IJCCN International Journal of Computer Communications and Networks,
VOLUME 4, ISSUE 1, FEBRUARY 2014, ISSN 2289-3369
WWW.IARTC.NET

IMAGE PROCESSING AND FACE DETECTION SYSTEM WITH FACE RECOGNITION BASED FACE ALGORITHM EIGENVECTORS PRINCIPAL COMPONENT ANALYSIS (PCA)

[1, 2)] **Setyawan Widyarto,** [2)] **Dwi Hartanto, Jose Rizal, Dimas Satya Wicaksono**
1111 600 233, 1111 600 308, 1111 600 266
[1,2)]*Faculty of Computer Science and Information Technology, Universiti Selangor, Selangor, Malaysia*
[2)]*Fakultas Teknologi Informasi, Universitas Budi Luhur Jakarta, Indonesia*
gandungdwi@yahoo.co.id

ABSTRACT

The face is one of the easiest ways to distinguish the individual identity of each other. Face recognition is a personal identification system that uses personal characteristics of a person to identify the person's identity. Human face recognition procedure basically consists of two phases, namely face detection, where this process takes place very rapidly in humans, except under conditions where the object is located at a short distance away, the next is the introduction, which recognize a face as individuals. Stage is then replicated and developed as a model for facial image recognition (face recognition) is one of the much-studied biometrics technology and developed by experts. There are two kinds of methods that are currently popular in developed face recognition pattern namely, Eigenface method and Fisherface method. Facial image recognition Eigenface method is based on the reduction of face-dimensional space using Principal Component Analysis (PCA) for facial features. The main purpose of the use of PCA on face recognition using eigenfaces was formed (face space) by finding the eigenvector corresponding to the largest eigenvalue of the face image.

Keywords: ***face detection, Eigenface, PCA, matlab***

I. INTRODUCTION

Growing information technology today has produced many different applications using a face image as a source of information. This is because in general a facial image can provide specific information related to personal identification based face recognition that can be used in an electronic security system. Advantage that the security of the system based on face recognition is the ability of the security that is relatively difficult to penetrate (Soelaiman, 2006).

Face recognition is a pattern recognition approach for personal identification purposes in addition to other biometric approaches such as fingerprint recognition, signature, retina and so forth. Facial image recognition associated with the object that is never the same, because of the parts that can be changed. These changes can be caused by facial expressions, intensity of light and camera angles, or change accessories on the face. In this regard, the same object with some differences should be recognized as the same object. Broadly speaking, the methods used in the process of face recognition, there are three kinds of holistic methods, methods based on characteristics, and hybrid method. Among holistic methods, methods based on appearance (appearance basedmethode) is a very successful technique for face recognition in recent years. When using a method based appearance, image size nxm Pikes described as a vector in the space dimension nxm (Rnxm).

In practice, nxm dimensional space is too large to make a quick facial image recognition. To solve this problem is usually done using dimension reduction technique (Yambor, 2000). Dimension reduction techniques which are well known Eigenfaces technique-oriented method of Principal Component Analysis (Belhumeur et al, 1997). PCA which will be integrated with a linear classification technique would also be used in the feature extraction stage in the process of introducing image of a human face (Marti, 2007).

1.1 LIMITATION OF PROBLEMS

In this task the problem is limited to the discussion:

- digital image processing method to be used is the Principal Component Analysis or PCA (Principal Components Analysis).
- facial image used in the test is the image of a face looking straight ahead.
- auxiliary program used is Matlab version 7.11.0
- Issues that will be examined is limited to: the influence of the number of training images, the effect of the amount of the main components of the face recognition results, and the determination of threshold values using standard deviation.

1.2 PURPOSE

The aim of this project is to make program aids recognition of human faces using principal component analysis (PCA) and to produce conclusions about the appropriateness of PCA as a method of face recognition.

II. THEORETICAL

2.1 FACE

The face is a part of the human body are the focus of attention in social interactions, facial play a vital role by demonstrating identity and emotion. The human ability to know one of his very unusual. We can recognize thousands of faces because of the frequency of interaction very often or even just a glimpse of the very long time span.

Even though we were able to identify a person of any changes in the person because of age or wear glasses or hair style changes. Therefore, the face is used as an organ of the human body is used as an indication of the introduction of one or face recognition.

2.2 CONCEPT OF FACE RECOGNITION

Face recognition is a method of face recognition-oriented. The introduction can be divided into two parts: Recognized or unrecognized, after comparison with previous patterns stored in the database. This method should also be able to recognize objects instead of faces.

Calculation model of face recognition has some issue. Difficulties arise when the face represented in a pattern that contains unique information that differentiates it from other faces. The method of face recognition using two procedures, namely:

a. Introduction by recognizing facial contour shape of the nose, eyes and mouth and shape of the correlation between the two. The characteristics of the organ is then expressed as a vector
b. Principal component analysis, based on information from this concept, finding the best model calculations describe the shape of the face by quoting the most relevant information contained in those faces.

Behind easy to recognize faces, there are some problems that may arise in the process of face recognition, namely a change of scale, position changes, lighting changes, or a change in detail and facial expressions..

2.3 VECTOR AND THE FACE

A face, which is also an image, can be seen as a vector. If pxl image pixel size, the number of components of the vector is pxl. Construction of vector of an image is formed by a simple merging all lines from an image and placed side by side with the other, as seen in Figure 1 below:

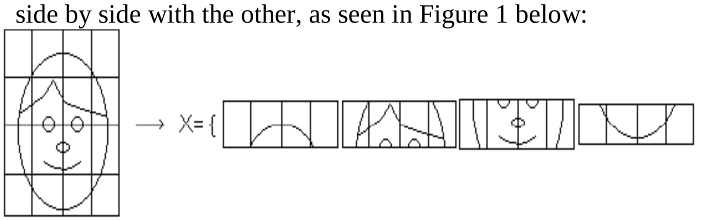

Figure 1. Formation Vectors of A Picture

Vector face have been described previously is part of an image space is the space of the whole image has dimension pxl pixels. All faces resemble each other in which these faces have two eyes, one nose, one mouth, two ears, etc. are located in the same place. Consequently all faces vector lies in places very close together in the image space. Basis vector of the face space is referred to the main component (principal component).

2.4 PRINCIPLE COMPONENT ANALYSIS (PCA)

Based on the conclusions given by Kirby and Sirovich (1990), demonstrated that the optimal basis that can be used to represent a vector image given by the eigenspace formed from the eigenvalues are not zero (nonzero eigenvalues) of the covariance matrix of the entire image.

By using the basis formed by the eigenspace dimension reduction can be done by performing a linear transformation of a space high dimension into a lower-dimensional space. To determine a lower dimension with errors (error, information loss) that minimum can be done by selecting a number of the largest eigenvalues of the high dimensional space.

Stage is a general description of the method called Principal Component Analysis (Principal Component Analysis). Turk and Pentland (1991) says that if there are M fruit facial image is $\Gamma_1, \Gamma_2, \ldots, \Gamma_M$ used in training and each face image is viewed as a vector of length N faces (rows x columns). The average face of the training data set can be obtained by equation (1) as follows:

$$\Psi = \sum_{i=1}^{M} \Gamma_i \quad (1)$$

Then, each face vector is reduced by an average face, with equation (2).

$$\Phi_i = \Gamma_i - \Psi \quad (2)$$

Set of vectors is a huge requirement for principal component analysis, which is looking for a set of m (m << N), orthonormal vectors μk corresponding to the eigenvalue λk distribution best describes the data. Where the collection of eigen vector with eigenvalue best a basis vector of the PCA or commonly called the Eigenfaces. Vector and scalar λk μk respectively are the eigenvectors and eigenvalues of the covariance matrix of the training data (Gonzalez at all, 2002). Covariance matrix sought by equation (3).

$$C = \frac{1}{M}\sum_{i=1}^{M} \Phi_i \Phi_i^T = AA^T \quad (3)$$

where the matrix $A = [\Phi_1, \Phi_2, \ldots, \Phi_M]$ .Covariance matrix of the data (C) will be sized NxN, and used to find the N eigenvectors and eigenvalues. This value has a huge dimension to the image. For it is computing, it needs a simpler method to obtain eigenvectors eigenvector-this.

To overcome this, you can use the smaller dimensions of the matrix is mxm for the PCA, where M is the number of images used in training. By Yambor (2000), the way is known as Snapshot Eigenface method. With this analysis, the calculation is reduced dramatically, from the size of the number of pixels N to M (M << N). Solving method is used when the amount of training data are much smaller than the dimensions of the face vector.

2.5 METHOD EIGENVECTORS FACE

Eigen face is one of the face recognition algorithm based on the Principle Component Analysis (PCA), which was developed at MIT. Eigen Face Algorithms overall pretty simple. Training Image represented in a vector flat (combined vector) and merged together into a single matrix. Eigen Vector then extracted and stored in a temporary file or database. Training then later projected image in feature space, called the face space defined by the eigen vectors

In the method of eigen face preliminary processing on a two-dimensional image that is used needs to be done. Initial processing goal is to accelerate the performance and reduce the size of memory used in object recognition three (3) dimensions. Initial processing is done by extracting the characteristics of the image being dimensionless N-dimensional image of M where **M <N**. Two-dimensional image of three-dimensional objects with different viewpoints are recognized collected to represent the object as a reference image. Feature extraction is performed on a collection of objects to get information cirri. The results are then used for feature extraction process of the introduction of three-dimensional objects.

Understanding eigenvalues and eigen vectors can be better understood when viewed in the form of physical problems. Suppose there is a piece of two-dimensional elastic membrane that can be expressed in the coordinates x and y, the membrane gets physical treatment that can be pushed, pulled and rotated about the origin. If the membrane is treated above, after the initial deformation point on the membrane plane (x, y) changes position to (X, Y) and we can say there is a matrix M that describes the shape of the deformation.

Changes in the position vector can be expressed as **R = μr = μ** constant where vector **R** as eigen vectors and μ is called the eigenvalues of the transformation matrix **M**. To illustrate the determination of eigenvalues, review the form of linear transformations in two (2) dimensions:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} 5 & -2 \\ -2 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Terms of eigen vector R = μr above form can be written as:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} 5 & -2 \\ -2 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \mu \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \mu x \\ \mu y \end{bmatrix}$$

Or it can also be written as:

$$\begin{aligned} (5-\mu)x - 2y &= 0 \\ -2x + (2-\mu)y &= 0 \end{aligned} \qquad (1)$$

If the problem is solved as a form of homogeneous equations with the determinant, obtained x = 0 and y = 0. Unless the determinant of the coefficients equal to 0, then the eigenvalues obtained are **μ = 1** or **μ = 6** and eigen vectors **2x-y = 0** and **x +2 y = 0**, as can be seen in Figure 2.

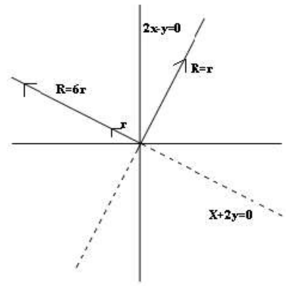

Figure 2. Physical interpretation of eigenvalues and Eigen Vectors

Interpretation of these forms are the eigenvalues of the transformation matrix gives information how the deformation field is given, while the eigen vectors of the transformation matrix deformation change direction information field.

2.6 IMAGE REPRESENTATION OF REFERENCE

Describe an object in a room full image (spatial) is not optimal when used to recognize objects in three (3) dimensions. Shape representation of the reference image is used Eigenface representation or eigenspace. The basic idea of this representation is representing a set of images or characteristics of an image in a space of transformation in which each trait was not correlated (Chandra et al., 1997).

Eigenspace representation form can be obtained by transforming the Karhunen-Loeve or Principal Component Analysis (PCA) to a set of reference image. The results of this transformation is the orthonormal basis vectors are used to form a sub-vector space called the characteristic space.

An image space can be viewed as a vector (Rhomdani, 1997). An image with width w and height h will form a vector that has as many components with size w * h 1 * (w * h). The vector was prepared by incorporating the image rows are arranged next to one another, as shown in Figure 3 below.

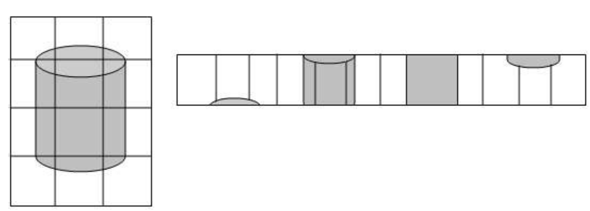

Figure 3. Vector image formation of a 2D image

Vector reference image is transformed using Karhunen-Loeve transformation or Principal Component Analysis (PCA), which will result in a number of orthonormal basis vectors. Orthonormal basis vectors are used to form the characteristic space for the object to be recognized. Characteristics xxtraction of an image is done by mapping the input image into the traits that have been produced.

2.7 FACE RECOGNITION SYSTEM USING PCA

A face in the form of two-dimensional images can be viewed as one-dimensional vectors. If the length and width of the image is the image of w is h, then the number of components of the vector 1 is WxH dimensions. Each pixel is encoded by a single component of the vector and the vectors are arranged as a column, as shown in Figure 4.

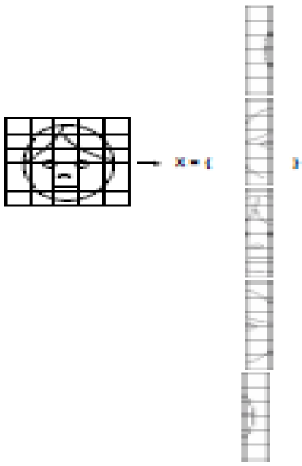

Figure 4. Changes in two-dimensional images into one-dimensional vector

Vector face in Figure 1 are then put into a room, the room is called space image, the image space to hold a number of size WxH pixels. Characters identifying vectors of any face is only located in certain parts of the image space. Therefore, the image space is too much to describe each face in it, and then it is necessary to build a new space, namely the face. Principal Component Analysis (Principal Component Analysis) or PCA is a method that involves a mathematical procedure that converts and transforms a large number of correlated variables into a small number of uncorrelated variables, without losing important information in it. The process of the face recognition system using PCA can be seen more clearly in Figure 5.

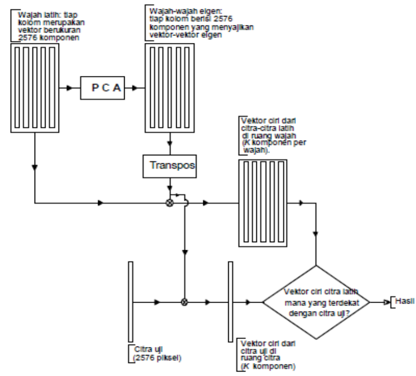

Figure 5. Face recognition system using PCA

2.8 PCA IN STATISTICS

PCA calculations began with the formation of the covariance matrix of a set of training images in the data-base. To make the covariance matrix can be done by subtracting each column of a matrix space image with the average respectively to obtain a matrix transformation, then multiplying the result with matrix transpose matrix itself. If we let the transformation matrix is X, then the matrix is covariance ΣX = XXT.

Basically, the covariance matrix ΣX = XXT □ □ of a set of training face is not shaped diagonal

$$\Sigma_X = XX^T = \begin{vmatrix} \sigma_{11}^X & \sigma_{12}^X & \cdots & \sigma_{1,w\times h}^X \\ \sigma_{21}^X & \sigma_{22}^X & \cdots & \sigma_{22}^X \\ \cdots & \cdots & \cdots & \cdots \\ \sigma_{w\times h,1}^X & \sigma_{w\times h,2}^X & \cdots & \sigma_{w\times h,w\times h}^X \end{vmatrix} \quad (1)$$

$\sigma\, i\, j$ presents the covariance between pixel i and pixel j. The relationship between the coefficient covariance and correlation coefficients are:

$$r_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{ii} \cdot \sigma_{jj}}} \quad (2)$$

so that the correlation coefficient is the normalization of the coefficient covariance. The aim is to build a facial room, with each of the components are not correlated. This means that the covariance matrix of the new component to be shaped diagonal:

$$\Sigma_Y = YY^T = \begin{bmatrix} \sigma_{11}^Y & 0 & \cdots & 0 \\ 0 & \sigma_{22}^Y & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \sigma_{w\times h,w\times h}^Y \end{bmatrix} \quad (3)$$

with yi is a column vector describing the axis xi face facial room, yi also called principal components; X is a matrix containing the number of face training, xi, Y is a matrix containing the vectors yi. Diagonal form of the covariance matrix shows that the variance a variable by variable itself would be worth a maximum, whereas a variable covariance with other variables will be zero, every variable is not correlated with each other anymore. Thus, the formation of new faces this space is to find the direction of the axis which maximizes the variance.

The main component of the matrix X can be computed linearly. If P is a transformation matrix, then:

$$\boldsymbol{Y = PT\ X} \textbf{ and } \boldsymbol{X = PY} \qquad (4)$$

essentially P = P-1, for each column of the vector P are mutually orthonormal, with: PT P = I, here I is an identity matrix. It is the question now is whether it should be if **ΣY** P is a diagonal matrix.

$$\Sigma_Y = YY^T = P^T XX^T P = P^T \Sigma_X P \qquad (5)$$

$$\Sigma_X P = \Lambda P \qquad (6)$$

is a diagonal matrix with which contains a number of eigenvalues of the matrix ΣX. Thus, ΣY is diagonal and contains a number of eigenvalues of ΣX, since each element of the diagonal is the variance of the component ΣY faces practiced in the face, which is also an eigenvalue of the matrix ΣX.

$$\Sigma_Y = P^T \Lambda P = \Lambda P^T P = \Lambda \qquad (7)$$

III.DISCUSSION MATERIALS

The system is built using a PC platform with writing a computer program using Matlab R2009a. The system was tested using a variety of test image database consisting of 10 people with each one consisting of 1 image. The system was also tested using the train database consisting of 10 people with each consisting of 2 images. The size of each image is 180x200 pixels.

The program is made generally consists of two modules, namely training module and testing module. Training modules produce image parameter eigenvector and the average for the PCA.

3.1 SPECIFICATION FOR DESIGN

A face image recognition generally has several stages, the first stage is the image acquisition stage, followed by a computation phase, Eigenface. Once both phases are completed, the stage can proceed with phase three, the face recognition.

In the process of face recognition with Eigenface method is all that is used as the input image and the image used lo introduction phase should have the dimensions of the same size and orientation.

In this task used 20 faces consisting of 10 people. Everyone was taken by 2 sample images with different expressions and positions using a digital camera image input. The image taken is stored using JPG format and color images with size 180 X 200 Pixels. Here's a face image are taken as sample and used in this task:

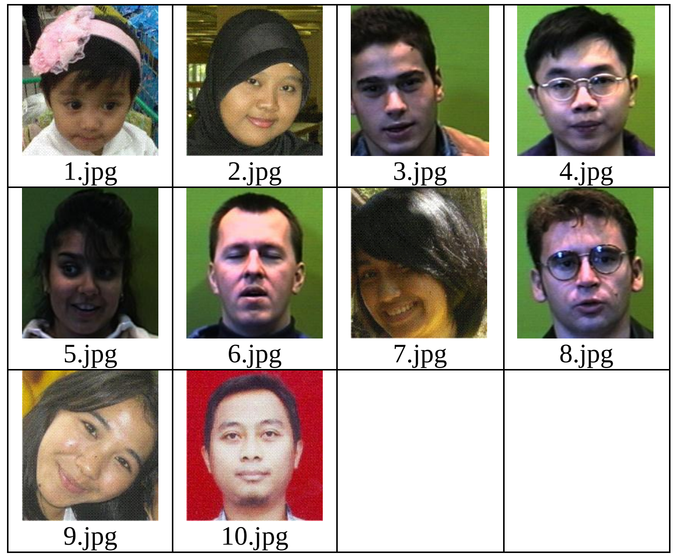

Figure 6. Image On Module Testdatabase

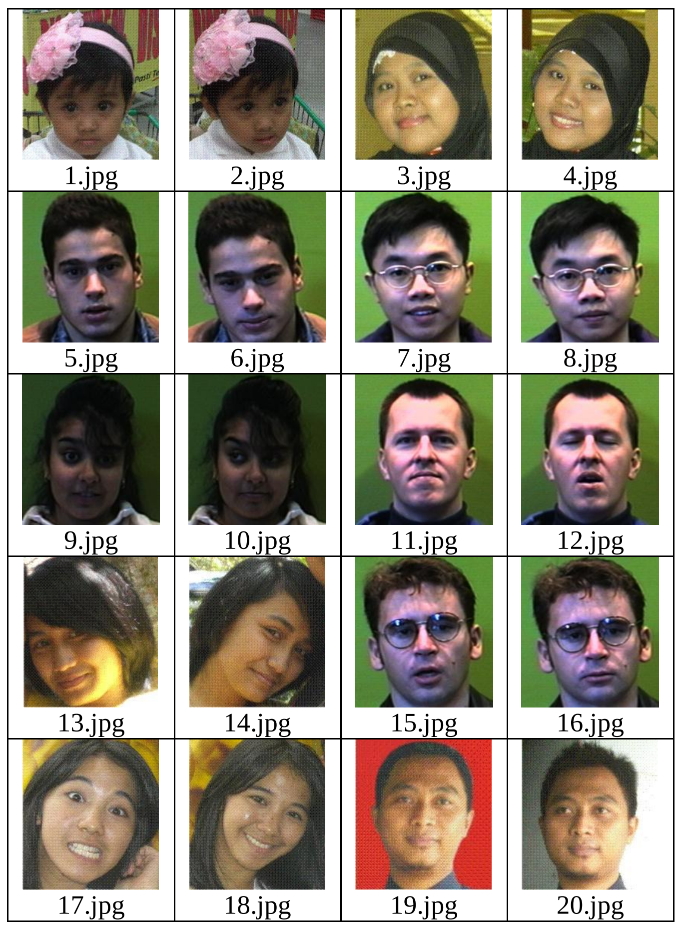

Figure 7. Image On Module Traindatabase

3.2 ANALYSIS OF NEED INPUT

Input for face recognition has been provided ten (10) face images of each person from the front with various poses as depicted in Figure 6.

The image will be used on the input training (learning input) is as much as 20 images and 10 images will be used for testing input (input test).

a. Input training (learning input) consists of 20 pieces of a picture or image of a person's face with color images.
b. Input testing (test input) selected 10 input images are not included (used) on the training input.

3.3 OUTPUT NEEDS ANALYSIS

The output generated from the application of face recognition using Euclidean distance function is the Euclidean distance function assessment on compatibility calculation training pattern testing patterns to recognize faces.

3.4 FUNCTIONS REQUIRED

Functions - the functions required by the system to be applied to the MatLab, among others:

a. Imread function to read the image / images.
b. Imresize function to convert lower resolution original image padad
c. Reshape function is used to change the dimensions of the matrix to the size of a one-dimensional
d. Mean function is used for counting the value - average.
e. Min function to find the minimum value for the final decision making process.

IV. TESTING SYSTEM

High percentage of low recognition PCA method depends on several variables as follows:
1. Number of images overall training
2. The number of training images contained in a single class
3. The number of features that are taken for the count

Class is meant here is the image-image training set which is owned by the same person. Each class is separated by person identity, in this case the name of the person concerned examples of the image shown in figure 7.

4.1 Folder allocation of Controlling Face

The first step is to determine the location where the sample was on the face of the equalizer traindatabase folder. Then click OK after a specified folder to go (TrainDatabase).

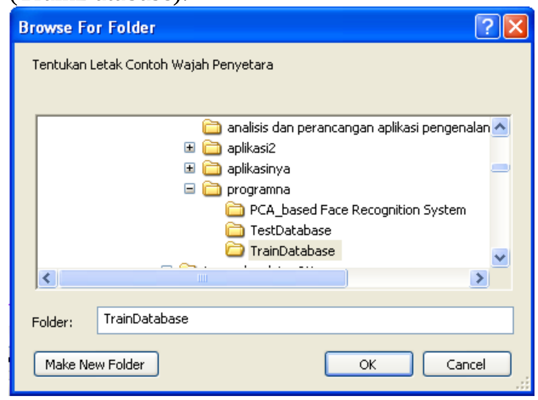

Figure 8. Determining the location of the folder TrainDatabase

4.2 Determining the Layout / Folder Sample Will Face in Test

The next step is to determine the location of the sample to be tested is the face on TestDatabase folder. Then click OK after a specified folder to go (TestDatabase)

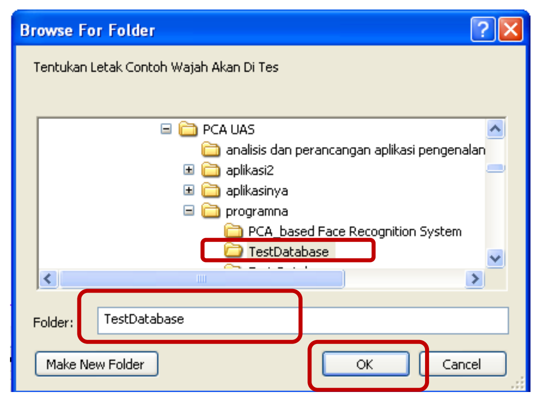

Figure 9. Determining the location of the folder TestDatabase

4.3 Determining the Sample File Name Which Will Face in Test

Next is to determine the name of the file to be on the test. Enter the file name to write down the serial number of examples of facial images and then click OK.

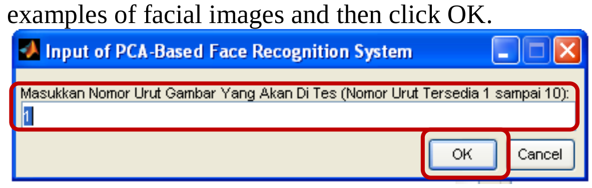

Figure 10. Determining Number Picture To The Test

After that it will get the following results:

Figure 1 (Figure A Tested) and Figure 2 (Figure Resemble)

From the above test results conducted on a face image 1.jpg in a folder and do equivalency approach TestDatabase percent similarity or proximity equalization up to 70% of the images and pictures 1.jpg 2.jpg TrainDatabase folder and the folder 1.jpg image obtained TrainDatabase.

1.jpg

Figure 12. Sample Image Facial In Test (TestDatabase)

1.jpg 2.jpg

Figure 13. Sample Image Face equalizer (TrainDatabase)

The second test performed on a file in the folder 9.jpg TestDatabase and done equivalency approaching equivalency percentage similarity or proximity to 70% of the image and the image 18.jpg 17.jpg TrainDatabase folder and the folder 1.jpg image obtained TrainDatabase.

9.jpg

Figure 14. Sample Image Facial In Test (TestDatabase)

17.jpg 18.jpg

Figure 15. Sample Image Face equalizer (TrainDatabase)

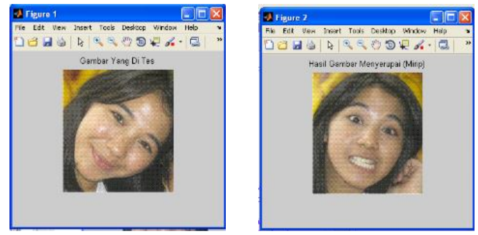

Figure 16. Test Results for Two

4.4 SOURCE PROGRAM

A. Source Code Example.m

```
clear all
clc
close all

TrainDatabasePath = uigetdir('D:\1 Bahan Kuliah
MKOM\Semester 2\Teknik Komputasi\PCA
UAS\programna\TrainDatabase', 'Tentukan Letak Contoh
Wajah Penyetara' );
TestDatabasePath = uigetdir('D:\1 Bahan Kuliah
MKOM\Semester 2\Teknik Komputasi\PCA
UAS\programna\TestDatabase', 'Tentukan Letak Contoh Wajah
Akan Di Tes');

prompt = {'Masukkan Nomor Urut Gambar Yang Akan Di Tes
(Nomor Urut Tersedia 1 sampai 10):'};
dlg_title = 'Input of PCA-Based Face Recognition System';
num_lines= 1;
def = {'1'};

TestImage  = inputdlg(prompt,dlg_title,num_lines,def);
TestImage = strcat(TestDatabasePath,'\',char(TestImage),'.jpg');
im = imread(TestImage);

T = CreateDatabase(TrainDatabasePath);
[m, A, Eigenfaces] = EigenfaceCore(T);
OutputName = Recognition(TestImage, m, A, Eigenfaces);

SelectedImage = strcat(TrainDatabasePath,'\',OutputName);
SelectedImage = imread(SelectedImage);

imshow(im)
title('Gambar Yang Di Tes');
figure,imshow(SelectedImage);
title('Hasil Gambar Menyerupai (Mirip)');

str = strcat('Matched image is :  ',OutputName);
disp(str)
```

B. Source Code EigenFaceCore.m

```
function [m, A, Eigenfaces] = EigenfaceCore(T)

m = mean(T,2);
Train_Number = size(T,2);

A = [];
for i = 1 : Train_Number
    temp = double(T(:,i)) - m;
    A = [A temp];
```

```
end

L = A'*A;
[V D] = eig(L);

L_eig_vec = [];
for i = 1 : size(V,2)
   if( D(i,i)>1 )
      L_eig_vec = [L_eig_vec V(:,i)];
   end
end

Eigenfaces = A * L_eig_vec;
```

C. Source Code CreateDatabase.m

```
function T = CreateDatabase(TrainDatabasePath)

TrainFiles = dir(TrainDatabasePath);
Train_Number = 0;

for i = 1:size(TrainFiles,1)
   if
not(strcmp(TrainFiles(i).name,'.')|strcmp(TrainFiles(i).name,'..')
|strcmp(TrainFiles(i).name,'Thumbs.db'))
      Train_Number = Train_Number + 1; % Number of all
images in the training database
   end
end

T = [];
for i = 1 : Train_Number

   str = int2str(i);
   str = strcat('\',str,'.jpg');
   str = strcat(TrainDatabasePath,str);

   img = imread(str);
   img = rgb2gray(img);

   [irow icol] = size(img);

   temp = reshape(img',irow*icol,1);
   T = [T temp];
end
```

D. Source Code Recognation.m

```
function OutputName = Recognition(TestImage, m, A,
Eigenfaces)

ProjectedImages = [];
Train_Number = size(Eigenfaces,2);
for i = 1 : Train_Number
   temp = Eigenfaces'*A(:,i);
   ProjectedImages = [ProjectedImages temp];
end

InputImage = imread(TestImage);
temp = InputImage(:,:,1);
```

```
[irow icol] = size(temp);
InImage = reshape(temp',irow*icol,1);
Difference = double(InImage)-m;
ProjectedTestImage = Eigenfaces'*Difference;

Euc_dist = [];
for i = 1 : Train_Number
   q = ProjectedImages(:,i);
   temp = ( norm( ProjectedTestImage - q ) )^2;
   Euc_dist = [Euc_dist temp];
end

[Euc_dist_min , Recognized_index] = min(Euc_dist);
OutputName = strcat(int2str(Recognized_index),'.jpg');
```

V. CONCLUSION

From the experimental results obtained in general when the training image variation is high (eg, illumination and expression), the use of PCA would provide a higher contribution. Even with a number of features that are slightly PCA gives better results. Taking the number of features that are calculated is included. If too little or too much will lower percentage recognition. The number of features that truly optimal can be obtained by conducting experiments repeatedly. From the experiments that have been conducted, the number of features that are optimal is between 30-50 for a database with 10 people, each training image 5, with a maximum of 78% recognition rate.

REFERENCES

Usman Ahmad, *Pengolahan Citra Digital dan Teknik Pemrogramannya*, penerbit Graha ilmu, Yogyaarta, 2005

PCA Based Face Recognition System, http://www.mathworks.com/matlabcentral/filee exchange/17032, Desember 2008.

Fauzan, Z. dan F. Taufiq, *Pengolahan Citra Menggunakan Delphi*, Edisi Pertama, Graha Ilmu, 2008.

Gunadi, K. dan Sonny R.P., *Pembuatan Perangkat Lunak Pengenalan Wajah MenggunakanPrincipal Component Analysis.*

Siang, J.J., *Jaringan Syaraf Tiruan dan Pemogramannya menggunakan Matlab*, ANDI, Yogyakarta, 2005.

R. Chellappa, C. Wilson, and S. Sirohey, "Human and machine recognition: A survey," *Proceedings of the IEEE*, vol. 83, no. 5, pp. 705--740, 1995.

M. Turk and A. Pentland, "Eigenfaces for recognition." *Journal of Cognitive Neuroscience*, Vol. 3, pp. 71-86, 1991.

K. Etemad and R. Chellappa, "Discriminant Analysis for Recognition of Human Face Images," *Journal of Optical Society of America A*, pp. 1724-1733, Aug. 1997.

P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection", *IEEE Trans. PAMI*, Vol. 19, No. 7, July 1997.