Review Article

# A Method for Face Recognition from Facial Expression

**Sarbani Ghosh and Samir K. Bandyopadhyay***

Department of Computer Science & Engineering, University of Calcutta,
Kolkata, India

***Corresponding Email:** skb1@vsnl.com

## ABSTRACT

Facial expressions play a major role in Face Recognition Systems and image processing techniques of Human Machine Interface. There are several techniques for facial features selection like Principal Component Analysis, Distance calculation among face components, Template Matching. This algorithm describes a simple template matching based facial feature selection technique and detects facial expressions based on distances between facial features using a set of image databases. The algorithm involves three stages: Pre Processing, Facial Feature Extraction and Distance Calculations. Then, we can identify whether a human is smiling or not using the measurement of Euclidean distances between pairs of eyes and mouth region of that face.

**Keywords**: Facial feature detection, Template matching, Normalized cross-correlation.

## INTRODUCTION

Facial Feature Detection system is fast becoming a familiar feature in 'apps' and on websites on different purposes. Human face identification and detection is often the first step in applications such as video surveillance, human computer interface, and face recognition and image database management[1]. Furthermore facial feature characteristics are very much effective in both biometric identification which automatically identifies a person from a digital image or a video image. Facial expressions are used not only to express our emotions, but also to provide important communicative cues during social interaction, such as our level of interest[2]. Facial features, eye feature has more application domain. It is reported that facial expressions have considerable effects on listener, near about 55 % effect of the spoken words depend on eye movements and facial expressions of the speaker.

This paper represents a computationally efficient algorithm for facial feature selection based on template matching method which further leads to classification of facial expressions by calculating the distance between eye pair and mouth. In a smiley face distance between eye pair and mouth must be greater than normal face (without any notable emotion). Thus it can be identified whether a person is happy or not. Any face image database with various expressions can be taken for application. At first, minimal pre-processing including gray scale conversion is done on the image. After that, matching between original image and template image is done using normalized cross-correlation technique. Each matching area is bounded by box to identify that region of interest. Then the mid points between the eye regions are found and the distance between the mid points and the corners of the mouth region is calculated. On the basis of the distances between these features, emotions are recognized.

## LITERATURE REVIEW

Human-like robots and machines that are expected to enjoy truly intelligent and transparent communications with human can be created using automatic facial expression recognition with a set of specific desired accuracy and performance requirements. Expression recognition involves a variety of subjects such as perceptual recognition, machine learning, affective computing etc.

One case study uses skin color range of human face to localize face area. After face detection, various facial features are identified by calculating the ratio of width of multiple regions in human face. Finally the test image is partitioned into a set of sub-images and each of these sub-images is matched against a set of sub-pattern training set. Partitioning is done using Aw-SpPCA algorithm. Given as input any emotion of face, this pattern training set will classify the particular emotion[1].

Face component extraction by dividing the face region into eye pair and mouth region and measurement of Euclidean distance among various facial features is also adopted by a case study. Similar study is done by Neha Gupta to detect emotions. This research includes four steps: pre-processing, edge detection, feature extraction and distance measurement among the features to classify different emotions. This type of approach is classified as Geometric Approach[3].

Another research includes Face detection method using segmentation technique. First, the face area of the test image is detected using skin color detection. RGB color space is transformed into yCbCr color space in the image and then skin blocks quantization is done to detect skin color blocks. As next step, a face cropping algorithm is used to localize the face region. Then, different facial features are extracted using segmentation of each component region (eyes, nose, mouth). Finally, vertical & angular distances between various facial features are measured and based on this any unique facial expression is identified. This approach can be used in any biometric recognition system[2].

A template matching based facial feature detection technique is used in a different case study[4,7,8].

Different methods of face detection and their comparative study are done in another review work. Face detection methods are divided into two primary techniques: Feature based & View based methods[5,9].

Gabor filters are used to extract facial features in another study. This approach is called Appearance based approach. This classification based facial expression recognition method uses a bank of multilayer perceptron neural networks. Feature size reduction is done by Principal Component Analysis (PCA)[6,10].

Thus existing works primarily focused in detecting facial features and they are served as input to emotion recognition algorithm. In this study, a template based feature detection technique is used for facial feature selection and then distance between eye and mouth regions is measured.

### Design

(See image 1.)

## Steps

### 1. Acquisition of input image

Read the Input Human Face Image. If the Input Image is color (RGB), then convert it to Gray scale image.

### 2. Acquisition of template image

Main feature of a Human Face to be extracted are: Left Eye, Right Eye, Nose and Mouth. Read 4 different Template Image for the corresponding features. If the Template Images are color (RGB), convert them to corresponding Gray scale Images respectively.

### 3. Normalized 2D - Cross correlation between input image and template images

Correlation mask $w$ (x, y) of size m*n, with an image $f$ (x, y) may be expressed in the form

$$C(x,y) = \sum_{s}\sum_{t} w(s,t)\, f(x+s,y+t)$$

Where the limits of summation are taken over the region shared by $w$ and $f$. This equation is evaluated for all values of the displacement variables x and y so that all elements of $w$ visit every pixel of $f$, where $f$ is assumed to be larger than $w$. When the input image $f$ is larger than the template image, we say that it is *normalized*. Here $w$ is referred to as a *template* and correlation is referred to as *template matching*. The values of template cannot all be the same. The resulting matrix C contains the correlation coefficients, which can range in value from -1.0 to 1.0.Perform 2D – Cross Correlation between Gray scale Input Image and 4 different Gray scale Template Images separately.

### 4. Determine the maximum correlation value

Following is performed for all 4 2D – Cross Correlation operations mentioned in the previous step:

The Maximum Value of C occurs when the corresponding normalized region in f are identical i.e where the maximum correlation occurs. Find the corresponding rectangular region where maximum matching is found. Pixel coordinate of top left corner (let $(x_1, y_1)$), width (let w), height (let h) of the rectangle are found.

### 5. Draw a boundary around the region

Following is performed for all 4 2D – Cross Correlation operations mentioned in step 3:

Draw Rectangle with pixel coordinates $(x_1, y_1)$, $(x_2, y_1)$, $(x_1, y_2)$, $(x_2, y_2)$, where $x_2=x_1+w$ and $y_2=y_1+h$.

### 6. Determine the middle point of the rectangle around left eye

If the pixel coordinates of the drawn rectangle around left eye are $(x_1, y_1)$, $(x_2, y_1)$, $(x_1, y_2)$, $(x_2, y_2)$, then middle point coordinate of the rectangle are $(x_{mid1}, y_{mid1})$ where $x_{mid1} = (x_1+x_2)/2$ and $y_{mid1} = (y_1+y_2)/2$.

### 7. Determine Euclidian distance between middle point of rectangle around left eye and top left corner pixel coordinate of rectangle around mouth

If the middle point pixel coordinate of rectangle around left eye is $(x_{mid1}, y_{mid1})$ and top left corner pixel coordinate of rectangle around mouth is $(x_1', y_1')$ then Euclidian distance between these 2 points is $= \sqrt{( x_{mid1} - x_1')^2 + ( y_{mid1} - y_1')^2}$ unit.

### 8. Determine the middle point of the rectangle around right eye

If the pixel coordinates of the drawn rectangle around left eye are $(x_1, y_1)$, $(x_2, y_1)$, $(x_1, y_2)$, $(x_2, y_2)$, then middle point coordinate of the rectangle are $(x_{mid2}, y_{mid2})$ where $x_{mid2} = (x_1+x_2)/2$ and $y_{mid2} = (y_1+y_2)/2$.

9. Determine Euclidian distance between middle point of rectangle around right eye and top right corner pixel coordinate of rectangle around mouth

If the middle point pixel coordinate of rectangle around right eye is ($x_{mid2}$, $y_{mid2}$) and top right corner pixel coordinate of rectangle around mouth is ($x_1$'',$y_1$'') then Euclidian distance between these 2 points is $= \sqrt{\{( x_{mid2} - x_1'')^2 + ( y_{mid2} - y_1'')^2\}}$ unit.

## Algorithm

**Facial_Expression_Recognition (Input Image, Template Images)**

**Step1.** Start

**Step2.** Read Input Human Face Image.
If the Input Image is color (RGB), then
convert it to Gray scale Image and save the pixel values to a 2D array let gface.
Else
save the pixel values of the input image to a 2D array let gface.

**Step3.** Read Left eye template image.
If the template image is color (RGB), then
convert it to Gray scale Image and save the pixel values to a 2D array let gleft.
Else
save the pixel values of the input image to a 2D array let gleft.

**Step4.** Read Right eye template image.
If the template image is color (RGB), then
convert it to Gray scale Image and save the pixel values to a 2D array let gright.
Else
save the pixel values of the input image to a 2D array let gright.

**Step5.** Read Nose template image.
If the template image is color (RGB), then
convert it to Gray scale Image and save the pixel values to a 2D array let gnose.
Else
save the pixel values of the input image to a 2D array let gnose.

**Step6.** Read Mouth template image.
If the template image is color (RGB), then
convert it to Gray scale Image and save the pixel values to a 2D array let gmouth.
Else
save the pixel values of the input image to a 2D array let gmouth.

**Step7.** Declare 4 2D Array C1, C2, C3 & C4 of size m*n where m*n is the size of gface.

**Step8.** Calcualte C1[][] = 2D_norm_crosscorr(gleft,gface)
C2[][]= 2D_norm_crosscorr(gright,gface)

C3[][] = 2D_norm_crosscorr(gnose,gface)
C4[][] = 2D_norm_crosscorr(gmouth,gface)

**Step9.**

**Step10.** Call $(x_{11},y_{11},w_1,h_1)$ = Find_max(C1)
$(x_{21},y_{21},w_2,h_2)$ = Find_max(C2)
$(x_{31},y_{31},w_3,h_3)$ = Find_max(C3)
$(x_{41},y_{41},w_4,h_4)$ = Find_max(C4)
where $(x_{11},y_{11},w_1,h_1)$, $(x_{21},y_{21},w_2,h_2)$, $(x_{31},y_{31},w_3,h_3)$, $(x_{41},y_{41},w_4,h_4)$ are top – left pixel coordinate, width, height of the matched rectangular area around left eye, right eye, nose and mouth respectively.

**Step11.** Calculate $x_{12} = x_{11} + w_1$ & $y_{12} = y_{11} + h_1$
$x_{22}= x_{21} + w_2$ & $y_{22} = y_{21} + h_2$
$x_{32}= x_{31} + w_3$ & $y_{32} = y_{31} + h_3$
$x_{42}= x_{41} + w_4$ & $y_{42} = y_{41} + h_4$
where $(x_{12},y_{12})$, $(x_{22},y_{22})$, $(x_{32},y_{32})$, $(x_{42},y_{42})$ are bottom right pixel coordinate of the matched rectangular area around left eye, right eye, nose and mouth respectively.

**Step12.** Draw Boundary Rectangle around left eye in gface with top – left, top – right, bottom left and bottom – right pixel coordinates as $(x_{11},y_{11})$, $(x_{12},y_{11})$, $(x_{11},y_{12})$ & $(x_{12},y_{12})$ respectively.

Draw Boundary Rectangle around right eye in gface with top – left, top – right, bottom – left and bottom – right pixel coordinates as $(x_{21},y_{21})$, $(x_{22},y_{21})$, $(x_{21},y_{22})$ & $(x_{22},y_{22})$ respectively.

Draw Boundary Rectangle around nose in gface with top – left, top – right, bottom – left and bottom – right pixel coordinates as $(x_{31},y_{31})$, $(x_{32},y_{31})$, $(x_{31},y_{32})$ & $(x_{32},y_{32})$ respectively.

Draw Boundary Rectangle around mouth in gface with top – left, top – right, bottom – left and bottom – right pixel coordinates as $(x_{41},y_{41})$, $(x_{42},y_{41})$, $(x_{41},y_{42})$ & $(x_{42},y_{42})$ respectively.

Calculate middle point pixel coordinate $(x_{1mid},y_{1mid})$ of the boundary rectangle around left eye as $x_{1mid} = (x_{11}+x_{12})/2$ and $y_{1mid} = (y_{11}+y_{12})/2$.

**Step13.** Calculate Euclidian Distance between middle point pixel coordinate $(x_{1mid},y_{1mid})$ of the boundary rectangle around left eye and top – left pixel coordinate $(x_{41},y_{41})$ of the boundary rectangle around mouth as:

**Dist1** = $\sqrt{\{(x_{1mid} - x_{41})^2 + (y_{1mid} - y_{41})^2\}}$ unit.

**Step14.** Calculate middle point pixel coordinate $(x_{2mid},y_{2mid})$ of the boundary rectangle around right eye as $x_{2mid} = (x_{21}+x_{22})/2$ and $y_{2mid} = (y_{21}+y_{22})/2$.

**Step15.** Calculate Euclidian Distance between middle point pixel coordinate $(x_{2mid},y_{2mid})$ of the boundary rectangle around right eye and top – right pixel coordinate $(x_{42},y_{41})$ of the boundary rectangle around mouth as:

**Dist2** = $\sqrt{\{(x_{2mid} - x_{42})^2 + (y_{2mid} - y_{41})^2\}}$ unit.

**Step16.** Write the value of Dist1 and Dist2 in a output text file for comparison.
**Step17.** Repeat step 1 to 15 for another same human face but with smiling facial expression.
**Step18.** Compare both input face images according the distances measured between eyes &
mouth. The image with larger distance is considered as Happy face or smiling face, in general.
**Step19.** Exit

**2D_norm_crosscorr (Template Gray scale Image, Input Gray scale Image)**

**Step1.** Start
**Step2.** Perform 2D Cross Correlation between Template Image and Input Image pixel values and
return 2D array C of size m*n with values of the corresponding Cross Correlation, where m*n is the size of the Input Image.
**Step3.** End

**Find_max(C[ ][ ])**

**Step1.** Start
**Step2.** Find Maximum Value of 2D Array C[ ][ ] and determine the corresponding rectangular
region where the maximum value is found.
**Step3.** Find top – left position coordinate (x,y), width (w) and height (h) of the rectangular
region and return the values.
End

## EXPERIMENTAL RESULTS

### Test results

- Testing includes sets of images with relatively different lighting condition. Each set of image is the images of same human face with different emotions (Neutral & with smiling). We perform template matching on both faces of a set of image using different templates. Templates, i.e. pair of eyes, nose & mouth area used, may be the regions of the same test image or different image other than the test image.
- After template matching, Euclidian distances between midpoint of the rectangle region of eye areas & the top two corner points (left & right) of the rectangle region of mouth area are calculated on both images.
- The distances are compared and the distance with larger value primarily yields that the person is smiling. A neutral face has smaller distance than the smiling expression of the same face, generally.

## Test result: Template Matching

### Templates used

- Left eye template
- Right eye template
- Nose template
- Mouth template

### Case 1

Matched with templates of test image itself. (See figure 1-7.)

### Case 2

Matched with different templates. (See figure 8-14.)

### Test result

Euclidean Distance calculation. (See figure 15-18.)

## Euclidean distance calculation

### Neutral face

- Distance between Left eye & Mouth: 50.01unit
- Distance between Right eye & Mouth: 53.04 unit

### Smiling face

- Distance between Left eye & Mouth: 54.08 unit
- Distance between Right eye & Mouth: 60.02 unit

Smiling face has larger distance between eyes & mouth than Neutral face.

### Performance analysis

Since the main purpose of work is smile recognition, therefore sample pictures must be taken with two different emotions, i.e. with smile or without smile. This research is based on some common assumption:

- Although templates of eyes, nose or mouth from different images can be used to match with a test image, best possible matching & region of interest detection can occur in the case where templates images are cropped from the same test image.
- Distance between eyes & mouth depends on the size of the template image & finally on the size of the rectangle region of interest. There may be some cases where, distance between eyes & mouth will be smaller than the distance measured with the same neutral face, due to very small size of template images, so as to size of region of interest. Generally, the template images should cover the facial features broadly so that the whole eye region or nose region or mouth region is covered.
- Templates of neutral image cannot be matched on a test image of smiling face & vice versa. Both set of templates should be different.

### Performance of the Template matching algorithm

(See case 1-3.)

### Future work

A primary work is done to develop an Emotion Detection system which will be able to identify many different types of Emotion or facial expression in human face. Here, in the specified algorithm, we first tried to extract all the regions of interest i.e., left eye, right eye, nose & mouth region using template matching technique. Then, we measured the vertical distance between two eyes & mouth to detect one expression that whether a person is smiling or not.

## Next Steps

### Principal component analysis

We will partition the whole face into equal sized sub-images or sub-regions and each of these sub-regions will be matched against a set of sub-pattern training set database using PCA algorithm (i.e. Eigen vectors with greater Eigen values).

### Measurement of distances among other facial features

To identify many more different types of expression in human face, further we have to measure distances between multiple face components like:

- Right eye-left eye
- Left eye-nose peak
- Right eye-nose peak
- Nose peak-mouth
- Nose height & width

All of these features will define the uniqueness of a whole facial component.

## CLASSIFICATION

Finally all the image sets with different emotions will be classified according to these training set and each type of emotion will be identified as a different emotion with different name. This classification can be done using *k*-NN (*K* Nearest Neighbors) classifier. Given any type of input image with various emotions, the system may identify the emotions individually & accurately.

## CONCLUSION

Facial expression recognition or emotion detection system has numerous applications in image processing domains, security applications domain or any type of biometric system. This research work is a primary step of an emotion detection system using which one specific emotion can be identified after extracting different facial features. Future work will be done to classify more different types of expressions in human face and identify each emotion properly. The proposed approach will help to define uniqueness of human face component accurately to some extent which can be used greatly in biometric recognition system. Also the algorithm can be used further in face recognition systems, machine learning system and other types of image processing applications.

## REFERENCES

1. Md. Zahangir Alom, Mei-Lan Piao, Md. Ashraful Alam, Nam Kim and Jae-Hyeung Park. Facial Expressions Recognition from Complex Background using Face Context and Adaptively Weighted sub-Pattern PCA, *World Academy of Science, Engineering and Technology*, Vol:6 2012, Page 3-20.
2. Dewi Agushinta R, Adang Suhendra, Sarifuddin Madenda, and Suryadi H.S. Face Component Extraction Using Segmentation Method Face Recognition System, *Journal of engineering Trends in Computing & Information Science*, Vol. 2 No. 2.
3. Neha Guptaand Prof. Navneet Kaur Design and Implementation of Emotion Recognition System by Using Matlab. On *International Journal of Engineering Research and*

*Applications* (IJERA) Vol. 3, Issue 4, Jul-Aug 2013, Page2002-2006.
4. MRS. Sunita Roy and PROF. Samir Kumar Bandyopadhyay. Extraction of facial features using a simple template based method, on *International Journal of Engineering Science and Technology* (IJEST).
5. Mrs. Sunita Roy, Mr. Sudipta Roy, and Prof. Samir K. Bandyopadhyay. A Tutorial Review on Face Detection, *International Journal of Engineering Science and Technology* (IJEST), Volume 1 Issue 8, October-2012.
6. Lajevardi, S.M.and Lech, M. Facial Expression Recognition Using Neural Networks and Log-Gabor Filters, Digital Image Computing: Techniques and Applications IEEE 2008.
7. W.-K. Liao and G. Medioni. 3D face tracking and expression tem using subspace model analysis," in IEEE Conf. Systems, Man inference from a 2D sequence using manifold learning, CVPR, and Cybernetics, vol. 2, 2005.
8. T. Kanade, J. F. Cohn, and Y. Tian. Comprehensive database for spontaneous facial action modeling and understanding, TPAMI, facial expression analysis," in FG, 2000.
9. Y.-L. Tian, T. Kanade, and J. Cohn. Recognizing lower face action units for facial expression analysis, in FG, March 2000, Page. 484.
10. M. Pantic and I. Patras. Dynamics of facial expression: Recognition of facial actions and their temporal segments from face profile image sequences, *SMC-B*, vol. 36, no. 2, pp. 433–449, 2006.

**Case 1.** Matched with templates of the test image itself

| No. | Type of Emotion | No. of Input Images | Recognized | Result (%) |
|---|---|---|---|---|
| 1 | Neutral | 20 | 20 | 100 |
| 2 | Happy | 10 | 10 | 100 |

**Case 2.** Matched with templates of different images other than the test image

| No. | Type of Emotion | No. of Input Images | Recognized | Result (%) |
|---|---|---|---|---|
| 1 | Neutral | 20 | 15 | 75 |
| 2 | Happy | 10 | 7 | 70 |

**Case 3.** Performance of smiling face identification algorithm after template matching

| No. of sets of images | No. of Neutral faces | No. of Happy faces | No. of sets of images where distance between eyes & mouth is larger for one face than the other in the particular set | No. of sets of images where smiling face is identified | Result |
|---|---|---|---|---|---|
| 10 | 10 | 10 | 8 | 8 | 80 |

**Figure 1.** Test Image

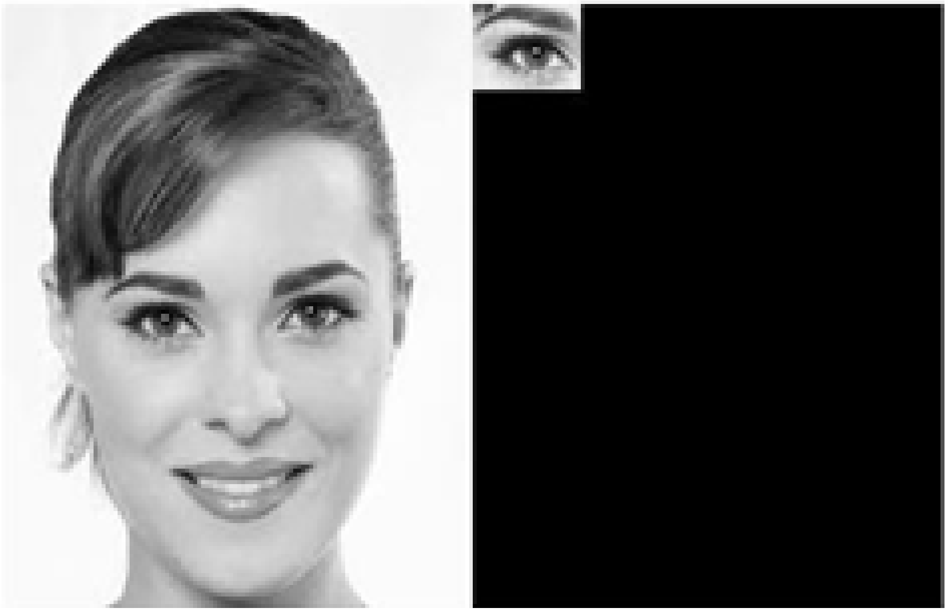

**Figure 2.** Left eye template

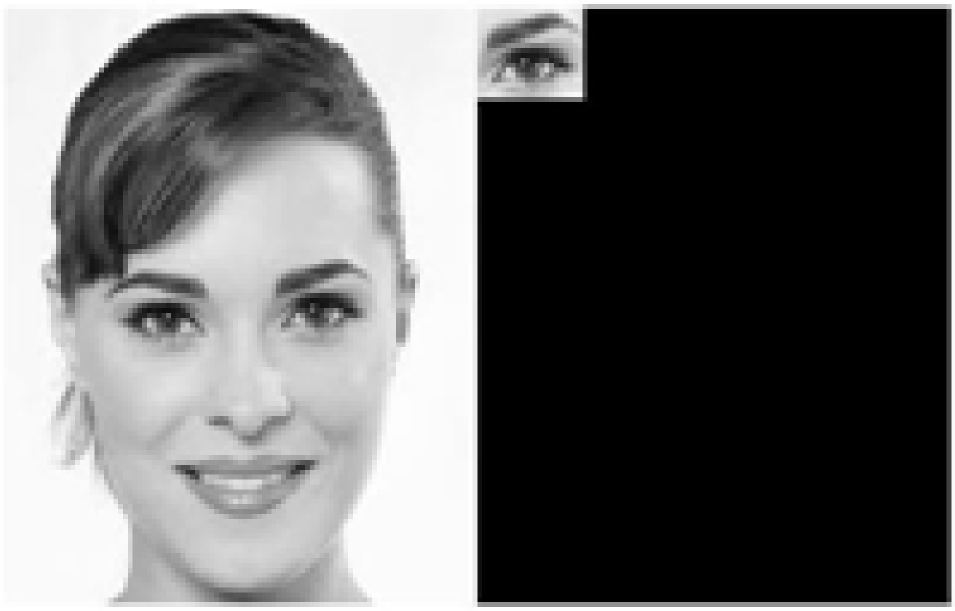

**Figure 3.** Right eye template

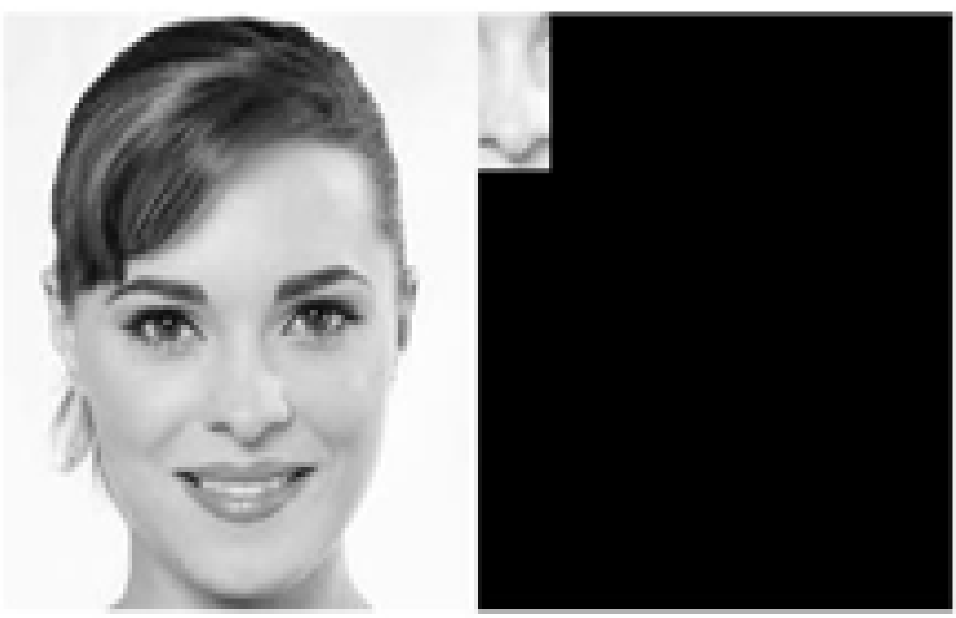

**Figure 4.** Nose template

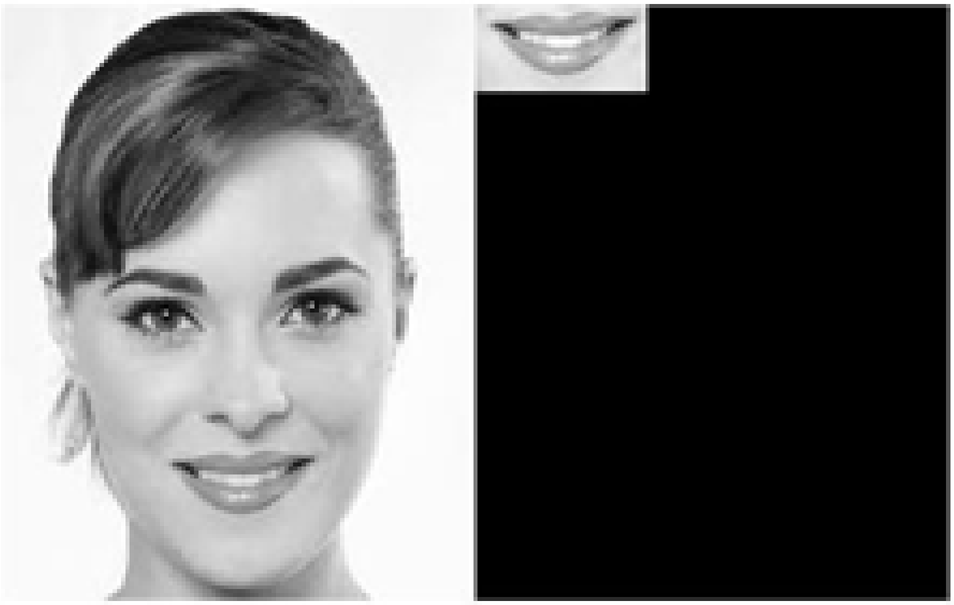

**Figure 5.** Mouth template

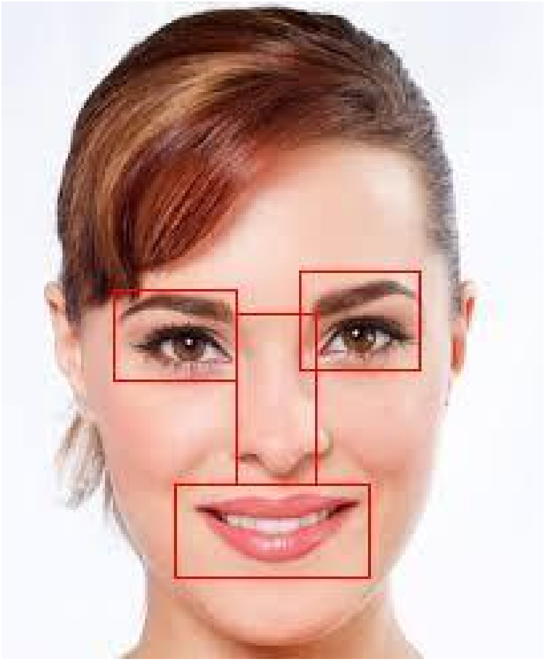

**Figure 6.** Output Image

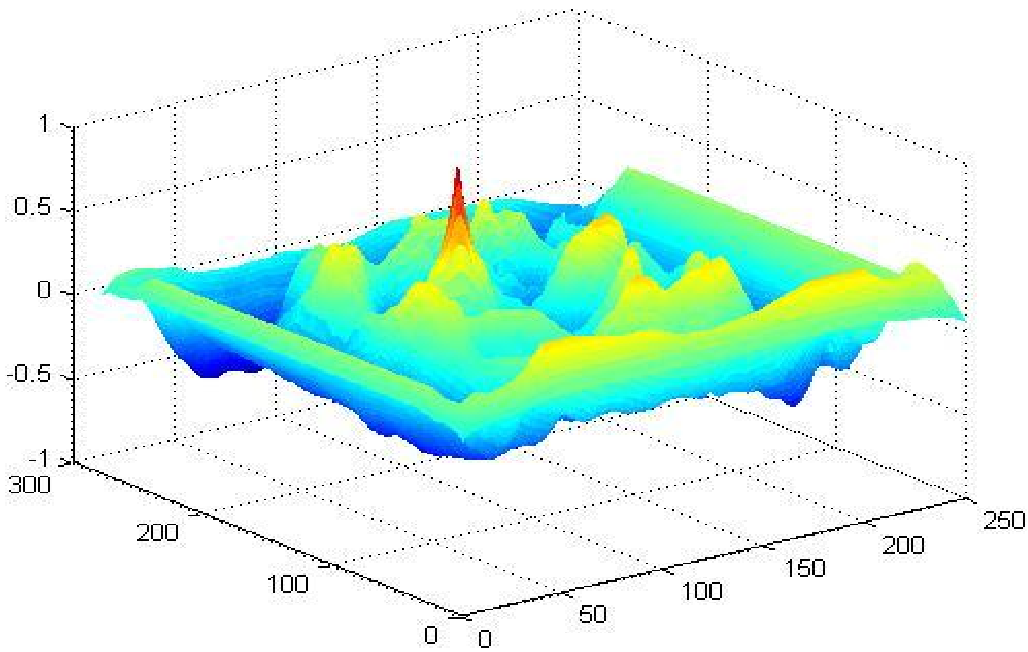


**Figure 7.** Peak normalized cross-correlation value when matched with left eye template

**Figure 8.** Test image

**Figure 9.** Templates of this image

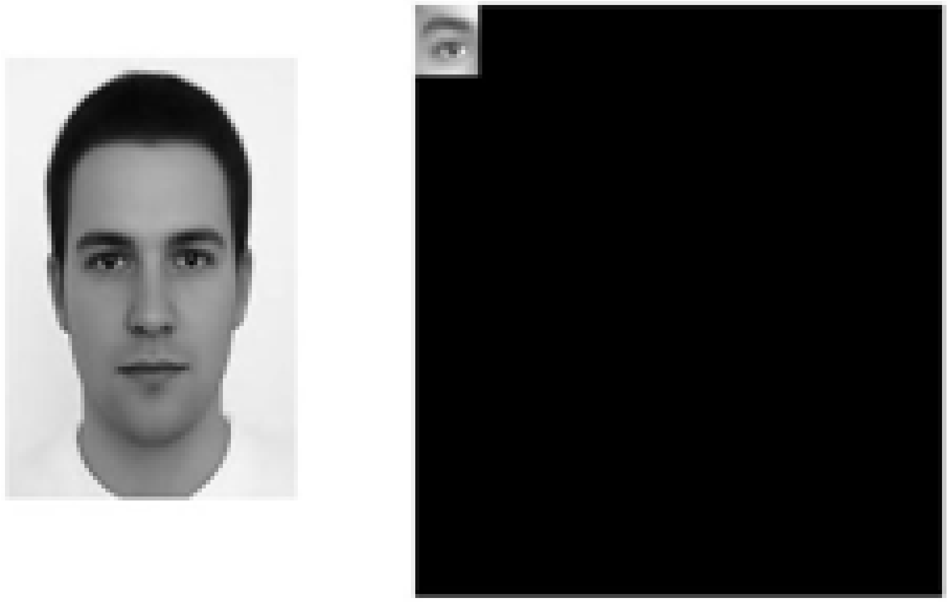

**Figure 10.** Left eye template

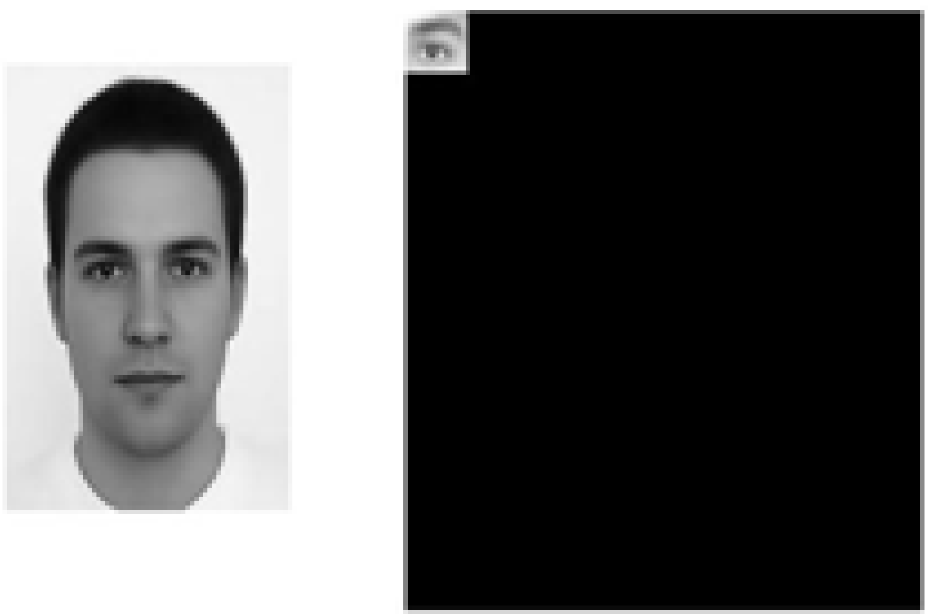

**Figure 11.** Right eye template

**Figure 12.** Nose template

**Figure 13.** Mouth template

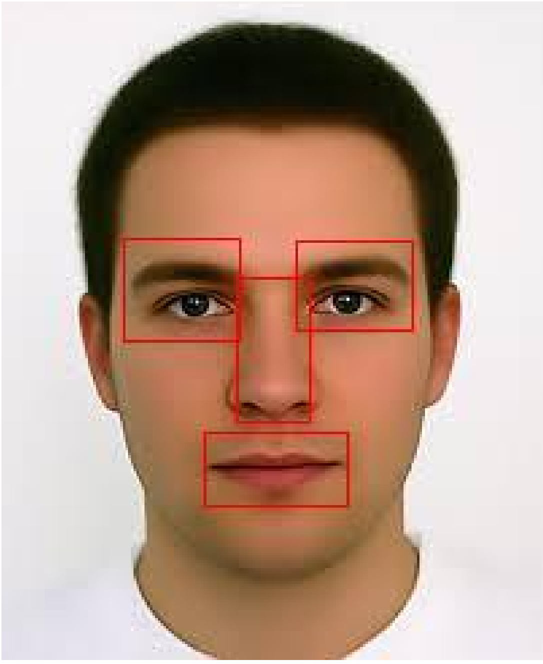

**Figure 14.** Output image

**Figure 15.** Neutral face

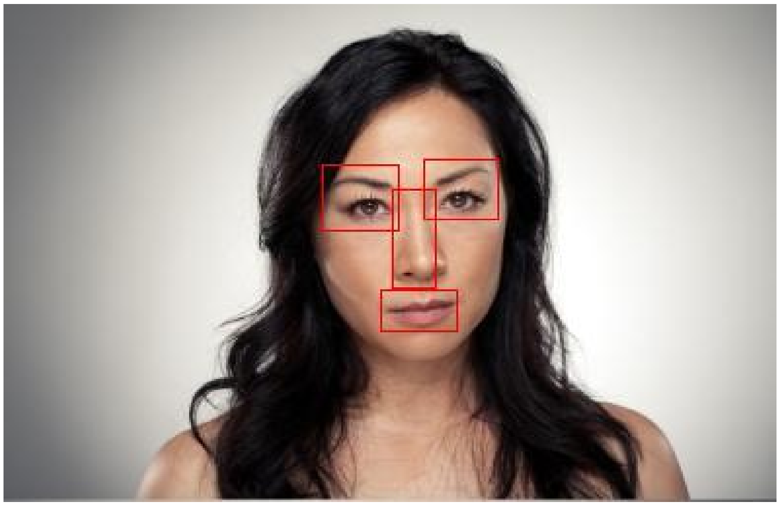

**Figure 16.** Neutral face template matching

**Figure 17.** Smiling face

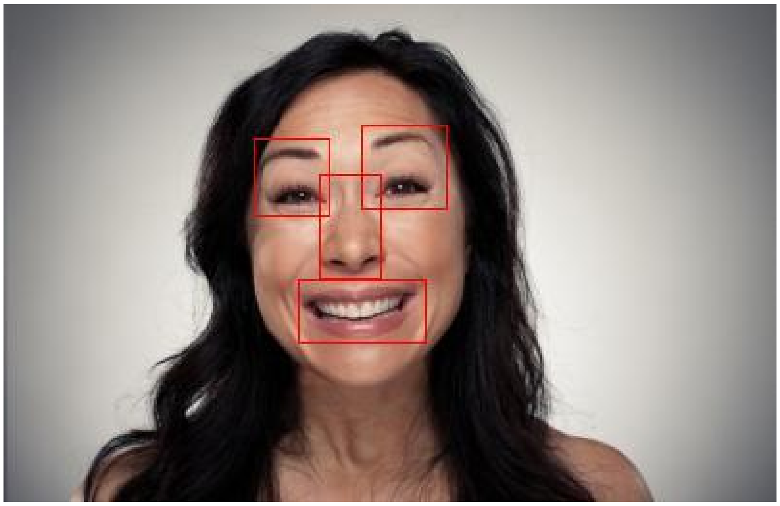

**Figure 18.** Smiling face template matching

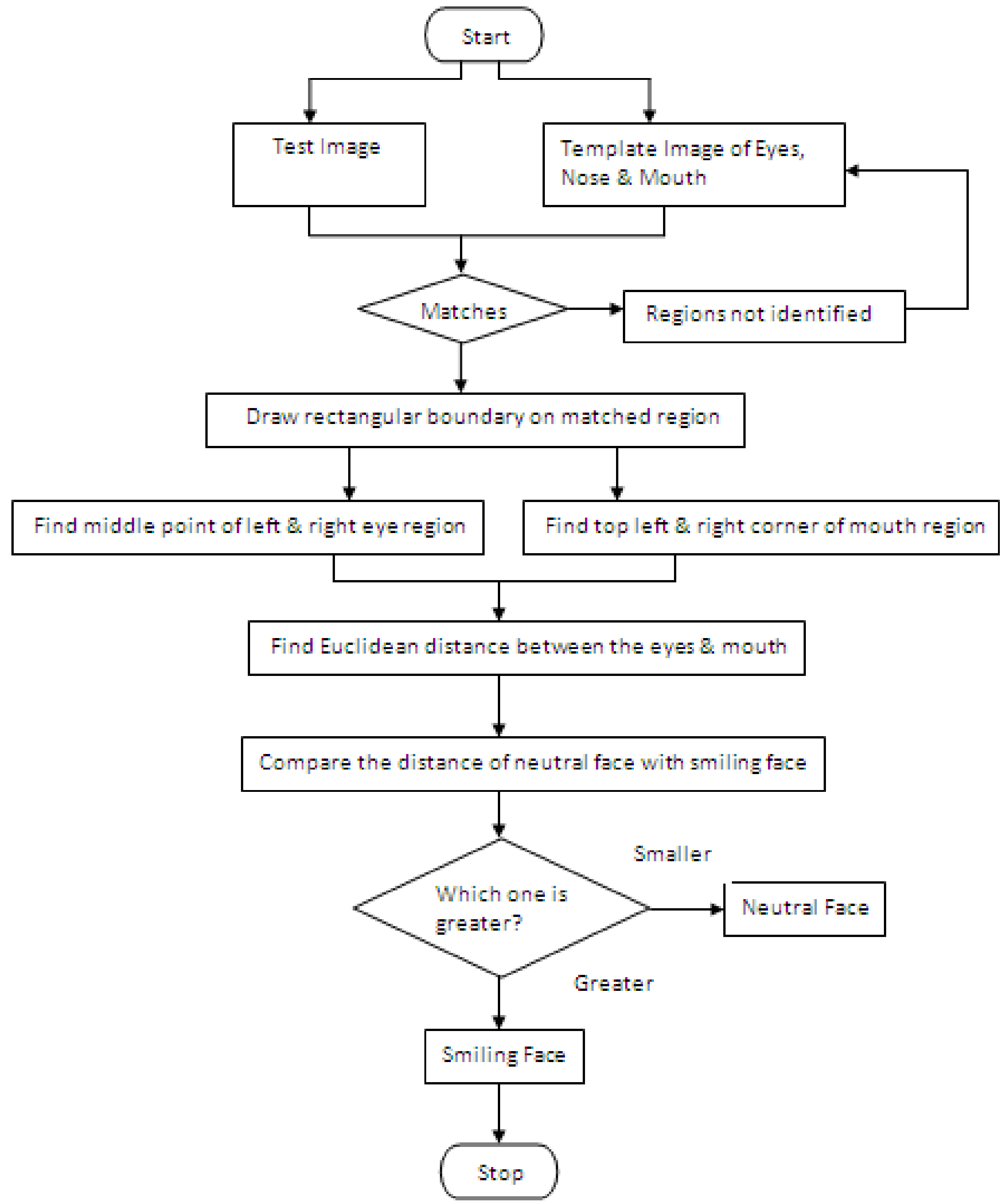
Start
Test Image
Template Image of Eyes, Nose & Mouth
Matches
Regions not identified
Draw rectangular boundary on matched region
Find middle point of left & right eye region
Find top left & right corner of mouth region
Find Euclidean distance between the eyes & mouth
Compare the distance of neutral face with smiling face
Which one is greater?
Smaller
Neutral Face
Greater
Smiling Face
Stop

**Image 1.**