ABSTRACT — This Most of the common person verification and identification methods (Password, Personal Identification Number (PIN) systems or other similar techniques) used today are not unique, and is possible for somebody to forget, lose or even have it stolen for somebody. Design Authentication system, which use pattern recognition techniques to identify people by using their characteristics. This project concentrates on face recognition using a neural network.

KEYWORDS — face recognition, Principal Component Analysis (PCA), Artificial Neural Network (ANN)

I. INTRODUCTION

The face is a complex multidimensional structure and needs good computing — techniques for recognition. The face is our primary and first focus of attention in social life and playing an important role in the identity of the individual. We can recognize a number of faces learned throughout our lifespan and identify that faces at a glance even after years although there may be variations in faces due to ageing and distractions. Face recognition is a part of a wide area of pattern recognition technology. Recognition and especially face recognition covers a range of activities from many aspects of our life.

II. METHODOLOGY

This Chapter concentrate on the method of face recognition and how the system recognize faces starting from the input image to the output result. The process goes through stages are shown in figure [1]}

Figure 1: Methodology block diagram.

There are four basic steps involved in the proposed model, after inserting the image to Matlab platform, the first step is image preprocessing basically in image processing system, image acquisition and enhancement are steps that need to be done. All the images are collected from people around us. The proposed model requires converting the image into a format capable of being manipulated by the computer. The second step is PCA. The images are converted into matrixes and vectors using PCA. The Third step is neural network; the neural network model is developed. After the NN model successfully developed, the output result is a fourth and final step. All that done by MATLAB programming

Dataset: The dataset was used in this project is a small one consists of 12 images, three persons with three images for each one. All of the images were taken in a standing position and on identical background, as shown in Figure [2].

Figure (2): Dataset

Principal Component Analysis (PCA): It is a way of identifying patterns in data, and expressing the data in such a way as to highlight their similarities and differences. Since patterns in data can be hard to find in data of high dimension, where the luxury of the graphical representation is not available, PCA is a powerful tool for analyzing data. [3]

The other main advantage of PCA is that once you have found these patterns in the data, and you compress the data, i.e. by reducing the number of dimensions, without much loss of information.

In this project, the principal component analysis (PCA) is used as a feature extraction algorithm. The principal component analysis (PCA) is one of the
most successful techniques that have been used in image recognition and compression.

The purpose of PCA is to reduce the large dimensionality of the data. MR image recognition systems find the identity of a given test image according to their memory. The memory of an MR image recognizer is generally simulated by a training set. In this project, the training database consists of a set of MR images. Thus, the task of the MR image recognizer is to find the most similar feature vector among the training set to the feature vector of a given test image.

In the training phase, feature vectors are extracted for each image in the training set. Let $\Omega_1$ be a training image of image one which has a pixel resolution of $M \times N$ ($M$ rows, $N$ columns). In order to extract PCA features of $\Omega_1$, first convert the image into a pixel vector $\Phi_1$ by concatenating each of the $M$ rows into a single vector. The length (or, dimensionality) of the vector $\Phi_1$ will be $M \times N$. In this project, the PCA algorithm is used as a dimensionality reduction technique which transforms the vector $\Phi_1$ to a vector $\omega_1$ which has a dimensionality $d$ where $d \ll M \times N$. For each training image $\Omega_i$, these feature vectors $\omega_i$ are calculated and stored for training. Our image after edge detection and gaps filling are [64 -by- 64] as input to PCA algorithm the output of PCA will be [64 -by- 1], this data form represents the hole image and ready to enter the ANN for training and simulation to get true and cost regression. The algorithm as follow:

**Step 1:** Get some data the image got 2dimensions matrix, a binary image with edge values.

**Step 2:** Subtract the mean for PCA to work properly, you have to subtract the mean from each of the data dimensions. The mean subtracted is the average across each dimension. So, all the $m$ values have $m$ (the mean of the $x$ values of all the data points) subtracted, and all the $n$ values have $n$ subtracted from them. This produces a data set whose mean is zero.

**Step 3:** Calculate the covariance matrix recall that covariance is always measured between 2 dimensions. If we have a dataset with more than two dimensions, there is more than one covariance measurement that can be calculated. Since the data is two dimensional, the covariance matrix will be $2 \times 2$. So, since the non-diagonal elements in this covariance matrix are positive, we should expect that both the $m$ and $n$ variable increase together.

**Step 4:** Calculate the eigenvectors and eigenvalues of the covariance matrix

Since the covariance matrix is square, we can calculate the eigenvectors and eigenvalues for this matrix. These are rather important, by this process of taking the eigenvectors of the covariance matrix, we have been able to extract lines that characterize the data. The rest of the steps involve transforming the data so that it is expressed in terms of the $m$ lines.

**Step 5:** Choosing components and forming a feature vector

Here is where the notion of data compression and reduced dimensionality comes into it. In general, once eigenvectors are found from the covariance matrix, the next step is to order them by eigenvalues, highest to lowest. This gives you the components in order of significance. To form a feature vector, which is just a fancy name for a matrix of vectors. This is constructed by taking the eigenvectors that you want to keep from the list of eigenvectors and forming a matrix with these eigenvectors in the columns.

**Step 6:** Deriving the new data set. This final step in PCA, and is also the easiest. Once we have chosen the components (eigenvectors) that we wish to keep in our data and formed a feature vector, we simply take the transpose of the vector and multiply it on the left of the original data set, transposed. [4]

C) Neural Network model: In this model, used the fitting network as the proper one for our data which were extracted from PCA. The fitting network creates a back propagation network. Back propagation has fast learning property. [10]

`trainlm` is the default training function because it is very fast, but it requires a lot of memory to run. If you get an "out-of-memory" error when training, try doing one of these approaches:

1. Slow `trainlm` training, but reduce memory requirements, by setting NET.trainParam.mem_reduce to 2 or more.
2. Use `trainbfg`, which is slower but more memory efficient than `trainlm`.
3. Use `trainrp`, which is slower but more memory efficient than `trainbfg`.

The learning function `BLF` can be either of the backpropagation learning functions such as `learned` or `learngdm`. The performance function can be any of the differentiable performance functions such as `mserror`.

Feed-forward networks consist of $n$ layers using the dotprod weight function, `netsum` net input function, and the specified transfer functions.

The first layer has weights coming from the input. Each subsequent layer has a weight coming from the previous layer. All layers have biases. The last layer is the network output. [5]

Neural network was used in this project is backpropagation feed forward, it has three layers one hidden layer with ten neuron and input, output layer with sixty four neuron for each.
The sixty-four input neuron was selected according to the output of PCA which is [512-by-1] matrix. The hidden layer is ten neurons because it reduces the mean squared error due to the complexity of calculation which is produced from the large number of neuron. Output layer must be same number of the input layer neurons to give an efficient and accurate linear regression between the output matrix and target matrix. [6]

d) Classification and Detection: Classification and Detection accomplish by observing several parameters resulted from network training such as: performance, fitting, mean squared error or linear regression between the output and the target which is made before as reference data. In this project we depend on linear regression. Depend on criteria linear regression is easy to observe and calculated. Also it has high efficiency to show the difference between data and it can be trusted.

e) Output Phase: The output phase involves the application of the model to observe how well the model reacts to the untrained data and also the detection based on the data itself. The result correlation coefficient (r) is compared with based coefficient made in the training phase. Now we have classes for each (r) to give the status of the tested image and find the specific output result of that tested image.

CONCLUSION

Face recognition is one of the pattern recognition techniques. The most important advantage of face recognition is that it identifies people by using their characteristics. The system that has been designed is used for classifying face images from a known dataset. Based on the result that we reached in chapter four, it can be concluded that this project has achieved the objectives and problem statements as set earlier in chapter one. Experimental result indicates that the system is workable with high accuracy.

REFERENCES