the recognition result $r(i)$. The confidence is defined as the number of votes of the result class divided by $J-1$:
$$c(i) = \frac{v(r(i))}{J-1}, \tag{11}$$
where $J-1$ is the maximum number of vote a class could obtain.

- **Neural network**: We choose a binary vector of size J for the target representation in training phase. In the binary vector, the target class is set to one and the others are set to zero. In recognition phase, the class j with output value closest to 1 in the output vector is chosen as the result and the output value is chosen as the confidence.

The weights in FRCM are evaluated in our testing for different algorithms under ORL and Yale face database. We take the average accuracy for the algorithms as weights (shown in table 4 and table 5 respectively). The use of weights in the voting machine further reduces the chance for an expert who performs poorly on average from affecting the ensemble result even if it has high confidence on the result. After collecting the **result** r and **confidences** c from the five experts, the voting machine assembles the results by calculating the **score** s of each class as follows:
$$s(j) = \sum_{i=1}^{5} w(i) * c(i), \forall j \in r(i). \tag{12}$$
Score is a multiple of the confidence of an expert on a result and the weight of the expert. We define the score in such a way that only experts with high performance on average and high confidence on the result would take most significant score in the final decision.

4 Distributed Face Recognition System

We have implemented a Face Recognition System which is an implementation of the FRCM mentioned above. The system consists of two main modules: 1) **Face Detection** and 2) **Face**

Recognition. Face is first extracted from a web-camera by Face Detection, the detected face is passed to the Face Recognition modules for further recognition. Face detection is another challenging problem which would not be explained in detail in this paper. In our implementation, we employ the color model to identify possible human face and use neural network to validate the human face. The face image found is then passed into the FRCM. The final ensemble result from the FRCM will be shown to user. Figure 3 gives a snapshot of the system and Table 1 lists some of the implementation details of the system for reference.

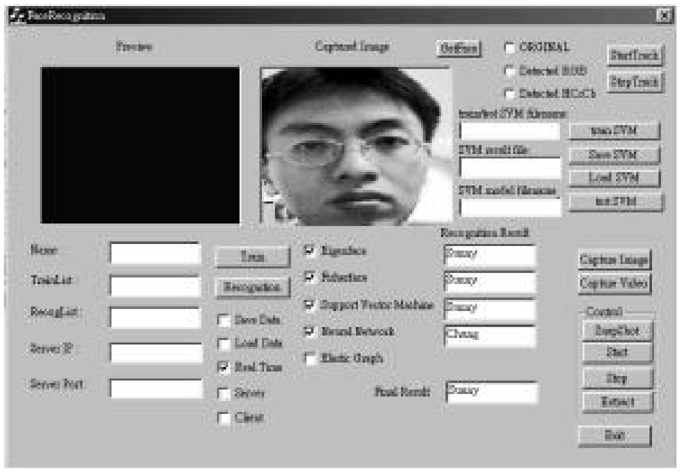

Figure 3: System Snapshot

Implementation	**Detail Description**
Committee Machine	Static structure with 5 expert
	Average Performance used as weight function
	Adopt client/server approach
Eigenface	50 eigenvectors used
EGM	40 Gabor filters (8 orient. & 5 freq.) used
	12 fiducial points selected (Fig 2) manually
SVM	Polynomial kernel function used
	"One-against-One" approach used
Neural Network	Feed forward backpropagation network
	Fisher projection used as feature vector
	40 hidden nodes used
Programming Lang.	Visual C++
Operating System	Windows

Table 1: Implementation details of FRCM

As face recognition is becoming more mature and popular nowadays. It is desirable to have face recognition application on some mobile devices like notebook, Personal Data

Assistant (PDA) or even mobile phone in the near future. However, face recognition requires large memory storage and high CPU power which is not available on mobile device.

4.1 Memory and Processor Requirement

Huge **memory** is required to store a number of models or projections for different algorithms. Table 2 lists some of the storage requirements of ORL database and Yale database for the five algorithms mentioned. From the table, SVM requires largest storage (55MB and 34MB respectively). This is due to the fact that it stores most information from all the training set images. Other methods, except neural network, require less storage but they still requires several megabytes memory. Therefore, it is not feasible to have a face recognition application on some memory limited machine like PDA, mobile phone.

Algorithm	**ORL Database**	**Yale Database**
Eigenface	5MB	8MB
Fisherface	4MB	2MB
EGM	2MB	1MB
SVM	55MB	34MB
Neural Network	42KB	18KB

Table 2: Storage requirements for different algorithms

Apart from memory requirement, **processing** power is another important consideration. It is desirable to have a real-time response from the machine. However, most mobile device do not have high processing power to cope with the need. Take PDA as an example, the most advanced model for Pocket PC only has 400MHz processor (X-Scale processor). For other models, popular processor used has processing power of 33MHz, 66MHz or 200MHz, which is not enough for real-time face recognition.

4.2 Distributed System

Distributed system would be a solution to solve the storage and processing limitation on mobile device. We adopt client-server approach for the implementation of FRCM. The most memory and CPU consuming job can be processed in the server. Whenever a client receives an image from Face Detection Module, it distributes the image into different experts in server for recognition. The experts can be located in a single server or distinct servers. Once the experts finish recognition, the corresponding results and confidences are returned to the voting machine in the client. Final decision would be made in client and shown to user.

With the distributed system, client do not need high processing power and memory. Client only need to 1) capture image, 2) send the image and 4) ensemble the results as shown in Figure 4. Therefore, it is feasible to employ face recognition on mobile device.

We have implemented the client in a way that several threads are created to handle different experts in FRCM. Client can connect to distinct servers to speed up processing time. The client is implemented on a notebook for testing. The processing time is listed in Table

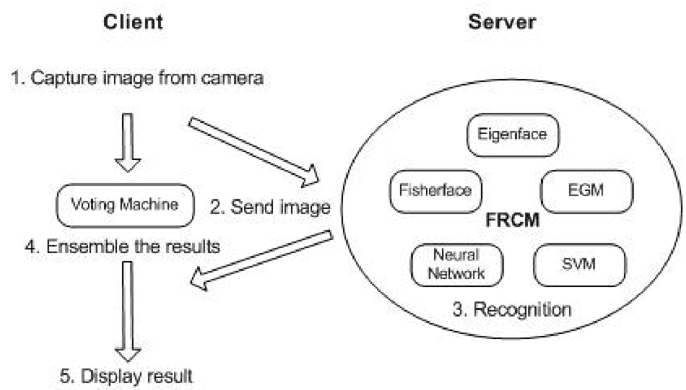

Figure 4: Distributed Face Recognition System

3. We have tested the system on two machines, a desktop and a notebook with processor 1400MHz and 300MHz respectively. The result shows that there is no big difference in the processing time for recognition on a desktop or notebook computer. The time required differs by one second only. However, long time is required for loading the models and projections on a notebook computer. With the use of distributed system, the startup time is greatly reduced.

At current state, we have not implemented client program on PDA or mobile phone. However, we believe that the proposed distributed architecture can greatly increase the recognition time for PDA and mobile phone as their processing power is much lower than the notebook computer.

Machine for Testing	**Time (S+R)**	**Time (R)**
PIV 1400MHz(Desktop)	13s	1s
PII 300MHz (Notebook)	93s	2s
PII 300MHz Client + PIV 1400MHz Server	16s	2s

Table 3: Processing time: S: Startup, R: Recognition

5 Experimental Results

Two sets of experiments are presented to evaluate the performance of FRCM and individual algorithms. We adopt leaving-one-out cross validation method for the experiment. For a given sample of n images in a class, a classifier is trained using $(n-1)$ images in that class and tested on the remaining single case. The test repeats n times, each time training a classifier with leaving-one-out. Thus, all images are used for training and testing to produce a thorough result.

5.1 The ORL Database of Faces

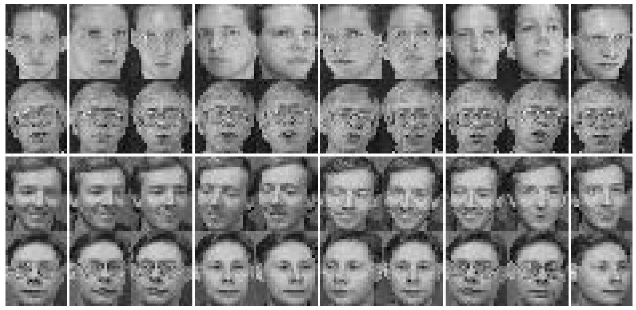

Figure 5: Snapshot of ORL database

First experiment is performed on the ORL face database from AT&T Laboratories Cambridge. The images are gray-scale with a resolution of 92×112 pixels. The database contains 400 images, including 40 distinct people, each with 10 images that vary in position, rotation, scale and expression. The images are taken under constant lighting condition. Figure 5 shows a snapshot of 4 individuals.

From the ORL result shown in Table 4, FRCM (98.8%) has improvement in accuracy over the individual algorithms in the testing. We notice that Fisherface and SVM obtain higher accuracy (over 97%) than the others. This is due to the fact that both Fisherface and SVM inherits better classification ability in general cases. We can see the effect of the committee machine in image set 7 that none of the experts has 100% accuracy but FRCM achieves it. The result also demonstrates that with the use of confidence and weight function, poor result from some experts would not affect the ensemble result significantly.

Image Set	Eigenface	Fisherface	EGM	SVM	NN	FRCM
1	92.5%	**100.0%**	90.0%	95.0%	92.5%	**95.0%**
2	85.0%	**100.0%**	72.5%	**100.0%**	95.0%	**100.0%**
3	87.5%	**100.0%**	85.0%	**100.0%**	95.0%	**100.0%**
4	90.0%	97.5%	70.0%	**100.0%**	92.5%	**100.0%**
5	85[.0%	**100.0%**	82.5%	**100.0%**	95.0%	**100.0%**
6	87.5%	**97.5%**	70.0%	**97.5%**	92.5%	**97.5%**
7	82.5%	**95.0%**	75.0%	**95.0%**	**95.0%**	**100.0%**
8	92.5%	95.0%	80.0%	**97.5%**	90.0%	**97.5%**
9	90.0%	**100.0%**	72.5%	97.5%	90.0%	**100.0%**
10	85.0%	**97.5%**	80.0%	95.0%	92.5%	**97.5%**
Average	87.5%	**98.3%**	77.8%	97.8%	93.0%	**98.8%**

Table 4: ORL Result

Figure 6: Snapshot of cropped Yale database

5.2 Yale Face Database

Second experiment is performed on Yale face database from Yale University. The images are gray-scale and are cropped to a resolution of 116×136 pixels. The database contains 165 images, including 15 distinct people, each with 11 images that vary in both expression and lighting. A snapshot of 4 individuals in the database is shown in Figure 6. The result of FRCM on Yale database is given in Table 5.

Image Set	**Eigenface**	**Fisherface**	**EGM**	**SVM**	**NN**	**FRCM**
centerlight	53.3%	**93.3%**	66.7%	86.7%	73.3%	**93.3%**
glasses	80.0%	**100.0%**	53.3%	86.7%	86.7%	**100.0%**
happy	93.3%	**100.0%**	80.0%	**100.0%**	93.3%	**100.0%**
leftlight	26.7%	26.7%	**33.3%**	26.7%	26.7%	**33.3%**
noglasses	**100.0%**	**100.0%**	80.0%	**100.0%**	**100.0%**	**100.0%**
normal	86.7%	**100.0%**	86.7%	**100.0%**	93.3%	**100.0%**
rightlight	26.7%	**40.0%**	**40.0%**	13.3%	26.7%	**33.3%**
sad	86.7%	93.3%	93.3%	**100.0%**	93.3%	**100.0%**
sleepy	86.7%	**100.0%**	73.3%	**100.0%**	**100.0%**	**100.0%**
surprised	**86.7%**	66.7%	33.3%	73.3%	66.7%	**86.67%**
wink	**100.0%**	**100.0%**	66.7%	93.3%	93.3%	**100.0%**
Average	75.2%	**83.6%**	64.2%	80.0%	77.6%	**86.1%**
No Night	85.9%	**94.8%**	70.4%	93.3%	88.9%	**97.8%**

Table 5: Yale Result

From the Yale result, FRCM (86.1%) also outperforms all the individuals on average. The main reason for some non-satisfactory result is due to the fact that Yale database contains variations in left and right lighting (4^{th} and 7^{th} column in Fig. 6). The accuracy for both leftlight and rightlight in FRCM is 33.0% only. For algorithms taking the whole image as input like Eigenface, the accuracy would drop significantly because the lighting would greatly affect the pixel values. We notice that EGM works relatively better in the light testings than other algorithms. This is due to the use of Gabor wavelet transformation of fiducial points in EGM rather than in the whole image. Without the lighting variations, FRCM achieves 97.8% accuracy, which is comparable to the ORL result (98.8%).

6 Conclusion and Future Work

In this paper, we perform a comprehensive experiment on five well-known face recognition algorithms to compare the accuracy of the algorithms under the same framework. We conclude that Fisherface and SVM are the best classifiers among them. Both achieves over 93% accuracy in general cases. However, none of them has high accuracy under lighting variation in Yale test.

Besides, we propose a Face Recognition Committee Machine. We introduce the use of confidence on experts' results and weight function on the committee machine which can reduce the chance for poor result of certain expert from affecting the ensemble result. The success has been demonstrated on the result of ORL and Yale test. In both tests, FRCM outperforms all other individual algorithms. This shows that the use of committee machine works in improving the accuracy of recognition. In our experiments, FRCM achieves 98.8% accuracy in ORL test and 97.8% accuracy in Yale test (without lighting variation).

We have also implemented a distributed face recognition system for real-time face recognition. The system consists of face detection and face recognition modules which can detect any human face and recognize the face with the FRCM. We have proposed a distributed architecture for the FRCM so that we can employ the system on mobile devices with low storage and low processing power.

In the Yale test, we notice that FRCM doest not perform satisfactorily on rightlight and leftlight testing. The reason for this is due to the lack of an expert in the committee machine which can accurately recognize a face under various lighting condition. Our future work will focus on including an expert for lighting variation like Illumination Cone [16] in order to make further improvement.

References

[1] L. Sirovich and M. Kirby, "A low-dimensional procedure for the characterization of human faces," in *J. Opt. Soc. Amer. A*, 1987, vol. 4, pp. 519–524.

[2] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman, "Eigenfaces vs. fisherfaces: recognition using class specific linear projection," in *IEEE Trans. on Pattern Analysis and Machine Intelligence*, July 1997, vol. 19, pp. 711–720.

[3] M. Lades, J.C. Vorbruggen, C. von der Malsburg, R. P. Wurtz, and W. Konen, "Distortion invariant object recognition in the dynamic link architecture," in *IEEE Trans. on Computers*, March 1993, vol. 42, pp. 300–311.

[4] Vladimir Vapnik, *Statistical Learning Theory*, John Wiley and Sons, Inc., 1998.

[5] Guo Guodong, S.Z. Li, and Chan Kapluk, "Face recognition by support vector machines," in *Proc. Fourth IEEE International Conf. on Automatic Face and Gesture Recognition 2000*, 2000, pp. 196–201.

[6] Xi Dihua, I.T. Podolak, and Lee Seong-Whan, "Facial component extraction and face recognition with support vector machines," in *Proc. Fifth IEEE International Conf. on Automatic Face and Gesture Recognition, 2002*, May 2002, pp. 76–81.

[7] M. Turk and A. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1991, pp. 586–591.

[8] J. M. Wiskott, L.and Fellous, N. Kuiger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," in *IEEE Trans. on Pattern Analysis and Machine Intelligence*, July 1997, vol. 19, pp. 775–779.

[9] J. Buhmann, J. Lange, and C. von der Malsburg, "Distortion invariant object recognition by matching hierarchically labeled graphs," in *Proc. IJCNN. International Joint Conf. on Neural Network, 1989*, 1989, pp. 155–159.

[10] E. Osuna, R. Freund, and F. Girosit, "Training support vector machines: an application to face detection," in *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition 1997*, 1997, pp. 130–136.

[11] Kim K.I., Kim J., and Jung K., "Recognition of facial images using support vector machines," in *Proc. 11th IEEE Signal Processing Workshop on Statistical Signal Processing 2001*, 2001, pp. 468–471.

[12] J. C. Burges Christopher, "A tutorial on support vector machines for pattern recognition," in *Data Mining and Knowledge Discovery*, 1998, vol. 2, pp. 121–167.

[13] Zhang B.T. and Joung J.G., "Building optimal committees of genetic programs," in *Parallel Problem Solving from Nature – PPSN VI*, 2000, pp. 231–240.

[14] Min Su and M. Basu, "Gating improves neural network performance," in *Proc. IEEE Conf. on IJCNN '01*, 2001, vol. 3, pp. 2159–2164.

[15] Jun Zhang, Yan Yong, and M. Lades, "Face recognition: eigenface, elastic matching, and neural nets," in *Proc. IEEE*, 1997, vol. 85, pp. 1423–1435.

[16] A.S. Georghiades, P.N. Belhumeur, and D.J. Kriegman, "From few to many: illumination cone models for face recognition under variable lighting and pose," in *IEEE Trans. on Pattern Analysis and Machine Intelligence*, June 2001, vol. 23, pp. 643–660.