Face Recognition Based On Vector Quantization Using Fuzzy Neuro Clustering

Elizabeth B. Varghese, M. Wilscy

Abstract—A face recognition system is a computer application for automatically identifying or verifying a person from a digital image or a video frame. A lot of algorithms have been proposed for face recognition. Vector Quantization (VQ) based face recognition is a novel approach for face recognition. Here a new codebook generation for VQ based face recognition using Integrated Adaptive Fuzzy Clustering (IAFC) is proposed. IAFC is a fuzzy neural network which incorporates a fuzzy learning rule into a competitive neural network. The performance of proposed algorithm is demonstrated by using publicly available AT&T database, Yale database, Indian Face database and a small face database, DCSKU database created in our lab. In all the databases the proposed approach got a higher recognition rate than most of the existing methods. In terms of Equal Error Rate (ERR) also the proposed codebook is better than the existing methods.

Keywords—Face Recognition, Vector Quantization, Integrated Adaptive Fuzzy Clustering, Self Organization Map.

I. INTRODUCTION

FACE recognition is a biometric approach that employs automated methods to verify or recognize the identity of a person based on his/her physiological characteristics. Machine Recognition of faces from still and video images is emerging as an active research area spanning several disciplines such as image processing, pattern recognition, computer vision, neural networks etc. Face recognition technology has numerous commercial and law enforcement applications. Applications range from static matching of controlled format photographs such as passports, credit cards, photo IDs, driver's licenses to real time matching of surveillance video images [1].

A lot of algorithms have been proposed for solving face recognition problem [2]. Among these algorithms Eigenfaces based approach [3], [4], [5] which uses Principal Component Analysis (PCA) was the most common one. Fisherfaces [4] which uses Linear Discriminant Analysis (LDA) were also used. Later PCA and LDA are used for feature extraction [4], [6]. Bartlett et al. [7] provided two architectures of Independent Component Analysis for face recognition process. A specific kind of genetic algorithm called Evolutionary Pursuit (EP) [8] was also used in face recognition. In Elastic Bunch Graph Matching (EBGM) all human faces share a similar topological structure [9]. Here

F. Elizabeth B Varghese is with Mar Baselios College of Engineering, Nalanchira, Trivandrum, Kerala, India (Phone: +919995276967, e-mail: eliza.b.varghese@gmail.com).

S. M. Wilscy, is with Department of Computer Science, University of Kerala, Karyavattom Campus,Trivandrum, Kerala, India (Phone: +919447462529, email: wilsyphilipose@gmail.com).

faces are represented as graphs, with nodes positioned at facial points (eyes, nose...) and edges labeled with 2-D distance vectors. Support Vector Machines (SVM) [10] can be used to classify the face images. Neural Network based [11] approaches are also available for classification of face images. Hidden Markov Models (HMM) [12] uses the statistical properties of face images for face recognition. Feature-based approach [13] such as Local Feature Analysis (LFA) [14] and local autocorrelations and multiscale integration technique [15] are also present.

Vector Quantization (VQ) [16] is used in face recognition. VQ algorithm [16] is well known in the field of image compression. In VQ a codebook is prepared in advance using which the data or image is compressed. This codebook information is used in face recognition, as follows. The face image is divided into blocks and the size of each block is same as that of the size of the codevectors in the codebook. Each block of the face image is then compared with the codevectors. Matched frequencies of each of the codevectors are counted. Then a histogram is obtained from these matched frequencies. This histogram can be termed as *codevector histogram*. This codevector histogram can be used as the feature vector of the human face. Thus using VQ the dimensionality of faces is reduced.

Many works are proposed in VQ based face recognition. Kotani et al. [17] have proposed a VQ-based face recognition method called VQ histogram method by using a systematically organized codebook for 4x4 blocks with 33 codevectors. Chen et al [18] proposed another face recognition system based a codebook which is a combination of a systematically organized codebook and a codebook created by Kohonen's Self Organizing Maps (SOM) [19]. A codebook is very important since it directly affects the quality of VQ processing. In [18] a codebook is created from two different codebooks. The first codebook is an intensity variation codebook and the second codebook is created using Kohonen's self organizing map [19]. The Kohonen self-organizing feature map [19] has to assume the number of clusters a priori and to initialize the cluster centroids. SOM guarantee convergence of weights by ensuring decrease in learning rates. Such learning rates, however, do not consider the similarity of the input pattern to the prototype of the corresponding cluster [20].The learning rule of Kohonen's SOM considers only the Euclidean distance between an input pattern and the centroid of the cluster. So in certain cases an input pattern is found to be outlier of the corresponding cluster even if the pattern belongs to that cluster.

In this work an improved codebook design method for VQ-based face recognition is proposed. The proposed codebook is also a combination of two different codebooks: the first codebook is being the intensity variation codebook same as in [18]. A second codebook is needed because the first codebook doesn't consider any of the facial features. The second codebook is created using Integrated Adaptive Fuzzy Clustering Method (IAFC) [20]. In IAFC a fuzzy membership value is incorporated in the Kohonen's SOM learning rule. This fuzzy membership value of the input pattern provides additional information for correct categorization of the input patterns. Moreover IAFC does not assume the number of clusters in the data set a priori. It updates the clusters during processing of data [20]. Because of this adaptive nature correct categorization of the clusters is possible. The intensity variation codebook and the codebook generated using IAFC are combined to form a single codebook which consists of 2x2 codevectors.

The performance evaluation of the proposed algorithm is done using publicly available AT&T database, Yale database, Indian Face database and some images which were taken by us called the DCSKU database. The results show that the proposed codebook approach is efficient in recognition. A recognition rate of 99.25%, 98.18%, and 96.83% is obtained for AT & T, Yale and Indian Face databases respectively. In all the databases the proposed approach got a higher recognition rate than the SOM codebook method and other existing face recognition methods in literature. The proposed approach is efficient in terms of Equal Error Rate (ERR) also. It has got an ERR of 3.5%, 6%, and 4% for AT & T, Yale and Indian Face respectively. The ERR is also obtained for the face recognition based on SOM codebook which is 4%, 7% and 5.5% for AT & T, Yale and Indian Face respectively. The results show that the proposed approach has a lower ERR when compared to SOM Codebook method. In the case DCSKU database also the proposed approach is better than the SOM Codebook method.

The rest of the paper is organized as follows: The proposed codebook generation is explained in Section II. Face Recognition using the proposed codebook is discussed in Section III. Experimental results and discussions are given in Section IV. In Section V conclusion is presented.

II. PROPOSED CODEBOOK GENERATION FOR FACE RECOGNITION SYSTEM

VQ is a well known algorithm in image compression and which is used for face recognition in this work. In VQ based face recognition a codebook is prepared in advance from a set of training images. The database of faces are converted to codevector histograms and stored. A codevector histogram gives the frequency of the codevectors in the face image. Hence each face is converted to a feature vector, which is the corresponding codevector histogram

The proposed codebook for Face Recognition is a combination of two different codebooks: the Intensity Variation codebook [18] and another codebook generated by

IAFC clustering. The Intensity Variation codebook captures the intensity variation among the pixels in a 2x2 block and the second codebook captures the facial features and they are combined to generate the final codebook. The various steps in the codebook generation are explained in the following subsections.

A. Intensity Variation Codebook

Nakayama et al. [21] have developed a classification method for 2x2 codebook design for image compression. Fig. 1 shows all categories for the 2x2 image block patterns without considering the location of pixels. In a 2x2 block, pixel intensities are marked by alphabet 'a', 'b', 'c', 'd', and a > b > c > d is prescribed. It is demonstrated in [21] that the number of typical patterns for all 2x2 image block is only 11. The possible arrangements of these 11 typical patterns are only 75 as shown in Fig. 1.

By the similar consideration, Chen et al. [22] classified and analyzed the code patterns in face images. They found that, code patterns belong to categories 7, 10, and 11 shown in Fig. 1 are almost not occurring in face images.
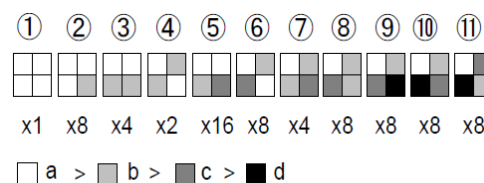

Fig. 1 Categories of 2x2 code patterns

Based on this observation, we have created a new codebook for 2x2 code patterns for face images as follows.
1. Generate codevectors by varying the intensity among the 2x2 blocks by 1 or 2 .The pixel intensities are obtained by changing the direction of intensity variation as shown in Fig. 2 where a, b, c, d are pixel intensities
2. Generate codevectors from the code patterns of category No.1, 2, 3, 4, 5, 6, 8, and 9, with the intensity differences among the blocks set to be 1 to 10.

Thus a large codebook of size 2024, known as the Intensity Variation codebook is created. Since this codebook considers only the intensity variation among the pixels and not the facial features we need a second codebook from the face images. The second codebook is generated using an adaptive clustering technique called the Integrated Adaptive Fuzzy Clustering.

B. Second Codebook Using IAFC

The second codebook in the proposed approach takes care of the facial features. It is generated from a set of training images using the Integrated Adaptive Fuzzy Clustering method. The training images are preprocessed to eliminate very high frequency components as well as the dc and very low frequency components, retaining only the relevant facial features to recognize the faces. The resulting 2x2 pattern vectors from the overlapping blocks form the *input patterns*.

These input patterns are clustered and the centroid of each cluster is selected as a codevector. The lAFC model is a fuzzy neural network which incorporates a fuzzy learning rule into a competitive neural network [20]. For each input pattern, IAFC consists of three major procedures:

i. Deciding the winning cluster,
ii. Performing the vigilance test
iii. Updating the centroid of the winning cluster.

1. Deciding the Winning Cluster

The input pattern X is normalized to find the winning cluster. To find the winner using dot product normalized weight vector is used. The weight vectors are initialized by the intensity variation codebook. For an input pattern we have to find the winner neuron close to the input pattern.

Fig. 2 Code patterns by changing the direction of intensity variation

A dot product operation used to find the winner is shown below.

$$Winner = I \bullet b_i \tag{1}$$

where I is the normalized input pattern, b_i is the normalized weights from the input neurons to the ith output cluster. The output neuron that receives the largest value for (1) wins the competition. In this process, the winner is decided by the angle between the input pattern and the centroids of clusters. This can cause misclassifications because a cluster of which the direction of the centroid vector has the smallest angle with the input vector wins the competition even though its centroid is located farther from the input pattern than other cluster centroids. In such a case, Euclidean distance can be used as a better similarity measure to determine a winner. So in IAFC a combined similarity measure is used to find a winner. For that we have to calculate the fuzzy membership value, $\mu_{i..}$. The fuzzy membership value accounts for the degree of correspondence between the input pattern and the existing cluster centroids. μ_i, is calculated as follows:

$$\mu_i = \frac{\left(\dfrac{1}{\|X - V_i\|^2}\right)^{1/m-1}}{\sum_{j=1}^{n}\left(\dfrac{1}{\|X - V_j\|^2}\right)^{1/m-1}} \tag{2}$$

where m is a weight exponent (also called fuzzifier) which is experimentally set to 2 [20] and n is the number of clusters. V_i is the ith winning cluster and V_j is the other existing cluster centroids. After deciding a winner by the dot product, the IAFC algorithm compares the fuzzy membership value, μ_i of the input pattern in the winning cluster with a threshold parameter σ for fuzzy membership value. If the fuzzy membership value is less than the value of the parameter σ, the angle between the input pattern and the cluster centroid is the dominant similarity measure to decide a winner. On the other hand, if the parameter σ is high, the Euclidean distance between the input pattern and the cluster centroid is the dominant similarity measure to decide a winner.

2. Performing the Vigilance Test and Updating the Centroid

After selecting a winning cluster, IAFC performs the vigilance test according to the criterion:

$$e^{\gamma \mu_i}\|X - V_i\| \leq \tau \tag{3}$$

where γ is a multiplicative factor that controls the shape of clusters, X is the input pattern, V_i is the centroid of the ith winning cluster, τ is the vigilance parameter which is set by the user and μ_i is the fuzzy membership value. The value of γ is normally chosen to be 1 [20]. The vigilance parameter τ controls the size of clusters. If the value of vigilance parameter is large, the size of clusters is large and vice versa.

The incorporation of μ_i yields more flexibility to the clusters formed since it considers not only the distance between the input pattern and the winning cluster centroid but also the relative proximity of the input pattern to the other existing cluster centroids as a degree of similarity. This may yield better classifications in certain cases since an input data point can be assigned to a winning cluster due to its large fuzzy membership value in that cluster whereas the same data point may be considered as an outlier when Euclidean distance is used. The extent of outlier location is determined by the vigilance parameter that a user can decide.

$$V_i^{new} = V_i^{old} + \lambda_{fuzzy}(X - V_i^{old}) \tag{4}$$

where V_i^{old} is the centroid of the winning cluster and

λ_{fuzzy} is $[f(l).\ \pi(X;\ V_i^{old};\ \tau).\ \mu_i^2]$.

The learning rule in (4) also incorporates the fuzzy

membership value (μ_i), an intracluster membership value (π), and a function of the number of iterations (f (l)) into a Kohonen-type learning rule. Because of the effect of the fuzzy membership value in the vigilance criteria, cluster centroids can drift around. Such a phenomenon can prevent weights from converging fast. The use of the intracluster membership value and the function of the number of iterations eliminate the problem [20]. The intracluster membership value (π) is decided by the distance between the input pattern and the centroid of the chosen cluster. The combination of the π-function and a function of the number of iterations guarantee weights to converge. The function *f(l)* is calculated as follows:

$$f(l) = \frac{1}{k(l-1)+1} \tag{5}$$

If a winning cluster satisfies the vigilance criterion, the input pattern is included in that winning cluster and the centroid of a winning cluster is updated using the learning rule as follows:

Where k is a constant and l is the number of iterations. The intracluster membership value is determined as follows:

$$\pi\left(X; V_i^{old}; \tau\right) = \begin{cases} 1 - 2\left(\dfrac{\left\|X - V_i^{old}\right\|}{\tau}\right)^2, & 0 \le \left\|X - V_i^{old}\right\| \le \tau/2 \\ 2\left(1 - \dfrac{\left\|X - V_i^{old}\right\|}{\tau}\right)^2, & \tau/2 \le \left\|X - V_i^{old}\right\| \le \tau \\ 0, & \left\|X - V_i^{old}\right\| \ge \tau \end{cases} \tag{6}$$

If the vigilance criterion is not satisfied then the input pattern is added to the winning cluster and the centroid of the winning cluster is not updated using the learning rule.

IAFC algorithm for the codebook design can be summarized by the following steps:

Step 1: Initialize parameters τ and σ.
Step 2: Pre-process the face images in the training set to intensity variation vectors called input patterns.
Step 3: Initialize the weight vectors with the Intensity Variation codebook.
Step 4: Select a new input pattern X.
Step 5: Calculate the fuzzy membership value, μ_i of the input pattern, X in the winning cluster using (2).
Step 6: Decide a winning cluster (best matching codevector) using the combined similarity measure.
If $\mu_i < \sigma$, the winner neuron is selected from the highest dot product or smallest angle using (1).
else the winner neuron is selected from the minimum Euclidean distance.

Step 7: The input pattern X is included in this winning cluster and performs the vigilance test using (3).
If the criterion is satisfied then update the weights using (4)
Else the weights are not updated.

Step 8: If all the input patterns are processed go to step 9
Else go to step 4
Step 9: Stop

The centroids of the clusters generated are the codevectors in the second codebook

C. Proposed Codebook Generation

The generated codevectors using IAFC procedure are combined with the intensity variation codebook to get the proposed codebook. Since IAFC does not assume the number of clusters a priori, we may not get a codebook with the same size as that of an Intensity Variation Codebook. By combining the intensity variation codebook and the codebook using IAFC, we get a codebook of some size. A set of training images are divided into patterns of the same size as that of the codevectors and they are matched with the codevectors in the codebook. The top matching N codevectors are selected to form the proposed codebook. The proposed system is evaluated with N varying from 50 to 110.

III. FACE RECOGNITION USING THE PROPOSED CODEBOOK

For recognizing faces a database of known faces are formed and these face images are converted to corresponding codevector histogram using the proposed codebook and stored. The generation of the codevector histogram is explained in Section III *B*. During testing, the input test face image is also converted to codevector histogram and Manhattan distance is calculated between the input face's codevector histogram with the stored codevector histograms. The face with the minimum distance is considered to be the matched face. Fig. 3 shows the block diagram of the face recognition process.

A. Pre-processing

Initially a low pass filtering is carried out using a simple 2D mean filter [17] for eliminating the high frequency noise components in the input face images.
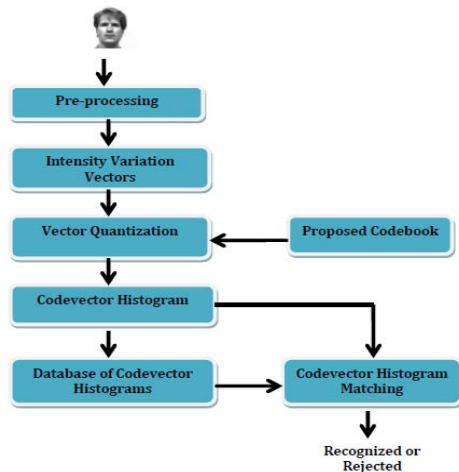
Fig. 3 Face Recognition using the Proposed Codebook

By this filtering, detailed facial features degrading recognition performance such as wrinkles, are excluded. Only the important personal features, such as the rough shape of facial parts are retained. The image is then divided into 2x2 overlapping blocks. Minimum intensity in each block is subtracted from each pixel in the block for excluding dc and very low frequency components, such as shade variations due to small variations in lighting conditions and the resulting 2x2.

B. Codevector Histogram Generation

In the case of face recognition, initially the face images in the database are pre-processed as explained in section 3.1 to get the intensity variation vectors. The vectors generated after pre-processing each input image is matched to the codebook using Manhattan distance measure and the frequency of matching of each codevector is noted. This frequency plot gives the codevector histogram of that image. Codevector histogram of a face image with a codebook of size 80 is shown in Fig. 4.

IV. RESULTS AND DISCUSSIONS

The performance of the system is evaluated using three publicly available face databases AT & T database [23], Yale database [24], the Indian Face Database [25] and a small database created in our department called the DCSKU database.

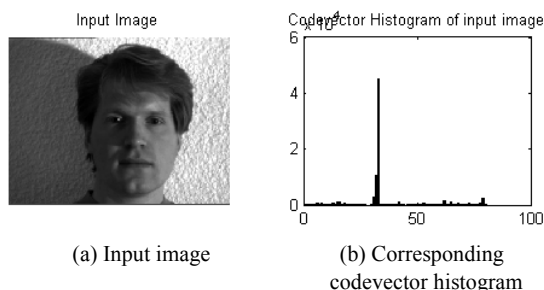(a) Input image   (b) Corresponding codevector histogram

Fig. 4 Codevector Histogram of a face image

The AT & T database contains 400 images in .pgm format of 40 persons. There are 10 different images of each of 40 distinct subjects. The images were taken at different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses) [23]. The Yale Face Database contains 165 grayscale images in GIF format of 15 individuals. There are 11 images per subject, one per different facial expression or configuration: center-light, with glasses, happy, left-light, without glasses, normal, right-light, sad, sleepy, surprised, and wink [24]. The Indian Face database [25] contains 10 different images of each of 60 distinct subjects. All the images were taken against a bright homogeneous background with the subjects in an upright, frontal position. The files are in JPEG format. The different orientations of the face are looking front, looking left, looking right, looking up, looking up towards left, looking up towards right, looking down. In addition to the images in pose, images with four emotions: neutral, smile, laughter, sad/disgust. The DCSKU database consists of 33 images of 3 different persons. These images were in different foreground, background, closed eyes, and in different poses. The images were in JPEG format. There are 11 different images of each of the person.

For the evaluation of the proposed method, a subset of face images of each person from these databases are converted to their respective codevector histograms and stored as a database of known faces. The remaining faces and other unknown faces are used for testing. A test image is also converted to its codevector histogram and is matched to the database of known faces using Manhattan distance measure. In experiments without rejection criteria the closest matching image from the database of 'known faces' is returned as identity. But in experiments with rejection criteria test faces will be rejected as unknown persons if the Manhattan distance between the codevector histograms exceeds a threshold. The results of experiments without rejection and with rejection are explained in the following sections.

A. Experiments without Rejection Criteria

Five images were selected from each person's 10 images in the case of AT & T and Indian Face and from 11 images in the case of Yale and DCSKU and stored as known faces in the database. The remaining images are used for testing. So 200 images from AT & T are stored as known faces and the remaining 200 is used for testing. In the case of Yale 75 images stored as known faces and 90 images are used for testing and in Indian Face database 300 images stored as known faces and the remaining 300 are used for testing. In the case of DCSKU 15 images stored as known faces and the remaining 18 images are used for testing. After testing the recognition rate is calculated as follows.

Recognition Rate = 100 * (Number of face images correctly recognized / Total number of face images tested)

1. Selection of τ and σ Parameters

In the case of the proposed method using IAFC, the parameters τ and σ plays an important role in the adaptive nature of clustering. Experiments are conducted to find the optimum τ and σ value. Experiments are done by varying the values for the vigilance parameter, τ. The recognition rate is plotted as shown in Fig. 5. It is clear from the figure that for the value, τ =2, a higher recognition rate is achieved. It is the parameter τ that determines the size of the clusters. So for the rest of the experiments, the vigilance parameter, τ is taken as 2.
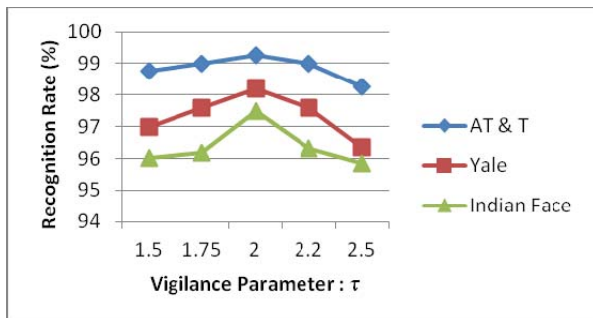

Fig. 5 Recognition Rate for different values of τ

By varying the values of σ, the threshold for the fuzzy membership value, recognition rates are calculated by fixing the value for τ as 2. The results are shown in Fig. 6.
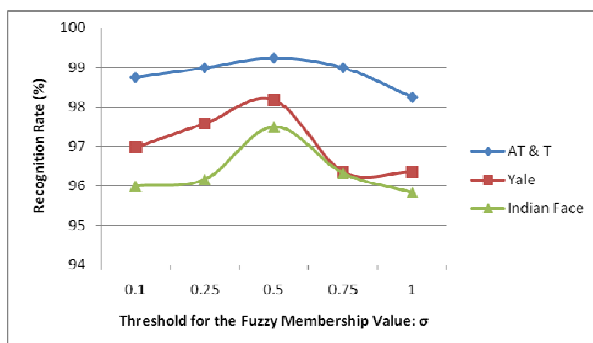

Fig. 6 Recognition Rates for different values of σ

From the figure it is clear that highest recognition rate is achieved when the value of σ is 0.5. So the rest of the experiments are conducted by taking σ as 0.5.

With the above optimum value for τ and σ, Recognition Rates are evaluated for different codebook sizes using the proposed codebook and these results are compared with that of SOM codebook. Figs. 7-9 show the comparison between the Face Recognition system with proposed codebook and SOM codebook, for AT & T, Yale and IndianFace respectively.
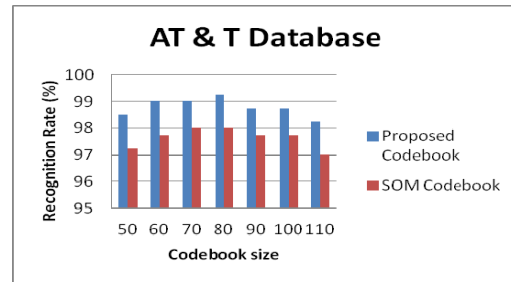

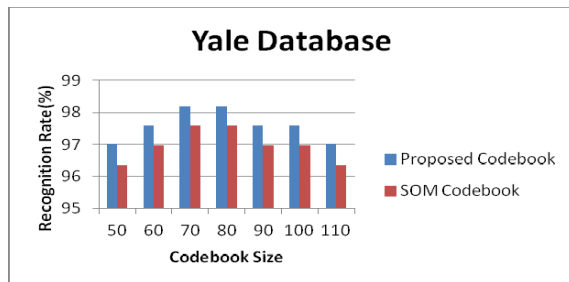Fig. 7 Comparison of the Recognition Rate Using AT & T Database

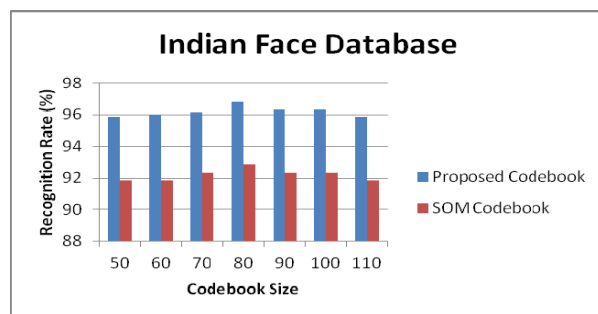Fig. 8 Comparison of the Recognition Rate Using Yale Database

Fig. 9 Comparison of the Recognition Rate Using Indian Face Database

In the case of AT & T, Yale and Indian Face databases, recognition rates are calculated for different values of codebook sizes such as 50, 60, 70, 80, 90, 100 and 110. From Figs. 7-9, it is clear that the recognition rates using the proposed codebook are always higher than the recognition rates using the SOM codebook. We get a higher recognition rate of 99.25% for AT & T, 98.18% for Yale and 96.83% for the Indian Face databases using the proposed codebook. From the results it is clear that the proposed codebook is more efficient in representing the facial features than the method using SOM codebook.

As the codebook size increases, noise corrupted codevectors may distort the codevector histogram. On the contrary, if codebook size is small, the codevector histogram cannot sufficiently discriminate between faces. It can be noted that a codebook size 80 gives higher recognition rate in all the cases.

The results for the DCSKU database are shown in Table I. In the case of the proposed codebook 14 images are correctly matched out of the 18 images which are used for testing. But

using the SOM Codebook only 13 images are correctly matched out of the 18 images used for testing. So from the results it is clear that the proposed method using IAFC is a good recognition system.

Figs. 10 and 11 show the results of recognition of some special cases using the SOM codebook and the proposed codebook for the database AT & T. The superiority of the proposed method compared to the SOM codebook based method is demonstrated in these examples. The face that is not correctly recognized using the SOM based method is found to be correctly recognized using the proposed method. The respective codevector histograms are also shown to the right of the faces. In these cases the size of the codebook is taken as 80.

TABLE I
COMPARISON OF THE PROPOSED CODEBOOK METHOD AND SOM CODEBOOK METHOD USING DCSKU DATABASE

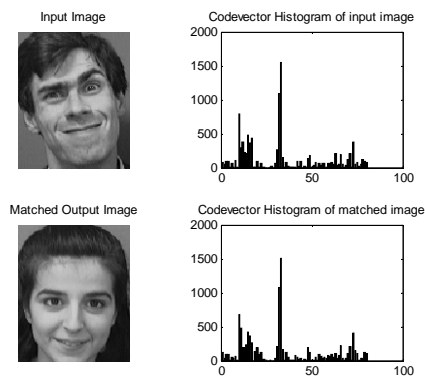|	Number of Images			
	Training	Testing	Correctly Matched	Not Matched
Proposed Codebook	15	18	14	4
SOM Codebook	15	18	13	5

2. Special Cases

Fig. 10 Incorrectly Matched face and their corresponding codevector histograms in AT & T database with SOM codebook
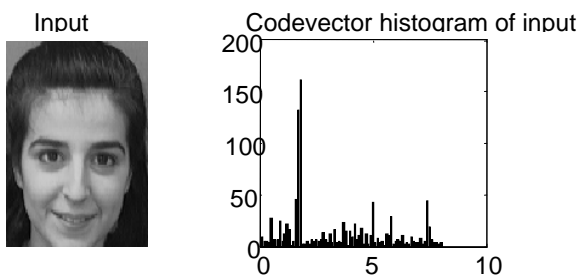

Fig. 11 Codevector Histogram of the incorrectly matched face image using the proposed codebook

It can be noted from Fig. 10 that the input test face is incorrectly matched to the face of a lady whose codevector histogram is somewhat similar to the input face image. In SOM codebook method the two faces cannot be discriminated using the codevector histograms. But in the case of the proposed method the test face is correctly recognized as the codevector histogram is matched to the same person's image stored in the known faces database. The codevector histogram of that lady using the proposed method is shown in Fig. 12. By comparing the codevector histograms of Figs. 11 and 12, it is obvious that the proposed method the features are more prominent to discriminate between faces.

In many similar cases (in the different databases used) where the SOM method failed to recognize the test images, the proposed method could correctly identify the persons.

The above result analysis show that the face recognition system based on proposed codebook exhibits a very good performance in terms of recognition rate. The recognition rate can be increased by increasing the number of training samples. Compared to the SOM codebook method IAFC based face recognition has highest recognition rate.
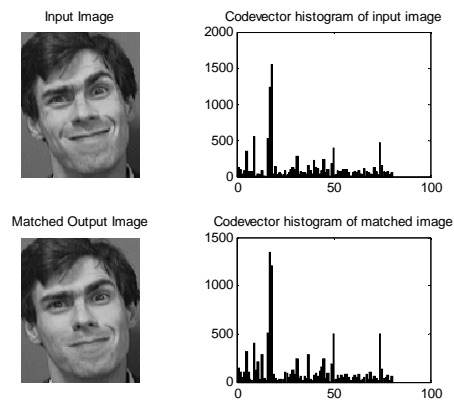

Fig. 12 Correctly matched face and their corresponding codevector histograms in AT & T database with proposed codebook

B. Experiments with Rejection Criteria

In many applications, mainly for face verification purposes, reliably recognizing known persons while rejecting unknown persons is required for a good verification system. This is a more challenging task and the proposed method is evaluated for this case with a rejection criteria incorporated in the system.

For the evaluation of the rejection performance of the system the databases are divided into two sets, one set termed as known faces and their feature vectors are stored in the database of known faces. The other set is unknown to the system called as unknown faces. In the case of AT & T database, the 40 persons are divided into two sets containing 20 known faces and 20 unknown faces. So, codevector histogram is generated from 100 images of the known 20 faces. The recognition and rejection performance of the system is then tested on all the 400 images of the database. The 15 persons in Yale database is divided into 8 known faces and 7 unknown faces. Codevector histogram is generated and stored for the 40 images of known faces. Testing is done for all the 165 images. The Indian Face database consists of 60

persons. Among them 30 faces are considered to be known and the remaining 30 are unknown. Feature vectors are generated for the 150 images and testing is done with all the 600 face images.

In experiments with rejection criteria test faces will be rejected as unknown persons if the Manhattan distance between the codevector histograms exceeds a threshold, *Th*. Rejection Rate and Recognition Rate are calculated for the three databases as follows

Recognition Rate = 100 * (Number of face images correctly recognized / Total number of face images tested)

Rejection Rate = 100 * (Number of face images correctly rejected / Total number of face images tested)

From the Recognition Rate and the Rejection Rate False Acceptance Rate (FAR) and False Rejection Rate (FRR) are calculated as follows:

$$FAR (\%) = 100 - Recognition\ Rate$$

$$FRR (\%) = 100 - Rejection\ Rate$$

Figs. 13-15 show the False Acceptance Rate (FAR) and False Rejection Rate (FRR) plots for the verification experiment for AT & T, Yale and Indian Face respectively. FAR and FRR are obtained to find the Equal Error Rate (ERR). ERR is the rate at which both FAR and FRR are equal. A system is said to be efficient if its ERR is low.

Table II summarizes the ERR for different databases using the proposed codebook and the SOM codebook

TABLE II
COMPARISON OF ERR FOR THE FACE RECOGNITION USING THE PROPOSED
CODEBOOK AND THE SOM CODEBOOK

Databases	Equal Error Rate (ERR) %	
	Proposed Codebook	SOM Codebook
AT & T	3.5	4
Yale	6	7
Indian Face	4	5.5

It can be noted that for all databases the ERR for the proposed approach is low. Hence this proposed system can be used as a good face verification system, too.
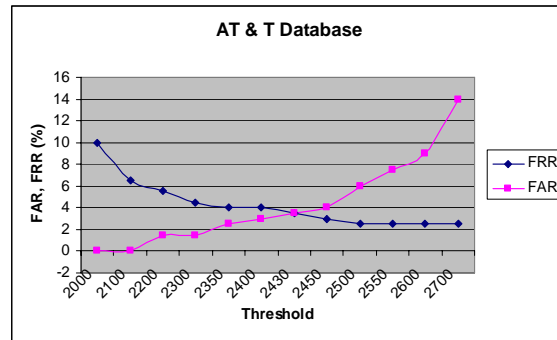

Fig. 13 False Acceptance Rate and False Rejection Rate (%) for AT & T database

A comparison of recognition rate of the propose method with other approaches on AT & T, Yale and Indian Face databases. From Tables III-V, it is clear that the performance of the proposed algorithm is evidently better than those we have listed.
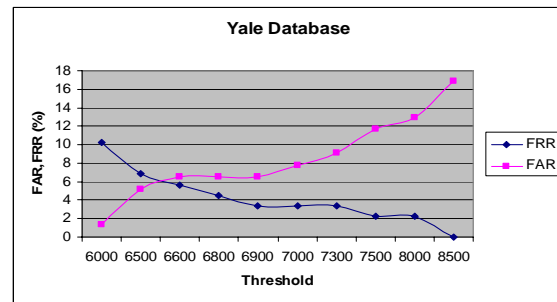

Fig. 14 False Acceptance Rate and False Rejection Rate (%) for Yale database

TABLE III
A COMPARISON OF RECOGNITION RATE WITH OTHER APPROACHES USING
YALE DATABASE

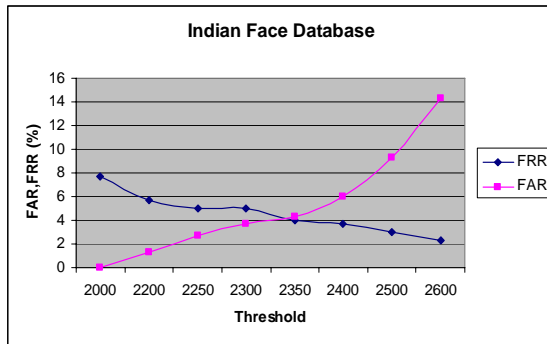| Algorithm	Recognition Rate	References
PCA	71.52	M. H Yang [33]
ICA	71.52	M. H Yang [33]
Kernel Eigenfaces	72.73	M. H Yang [33]
FLD	83.45	V More et al. [35]
2DPCA	84.24	Yang et al [32]
MFLD	91.40	V More et al. [35]
LBP	92.00	JayaPriya et al. [34]
FFLD	94.80	V More et al. [35]
VQ based FR using SOM	97.25	Chen et al. [18]
VQ based FR using IAFC	98.18	Proposed Approach

Fig. 15 False Acceptance Rate and False Rejection Rate (%) for Indian Face database

TABLE IV
A COMPARISON OF RECOGNITION RATE WITH OTHER APPROACHES USING AT &T DATABASE

Algorithm	Recognition Rate	References
PCA	84.86	Turk et al. [5]
ICA	85.00	Barlett et al [7]
KND	87.00	Liu et al. [26]
FSLDA	88.00	Li et al. [27]
R2DPCA	90.42	Xu et al. [28]
Global DCT	91.00	Hafed et al. [29]
PCA+SVM	91.20	Guo et al. [30]
LDA	91.30	Sahoolizadeh et al. [4]
PCA+LDA+BPNN	93.00	Sahoolizadeh et al. [4]
SIFT + Bow	93.49	Di Liu et al. [31]
2DPCA	96.00	Yang et al [32]
PCA+LDA+SFAM	97.00	A.A.Thomas et al.[6]
VQ based FR using SOM	98.00	Chen et al. [18]
VQ based FR using IAFC	99.25	Proposed Approach

V. CONCLUSION

Face recognition systems have been grabbing high attention in pattern recognition field. In this work a new, Fuzzy Neuro approach to Face Recognition based on Vector Quantization, method is presented. A simple and efficient codebook design algorithm for face recognition using vector quantization is proposed. The codebook is created from two different codebooks. One codebook is created by the intensity variation among the pixels. The other codebook is created from the face images using Integrated Adaptive Fuzzy Clustering (IAFC). These two codebooks are combined to get the proposed codebook. From the results it is clear that the proposed method is more efficient than most of the existing Face Recognition systems. For practical applications of face recognition, not a simple recognition rate but a False Acceptance Rate (FAR) and a False Rejection Rate (FRR) are more important. Equal Error Rate (ERR) is calculated from the plots of FAR and FRR. The ERR for the proposed approach is low for all the test databases used. Hence the proposed system is better for both recognition and rejection than the method that uses the SOM codebook. The proposed method is also superior to the other methods in literature.

TABLE V
A COMPARISON OF RECOGNITION RATE WITH OTHER APPROACHES USING INDIAN FACE DATABASE

Algorithm	Recognition Rate	References
ICA	74.6	E A Daoud [37]
PCA	75.9	E A Daoud [37]
BATCH-ILDA	82.5	E A Daoud [37]
PCA+Fourier Gabor Filter	82.8	E A Daoud [37]
SVM	84.2	E A Daoud [37]
LDA	84.5	E A Daoud [37]
VQ based FR using SOM	92.33	Chen et al. [18]
CST	96.4	D. Sudarshan et al [36]
VQ based FR using IAFC	96.83	Proposed Approach

REFERENCES

[1] R. Chellappa., C. L. Wilson., S. Sirohey: "Human and machine recognition of faces: a survey". Proc. IEEE, 83(5), pp. 705—740, 1995.
[2] S. Z. Li., A. K. Jain.: "Handbook of Face Recognition", Springer, New York, 2005.
[3] J. R. Solar, P. Navarreto, " Eigen space-based face recognition: a comparative study of different approaches, IEEE Tran., Systems man And Cybernetics- part c: Applications, Vol. 35,No. 3, 2005.
[4] Sahoolizadeh, H.; Ghassabeh, Y.A.;" Face recognition using eigen-faces, fisher-faces and neural networks, Cybernetic Intelligent Systems, 2008. CIS 2008.7th IEEE International Conference on 9-10 Sept. 2008
[5] M.Turk., A. Pentland.: "Eigenfaces for recognition". Journal of Cognitive Neuroscience, 3(1), pp.71--86, 1991
[6] A. A. Thomas and M. Wilscy, "Face Recognition Using Simplified Fuzzy ARTMAP", J.Signal and Image Processing, Vol.1, No.2, pp 134-136, December 2010.
[7] M.S. Bartlett, J.R. Movellan, T.J. Sejnowski, Face Recognition by Independent Component Analysis, IEEE Trans. on Neural Networks, Vol. 13, No. 6, November 2002, pp. 1450-1464
[8] C. Liu, H. Wechsler, Evolutionary Pursuit and Its Application to Face Recognition, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 22, No. 6, June 2000, pp. 570-582
[9] L. Wiskott, J.-M. Fellous, N. Krueuger, C. von der Malsburg, Face Recognition by Elastic Bunch Graph Matching, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, 1997, pp. 775-779
[10] P. J.Phillips.: "Support vector machines applied to face recognition". Advanced Neural Information. Processing. Systems. 11, 803–809, 1998.
[11] W. Zhao., R. Chellappa., P. J. Phillips., A. Rosenfeld.: "Face Recognition: A Literature Survey". ACM Computing Surveys, Vol. 35, No. 4, pp. 399–458, 2003.
[12] A.V. Nefian, M.H. Hayes, Hidden Markov Models for Face Recognition, Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'98, Vol. 5, 12-15 May 1998, Seattle, Washington, USA, pp. 2721-2724
[13] R. Brunelli., T. Poggio .: "Face recognition: features versus templates". IEEE Transactions on Pattern Analysis and Machine Intelligence, 15(10), pp.1042—1052, 1993.
[14] P. S. Penev., J. J. Atick.: "Local Feature Analysis: A general statistical theory for object representation". Network: Computation in Neural Systems, 7(3), pp.477—500, 1996.
[15] Goudail., E. Lange., T. Iwamoto., K. Kyuma., N. Otsu. : "Face recognition system using local autocorrelations and multiscale integration. IEEE Transaction on Pattern Analysis and Machine Intelligence, 18(10), pp. 1024—1028, 1996.
[16] K Sayood.: "Introduction to Data Compression". Morgan Kaufmann, San Francisco, 2000.
[17] K. Kotani., Q. Chen., and T. Ohmi.: "Face recognition using vector quantization histogram method". Proceedings of the 2002 Int. Conf. on Image Processing, Vol. II of III:II- 105-II-108, 2002
[18] Q. Chen., K. Kotani., F.F. Lee., and T. Ohmi.: "A VQ based fast face recognition algorithm using optimized codebook". Proeedings of the 2008 Int. Conf. on Wavelet Analysis and Pattern Recognition, 2008.
[19] T. Kohonen.: "The Self-Organizing Maps". Proceedings of the IEEE, Vol 78, No 9, September, 1990.

[20] Y. S. Kim., S Mitra., "An adaptive integrated fuzzy clustering model for pattern recognition". Journal. Fuzzy Sets and Systems.Vol (65) pp 297--310, Elsevier, Netherlands, 1994

[21] T. Nakayama., M. Konda., K. Takeuchi., K. Kotani., T. Ohmi.: "Still image compression with adaptive resolution vector quantization technique". Int. Journal of Intelligent Automation and Soft Computing, 10(2), pp.155—166, 2004

[22] Q. Chen., K. Kotani., F. F. Lee., and T. Ohmi.: "Face recognition using codebook designed by code classification". IEEE Int. Conf. on Signal and Image Processing, pp. 397—40,2006.

[23] AT&T, "The Database of Faces". http://www.cl.cam.ac.uk/research/ dtg/attarchive/facedatabase.html

[24] YaleFaceDatabase. http://cvc.yale.edu/projects/yalefaces/yalefaces.html

[25] TheIndianFaceDatabase.http://vis-www.cs.umass.edu/~vidit/IndianFaceDatabase/

[26] B. Liu, B. Liu, X. Sun, and J. Zhang, Eigenface classification using an extended kernel-based nonlinear discriminator, International Conference on Communications, Circuits and Systems, Volume 2, pp. 1123-1126, 2004.

[27] Y. Li, Z. Mu, X. Xu, Multimodal recognition based on face and ear International Conference on Wavelet Analysis and Pattern Recognition, Volume 3, pp. 1203 - 1207, 2007.

[28] C. Xu, H. Jiang, J. Yu, Robust two-dimensional principle component analysis, 27th Chinese Control Conference, pp.452-455, 2008.

[29] Z. M. Hafed and M. D. Levine, Face recognition using the discrete cosine transform, International Journal of Computer Vision, 43(3),2001.

[30] S. L. G. Guo and K. Chan, Face recognition by support vector machines, IEEE International Conference on Automatic Face and Gesture Recognition, pp. 196-201, 2000.

[31] D. L. Dong-mei., S.Z. Qiu., "Bag-of-Words Vector Quantization Based Face Identification", IEEE Second International Symposium on Electronic Commerce and Security, 2009

[32] J. Yang., D. Zhang., A. F. Frangi., and Jing-yu Yang., "Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition", IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. 26, pp 131-137, January 2004

[33] M.H. Yang, "Kernel Eigenfaces vs. Kernel Fisherfaces: Face Recognition Using Kernel Methods," Proc. Fifth IEEE Int'l Conf. Automatic Face and Gesture Recognition (RGR'02), pp. 215-220, May 2002.

[34] K.Jaya Priya., R.S Rajesh, "A Local Min-Max Binary Pattern Based Face Recognition Using Single Sample per Class", International Journal of Advanced Science and Technology Vol. 36, pp 41-49 November, 2011

[35] V More., A Wagh, "Improved Fisher Face Approach for Human Recognition System using Facial Biometrics", International Journal of Information and Communication Technology Research, Volume 2, pp 135-139, February 2012.

[36] D.Sudarshan S.,D Pooja S, "Principle Component Analysis of Low Level Feature Based Face Recognition System", International Journal of Recent Trends in Engineering, Vol 2, No. 3, pp 149-151, November 2009

[37] E A Daoud , "Enhancement of the Face Recognition Using a Modified Fourier-Gabor Filter", Int. J. Advance. Soft Comput. Appl., Vol. 1, No. 2,pp 119-131 November 2009

Elizabeth B Varghese is an Assistant Professor at Mar Baselios College of Engineering, Trivandrum, India. Her areas of research interests are Digital image Processing, Artificial Neural Networks, Pattern Recognition and Fuzzy systems.

M. Wilscy is a B.Sc (Engg) graduate in Electrical Engineering from TKM College of Engineering, Kerala, India, Master of Engineering (ME) from School of Automation & Computer Science, Indian Institute of Science, Bangalore, India and Ph.D from the Department of Computer Science and Engineering, Indian Institute of Technology, Madras, India. She is currently Professor and HoD, Dept. of Computer Science, University of Kerala, Trivandrum, India. Her areas of research interests are Digital Image Processing, Pattern Recognition, Neural Networks and Fuzzy Systems, and Intelligent systems.