CHAPTER 5

FACE RECOGNITION USING DB WAVELET AND FUZZY C-MEANS CLUSTERING

In the previous chapters, a brief description about the theory and experimental results of PCA, Gabor and Neural techniques for Face recognition were presented. In this chapter, some description about Daubechies wavelets and Fuzzy C means (FCM) clustering based Face Recognition are discussed.

Several recognition algorithms have been proposed in the previous chapters with minimum distance classification in the Eigen space analysis, Fisher Discriminant Analysis and Neural networks. Gabor Wavelet techniques work well for classifying frontal views of faces. However, they are not robust against pose and expression changes since global features are highly sensitive to translation and rotation of the face. To overcome these problems, in this chapter Daubechies wavelets, Fuzzy integral and BPNN based FCM algorithms are implemented.

Also, all the algorithms discussed in the earlier chapters are for recognizing a single image. To recognize a group of images in short duration, in this chapter, FCM with BPNN based algorithms are used.
5.1 DB WAVELET, FLDA AND FUZZY INTEGRAL APPROACH OF FACE RECOGNITION

5.1.1 Theory of DB Wavelet and Fuzzy Integral

In this section, a method for recognizing face images by combining wavelet decomposition, Fisher face method and fuzzy integral is developed and its basic block diagram (Keun-Chang 2007) is shown in Figure 5.1.

![Basic block diagram of DB wavelet and fuzzy integral based face recognition](image)

**Figure 5.1** Basic block diagram of DB wavelet and fuzzy integral based face recognition

Generally, wavelet decomposition helps to extract intrinsic features of images. Fisher face method is applied to these decomposed images to reduce the dimensionality and the last phase is concerned with aggregation of individual classifiers by fuzzy integral (Fatma et al 2008). The goal of the wavelet decomposition is to realize cleaning and compressing images, denoising and feature detection of images.

A Daubechies Wavelet is a non-linear filter used in image processing. This proposed Db wavelet decomposition approach comprises of four main stages namely approximation, horizontal, vertical and diagonal images. The low-frequency components contribute to the global description, while the high-frequency components contribute to the finer details required for the identification task. FLDA is a statistically motivated technique which maximizes the ratio of the determinant between-class scatter matrix and
within-class scatter matrix and in this sense attempts to involve information about classes of the patterns under consideration.

Also, FLDA is capable of forming well-separated classes in a low dimensional sub space, even under severe variation in lighting and facial expressions. Then the fusion of individual classifiers is realized through fuzzy integral. The ability of the fuzzy integral is to combine the results of multiple sources of information.

5.1.2 The Daubechies (Db) Wavelet Transform

Actually, Wavelet transform decomposition provides local information in space and frequency domains, but Fourier Transform supports only global information in frequency domain. Among various wavelet forms Daubechies proves to be the simpler and more efficient one. The Db Wavelet transform has four wavelet coefficients g0…. g3 and scaling function coefficients h0…. h3 as shown in Figure 5.2.

![Figure 5.2 Db wavelet transform](image-url)
The scaling function coefficients are:

\[ h_0 = \frac{1 + \sqrt{3}}{4\sqrt{2}} \]
\[ h_1 = \frac{3 + \sqrt{3}}{4\sqrt{2}} \]
\[ h_2 = \frac{3 - \sqrt{3}}{4\sqrt{2}} \]
\[ h_3 = \frac{1 - \sqrt{3}}{4\sqrt{2}} \]  (5.1)

Each step of the wavelet transform applies the scaling function to the data input. If the original data set has N values, then the scaling function will be applied in the wavelet transform step to calculate N/2 smoothed values. In the ordered wavelet transform, the smoothed values are stored in the lower half of the N element input vector. The wavelet function coefficient values are given by

\[ g_0 = h_3, \quad g_1 = -h_2, \quad g_2 = h_1, \quad g_3 = -h_0. \]  (5.2)

Each step of the wavelet transform applies the wavelet function to the input data. If the original data set has N values, the wavelet function will be applied to calculate N/2 differences to show the reflecting change in the data. In the ordered wavelet transform the wavelet values are stored in the upper half of the N element input vector. The scaling and wavelet functions are calculated by taking the inner product of the coefficients and four data values (Keun-Chang Kwak 2007). The equations are shown below.

Daubechies scaling function

\[ a_i = h_0s(2i) + h_1s(2i+1) + h_2s(2i+2) + h_3s(2i+3) \]
\[ a[i] = h_0s[2i] + h_1s[2i+1] + h_2s[2i+2] + h_3s[2i+3] \]  (5.3)
Daubechies wavelet function:

\[ C_i = g_0 s(2i) + g_1 s(2i+1) + g_2 s(2i+2) + g_3 s(2i+3) \]

\[ C[i] = g_0 s[2i] + g_1 s[2i+1] + g_2 s[2i+2] + g_3 s[2i+3] \]  \hspace{1cm} (5.4)

Each iteration in the wavelet transform step calculates a scaling function value and a wavelet function value. The index ‘i’ is incremented by two with each iteration, and new scaling and wavelet function values are calculated. This decomposition generates coefficient matrices of the one-level approximation and horizontal, vertical and diagonal details respectively. From the obtained coefficients, the approximation and three detailed images via the high-pass and low-pass filtering are constructed as shown in Figure 5.3.

\[ \text{Figure 5.3 Basic block diagram of Db wavelet analysis} \]

Here, a feature vector containing important features of face image is constructed by Db wavelets and the vector size is reduced by Fisher linear discriminant analysis. Finally, the output of each classifier is aggregated and the degree of importance of each classifier using fuzzy integral is based on the highest value is declared, as the output of the classifier.
5.1.3 Fuzzy Integral Approaches

In fuzzy Integral, the membership grades based on the distance information produced in the previous phase are generated. The membership grades can be determined by using the distance between the test image and those images existing in the training set.

\[ \mu_{ij} = \frac{1}{1+(\frac{d_{ij}}{d_i})} \]  \hspace{1cm} \text{(5.5)}

where ‘i’ is the number of classifier and ‘j’ is the index of the training face image. \( \overline{d}_i \) denotes an average distance between all distance values in \( i^{th} \) classifier, \( \overline{d}_{ij} \) is the Euclidean n distance of feature vector between a given test image and \( j^{th} \) training image in \( i^{th} \) classifier. The output values \( h(y_{ik}) \) of the \( k^{th} \) class in \( i^{th} \) classifier are as follows:

\[ h(y_{ik}) = \sum_{\mu_{ij} \in C_k} (\mu_{ij}) / N_k \]  \hspace{1cm} \text{(5.6)}

where \( N_k \) is the number of samples in \( k^{th} \) class \( C_k \).

5.1.4 Experimentation Results of FR using Db Wavelets, FLDA and Fuzzy Integral

Yale database of 165 images, consisting of 11 persons with 15 images taken at different lighting conditions, with various expressions is considered as the training set. A sample database is as shown in Figure 5.4.
At first training set images are preprocessed i.e., converted to grayscale, then they are resized and reshaped. Afterwards, it is subjected to Db wavelet decomposition where it is cleaned and compressed. As a result of wavelet decomposition four sub images namely approximation, horizontal, vertical and diagonal are obtained. One sample input image and its sub images are as shown in Figures 5.5 and 5.6 respectively.
From the coefficients of all sub images a feature vector can be obtained. Then it is applied to FLDA algorithm as explained in section 3.2. The between class and within class matrices are calculated, from which the reduced FLD feature vector is calculated. Similar steps are applied to all test images and training images.

The next step is the application of fuzzy integral (FI). FI of Sugeno type makes use of the ‘min’ and ‘max’ functions. Among the calculated minimum classifier values, it finds the maximum value, which is the resultant output, i.e., the best match of the given image. The degree of importance of the classifier is now calculated and is aggregated using fuzzy integral. From the resulting classes, the class with the highest output is considered to be the output of the classifier.

Then the values of Euclidean distance are computed using the feature vectors produced from the training image set and the given test image. Euclidean distance and the recognized index from the MatLab simulation are shown in Figure 5.7.

Figure 5.6 Sub images at level 2
This proposed method makes use of the appearance-based face recognition. For the purpose of preprocessing and feature extraction, the image is applied to fisher face, with the help of which the sensitivity caused by varying illumination and viewing conditions associated with the original image are reduced. Since not only the approximation image, but also the other sub images resulting from wavelet decomposition are considered, it increases the accuracy of the recognition. Finally, it has been experimentally demonstrated that the aggregation of classifiers resulting from fisher face operating on four sub image sets generated by wavelet decomposition leads to better classification results.

Experiments are carried over on the face images of different databases by DB wavelets, FLDA and FI method. The results of Yale database are presented as in Figures 5.8.
Subject 01

(a)

Recognized index = 1, and $Euc\_dist\_min\_a = 3.9745e-014$

The input image matches with Person 1 – CORRECT RESULT

Subject 02

(b)

Recognized index = 2 and $Euc\_dist\_min\_a = 2.4364e-015$

Input image matches with Person 2 – CORRECT RESULT

Subject 03

(c)

Recognized index = 4 and $Euc\_dist\_min\_a = 4.9960e-016$

Input image matches with Person 4 – WRONG RESULT

Figure 5.8 Result analysis of DB wavelets and FLDA method
This DB wavelets, FLDA and FI method of face recognition, when compared with other methods, has the advantages of low computational cost and requires less time for classification for small number of database images. Further, the variation in illumination, pose or facial expression does not affect the recognition process. The fuzzy integral data fusion technique also has its limitation. The number of fuzzy measures increases exponentially with the number of parameters. One of the distinguishing features of fuzzy measure and fuzzy integral technique is that it is able to represent certain interactions between criteria.

5.2 FACE RECOGNITION USING FUZZY C-MEANS CLUSTERING AND NEURAL NETWORKS

5.2.1 Theory of FCM

Fuzzy C-mean (FCM) is one of the most used methods for image segmentation and its success chiefly attributes to the introduction of fuzziness for the belongingness of each image pixel. Compared with crisp or hard segmentation methods, FCM is able to retain more information from the original image. The FCM algorithm has successfully been applied to a wide variety of clustering problems and it works better for unlabeled data. Fuzzy clustering algorithms partition methods that can be used to assign objects of the data set to their cluster (Jianming Lu et al 2007). These algorithms optimize an objective function that evaluates a given fuzzy assignment of objects to clusters. A cluster is a collection of objects which are similar in some way. Clustering is the process of grouping similar objects into clusters. Here, clusters are grouped using distance measures like Euclidean distance.

Fuzzy C-Means clustering (Adini et al 1997) is an algorithm in which each image is associated with a cluster through a membership degree. The purpose of using FCM is to decrease the data size by grouping similar
elements. FCM employs fuzzy partitioning such that a given data point can belong to several groups with a degree specified by membership grades between 0 and 1. Clustering is a mathematical tool that attempts to discover structures or certain patterns in a data set, where the objects inside each cluster show a certain degree of similarity.

Actually, clustering is an unsupervised learning task that aims at decomposing a given set of objects into subgroups or clusters based on similarity. Each data is a member of every cluster but with a certain degree known as membership value. This method is frequently used in pattern recognition as Fuzzy partitioning is carried out through an iterative procedure that updates membership $u_{ij}$ and the cluster centroids $c_j$. Apart from assigning a data point to clusters in shares, membership degrees can also express how ambiguously a data point should belong to a cluster.

The aim of the FCM is to minimize the Cost Function $J$ which uses squared Euclidean distance as the dissimilarity measure between a vector $x_k$ in group $i$ and its corresponding cluster center $c_i$ (Jianming Lu et al 2007). The cost function is given by,

$$J(U, c_1, c_2, \ldots, c_c) = \sum_{i=1}^{c} j_i = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^m d_{ij}^2$$

(5.7)

where $u_{ij}$ is between 0 and 1; $c_i$ is the cluster center of fuzzy group $i$; $d_{ij}=c_i-x_j$ is the Euclidean distance between $i^{th}$ cluster center and $j^{th}$ data point and ‘$m$’ is a weighting exponent. Here, preprocessing is done for image normalization and FLDA is used to simplify the given dataset into lower dimension while retaining the important characteristics of dataset.

The entire process of BPNN and FCM based face recognition is as shown in Figure 5.9.
FCM produces the idea of uncertainty of belonging, described by a membership function and it enables an individual to belong to several networks. Defuzzification is employed here for the purpose of reducing a number of elements in the cluster by comparing their Euclidean distance with that of the threshold value.

The problem of fuzzy clustering is to find a fuzzy pseudo partition and the associated cluster centers, by which the structure of the data is represented as good as possible. This requires some criterion expressing the general idea being strong within clusters and weak between clusters. Fuzzy clustering has been adapted successfully to solve this problem, which is a
hard combinatorial problem. However, when the problem becomes larger, fuzzy clustering algorithms may result in uneven distribution of suppliers.

An important criterion for any clustering algorithm is the “distance measure” that helps in determining the similarity of two elements. Here it is by the Euclidean distance. Fuzzy clustering allows each feature vector to belong to more than one cluster with different membership degrees (between 0 and 1) and vague or fuzzy boundaries between clusters.

One popular technique involves using the FCM algorithm is to compute the membership values for different classes before the final segmentation. Fuzzy C- means clustering is a data clustering algorithm in which each data point belongs to a cluster to a degree specified by a membership grade. Fuzzy C -means partitions a collection of n vectors into C fuzzy groups, and finds a cluster center in each group such that a cost function of dissimilarity measure is minimized.

The algorithm performs extremely well in situations of large variability of cluster shapes, densities and number of data points in each cluster. The FCM algorithm always converges to a strict local minimum of objective function, but different choices of the initial fuzzy membership can lead to different local minima. This implies that the FCM clustering algorithm could render a different partition matrix for a different random initial membership, especially when the given number of clusters is large. The FCM algorithm assigns membership value to a data sample based on its proximity to the cluster prototypes in the feature space.

In the FCM-based segmentation algorithm, feature vectors are assumed to be independent of each other and independent of their spatial coordinates.
The main objective of fuzzy C-means clustering algorithm is that it tries to minimize total intra-cluster variance. Thus, the incorporation of local spatial interaction between adjacent pixels in the fuzzy clustering process can produce more meaningful classification, as well as help to resolve classification ambiguities due to overlap in intensity value between clusters or noise corruption.

Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, with the update of membership $u_{ij}$ and the cluster centers $c_j$. This iteration will stop when, where $C$ is a termination criterion between 0 and 1, whereas $k$ is the iteration steps. This procedure converges to a local minimum or a saddle point of $J_m$.

### 5.2.2 FCM Algorithm

Fuzzy C-means (FCM) is a method of clustering which allows one piece of data to belong to more than one cluster (i.e.,) each data is a member of every cluster but with a certain degree known as membership value. Fuzzy partitioning is carried out through an iterative procedure that updates membership $u_{ij}$ and the cluster centroids $c_j$.

1. Choose the number of clusters $c$, with $1 < c < n$, the fuzziness exponent $m$, with $m > 1$, value for the stopping criterion.

2. Initialise all $u_{ij}$, membership values randomly and form a membership matrix $U^0$.

3. At step $k$: Compute centroids, $c_i$ using

$$c_i = \frac{\sum_{j=1}^{N} u_{ij}^m x_j}{\sum_{j=1}^{N} u_{ij}^m} \quad i = 1, 2, ..., c$$

(5.8)
4. Calculate objective function $J$ using

$$ J(M,C) = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^m d_{ij}^2 $$

(5.9)

where $m$ is the degree of Fuzziness.

5. Compute new membership values, $u_{ij}$ using

$$ u_{ij} = \frac{1}{\sum_{k=1}^{c} \left( \frac{\left\| x_j - c_j \right\|^2}{\left\| x_i - c_k \right\|^2} \right)^{\frac{1}{m-1}}} \quad j=1,2,\ldots,n $$

(5.10)

where

$$ d_j = \left\| C_i - X_j \right\| $$

6. Update $U_{k+1}$ from $U_k$

7. Repeat steps 2-6 until change of membership values is very small, $U_{k+1} - U_k < \varepsilon$ where $\varepsilon$ is some small value, typically 0.01.

8. At each iteration, the objective function is minimized to find the best location for the clusters.

The FCM algorithm is simply iteration of two conditions as given by $c_i$ and $u_{ij}$ as per the equations (5.8) and (5.10). In this work, total number of images taken is around 400 and the number of clusters is 7. The membership matrix is initialized with random values. Then the center for each cluster, membership matrix and the objective function are calculated and this process is repeated until the objective function is converged to the threshold value. Thus the similar face images are clustered into groups by FCM and thereby the training time taken for next process, i.e. BPNN is reduced.
Difficulties with Fuzzy Clustering: The optimal number of clusters to be created has to be determined. The number of clusters cannot always be defined a priori and a good cluster validity criterion has to be found. The character and location of cluster prototypes (centers) is not necessarily known a priori, and initial guesses have to be made. The data characterized by large variability in cluster shape, cluster density, and the number of points (feature vectors) in different clusters have to be handled. However, when the problem becomes larger, fuzzy clustering algorithms may result uneven distribution of suppliers.

5.2.3 Experimentation and Results of FCM

Various algorithms implemented in this section are PCA, FCM clustering and BPNN. A sample of 20 face images of ORL database and test images are taken as shown in Figures 5.10 and 5.11. At first the size of database images and test images are 112×92, then it is reduced to a size of 20×19 by PCA. The output of PCA in reduced dimensionality for the columns 1 through 11 is shown in Figure 5.12.

![Figure 5.10 Sample training set](image)
The output of PCA is given as input for FCM, which finds clusters in the data set. Here, the Data size is M*N, where M is the number of data points and N is the number of coordinates for each data point. The membership function matrix U contains the grade of membership of each data point in each cluster. Grades between 0 and 1 indicate that the data point has partial membership in a cluster.
At each iteration, the objective function is minimized to find the best location for the clusters and its values are returned in objective function. Thus the size $20 \times 19$ of PCA is reduced to $7 \times 20$ by FCM algorithm. The output of FCM for the given 20 standard database image is shown in Figure 5.13. Then the dataset of FCM output $7 \times 20$ is given to BPNN training.

![Figure 5.13 The output of FCM](image)

The final output of the PCA, FCM with BPNN for 20 images is given by,

<table>
<thead>
<tr>
<th>Face 1 is Non-registrant</th>
<th>Face 6 is Registrant</th>
<th>Face 11 is Registrant</th>
<th>Face 16 is Registrant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Face 2 is Registrant</td>
<td>Face 7 is Registrant</td>
<td>Face 12 is Registrant</td>
<td>Face 17 is Registrant</td>
</tr>
<tr>
<td>Face 3 is Registrant</td>
<td>Face 8 is a Non-registrant</td>
<td>Face 13 is Registrant</td>
<td>Face 18 is Registrant</td>
</tr>
<tr>
<td>Face 4 is Registrant</td>
<td>Face 9 is Registrant</td>
<td>Face 14 is Registrant</td>
<td>Face 19 is Registrant</td>
</tr>
<tr>
<td>Face 5 is Registrant</td>
<td>Face 10 is a Non-registrant</td>
<td>Face 15 is Registrant</td>
<td>Face 20 is Registrant</td>
</tr>
</tbody>
</table>

Execution Time taken for this analysis is 42.3 seconds

Here, the programme is trained for a database of group of 50, 100, 200, 300 and 400 images. As the number of database images is increased, various problems in FCM and neural network are crop up. The value of recognition rate is going on decreasing and execution time is increased, as the number of images is increased. A set of same images with different expressions of ORL database is shown in Figure 5.14.
Figure 5.14  Same images with different expression in the Test database

Image 1 is Non registrant  Image2 is Registrant  Image 3 is Registrant  Image 4 is Registrant
Image 5 is Registrant  Image 6 is Registrant  Image 7 is Registrant  Image 8 is Registrant
Image 9 is Registrant  Image 10 is Registrant  Image 11 is Non registrant  Image 12 is Registrant
Image 13 is Registrant  Image 14 is a Registrant  Image 15 is Registrant  Image 16 is Registrant
Image 17 is Non registrant  Image 18 is Registrant  Image 19 is a Registrant  Image 20 is Registrant

Execution Time taken for this analysis is 36.40 seconds

The performance of BPNN with and without FCM for ORL database is demonstrated in Table 5.1. Performance metrics for this work are recognition rate and execution time. Analysis for this comparison is shown in Figure 5.15. Recognition rate is calculated based on the number of images recognized correctly to the total number of database images. Execution time is calculated in MATLAB program by using Tic and Tac comments. It is observed that recognition rate is approximately increased by 10-15 %, the execution time is reduced by 10 -20 %% for BPNN with FCM algorithm.
Table 5.1  Comparison of time and recognition rate for BPNN with and without FCM

<table>
<thead>
<tr>
<th>No. of Images</th>
<th>BPNN without FCM</th>
<th>BPNN with FCM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Execution Time (Sec)</td>
<td>Recognition rate (%)</td>
</tr>
<tr>
<td>50</td>
<td>30.09</td>
<td>94</td>
</tr>
<tr>
<td>100</td>
<td>39.51</td>
<td>90</td>
</tr>
<tr>
<td>200</td>
<td>45.26</td>
<td>88</td>
</tr>
<tr>
<td>300</td>
<td>51.02</td>
<td>86</td>
</tr>
<tr>
<td>400</td>
<td>60.26</td>
<td>82</td>
</tr>
</tbody>
</table>

Figure 5.15  Comparison of execution time and recognition rate for BPNN with and without FCM
Table 5.2  Comparison of recognition rate (DW+FLDA) and (FCM+BPNN)

<table>
<thead>
<tr>
<th>No. of images</th>
<th>Recognition rate (%)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DB+FLDA</td>
<td>FCM+BPNN</td>
</tr>
<tr>
<td>50</td>
<td>93</td>
<td>98</td>
</tr>
<tr>
<td>100</td>
<td>89</td>
<td>95</td>
</tr>
<tr>
<td>200</td>
<td>88</td>
<td>94</td>
</tr>
<tr>
<td>300</td>
<td>85</td>
<td>92</td>
</tr>
<tr>
<td>400</td>
<td>81</td>
<td>89</td>
</tr>
</tbody>
</table>

Figure 5.16  Comparison of recognition rate (DW+ FLDA) and (FCM+ BPNN)
Table 5.3  Comparison of Execution Time (DW+FLDA) and (FCM+BPNN)

<table>
<thead>
<tr>
<th>No. of Images</th>
<th>Execution time ( sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DB wavelets+ FLDA</td>
</tr>
<tr>
<td>50</td>
<td>35.69</td>
</tr>
<tr>
<td>100</td>
<td>44.05</td>
</tr>
<tr>
<td>200</td>
<td>53.02</td>
</tr>
<tr>
<td>300</td>
<td>64.16</td>
</tr>
<tr>
<td>400</td>
<td>76.54</td>
</tr>
</tbody>
</table>

Figure 5.17  Comparison of Execution time (DW + FLDA) and (FCM + BPNN)
Table 5.4  Comparison of Accuracy for different database images

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>No of Images</th>
<th>ORL</th>
<th>Yale</th>
<th>FERET</th>
<th>Real time images</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20</td>
<td>95</td>
<td>90</td>
<td>90</td>
<td>85</td>
</tr>
<tr>
<td>2</td>
<td>40</td>
<td>92.5</td>
<td>87.5</td>
<td>87.5</td>
<td>82.5</td>
</tr>
<tr>
<td>3</td>
<td>60</td>
<td>90.0</td>
<td>85</td>
<td>85</td>
<td>81.66</td>
</tr>
<tr>
<td>4</td>
<td>80</td>
<td>88.75</td>
<td>85</td>
<td>83.75</td>
<td>80</td>
</tr>
<tr>
<td>5</td>
<td>100</td>
<td>88</td>
<td>84</td>
<td>82</td>
<td>79</td>
</tr>
</tbody>
</table>

Figure 5.18  Comparison of Accuracy for different database images
5.2.4 Summary

In this chapter, techniques for the recognition of different database face images based on PCA, FCM clustering and BPNN are presented. Dimensionality of face images are reduced by PCA and FCM is used to decrease the number of segments by grouping similar elements. Due to this data reduction of PCA and FCM, total computational time taken by BPNN is reduced. Based on execution time and recognition rate, a comparison is made by BPNN with and without FCM for a set of 50, 100, 200, 300 and 400 database images. It is concluded that recognition is better for BPNN with FCM which has 15% improvement in recognition rate and 10% reduction in execution time approximately.