AVERAGE HALF FACE RECOGNITION BY ELASTIC BUNCH GRAPH MATCHING BASED ON DISTANCE MEASUREMENT

Vibha Sharma

Research Scholar

D.A.V.I.E.T

Jalandhar,India

Rajiv Vashisht

Assistant Professor

D.A.V.I.E.T

Jalandhar,India

Abstract

Average-half-face experiments the overall accuracy of the system is better than using the original full face image. Clearly experiment shows that half face data produces higher recognition accuracy [5]. The average-half-face contain the data exactly half of the full face and thus results in storage and computational time saving. The information stored in average-half-face may be more discriminatory for face identification, especially for 3D databases [6]. Accordingly this paper is a review on the use of average-half-face and we described a system for Average-half-face recognition based on the extraction of facial fudicial points such as head, nose and ear and measuring the Euclidean distance between these features using Elastic bunch Graph matching algorithm. In this facial fudicial features on the face are head, nose and ear which are described by set of wavelet (jets) components. Image graph is a bunch graph, which is constructed between the jets. Recognition is based on the Euclidean distance measurement using bunch graph. The distance is considered as a unique factor for the specific features for each person.

Keywords

Jets,   Average-half-face,   Canny,   Euclidean,   EBGM

1. Introduction

Ramanathan et al. [3] first discuss the use of 'Half-face' which is exactly one half of the full face is better to use for the problem of non-uniform illumination on faces. The use of one half of the face, selecting either left-half or right-half shows interesting results. The accuracy of the eigenfaces recognition when using only one half of the face (either the right or left half-face) is less than, or equivalent to, using full face. The accuracy has its origin in the averaging operation, which produces a new face that contains a set of features that are more discriminatory than those of full face [6]. Josh Harguess et al. [5] introduced the concept of average-half-face and experiments the accuracy performance with average-half-face is better than using original full face. josh Harguess and J.K Aggarwal [6] experiments average-half-face with 6 different algorithms applied to two- and three-dimensional (2D and 3D) databases. Out of 18 experiments, there are only two clear instances out of a total of 18 experiments that give evidence of the full face producing higher accuracy [6]. C.Gnanaprakasam et al. [8] utilizes average-half-face for the research experiments the accuracy with average-half-face using wavelets shows the substantial decrease in storage and computational time. The work presented by J.Harguess et al. [4] and

J.Harguess et al. [6] is a holistic approach attempts to identify faces using global representation, i.e. a description based on the entire image rather than on local features of the face. The limitations of the described methods are that these methods are computationally very expensive and suffer from the usual shortcomings of straightforward correlation based approaches, such as sensitivity to face occlusion, size, uneven illumination, clutter, noise and aging affect. We propose a method that utilizes the inherent symmetry of the human face for feature extraction that can be further utilized in face recognition. The inherent symmetry is utilized to create average-half-face from the full frontal face image. The average-half-face stores the data exactly half of the full face thus results in substantial decrease of storage and time. The information stored in the average-half-face contains more discriminatory feature than that of full face. We set our task to show the effect of using distance calculation between the extracted features on the average-half-face. The distance between the specific features is intrinsic property of the skin which do not affected by the impact of time as other face recognition system suffers such as a face recognition system that is based on skin segmentation. Under distance measurement our system is robust to change in facial expression. The measured distance is the intrinsic property of the facial surface including the Euclidean distance between the pairs of points on the face would not be altered when the facial expression change. They may be less affected by global changes in the appearance of the face range images such as change in expression, occlusion, hole and presence of noise [9]. The overall accuracy of our system is better than original full face image. Clearly, experiments show that average-half-face produces higher recognition accuracy. Thus our algorithm improves the overall accuracy of the face recognition system and is robust to the global changes in the appearance of the face.

2. AVERAGE-HALF-FACE

It has been observed that that human face is inherently symmetric and the symmetry of face can be exploited for face recognition. The average half face can be used for face recognition to produce better results. The average-half-face is constructed from the full frontal face image in two steps: First the face image is centered and divided in half and two halves are averaged together [6]. The face in the image is optimally centered so that the mean-squared error of the difference between the two sides of the face (with the columns of one of the sides of the face reversed) is minimum. For example, in a 2D face image with equal illumination on both sides of the face, this can be achieved with a simple search for the best center of the face. For 3D range images of faces, this can also be achieved with a simple search for the best center of the face, along with the knowledge that the tip of the nose is usually the closest point (to the sensor) and is therefore easily identifiable as a starting point for the search. It is important to note that this step of centering the face image is vital for full faces, average-half-faces and any other data that use the majority of the face for face recognition, and that, for 3D faces, correcting the orientation and centering the face image is simple. Further, this processing step is done off-line and does not adversely affect the computation time of the eigenfaces method with average-half-faces as compared to full faces. Second, the two halves of the face (the right and left half-faces) are averaged together. Note that the columns of the left half-face must be reversed so that the two half-faces are aligned before averaging [5]. As an example, Figure 1(a) displays a 3D face image from our database and Figure 1(b) displays its corresponding left face. Also in Figures 1(c) and 1(d), we display the right half-face and average-half-faces of the same 3D face image. The

consequence of using the average-half-face may result in substantial savings in storage and computation time the face recognition.

3. EFFECT Of USING EUCLIDEAN DISTANCE MEASUREMENT WITH EBGM

The perfomance of a 3D human face recognition algorithm that employ euclidean distance between facial features is better than stastical face recognition algorithms.using Euclidean distance , that is to measure the distance between facial features results into an algorithm that is robust to change in facial expression. A number of anthrometric facial proportions that are employed to characterize the shape of human face are based on-the-surface facial distances. A recent study suggests that it may be possible to model changes in facial expression as isometric deformations of the facial surface. Under distance measurement, the intrinsic properties of the facial surface including the Euclidcean distance between the pairs of points on the face would not be altered when the facial expression changes [9]. The use of distance measures makes the algorithm robust from the affect of global changes in the appearance of the face range images such as change in facial expression, occlusion, hole and presence of noise. Our 3D face recognition system that employ distances between the facial fudicial points as features performs significantly better than the statstical holistic 3D face recognition algorithm.

3.1. The EBGM Algorithm:

The Elastic Bunch Graph Matching (EBGM) relies on the concept that faces have non-linear characteristics which are not addressed by linear methods of analysis. A Gabor wavelet transform creates the architecture which projects the face onto an elastic grid. The Gabor 'jet' is a 'node' which describes the behavior around a given pixel. Recognition is based on the similarity of the Gabor filter's response at each Gabor node. In the elastic bunch graph matching algorithm represents the labeled graph. Edges are labeled with distance information and nodes are labeled with wavelets responses locally bundled in jets. Stored model graphs can be matched to new images to generate image graphs which can then be incorporated into a gallery and become model graphs. Wavelets as we use them are robust to moderate lighting changes and small shift deformations. Model graphs can be easily translated, scaled, oriented, or deformed during matching process thus compensating for a large part of the variance of the images. A bunch graph is created using EBGM. A bunch graph is created in two stages. Its qualitative structure as a graph (a set of nodes plus edges) as well as the assignment of corresponding labels (jets and distances) for one initial image is designer provided, whereas the bulk of the bunch graph is extracted semi-automatically from sample images by matching the embryonic bunch graph to them, less and less often intervening to correct incorrectly identified fudicial points. For the purpose of recognition, image graph can compared with model graphs at small computing cost by evaluating the mean jet similarity [2].
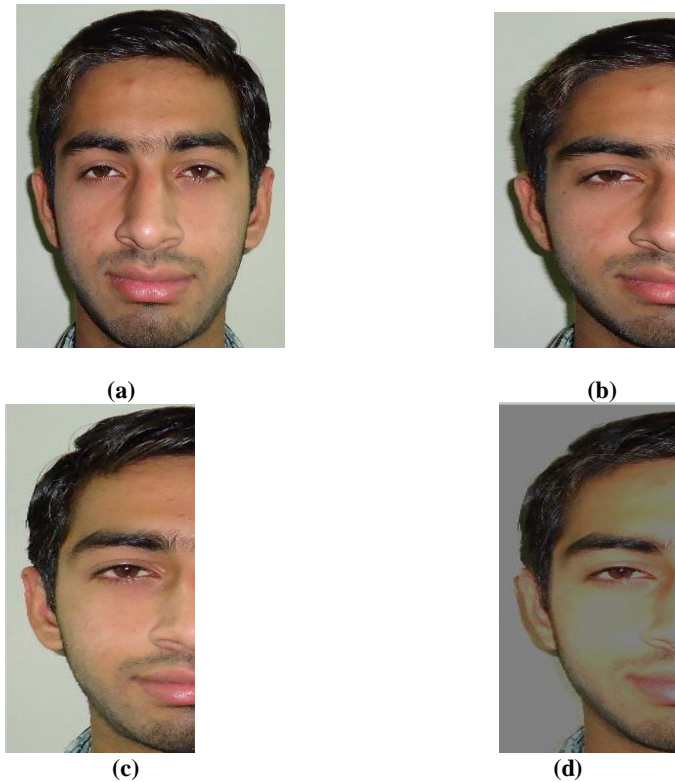
Figure 1. (a) 3D full-face image ; (b) its left half face; (c) its right half-face(d) its average-half-face

3.1.1. Gabor Wavelets:

Both the Wavelet and Fourier analysis are used to analyze frequency space properties of an image. The difference between the Wavelet and Fourier analysis is that wavelets operate on a localized image patch, while the Fourier transform operates over the entire image, respectively. Gabor wavelets are basically a sinusoid multiplied by a Gaussian. Thus, when a function is convolved with the Gabor wavelet, the frequency information near the center of the Gaussian is captured and frequency information far away from the center of the Gaussian has a negligible effect. One equation that specifies a one dimensional Gabor wavelet is

$$W(t,t_0,\omega) = e^{-\sigma(t-t_0)^2} \; e^{-t\omega(t-t_0)}$$

Convolution for the wavelet transform is defined as:

$$C(x(t)(t_o,\omega) = \int_{-\infty}^{\infty} x(t)W(t,t_0,\omega)dt$$

By expanding the wavelet function, the convolution looks a lot like the Fourier Transform

$$C(x(t))(t_o,\omega) = \int_{-\infty}^{\infty} x(t)e^{-\sigma(t-t_0)^2} \; e^{\omega(t-t_0)}$$

$$C(x(t))(t_o,\omega) = \int_{-\infty}^{\infty} x(t)e^{-\sigma(t-t_0)^2} \; \cos(\omega(t-t_0)) + i\int_{-\infty}^{\infty} x(t)e^{-\sigma(t-t_0)^2} \; \sin(\omega(t-t_0))$$

The EBGM algorithm uses a two dimensional form of Gabor wavelet for face reorganization. The wavelets used in EBGM have five parameters, control orientations, frequency, phase, size, and aspect ratio. In the EBGM algorithm, precomputed wavelet masks are used to perform the wavelet convolution. During convolution each mask is a two dimensional array that is used as a lookup table for wavelet values. The masks are centered over the correct location in the image, and each corresponding value is multiplied and added to the sum. To compute both the real and imaginary part of the wavelet, it is necessary to convolve the image with two masks that are out of phase by π/2, corresponding to the use of sine and cosine in the wavelet transform. In EBGM algorithm, cosine wavelets are thought to be a real part of wavelet and sine wavelets are thought to be the imaginary part of the wavelet [2].

3.1.1.1. Jets:

A jets describes a small patch of gray values in an image I(X) around a given pixel X=(x,y). It is based on a wavelet transformation. Gabor jets are a collection of complex Gabor coefficients from the same location in an image. The coefficients are generated using Gabor wavelets of a variety of different sizes, orientations, and frequencies. Gabor jets are based on 40 complex wavelets where each wavelet has a real and imaginary component. The Gabor jets use eight orientations, five frequencies, and two phases (real and imaginary). The Gabor jets are also referred as model jets [2].

3.1.1.2. Bunch Graph:

Bunch Graph is a data structure in which Model Jets are collected. The bunch graph has a node for every facial feature coordinate on face. The facial features used in this algorithm are tip of nose, ear and chin. The bunch graph serves as a database of facial features coordinates [2].

3.1.1.3. Face Graph:

A face graph is used to represent each image. the face graph nodes are placed at fudicial facial features such as nose, ear and chin and each node contain a Gabor jet extracted from that facial fudicial features. Figure 2 shows a wavelet convolution result for the average-half-face. The system proceeds with finding the distance between the nose and chin, chin and ear, ear and nose. The distance is measured in millimeters (mm). The distance can be calculated by using the Euclidean distance. The Euclidean Distance is probably the most widely used distance metric. The formula for Euclidean distance is:

$$\| x - y \|_e = \sqrt{| x_i - y_i |^2}$$

Euclidean distances for two dimensional images are: if p = (p1, p2) and q = (q1, q2) then the distance is given by

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2}$$

Euclidean distance for three dimensional images is:

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + (p_3 - q_3)^2}$$

3.1.1.4. Face Recognition

In this system the face can be recognized by using Elastic Bunch Graph Matching Algorithm. This algorithm is based on the jets which are made from the Gabor wavelets by applying convolution on it. The jets are different fudicial points like nose, ear and chin. Then bunch graph is created from the jets. The bunch graph is a database which contains information about different nodes. Then the face graph is created on for each image by extracting a jet for each fudicial point on the face. The last step of face recognition is calculating the distance between nose and ear, ear and chin, chin and nose. The face is recognized by using theses distances using EBGM [2].

3.1.1.4.1. Manual Generation of Graph

For the algorithm of facial recognition to recognize a facial image segment, it needs to match the image to a common set of marked facial images and find a similarity index between them. This process is basically divided into two sets and the first part is manual generation of face bunch graph for a person with a set of model graphs. Hence, initially a training set of facial images is taken, where each image is marked to a corresponding person. This set of image is then used to manually generate a face bunch graph by three basic steps discussed below. For the set of training images for a particular person, the fiducial points (ear, nose and chin) are automatically marked and the jet values of the Gabor wavelet transformed images of those points represent the node value of the fiducial points.

1. The edges are drawn between each pair of fiducial nodes already marked and the magnitude of each edge is the length of the edge, which is a calculated Euclidean distance between facial fudicial points.

2. These set of nodes are stacked and the set of edges between similar fiducial points are averaged to generate the face bunch graph corresponding to a particular person. This process is repeated for each individual who will form a model graph for recognition in the algorithm. In our analysis, for representative purpose we selected a set of 6 individuals and hence 6 face bunch graph is generated one for each of them. we choose out 3 fiducial points in the face [2].

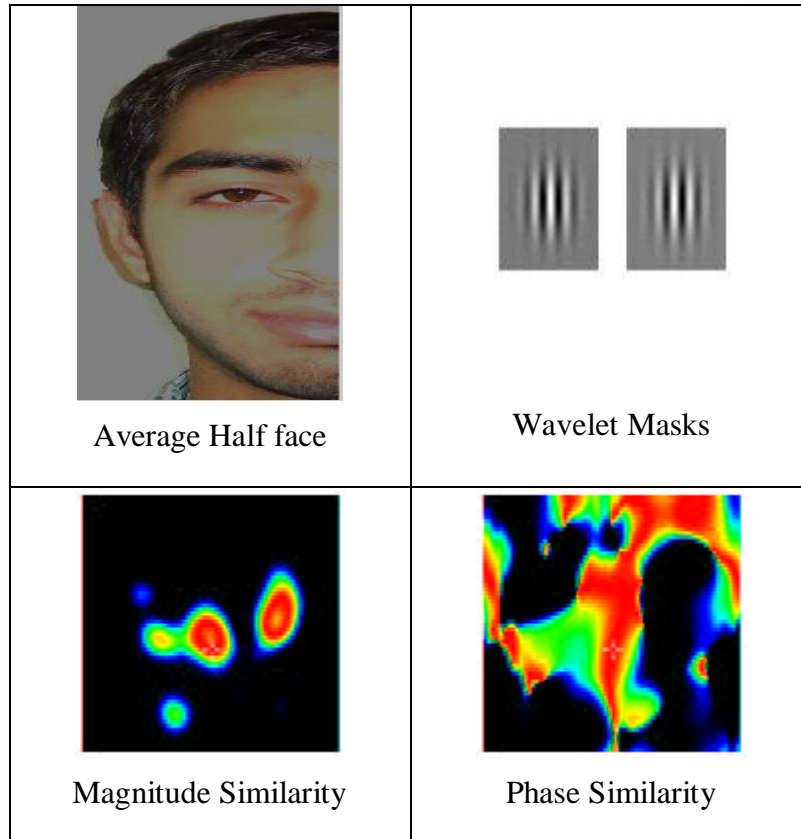|	
Average Half face	Wavelet Masks
Magnitude Similarity	Phase Similarity

Figure 2. Wavelet Convolution Results

3.1.1.4.2. Matching using Graph Similarity Function

After the model bunch graph is generated for the representative set of faces, the goal is to match each individual bunch graph with an input image graph and rate it using a similarity function that will analyze the degree of similarity or distortion between the facial structures represented by the graphs. The Elastic Bunch Graph Matching is responsible for e similarity between the input image graph and the face bunch g pose. Thus for an image graph G with nodes n=1, 2, 3 and edges e=1, 2, 3 matching is done between the corresponding parameters of the face bunch graph as:

$$S_B(G^I, \beta) = \frac{1}{N} \sum_N \max_m (\delta_\phi (J_n^I, J_n^{\beta m})) - \frac{\lambda}{E} \sum_e \frac{(\nabla x_e^{\rightarrow I} - \nabla x_e^{\rightarrow \beta})^2}{(\nabla x_e^{\rightarrow \beta})^2}$$

λ decides the relative importance between jets and metric structure at 1 for our analysis. This process is Elastic Bunch Graph Match pixel location of the fiducial points in the face bunch g the same pixel in the image graph is elastically extended in order to find a set of fiducial node points for which the similarity maximizes between the image graph and the face bunch graph.

In our analysis, the locality about a fiducial point is taken to be of a range (+/- 3, +/-3) pixels in x and y directions. Hence, taking the nodal location of the jets in the face bunch graph and Base and considering the locality restrictions, the similarity estimation is done for 9 pixel point per

node for the image graph, and the pixel point set where the similarity is maximum, is taken as the measure of similarity between the image graph and that particular face bunch graph [2].

4. ALGORITHM OUTLINE

4.1. Algorithm Description

Step 1: Read the image.

Step 2: Resize the image in particular Dimension

Step 3: Crop the image in which only facial areas have been extracted except background area.

Step 4: Divide the image into two halves i.e. left half and right half.

Step 5: Create the mirror image of the right half image.

Step 6: Create the average half image from left half & mirror image.

Step 7: Detect the edges from the average half face using Canny edge detection method.

Step 8: Automatic detection of head & plot the circle on the tip of head.

Step 9: Automatic detection of ear & plot the circle on the tip of ear.

Step 10: Automatic detection of nose & plot the circle on the tip of nose.

Step 11: Create a bunch graph on average-half-face between the pairs of point i.e. tip of head, tip of ear and tip of nose.

Step 12: Find the Euclidean distance between these three fudicial points.

Step 13: Identify the image from the data base.

5. EXPERIMENT

We applied the above algorithm to 3D image database, AR database and Yale database. The 3D Face database contains total of 60 frontal, 3D face images. There are total of 10 subjects with 6 images per subject representing change in illumination and facial expression. For experiment of algorithm, we maintained consistent use of database by forming the training data from the first 3 images per subject and using the remaining 3 images per subject for testing. The AR Face database consists of 18 frontal, 3D Face images. The Yale Face database consists of 12 grey scales, frontal 2D face images. The 3D Face database' range of images had different resolution along with different dpi and focal length. The face in the image is optimally centered so that the mean-squared error of the difference between the two sides of the face (with the columns of one of the sides of the face reversed) is minimum [5]. We divided the 3D dataset into three sets; training database, gallery and testing database. The Yale Face database's 2D range of images had vertical resolution of 96dpi and Horizontal resolution of 96dpi with focal length of 35mm.The face in the image is optimally centered so that the mean-squared error of the difference between

the two sides of the face (with the columns of one of the sides of the face reversed) is minimum [5]. The AR Face database's 3D range of images had a resolution of 96dpi and Horizontal resolution of 96dpi with focal length of 35mm.The face in the image is optimally centered so that the mean-squared error of the difference between the two sides of the face (with the columns of one of the sides of the face reversed) is minimum.

6. Results

The parameters considered for the research were recognition rate and computational time etc. A comparison is made on the performance of full face, average-half-face and average-half-face based on distance calculation between features. Table 1 show the comparison based on computational time. The experiment was validated on 3D, Yale and AR image face database. The face in the image is optimally centered so that the mean-squared error of the difference between the two sides of the face (with the columns of one of the sides of the face reversed) is minimum [5]. Table 1 shows the accuracy results for full face, average-half-face and average-half-face with distance calculation between features on the 3D, Yale and AR database. Table 2 shows a comparison based on different image database. Figure 3 shows accuracy comparison among different image databases. Figure 4 shows the output images that shows the face graph between features such as head, nose and ear. Figure 5 shows the calculated Euclidean distance between features such as head, nose and ear. The calculated distance was considered as a unique factor for the recognition of face.

Table 1. Parameter Comparison

Section	Computational Time (Sec)
Full Face	384
Average-half-face	125
Average-half-face DM	99

Table 2. Recognition Rate

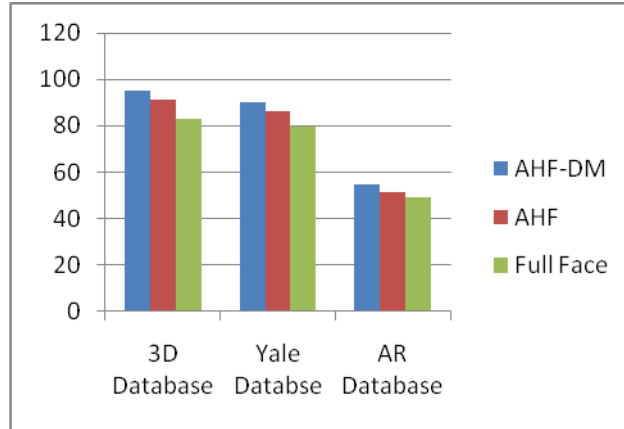| Database	A			B			C		
	3D			Yale			AR		
	Full	AHF	AHF-DM	Full	AHF	AHF-DM	Full	AHF	AHF-DM
Recognition Rate	83.0	91.5	95.3	79.7	86.2	89.9	49.4	51.7	55.0

Figure 3. Accuracy of AHF-DM, AHF and FF
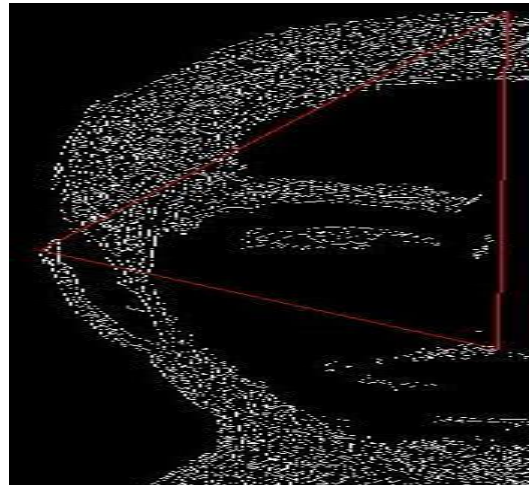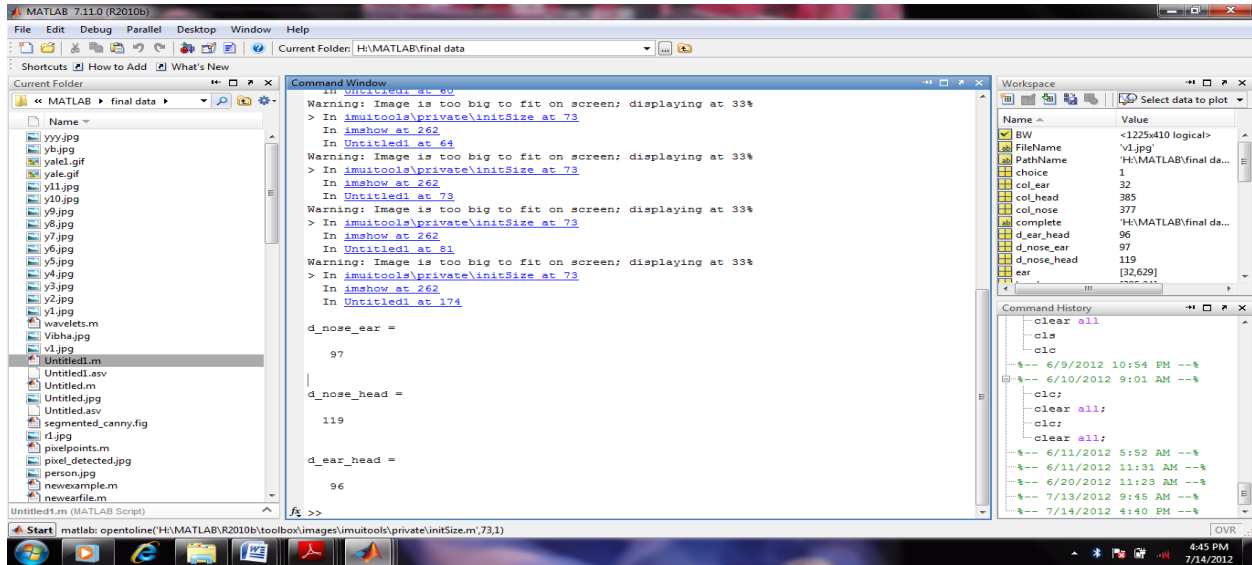

Figure 4. Output Image

Figure 5. Calculated Distances

6. CONCLUSION & FUTURE SCOPE

6.1. COCLUSION

This paper describes our implementation of Elastic Bunch Graph Matching algorithm used for 3D face recognition utilizing the average-half-face. The average-half-face produces better accuracy than using the original full faces. The face recognition system uses distance measurement between facial fudicial features by the implantation of EBGM that further enhances the accuracy of the face recognition system as compared to the statistical face recognition approaches as well as existing feature based approaches. A system that uses the distance measure for each image will perform better than existing systems that do not consider any distance parameter. It has shown that Distance measurement between the facial fudicial features provides the higher accuracy rate as compared to the AHF and full face. The system also results in the potential decrease in the storage space and computation time by the advent of average-half-face and use of distance measure between facial features. The EBGM face recognition algorithm uses the Gabor Wavelets transformation of the automatically detected facial fudicial features such as ear, nose and chin. Deep analyses of the source of the accuracy gain of the average-half-face. My approach increased the face recognition accuracy and more robust to effect of illumination, facial expressions, occlusions etc. Experimental results on a database of 109 range images showed that by using EBGM recognition rates as high as 94.17% are achieved.

6.2. Future Scope

The current system can be improved in many respects. The current system relies on the image database that is clicked from a particular distance which can be extended to clicking at any distance range. This would probably improve matching accuracy further and would provide more precise geometrical information which could be used to increase recognition performance. The work can be extended to the automatic detection of facial fudicial features such as eye, head and constructing the face graph between head, eye, nose.ear and chin for measuring the Euclidean

distance between these features for a more robust face recognition system. The system can further extended by the addition of distance classifiers. The different distance classifiers can be squared Euclidean distance, City-block distance, chebyshev distance etc. A system that uses different distance measures for each image will perform better than a system that only uses one. Face recognition can be applied in Security measure at Air Ports, Passport verification, Criminals list verification in police department, Visa Processing, Verification of Electoral identification and Card Security measure at ATM's. Face recognition has received substantial attention from researches in biometrics, pattern recognition field and computer vision communities. Thus the system can be combined with the biometric in future for a better face recognition system.

REFERENCES

[1] Laurenz Wiskott et al., (1999) "Face Recognition by Elastic Bunch Graph Matching" In Intelligent Biometric Techniques in Fingerprint and Face Recognition, eds. L.C. Jain et al., publ. CRC Press, ISBN 0-8493-2055-0, Chapter 11, pp. 355-396.

[2] D. S. Bolme, (2003) "Elastic bunch graph matching," Master's thesis, Colorado State University.

[3] N. Ramanathan, A. K. R. Chowdhury, and R. Chellappa, (2004) "Facial Similarity across Age, Disguise, Illumination and Pose", IEEE International Conference in Image Processing (ICIP), vol. 3, pp: 1999-2002, Singapore.

[4] Dong et at., (2007) "A half Face recognition Scheme", Eighth ACIS International Conference, pp. 355 – 358.

[5] Harguess et al., (2008) "3D Face Recognition with the Average-Half-Face", International Conference on Pattern Recognition ICPR, pp. 1-4.

[6] Josh Harguess and J.K. Aggarwal, (2009) "A Case for the Average-Half-Face in 2D and 3D for Face Recognition", In Proc IEEE Computer Society Workshop on Biometrics (in conjunction with CVPR), pp. 7-12.

[7] Raman Maini and Dr. Himanshu Aggarwal, (2009) "Study and Comparison of Various Image Edge Detection Techniques", International Journal of Image Processing (IJIP), Volume 3, Issue 1.

[8] C.Gnanaprakasam, S.Sumathi and R.RaniHema Malini, (2010) "Average-Half-Face in 2D and 3D Using Wavelets for Face Recognition", Proceedings of the 9th WSEAS international conference on Signal processing.

[9] Mini Singh Ahuja, Sumit Chhabra, (2011) "Effect of Distance Measures In PCA Based Face Recognition", International Journal of Enterprise Computing and Business Systems, Vol. 1 , 2.

[10] Hussein Rady, (2011) "Face Recognition using Principle Component Analysis with Different Distance Classifiers", IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.10.