MODIFICATION OF EXISTING FACE IMAGES BASED ON TEXTUAL DESCRIPTION THROUGH LOCAL GEOMETRICAL TRANSFORMATION

A thesis

submitted in partial fulfillment of the requirement for the Degree of

Master of Computer Science and Engineering

of

Jadavpur University

By

Mrinmoyi Pal

Registration Number: 129012 of 2014-15

Examination Roll Number: M4CSE1622

Class Roll Number: 001410502031

Under the Guidance of

Prof. Debotosh Bhattacharjee

Department of Computer Science and Engineering

Jadavpur University, Kolkata-700032

India

May 2016

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

CERTIFICATE OF RECOMMENDATION

This is to certify that the dissertation entitled "Modification of Existing Face Images Based on Textual Description through Local Geometrical Transformation" has been carried out by Mrinmoyi Pal (University Registration Number: 129012 of 2014-15, Examination Roll Number: M4CSE1622) under my guidance and supervision and be accepted in partial fulfillment of the requirement for the Degree of Master of Computer Science and Engineering. The research results presented in the thesis have not been included in any other paper submitted for the award of any degree in any other University or Institute.

……………………………………………………
Prof. Debotosh Bhattacharjee (Thesis Supervisor)
Department of Computer Science and Engineering
Jadavpur University, Kolkata-700032

Countersigned

……………………………………………………
Prof. Debesh Kumar Das
Head, Department of Computer Science and Engineering
Jadavpur University, Kolkata-700032

……………………………………………………
Prof. Sivaji Bandyopadhyay
Dean, Faculty of Engineering and Technology
Jadavpur University, Kolkata-700032

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

CERTIFICATE OF APPROVAL*

This is to certify that the thesis entitled "Modification of Existing Face Images Based on Textual Description through Local Geometrical Transformation" is a bona-fide record of work carried out by Mrinmoyi Pal in partial fulfillment of the requirements for the award of the degree of Master of Computer Science and Engineering in the Department of Computer Science and Engineering, Jadavpur University during the period of June 2015 to May 2016. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the thesis only for the purpose for which it has been submitted.

……………………………………………………………………………..
Signature of Examiner 1
Date:

……………………………………………………………………………..
Signature of Examiner 2
Date:

*Only in case the thesis is approved

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

Declaration of Originality and Compliance of Academic Ethics

I hereby declare that this thesis entitled "Modification of Existing Face Images Based on Textual Description through Local Geometrical Transformation" contains literature survey and original research work by the undersigned candidate, as part of her Degree of Master of Computer Science & Engineering.

All information has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name: Mrinmoyi Pal
Registration Number: 129012 of 2014-15
Examination Roll Number: M4CSE1622
Class Roll Number: 001410502031
Thesis Title: Modification of Existing Face Images Based on Textual Description through Local Geometrical Transformation

…..……………………………………..

Signature with Date

ACKNOWLEDGEMENT

I take this opportunity to express my deepest gratitude and appreciation to all those who have helped me directly or indirectly towards the successful completion of this thesis.

Foremost, I would like to express my sincere gratitude to my respected guide and teacher **Prof. Debotosh Bhattacharjee** (Professor, Department of Computer Science and Engineering, Jadavpur University) for his exemplary guidance, monitoring and constant encouragement throughout the course of this Project. I feel deeply honored that I got the opportunity to work under his guidance.

I would also wish to thank **Prof. Sivaji Bandyopadhyay** (Dean, Faculty of Engineering and Technology, Jadavpur University) and **Prof. Debesh Kumar Das** (Head of the Department of Computer Science and Engineering, Jadavpur University) for providing me all the facilities and for their support to the activities of this project.

During the last one year I had the pleasure to work in our departmental lab. I am grateful to all the members of this lab for their kind co-operation and help.

I would like to express my gratitude and indebtedness to my parents and all my family members for their unbreakable belief, constant encouragement, moral support and guidance.

Last, but not the least, I would like to thank all my classmates of Master of computer science and Engineering (MCSE) batch of 2014-2016, for their co-operation and support. Their wealth of experience has been a source of strength for me throughout the duration of my work.

……………………………………………..

Mrinmoyi Pal

Registration Number: 129012 of 2014-15

Examination Roll Number: M4CSE1622

Class Roll Number: 001410502031

Department of Computer Science and Engineering

Jadavpur University, Kolkata-700032

Contents

List of figures

CHAPTER: 1
INTRODUCTION

1.1. Background

Over the last several decades, face recognition has received notable importance as one of the most effective applications of image processing and analysis. The reason for this trend is its application in person identification and the access to viable technologies. Although several biometric technologies exist (e.g., voice-waves, retinal scan and iris scan), these techniques have not been welcomed by the common people till now. Face recognition has an advantage over other types of biometric recognition because of its non-contact connection, long-distance recognition, and user-friendliness. In particular, face detection has been applied to bio-information-based authentication technologies such as accurate face extraction for face recognition as well as some various areas for identifying criminals, automatic video surveillance, and many other security systems. In recent years increased the importance of modification of existing face images based on textual description has been observed. Modification of existing face images is important because the database may not always contain the picture of the person to be identified.

We describe a person regarding the characteristic features of important face components like eyes, eyebrows, nose and lips. The integration of these features provides the recognition of face as a whole. It is essential to modify these features based on the description given by an eyewitness. An alternative system is forensic sketch generation by interviewing a witness to gain textual descriptions of the suspect [1]. To generate more realistic forensic photo, we use existing images for modification based on textual description through geometrical local transformation. This approach needs a face adequate database which is used for local geometrical transformation. The face images generated after geometrical local transformation are more expressive than the photo sketches because the photo sketches may not be realistic. The difference between modalities is another problem for matching photos and sketches. Therefore, it is required to modify the facial features and generate the altered image.

1.2. Objective

The main objectives of this project work are the following:

1. Modification of facial features i.e. eyes, eyebrows, nose, and lip.
2. To change the existing image based on the description of height and width of these face components.
3. To generate the transformed image.

1.3. Challenges

There are many challenges in dealing with the modification of existing face images; here we list a few:

1. Human faces are naturally expressive, and appearance may vary. Variations in facial expressions can be a cause of confusion.
2. Lack of large scale supervised datasets is another problem.
3. Both speed and scalability of the system are of prime importance because results are expected in real-time.

1.4. Scope of the study

The work is tested on 30 male and female face images from ORL database. The study has been done using Matlab R2013a.

1.5 Organization of the Thesis

A literature review on the concept of transformation of facial features is presented in chapter 2. The methodology which was followed during this study is presented in Chapter 3. The results and discussions from this study are presented in Chapter 4. The conclusion of the results is presented in chapter 5. Chapter 6 contains references.

CHAPTER: 2
LITERATURE REVIEW

2.1. Overview

An exhaustive literature survey has been carried out on facial-feature transformation. Various researches and studies in this field started several decades back. Relevant literature to this research works is presented in this chapter. Based on this important review papers, scope, and objective of the present study are established.

This chapter gives an insight into the present state of knowledge about face image modification related issues by exploring the available literature.

2.2. Related works

In [1] a multiscale Markov Random Fields (MRF) model has been used for sketch synthesis from a given face photo and recognition. Only faces of the front view and neutral expression have been considered. In [2] 2D image warping has been used for synthesizing human facial expressions. Spatial displacement of face point has been obtained from test images. Methods of facial feature extraction have been explored in [3]. A new face has been constructed based on textual description with these face components. In [4] face images have been transformed along perceived dimensions. Prototype images have been used to define an axis of transformation. Nonlinear image shape transformation has been proposed in [5]. Bilinear, Quadratic, cubic, biquadratic and bicubic image transformation models have been considered in this paper. In [6] transformation between face images of the same person has been used. Content-based image retrieval has been utilized [6,7]. Image feature representation and extraction is the fundamental basis of content-based image retrieval. Text descriptions of the images have been handled in [7].

CHAPTER: 3
METHODOLOGY

3.1. Overview

At first, we consider the base image as the image which has the best similarity levels with the description. The modification will be based on-

Left Eye: Height, Width.

Right Eye: Height, Width.

Left Eyebrow: Height.

Right Eyebrow: Height.

Nose: Height, Width.

Lip: Height, Width.

The assessment has been performed on 30 male and female face images from ORL database. The modifications will be performed on these facial features i.e. eyes, eyebrows, nose, and lips. An eye witness is interviewed about the criminal's appearance, and the textual descriptions are noted. Based on the textual description we perform a local geometrical transformation on face images. We perform the following steps-

1. Preprocessing
2. Extraction of facial components from a face image
3. Store the facial components
4. Perform local geometrical transformation based on textual description
5. Reconstruct the face image with the transformed facial features

One possible integrated system architecture, as shown in Figure 3.1.

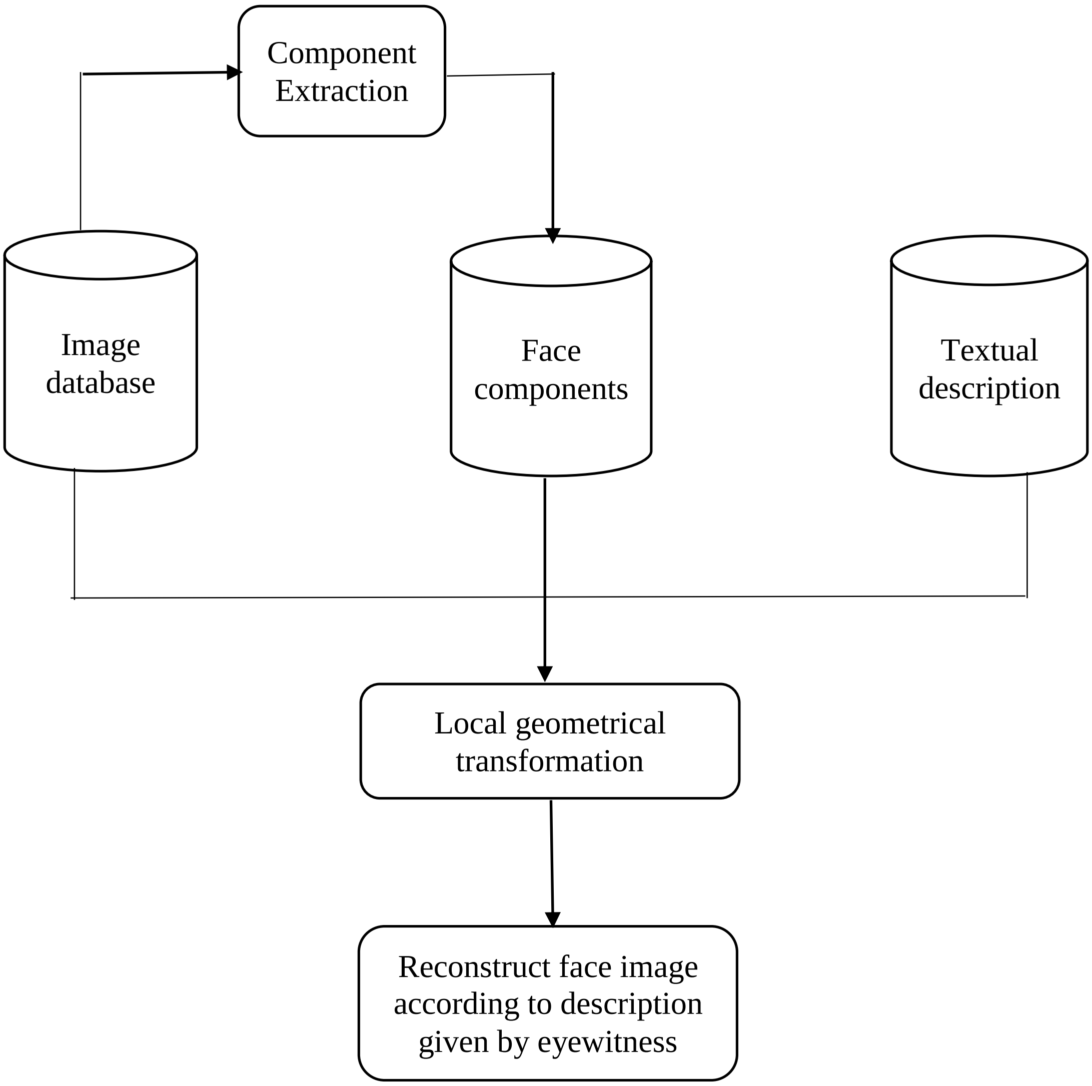


Figure 3.1. System architecture.

3.2 Preprocessing

Before extraction of the facial features, some preprocessing tasks on face photos are needed. The goal of preprocessing is to enhance the visual appearance of images. The faces are cropped from hair to chin and from right ear to left ear. In the preprocessing step, when photos are color, we first convert the RGB color space to the gray colorspace and all the faces are cropped with the dimension of 200 × 200 pixels. We may increase image contrast if needed. If required, noise removal is also performed.

3.2.1. Conversion to grayscale image

In a color image represented in RGB model, each color appears in its primary spectral components of red, green and blue. But for a grayscale image color varies in different shades of gray with the lightest being white and the darkest being black. For a grayscale image with 8-bit pixel intensity, pixel values may vary from 0 to 255 where 0 being black and 255, white. Grayscale transformations do not depend on the position of the pixel in the image. According to the proposed theory of International Journal of Computer Applications (0975 – 8887) Volume 7– No.2, September 2010 the following equation performs a very efficient RGB to grayscale conversion where each pixel intensity for grayscale from its corresponding RGB value can be calculated by Iy=0.333Fr+0.5Fg+0.1666Fb where Fr, Fg, Fb are the red, green and blue components.

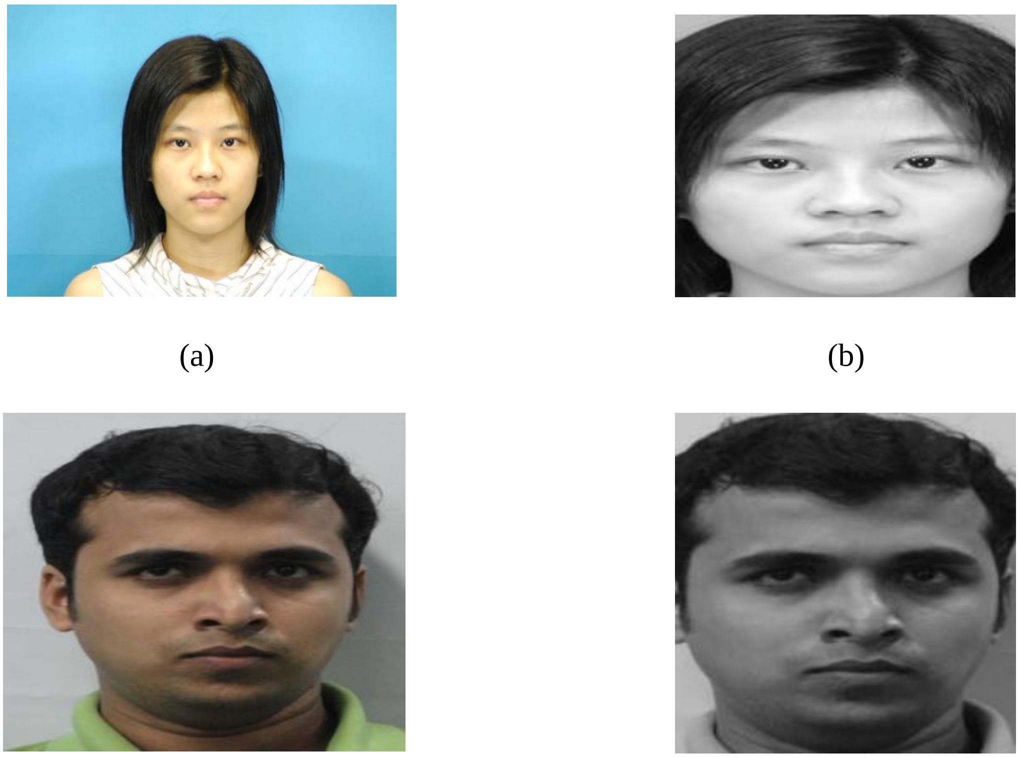
3.2.2. Resizing images

Resizing to a predefined size is very important. As our system has to deal with different images of different shapes and sizes, results will vary according to the scale. So images taken as input are resized to a predefined size. Images can be resized using several methods, e.g. nearest-neighbor interpolation, bilinear interpolation, bicubic interpolation, Fourier-based interpolation, edge-directed interpolation, vectorization. In this study, Bilinear Interpolation has been used to resize. We resize the image to a new width and height of 200*200 pixels. The output pixel value is a weighted average of pixels in the nearest 2-by-2 neighborhood.

3.2.3 Cropping Images

Cropping refers to removing unwanted areas and distracting elements from an image. Cropping an image extracts a region of interest from the original image. For cropping a portion of an image, we can select the region to be cropped by using the mouse. Now from the mouse event, we get the upper left corner and lower right corner of the selected portion and copy the intensity values of the selected region in a separate matrix as a new image.

Figure 3.2 shows some face images and their gray cropped images.



(a) (b)

(c) (d)

Figure 3.2 Original face images (shown in left) followed by gray cropped images (shown in right)

3.3. Extraction Methodology

To transform a face image, extraction of a facial component from images is required. The different seven components are:

A. Eyebrows (Left and Right)

B. Eyes (Left and Right)

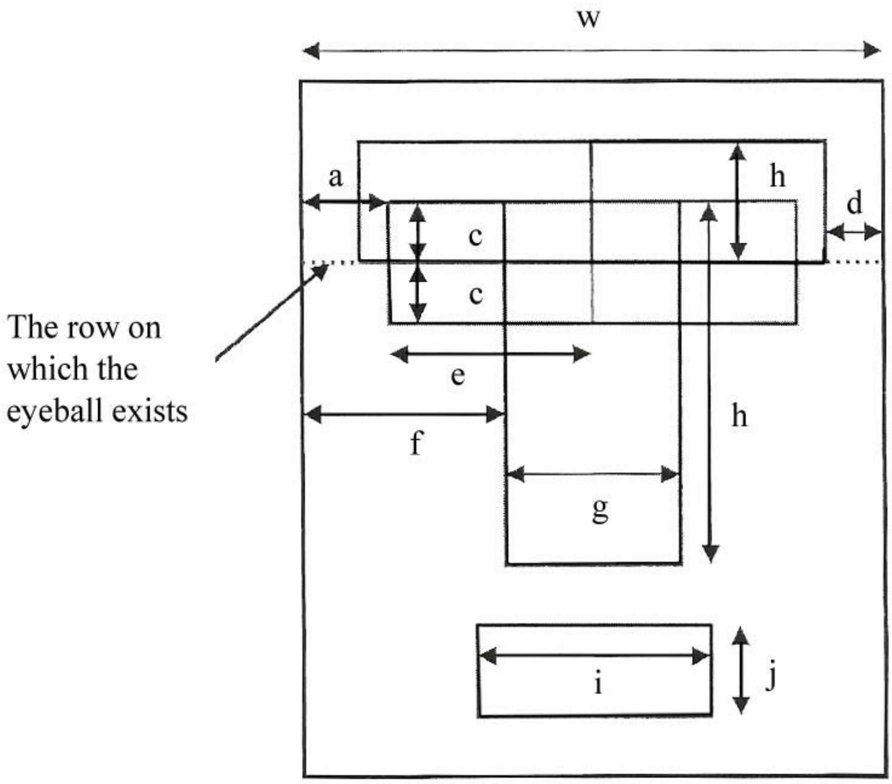
C. Nose

D. Lip

E. Face cutting

It is not possible to extract all the facial components with one algorithm. Therefore, we have used a separate algorithm for each facial component. These algorithms have been designed in [3].

The first and the most important step in facial component detection is to track the position of the components. Figure 3.3 shows a geometrical model of a face image to predict the region of interest or the approximate positions where the facial components may appear, and then the actual regions of the facial components are extracted by applying the proper algorithms over the area around the predicted regions.



a = 0.14W, b = 0.15W, c = 0.06, d = 0.05W, e = 0.36W, f = 0.32W, g = 0.40W, h = 0.55W, i = 0.50W, j = 0.25W

Figure 3.3 Geometrical model of face

Here we have considered two points (x1, y1) and (x2, y2) as the coordinates of the top left and bottom right corner of the predicted rectangular region for each facial components, respectively. All the calculations for the predicted regions in the geometrical model are shown on the width W of the face, where W is defined regarding the number of pixels.

3.3.1. Right eye extraction

It is assumed that the right eye lies in the right half of the face. Within the face region, it is assumed that maximum difference of intensity values between two adjacent pixels occurs in the middle of the eye. Therefore, if we get the row on which maximum difference occurs then we can predict the eye region. The steps to extract the right eye are as follows:

1. Extract the right half of the face.
2. Find the row (called Max_Index) on which the maximum difference between two adjacent intensities lies. The eyeball exists on the row Max_Index.
3. Now extract the portion of the image with the (x1, y1) and (x2, y2) values according to the geometric model shown in Fig. 3 where

   x1= Max_Index – (0.06 * W)

   y1= 0.14 * W

   x2= Max_Index + (0.06 * W)

   y2= W/2
4. Apply median filter.
5. Normalize by adding an integer N (N=127) with each pixel intensity values and get a new matrix.
6. Find edges by Canny's edge detection algorithm which yield a new matrix.
7. Using a boundary detection algorithm, get the actual right eye region.

Figure 3.44 shows the extraction of the actual right eye region of Fig. 2(b) from the predicted region of the right eye.

(a) (b)

Figure 3.4 (a) Predicted right eye region. (b) Extracted right eye region after applying the method

3.3.2. Left eye extraction

Left eye extraction method is similar to the right eye extraction method. The calculation for the prediction of the left eye region according to the geometrical model shown in Fig. 3 is:

x1= Max_Index – (0.06 * W)

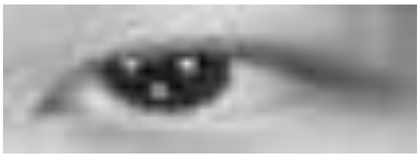
y1= W/2

x2= Max_Index + (0.06 * W)

y2= W – (0.05 * W)

Figure 3.5 shows the extraction of the actual left eye region of Fig. 2(b) from the predicted region of the left eye.

(a)



(b)

Figure 3.5 (a) Predicted left eye region. (b) Extracted left eye region after applying the method

3.3.3. Nose extraction

For extraction of the nose, we need to predict the nose region by using the geometric model shown in Fig.3 Then by finding the nostrils position, the left, right, and lower boundaries of the nose can be obtained. The upper boundary of the nose is calculated from its binary image. The steps to extract the nose are as follows:

1. Predict the nose region with the (x1, y1) and (x2, y2) values according to the geometric model shown in Fig. 3 where

   x1= Max_Index – (0.15 * W)

   y1= 0.32 * W

   x2= x1 + (0.55 * W)

   y2= y1 + (0.04 * W)

2. Get the upper boundary of the nose by discarding the upper portion from the binary image of it where no black pixel exists.

3. Extract the lower one-third portion of the Nose because the nostrils and the left and right boundaries of the nose can be obtained from this region.

4. Normalize by adding an integer N (N=127) with each pixel intensity values of it and get a new matrix.

5. Find the row (called Row_Nostrils) on which nostrils exist (by getting the minimum intensity)

6. Find the edges using Canny's edge detection algorithm and get the logical matrix.

7. Get the left, right and lower boundaries of the nose.

Figure 3.6 shows the extraction of actual nose region of Fig. 2(b) from the predicted region of the nose.

(a)

(b)

Figure 3.6 (a) Predicted nose region. (b) Extracted nose region after applying the method

3.3.4. Right eyebrow extraction

For extraction of eyebrow, we need to predict the eyebrow region by using the geometric model shown in fig 3. Extraction of eyebrow is not accurate in RGB but the extraction gives an accurate result in HSV scale. The steps to extract the right eyebrow are as follows:

1. Predict the right eyebrow region with the help of the (x1, y1) and (x2, y2) values according to the geometric model shown in Fig. 3 where

x1= Max_Index – (0.15 * W)

y1= 0.05 * W

x2= Max_Index

y2= W/2

2. Convert from RGB scale to HSV scale.

3. The following steps are performed to convert the V values to a binary scale:

   a. Let P = 0.25 Q = 0.25

   b. if |V – P| > Q, then T = 0 else T = 1

   c. if T = 1, then F = 1 – ((V – P) / Q*Q) else F = 0

   d. if F < 0.5, then C = 1 else C = 0

4. Copy the intensity value where the value of C is 1. The new matrix is the extracted right eye-brow in RGB.

Figure 3.7 shows the extracted right eyebrow of Figure3. 2(b) in RGB scale.

Figure 3.7 Extracted right eyebrow in RGB scale

3.3.5. Left eyebrow extraction

The extraction method of left eyebrow is same as the extraction of right eyebrow. However, prediction of the left eyebrow is different. The predicted region of the left eyebrow according to the geometric model of Fig. 3 can be described as:

x1= Max_Index – (0.15 * W)

y1= W/2

x2= Max_Index

y2= W – (0.05 * W)

Figure 3.8 shows the extracted left eyebrow of Fig. 2(b) in RGB scale.

Figure 3.8 Extracted left eyebrow in RGB scale

3.3.6. Lip extraction

Like the previous methods, here also the prediction of the lip region is made using the geometric model shown in Fig. 3. After prediction, the actual region of the lip is extracted from the predicted lip region. The steps to extract the lip are as follows:

1. Predict the lip region with the (x1, y1) and (x2, y2) values according to the geometric model shown in Fig. 3 where

   x1= NoseLx + (0.05 * W)

   y1= 0.25 * W

   x2= NoseLx + (0.30 * W)

   y2= y1 + (0.50 * W)

2. Apply median filter on gray image of Lip
3. Normalize Lip by adding an integer N (N = 127) with each pixel intensity values of Lip and get a new matrix.
4. Using a boundary detection algorithm, find the boundary of the Lip.

Figure 3.9 shows the extraction of actual lip region of Fig. 2(b) from the predicted region of the nose.

(a) (b)

Figure 3.9 (a) Predicted lip region. (b) Extracted lip region after applying the method

3.3.7. Face cutting extraction

To obtain a blank face from a complete face image, we replace all the facial components by its skin color. Figure 3.10 shows such an example.

(a) (b)

Figure 3.10 (a) A complete face image. (b) Blank face after replacing the facial components by the skin color.

3.4. Face component modification

In this phase, the geometric transformation of the facial features is performed. Here we analyze each facial component to determine the size of it regarding its height and weight, which are easily computable from the upper left and bottom right coordinates of its boundary box.

We have implemented the modification in two ways:

1. Scaling and translation of the features.
2. Image warping using 2-D affine geometric transformation and then scaling.

3.4.1. Scaling and translation

Image scaling changes the size of the face component by multiplying the coordinates of the points by scaling factors. It is the process of resizing an image. The scale operator performs a geometric transformation which can be used to shrink the size of an image. Image scaling can also be interpreted as image resampling or image reconstruction. Scaling compresses or expands an image along the coordinate directions. An image can be zoomed either through pixel replication or interpolation. Bilinear interpolation algorithm has been used to perform scaling procedure. The face components are resized to a new width and height.

After scaling, translation of the facial components is performed. Image translation moves a point to a new location by adding translation amounts to the coordinates of the point. To translate a facial feature, we translate every point of the feature by the same amount to the same direction.

3.4.1.1. Algorithm

Algorithm 1 depicts the steps for scaling, translation, and reconstruction of face image:

Algorithm Construct_Image1 (I, F)

{I is intensity matrix of a face component with order m*n and F is the intensity matrix of facial cutting.}

Begin

    Change the height and width of I. New height=H, new width=W.

    Detect the upper right corner (x,y) of I in F.

    r=x {r and c denotes the row and column of F.}

    For each row i of the matrix I (i=1 to H)

        r=r+1

        c=y

        For each column j of the matrix I (j=1 to W)

            F=I(i,j)

c=c+1

end for

end for

end [End of Algorithm]

3.4.2. Image warping using 2-D affine geometric transformation

An affine transformation is an important class of linear 2-D geometric transformations which maps variables (e.g. pixel intensity values located at position (x1,y1) in an input image) into new variables (e.g. (x2,y2) in an output image) by applying a linear combination of translation, rotation, scaling and/or shearing operations. It preserves parallelism of lines but does not preserve lengths and angles. If X and Y are affine spaces, then every affine transformation $f: X \longrightarrow Y$ is of the form $x \longrightarrow Mx + b$, where M is a linear transformation on X and b is a vector in Y. If an affine transformation is not a pure translation it keeps some point fixed, and that point can be chosen as the origin to make the transformation linear. Under certain assumptions, affine transformations can be used to approximate the effects of perspective projection [8]. A reflection that does not go through the origin is an affine transformation. All ordinary linear transformations are included in the set of affine transformations, and can be described as a simplified form of affine transformations. When using affine transformations, the homogeneous component of a coordinate vector will never be altered. Affine mapping coefficients can be uniquely determined from displacements of 3 vertices. Using an augmented matrix and an augmented vector, it is possible to represent both the translation and the linear map using a single matrix multiplication. The augmented matrix is called affine transformation matrix.

Image Warping is the process of manipulating an image such that any shapes portrayed in the image have been significantly distorted. While an image can be transformed in various ways, pure warping means that points are mapped to points without changing the colors. Warping transforms the face component according to the geometric transformation object defined by an affine transformation. We define the warp as a vector field that describes how each point in the source image should be translated to produce an output image.

3.4.2.1. Theoretical background

The affine transformation is any transformation that can be expressed in the form of a matrix multiplication (linear transformation) followed by a vector addition (translation). We can use an Affine Transformation to express: rotations (linear transformation), translations (vector addition) and scale operations (linear transformation).

We represent the affine transform by using a 3*3 matrix. If we want to transform a 2D vector [x y] by using A and B, we can do it equivalently with:

$$T = A \cdot \begin{bmatrix} x \\ y \end{bmatrix} + B$$

Where A is a 2*2 matrix and B is a 2*1 matrix.

Image warping is the act of distorting a source image into a destination image according to a mapping between source space (u,v) and destination space (x,y). The mapping is usually specified by the functions x(u,v) and y(u,v). There are two basic components to an image warp: spatial transformation and resampling through interpolation.

Image warping is used in image processing primarily for the correction of geometric distortions introduced by imperfect imaging systems [9]. In image processing, we do image warping typically to remove the distortions from an image, but we are introducing one in this work. Image warping is also used for artistic purposes and special effects in interactive paint programs. For image processing applications, the mapping may be derived given a model of the geometric distortions of a system, but more typically the mapping is inferred from a set of corresponding points in the source and destination images.

3.4.2.2. Equations

In homogeneous notation, 2-D points are represented by 3-vectors. For affine mappings, we denote points in source space by $P_s = (u,v,q)$ and points in the destination space by $P_d = (x,y,w)$.

A general 2-D affine mapping may be written algebraically as:

$$P_d = P_s M_{sd}$$

$$(x\ y\ 1) = (u\ v\ 1)\begin{pmatrix} a & d & 0 \\ b & e & 0 \\ c & f & 1 \end{pmatrix}$$

Transformation matrices are denoted M_{ab} where a is the initial coordinate system and b is the final coordinate system. We have chosen w = q = 1 without loss of generality. The 3*3 matrix M_{sd} has 6 degrees of freedom. Geometrically, the vectors (a,d) and (b,e) are basis vectors of the destination space, and (c,f) is the origin.

Since an affine mapping has 6 degrees of freedom, it may be defined geometrically by specifying the source and destination coordinates of three points.

3.4.2.3. Algorithm

Algorithm 2 depicts the steps for image warping implementation:

```
for (int u = 0; u < umax; u++)
{
        for (int v = 0; v < vmax; v++)
        {
                float x = fx(u,v);
                float y = fy(u,v);
                dst(x,y) = src(u,v);
        }
}
```

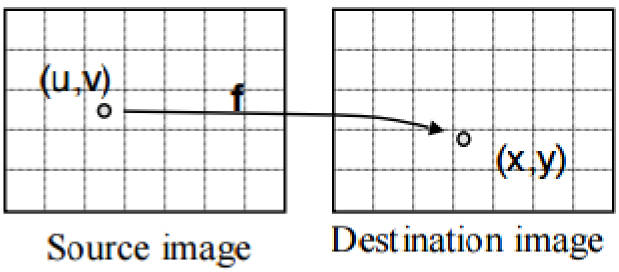


Figure 3.11 Image warping implementation

Algorithm 3 depicts the steps for warping using 2-D affine transformation and reconstruction of face image:

Algorithm Construct_Image2 (I, F)

{I is intensity matrix of a face component with order m*n and F is the intensity matrix of facial cutting.}

Begin

Create a geometric transformation object tform.

Transform I according to tform.

Change the height and width of I. New height=H, new width=W.

Detect the upper right corner (x,y) of I in F.

r=x {r and c denotes the row and column of F.}

For each row i of the matrix I (i=1 to H)

r=r+1

c=y

For each column j of the matrix I (j=1 to W)

F=I(i,j)

c=c+1

end for

end for

end [End of Algorithm]

3.5. Smoothing after transformation

When the modified face components are overlapped on the face cutting, there may generate dissimilarities between overlapping pixels. To make the overall face natural to the user, smoothing should be used. We can apply image stitching. Image stitching is a process of combining different images to form one single image [13]. The image stitching process can be divided into three main components – image calibration, registration and blending [14]. Image calibration provides a pixel-to-real-distance conversion factor (i.e. the calibration factor, pixels/cm), that allows image scaling to metric units. Image registration involves matching features [6] in a set of images or using direct alignment methods to search for image alignments. Image blending involves executing the adjustments figured out in the calibration stage, combined with remapping of the images to an output projection.

Alignment may be necessary to transform an image to match the viewpoint of the image it is being composted with. Alignment in simple terms is a change in the coordinates system so that it adopts a new coordinate system which outputs image matching the required viewpoint.

For image stitching, we must first choose a final compositing surface onto which to warp and place all of the aligned images [6]. To facilitate working with images at different resolutions, we may adopt a variant of the normalized device coordinates [11, 12].

There are two main approaches for image stitching.

1. Direct techniques: to shift or warp the images about each other and to look at how much the pixels agree [10].

2. Feature-based techniques: to find all corresponding feature points in an image pair by comparing all features in one image against all features in the other using one of the local descriptors [15].

The issue with image stitching is that in the real life case the intensity varies across the whole image and so does the contrast and intensity across the frames.

4.1 Overview

In this section images of faces after geometrical transformation of features are presented. The amount of changes in facial features is also projected in this section.

In this study, a total of 30 images was considered. Among all the images, 23 images are of male and seven are of the female.

4.2 Study on modifications due to Scaling and translation

In this report, the following modifications of features are considered for a face image: (i) eye width change, (ii) eyebrow width change, (iii) nose height increase and width change, (iv) lip height and width change. The effects of glasses are not considered for this study.

4.2.1 Results of modifications due to Scaling and translation

Figure 4.1 and 4.2 presents the changes of the facial features of figure 3.10(a) and figure 3.2(b) respectively based on the textual description.

(a) (b)

Figure 4.1 (a) Original face image. (b) Constructed face image after performing changes.

(a) (b)

Figure 4.2 (a) Original face image. (b) Constructed face image after performing changes.

4.2.1.1 Amount of changes

Amount of changes in Figure 11 is as follows:

The width of the eye has been decreased by 20%. The width of eyebrow has been reduced by 10%. The height of nose has been increased by 20%, and width has been decreased by 53%. The height of lip has been decreased by 42%, and 52% has reduced width.

Amount of changes in Figure 12 is as follows:

The width of the eye has been decreased by 19%. The width of eyebrow has been increased by 50%. The height of nose has been increased by 28%, and width has been decreased by 2%. The height of lip has been decreased by 3%, and width has been increased by 30%.

4.3. Results of modifications due to Image warping using 2-D affine geometric transformation

Figure-13 shows the interface of modifications due to Image warping using 2-D affine geometric transformation. There are six buttons depicting the changes, one button for viewing original image and one list box for selecting an image. We have created a 2-D affine transformation object which is a 3*3 matrix. This matrix defines the 2-D forward affine transformation. The matrix used is as follows:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 20 & 20 & 1 \end{bmatrix}$$

After creating the geometric transformation object, we transform the image of face component according to that object. We generate six modified images.

In first modified image, the changes are as follows:

EYE: Width decrease 30% EYEBROW: Height increase 214% NOSE: Height decrease 33% Width decrease 18% LIP: Height decrease 35% Width decrease 47%

In second modified image, the changes are as follows:

EYE: Width decrease 10% EYEBROW: Height increase 257% NOSE: Height decrease 25% Width decrease 18% LIP: Height decrease 35% Width decrease 43%

In third modified image, the changes are as follows:

EYE: no change EYEBROW: Height increase 257% NOSE: Height decrease 25% Width decrease 10% LIP: Height decrease 35% Width decrease 39%

In fourth modified image, the changes are as follows:

EYE: Height increase 25% EYEBROW: Height increase 257% NOSE: Height decrease 25% Width decrease 2% LIP: Height decrease 35% Width decrease 45%

In fifth modified image, the changes are as follows:

EYE: Height increase 25% EYEBROW: Height increase 257% NOSE: Height decrease 17% Width decrease 2% LIP: Height decrease 35% Width decrease 41%

In sixth modified image, the changes are as follows:

EYE: Height increase 25% EYEBROW: Height increase 257% NOSE: Height decrease 8% Width decrease 2% LIP: Height decrease 35% Width decrease 37%

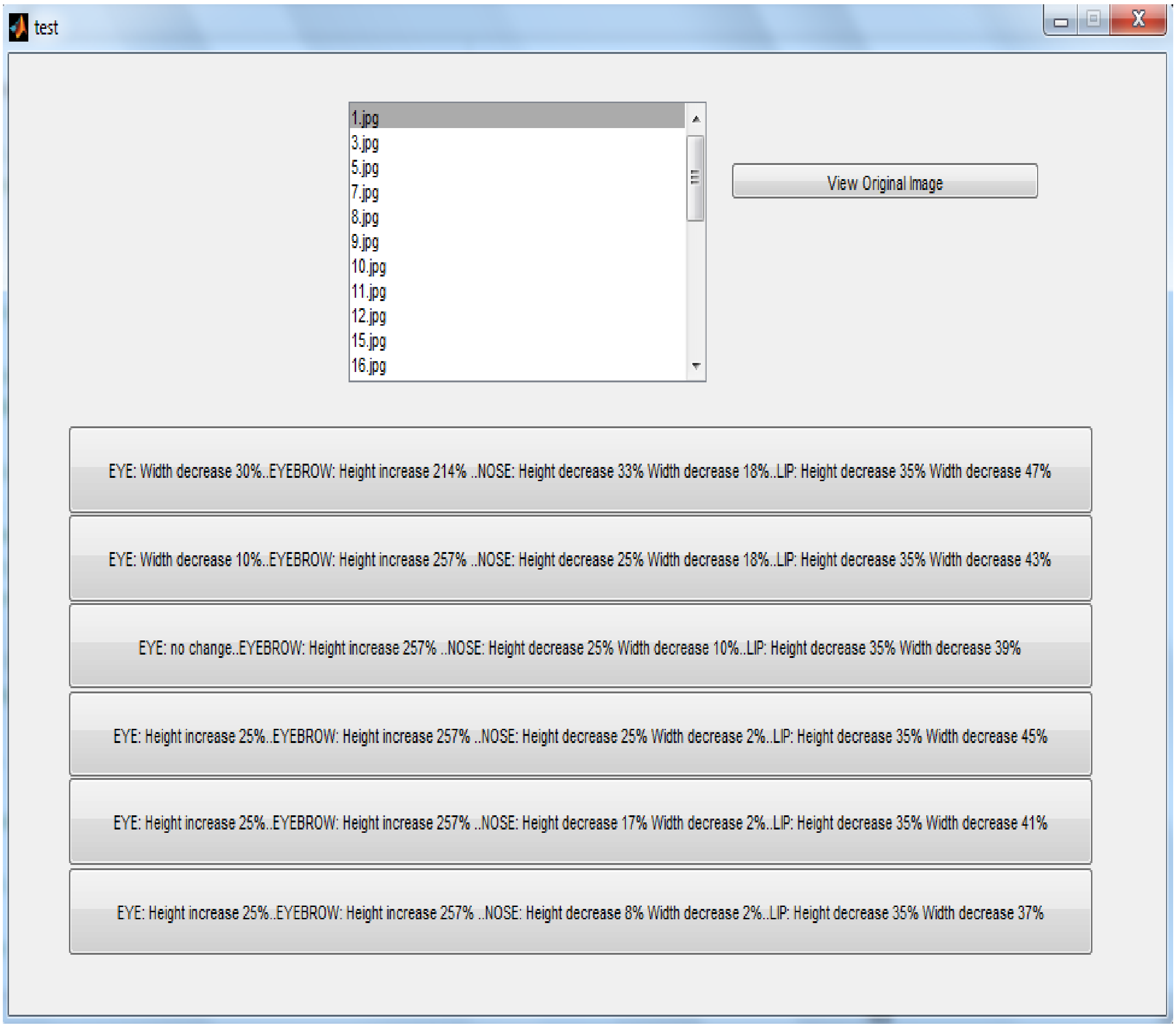


Figure 4.3. User interface of modifications due to Image warping using 2-D affine geometric transformation

Figure 4.4 presented the snapshot of the system when the button labeled as view original image is clicked.

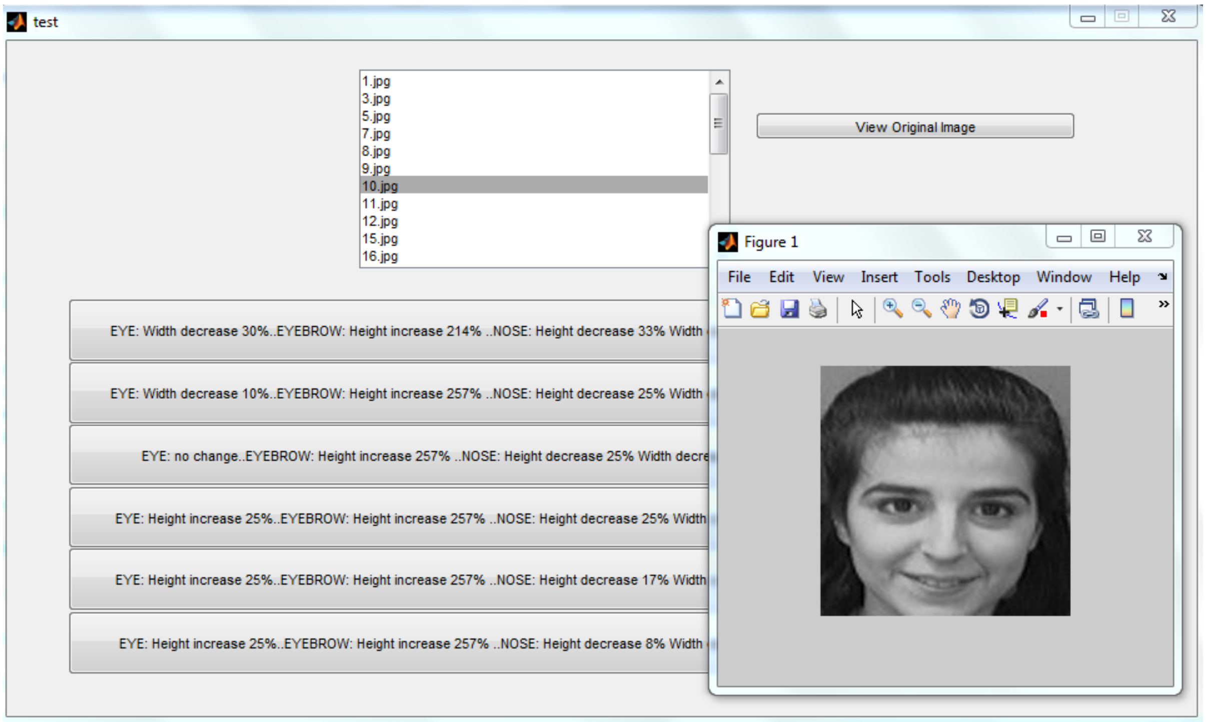


Figure 4.4. Snapshot of the system when View Original Image button is clicked

Figure 4.5 shows the modified images after image warping.

(a) EYE: Width decrease 30%
EYEBROW: Height increase 214%
NOSE: Height decrease 33% Width decrease 18%
LIP: Height decrease 35% Width decrease 47%

(b) EYE: Width decrease 10%
EYEBROW: Height increase 257%
NOSE: Height decrease 25% Width decrease 18%
LIP: Height decrease 35% Width decrease 43%

(c) EYE: no change
EYEBROW: Height increase 257%
NOSE: Height decrease 25% Width decrease 10%
LIP: Height decrease 35% Width decrease 39%

(d) EYE: Height increase 25%
EYEBROW: Height increase 257%
NOSE: Height decrease 25% Width decrease 2%
LIP: Height decrease 35% Width decrease 45%

(e) EYE: Height increases 25%
EYEBROW: Height increase 257%
NOSE: Height decrease 17% Width decrease 2%
LIP: Height decrease 35% Width decrease 41%

(f) EYE: Height increases 25%
EYEBROW: Height increase 257%
NOSE: Height decrease 8% Width decrease 2%
LIP: Height decrease 35% Width decrease 37%

Figure-4.5 (a) – (f) images after modification of Figure 11. (a)

In these six modified images, the upper left position of the face components is fixed. Only the size is varied. However, the position of a face component may differ when the images are of different persons. But in different modified images of the same person, the upper left position of face components is always fixed. The position of face components is noted at the time of extracting those features.

Figure 4.6 shows another set of modified face images.

(a) Original face image

(b) EYE: Width decrease 30%

EYEBROW: Height increase 214%

NOSE: Height decrease 33% Width decrease 18%

LIP: Height decrease 35% Width decrease 47%

(c) EYE: Width decrease 10%

EYEBROW: Height increase 257%

NOSE: Height decrease 25% Width decrease 18%

LIP: Height decrease 35% Width decrease 43%

(d) EYE: no change

EYEBROW: Height increase 257%

NOSE: Height decrease 25% Width decrease 10%

LIP: Height decrease 35% Width decrease 39%

(e) EYE: Height increase 25%

EYEBROW: Height increase 257%

NOSE: Height decrease 25% Width decrease 2%

LIP: Height decrease 35% Width decrease 45%

(f) EYE: Height increase 25%
EYEBROW: Height increase 257%
NOSE: Height decrease 17% Width decrease 2%
LIP: Height decrease 35% Width decrease 41%

(g) EYE: Height increase 25%
EYEBROW: Height increase 257%
NOSE: Height decrease 8% Width decrease 2%
LIP: Height decrease 35% Width decrease 37%

Figure-4.6 (a) – original image (b - (f) images after modification

Figure 4.7 shows the experiment on another image.

(a) Original face image

(b) EYE: Width decrease 30%

EYEBROW: Height increase 214%

NOSE: Height decrease 33% Width decrease 18%

LIP: Height decrease 35% Width decrease 47%

(c) EYE: Width decrease 10%

EYEBROW: Height increase 257%

NOSE: Height decrease 25% Width decrease 18%

LIP: Height decrease 35% Width decrease 43%

(d) EYE: no change

EYEBROW: Height increase 257%

NOSE: Height decrease 25% Width decrease 10%

LIP: Height decrease 35% Width decrease 39%

(e) EYE: Height increase 25%

EYEBROW: Height increase 257%

NOSE: Height decrease 25% Width decrease 2%
LIP: Height decrease 35% Width decrease 45%

(f) EYE: Height increase 25%

EYEBROW: Height increase 257%

NOSE: Height decrease 17% Width decrease 2%

LIP: Height decrease 35% Width decrease 41%

(g) EYE: Height increase 25%
EYEBROW: Height increase 257%
NOSE: Height decrease 8% Width decrease 2%
LIP: Height decrease 35% Width decrease 37%

Figure-4.7 (a) – original image (b - (f) images after modification

Figure 4.8 shows another set of modified images.

(a) Original face image

(b) EYE: Width decrease 30%

EYEBROW: Height increase 214%

NOSE: Height decrease 33% Width decrease 18%

LIP: Height decrease 35% Width decrease 47%

(c) EYE: Width decrease 10%

EYEBROW: Height increase 257%

NOSE: Height decrease 25% Width decrease 18%

LIP: Height decrease 35% Width decrease 43%

(d) EYE: no change

EYEBROW: Height increase 257%

NOSE: Height decrease 25% Width decrease 10%

LIP: Height decrease 35% Width decrease 39%

(e) EYE: Height increase 25%

EYEBROW: Height increase 257%

NOSE: Height decrease 25% Width decrease 2%

LIP: Height decrease 35% Width decrease 45%

(f) EYE: Height increase 25%

EYEBROW: Height increase 257%

NOSE: Height decrease 17% Width decrease 2%

LIP: Height decrease 35% Width decrease 41%

(g) EYE: Height increase 25%
EYEBROW: Height increase 257%
NOSE: Height decrease 8% Width decrease 2%
LIP: Height decrease 35% Width decrease 37%

Figure-4.8 (a) – original image (b - (f) images after modification

Figure 4.9 shows another set of modified images.

(a) Original face image

(b) EYE: Width decrease 30%

EYEBROW: Height increase 214%

NOSE: Height decrease 33% Width decrease 18%

LIP: Height decrease 35% Width decrease 47%

(c) EYE: Width decrease 10%

EYEBROW: Height increase 257%

NOSE: Height decrease 25% Width decrease 18%

LIP: Height decrease 35% Width decrease 43%

(d) EYE: no change

EYEBROW: Height increase 257%

NOSE: Height decrease 25% Width decrease 10%

LIP: Height decrease 35% Width decrease 39%

(e) EYE: Height increase 25%

EYEBROW: Height increase 257%

NOSE: Height decrease 25% Width decrease 2%

LIP: Height decrease 35% Width decrease 45%

(f) EYE: Height increase 25%

EYEBROW: Height increase 257%

NOSE: Height decrease 17% Width decrease 2%

LIP: Height decrease 35% Width decrease 41%

(g) EYE: Height increase 25%
EYEBROW: Height increase 257%
NOSE: Height decrease 8% Width decrease 2%
LIP: Height decrease 35% Width decrease 37%

Figure-4.9 (a) – original image (b - (f) images after modification

4.4. Performance analysis

When we use the transformation matrix as:

$$\begin{bmatrix} 5 & 7 & 0 \\ 3 & 50 & 0 \\ 10 & 15 & 1 \end{bmatrix}$$

Then we observe a significant amount of rotation. That is why we can use this transformation object for eyebrows and at most for the eyes. But the use of this transformation matrix for modification of nose and lips causes abnormality.

(a) Modified image when Transformation matrix is [1 0 0; 0 2 0; 20 20 1]

(b) Modified image when transformation matrix is [5 7 0;3 50 0;10 15 1]

Figure 4.10 Performance using different transformation matrices

CHAPTER: 5

CONCLUSIONS

5.1 Summary of the study

A total of 30 male and female images of different ages from ORL database was tested.

This system aims at constructing new faces based on the textual description of a face with the geometrically transformed features from the same face. Image scaling, translation, 2-D affine transformation, and image warping are used for local geometrical transformation of the facial features. For extraction of face components, we have used a geometric model of a face [3]. This system is helpful when the intended image is not present in the database. The system provides a user interface which does not need any proficiency of the user. This system can also be used as a mechanism to generate a database of different faces as it generates new modifies face images. This study can be very useful for criminal investigation where based on the description given by an eyewitness; we have to generate a face of a criminal which is not available in the database.

5.2 Future scope

In the present study, only some of the possible modifications have been performed. However, a range of modification of each component may be specified in future to improve the performance of the system. Face images with glasses have not been considered in this study; that can be considered in future. Future work should be connected with a survey of divergent scenarios consisting of actual situations.

CHAPTER: 6

REFERENCES

[1] Xiaogang Wang and Xiaoou Tang. (2009). Face Photo-Sketch Synthesis and Recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 31, No. 11.

[2] Mary Leung, Hung Yen Hui, and Irwin King. (1996): Facial Expression Synthesis by Radial Basis Function Network and Image Warping. IEEE International Conference on Neural Networks, Vol 3, pages 1400-1405.

[3] D. Bhattacharjee, S. Halder, M. Nasipuri, D.K. Basu, M. Kundu. (2009). Construction of Human Faces from Textual Descriptions, Springer, Soft Computing - A Fusion of Foundations, Methodologies, and Applications, Vol. 15, No. 3, pp. 429-447.

[4] Bernard Tiddeman, Michael Burt, and David Perrett. (2001). Prototyping and Transforming Facial Textures for Perception Research, IEEE Computer Graphics, and Application, Vol. 21 Issue 5, page 42-50.

[5] Yaun Y. Tang, and Ching Y. Suen. (1993). Image Transformation Approach to Nonlinear Shape Restoration, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 23, No. 1, page 155-172.

[6] Florent Perronnin, Jean-Luc Dugelay, and Kenneth Rose. (2004). A Probabilistic Model for Face Transformation with Application to Person Identification, EURASIP Journal on Applied Signal Processing:4, 510–521.

[7] Yong Rui and Thomas S. Huang. (1999). Image Retrieval: Current Techniques, Promising Directions, and Open Issues, Journal of Visual Communication and Image Representation 10, 39–62.

[8] G. Bebis, M. Georgiopoulos, N. da Vitoria Lobo, and M. Shah. (1999). Recognition by learning affine transformations, Pattern Recognition, Vol. 32, No. 10, pp. 1783-1799.

[9] Rafael C. Gonzalez, Paul Wintz. (1987). Digital Image Processing (second edition), Publisher: Addison Wesley.

[10] Richard Szeliski. (2006). Image Alignment and Stitching, Foundations and Trends in Computer Graphics and Vision, Vol. 2, No 1, 1–104.

[11] OpenGL-ARB, OpenGL Reference Manual: The Official Reference Document to OpenGL, Version 1.1., Second Edition, Addison-Wesley, Reading, MA, 1997.

[12] A. Watt, 3D Computer Graphics. Addison-Wesley, Third Edition, 1995.

[13] Pranoti Kale, K.R.Singh. (2015). A Technical Analysis of Image Stitching Algorithm, International Journal of Computer Science and Information Technologies, Vol. 6 (1), 284-288.

[14] K.Shashank, N.Siva Chaitanya, G.Manikanta, Ch.N.V.Balaji, V.V.S.Murthy. (2014). A Survey and Review Over Image Alignment and Stitching Methods, International Journal of Electronics & Communication Technology, vol.5, pp.50-52.

[15] Ebtsam Adel, Mohammed Elmogy, HazemElbakry. (2014). Image Stitching based on Feature Extraction Techniques: A Survey, International Journal of Computer Applications, vol.99, no.6, pp.1-8.