AUTOMATIC DETECTION OF SMILING FACE AND NEUTRAL FACE

Sreelekshmi A.N
Department of Computer Science, Sree Ayyappa College,Eramallikkara,chengannur, India.
sree.an1989@gmail.com

ABSTRACT— Facial feature points, such as the corners of the eyes, corners and outer mid points of the lips and nostrils, are generally referred to as facial salient points. Detection of facial feature points is often the first step in computer vision applications such as face identification, facial expression recognition, face tracking and lip reading etc. Facial features generally include salient points which can be tracked easily, like corners of the eyes, nostrils, lip corners etc. Currently, most of the applications for facial expressions tracking are manually giving points as initial feature points for tracking. Detection of facial features like eye, pupil, mouth, nose, nostrils, lip corners, eye corners etc., with different facial expression and illumination is a challenging task. In the present work, human eye detection is the first step in facial feature detection, since the face has a nice facial geometry, which can be estimated, based on the eyes position. . In this work, we propose expression identification based on fully automatic detection of facial features and deals with the development of a technique that can perform the classification of given a frontal face input image, identify the features and classify the face into either neutral or smiling.

KEYWORDS- Face detection, Feature extraction, , ROI,Expression classification, Shi-Tomasi’s corner detection,Ada-boost algorithm, Haar-like features.

INTRODUCTION

Human eye detection is the first step in facial features detection as the face has a nice facial geometry, which can be estimated, based on the eyes position. Eye detection using AdaBoost algorithm gives an estimated location of the face and eyes. AdaBoost algorithm uses a set of weak cascaded classifiers, makes the classification accuracy greatly improved through extensive studies. However, for some cases, the detection fails when it is detecting the features like eye, nose and mouth in the whole face image. It can be improved by providing the estimated search regions for eyes and nose. The use of mouth detection cascade is often found to be inaccurately detecting the mouth even after estimated search region is provided. Mouth detection using the concept of facial geometry is used in this work and the method has been tested on number of faces including a set of faces from database (FEI Face Database) and also some real images.

OVERVIEW OF THE PROPOSED METHOD

Human eye detection is the first step in facial features detection. It is because; the face has a nice facial geometry, which can be estimated, based on the eyes position. Eye detection using AdaBoost algorithm gives an estimated location of the face and eyes. AdaBoost algorithm uses a set of weak cascaded classifiers, makes the classification accuracy greatly improved through extensive studies. However, for some cases, the detection fails when it is detecting the features like eye, nose and mouth in the whole face image. It can be improved by providing the estimated search regions for eyes and nose. The use of mouth detection cascade is often found to be inaccurately detecting the mouth even after estimated search region is provided. Mouth detection using the concept of facial geometry is used in this work and the method has been tested on number of faces including a set of faces from database (FEI Face Database) [1].

Eye corners are having sharp edge information. In the proposed work, the eye corners within the Eye ROI (Region of Interest) are detected using Shi-Tomasi’s corner detection algorithm. The main downside of that method is that it needs to train the algorithm with the template patches. Lip muscles are highly deformable and subjected to change. The Shi-Tomasi feature corner detection algorithm takes care of the features which can be tracked easily[7]. Samples of smiling faces as well as faces wearing glasses are used to test the accuracy of detection the relevant features. Nostrils are detected by thresholding the gray scale image of nose and then finding the contours within the region. The center of the bounded rectangular region (bounding the contour) is the nostril. The methodology for detecting lip corner points is also explained in detail in the following chapter. The algorithm is tested with various frontal face images with different illumination. It is found to be highly accurate in spite of varying lighting conditions. The algorithm is also tested against different lips pose like, mouth closed and mouth open (smiling face) and found to be detecting accurately [2].

IMPLEMENTAION DETAILS

In our proposed approach, initially we detect the face using Viola and Jones' Boosting algorithm and a set of Haar-like cascade features. The eye search area is minimized by assuming the eyes' estimated position to be at the upper part of the face. Haar-like features cascade is used for the eye detection. It locates the rectangular regions containing eyes. Given the eyes ROI, an algorithm (shi-Thomasi's corner detection) is used to locate the eye corners. Next, the nose is detected using haar-like features. Having known the eyes corners, and the position of the nose, mouth location estimation is carried out based on the facial geometry. The next step is to locate the lips corners points, which are considered as good features for tracking lips movement. Finally, nostrils are detected from the nose ROI.

The block diagram of the proposed method is given bellow [1]:-

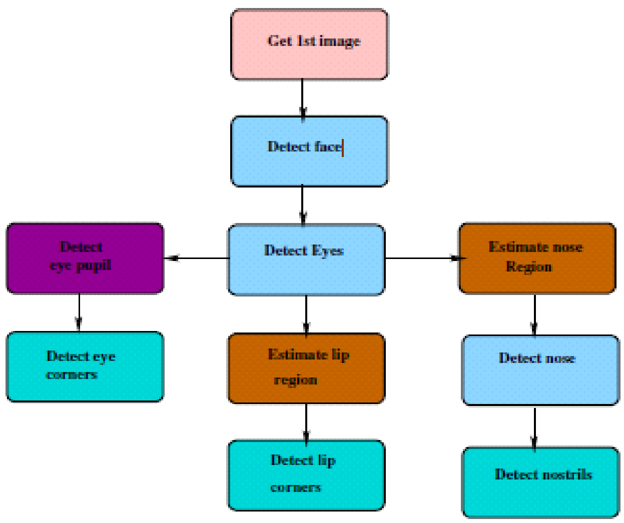

Figure.3.1: System Design

Automatic facial features detection approach

- **Face detection**

Viola and Jones' face detection algorithm is based on the Haar-like features. The four types of rectangular box are used to extract the features. The shaded area of the rectangular region is subtracted from the white area of the rectangle.

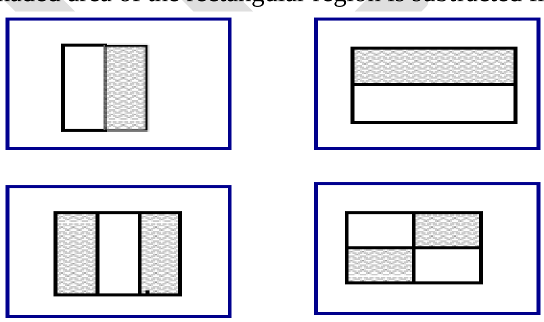

Figure 3.2.1:Haar-like features

These are the four types of Haar-like features, which is used for features detection. Upper two rectangles gives edge features, below left gives line features and the right one gives rectangle features [3].

- **Eye detection**

Here, eyes are detected from the face ROI using eye detection cascade of boosted tree classifiers with haar-like features. To improve the accuracy of eye detection, approximate location of eyes in the face is estimated.

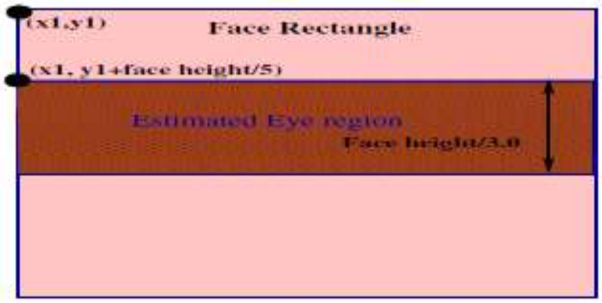

Figure 3.2.2: Estimated eye region

The eye corners are more stable than other eye features. But, accurate detection of eye corners is difficult. Also the eyes have eye lashes, which make the detection of eyes corners and pupil even more difficult [1].

- **Eye Corners Detection**

In the proposed work, eye corners are detected as follows[7].

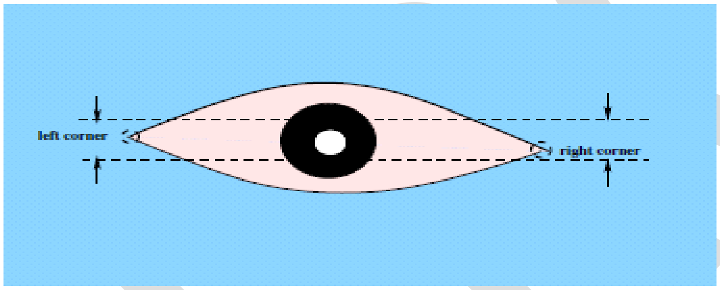

Figure 3.2.3: Estimated eye

1) Get the ROI image of the eye.
2) Extract the gray scale image of the ROI.
3) Using Shi-Tomasi's Good Features to Track method, obtain all the good corner features within the ROI.
4) Get the coordinate of the pupil detected in subsection 1 as ($x0$, $y0$). Considering each coordinates of the corners as (xi, yi), obtain the difference vectors ($x0-xi$) and ($y0 - yi$). Positive and negative values across x direction and y-direction will obtain.
5) Corners with positive difference are in the left of the pupil and those of negative difference are at the right of the pupil.
6) The eye corners usually are near to y-coordinate of the eye pupil and also mostly found slightly below the pupil. So taking a constrain in the y-direction (different along y direction in the range of ($+3\,to - 8$)), find the greatest $+ve$ value and $-ve$ value of the x vector.
7) The greatest $+ve$ value gives the left most corner point and the greatest $-ve$ value gives the right most corner point.
8) Resultant two extreme points are the required corner points of the eyes.

- **Nose Detection**

Nose is detected within the face region by using haar-like cascaded features and AdaBoost classification algorithm. To improve the nose detection accuracy, estimated region of nose is calculated in the given face image [1].

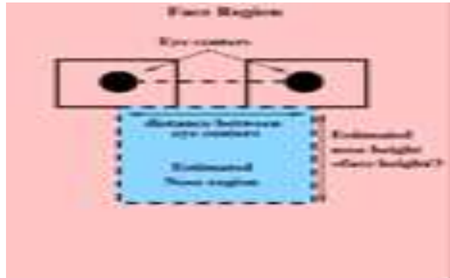

Figure 3.2.4: Estimated nose region

Nose position estimation is done as follows:

1) Get both the eyes ROI after eye detections are done.
2) Obtain the eyes center as $(x1, y1)$ and $(x2, y2)$ (centroid of the rectangle).
3) Obtain initial position of the rectangular region for nose as $(xn0, yn0)$. Where, $xn0$ = is the x coordinate of the center of eye1. And $yn0$ is the y coordinate of the initial point of eye1 + eye1 height.
4) The width of the nose rectangle is the difference $d = (x2 - x1)$.
5) The height is given by 1/3 of the face height.
6) The estimation of nose can also be one for tilted face by calculating the angle between two eye centers.
7) Angle θ can be given as $tan{-1}\ (y2-y1/x2-x1)$.

- Nostril Detection

Nostrils are relatively darker than the surrounding nose regions even under a wide range of lighting conditions. After nose is detected, the nostrils can be found by searching for two dark regions [7].

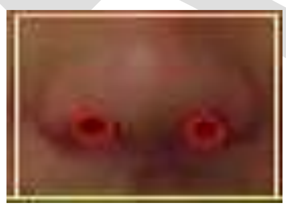

Figure 3.2.5: Estimated nostrils

The methods of obtaining two nostrils are given below:

1) Get the nose ROI after it is detected using haar-like cascaded features.
2) Extract gray scale image of the ROI.
3) Threshold the gray image. We have used a conventional thresholding method here.
4) Use morphological operations like erosion technique to remove the particles, small dis-joint parts etc.
5) Obtain rectangles bounding the contours. Find the centroid of both the rectangles.
6) The resultant centroids are the two nostrils.

- Mouth Detection

The proposed method of mouth detection uses a simple fundamental of facial geometry. From the facial geometry, we can easily observe that, the approximate width of the lips is same as the distance between two eyes centers. The mouth y-location starts after the nose tip. X-locations can be given as two eye centers x-locations. The height of the mouth is estimated at 3/4 of the nose height detected. The height can also be taken as equivalent to the height of the nose to avoid the elimination of lower lips edges, especially when a person is smiling [1].

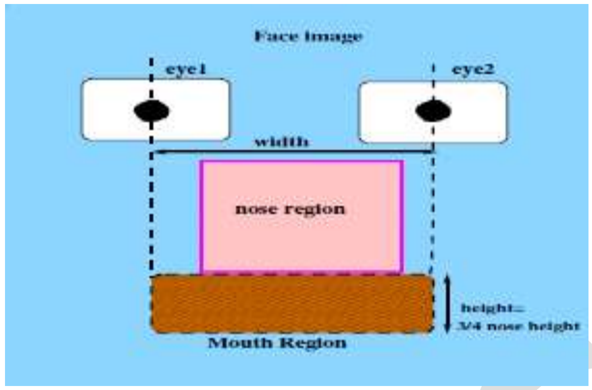

Figure 3.2.6: Estimated mouth region

- Mouth Corners Detection

We used Shi-Tomasi's corner detection method for detecting all the good/trackable features within the lip region. It is observed that, the threshold image of the mouth ROI contains a nice pattern from which the lip corner points can be detected easily [7].

Figure 3.2.7: Estimated mouth corners

The proposed method for detecting lips corners is given below:
1) Get the detected mouth ROI, which is done as stated in section E.
2) Extract gray-level image.
3) Threshold the gray-image. It gives nice edge information for a closed mouth and a contour region for an open mouth.
4) Using Shi-Tomasi's corner detection method, obtain all the corner points in the threshold image.
5) Considering the midpoint of the lip as $(xl0, yl0)$ obtain the difference $dx = xl0 - xli$ and $dy = yl0 - yli$. The obtained $-ve$ difference will give the points at the right and $+ve$ one will give those of are at the right.
6) Obtain the points at both extremes. Resultant points are the corners of the lips.

EXPERIMENTAL RESULT

Experiment is operated on more than 30 frontal face images of FEI database and also on few badly illuminated frontal face images. The database images used are of size 260 × 360. The experiment is also performed on different resolution images, like figure 1 shows the detection results of an image of size 640 × 480. The image is not well illuminated. It is observed that the algorithm is working well even in bad illuminated image.

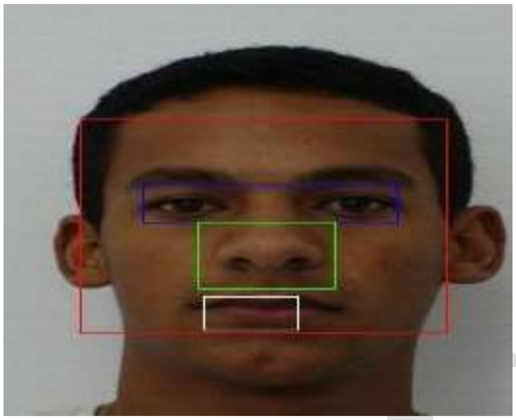

Figure 1: Region detection of bad lighting condition

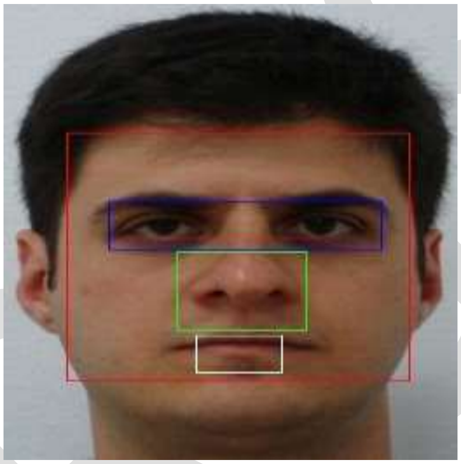

Figure 2: Region detection of good lighting condition

Figure 1 shows the experimental results of few frontal images from FEI database. The experimental result shows the accuracy of the methods used in the proposed work. Some results, for two different expressions (both on smiling and on neutral faces) are shown below. In both cases, it is found to be accurately detecting the features. We can observe in the figure 2 that, the eye pupils are getting detected even for very small eye images.

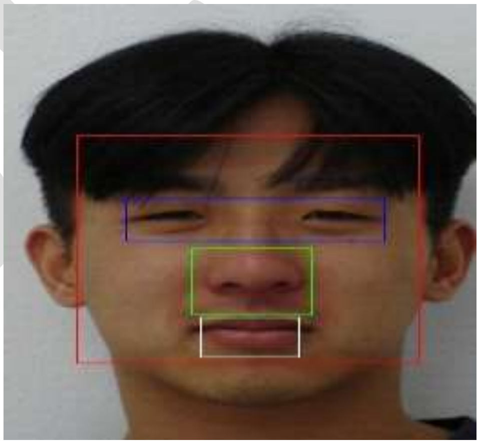

Figure 3: Face detection

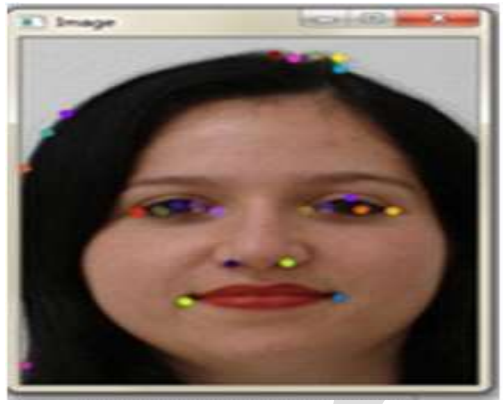

Figure 4: Corner detection

Some images of faces wearing spectacles are also found to be detecting eye features accurately. It is observed that even in presence of glass, eye feature detecting accurately. Fig 4 shows the Feature point extraction:

In both cases it is found to be accurately detecting the features. Finally Figure 5th is the image from the database.

Figure 5: Expression detection

Proposed system is tested over 30 frontal face images with two different facial expression (neutral face and smiling face).The results obtained are found to be 60% accurate for lip, lip corners, nose and nostrils detection. The eye corners are giving approximately 70% accurate results. The results obtained are found to be 85% accurate for expression detection (neutral and smiling face).

CONCLUSION

This paper focuses on the recognition of two major facial expressions: neutral and smiling. This work used a robust algorithm for automatic and accurate detection of different facial features. An improvement over AdaBoost detection of eyes, mouth and nose are done by estimating the probable region for each features. Geometrical interpretation of location of facial features, used in the algorithms is described with pictorial descriptions. It is observed that, with the use of facial geometry, the accuracy of features (eyes, nose and mouth) detection is greatly improved over that of using only AdaBoost algorithm in whole face image. The proposed lip detection algorithm is found to be accurately detecting the lips corners for both neutral face images and smiling face images.

REFERENCES:

[1] Anima Majumder, L. Behera and Venkatesh K Subramanian, "Automatic and Robust Detection of Facial Features in Frontal Face Images", IEEE 13th International Conference on Modeling and Simulation, 2011, pp. 331-336.

[2] P. Li, S. L. Phung, A. Bouzerdom, and F. H. C. Tivive, "Automatic Recognition of Smiling and Neutral Facial Expressions", IEEE Journal on Digital Image Computing: Techniques and Applications, 2010, pp. 581-586.

[3] M. Pantic and L. J. M. Rothkrantz. "Facial action recognition for facial expression analysis from static face images". IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol.34, no.3, 2004, pp. 1449–1461.

[4] B. Xiang and X. Cheng, "Eye detection based on improved ad adaboost algorithm," IEEE international conference on signal processing system (ICSPS), vol.2, 2010,pp.112-132.

[5] D. Vukadinovic and M. Pantic, "Fully automatic facial feature point detection using gabor feature based boosted classifiers," IEEE international conference on systems, man and cybernetics, vol. 2, 2006, pp. 1692– 1698.
[6] P. Viola and M. Jones, "Robust real-time object detection," international journal of computer vision, vol. 57, no. 2, 2002, pp. 137–154.
[7] C. Xu, Y. Zheng, and Z. Wang, "Semantic feature extraction for accurate eye corner detection," IEEE International Conference on Pattern Recognition (ICPR 2008), 2009, pp. 1–4.
[8] J. Shi and C. Tomasi, "Good features to track", IEEE Conference on Computer Vision and Pattern Recognition, 2002, pp. 593–600.
[9] C. Thomaz and G. Giraldi, "A new ranking method for principal components analysis and its application to face image analysis," Image and Vision Computing, vol. 28, no. 6, 2010, pp. 902–913.
[10] L. Ding and A. Martinez, "Precise detailed detection of faces and facial features,"IEEE international conference on Computer Vision and Pattern Recognition (CVPR 2008), 2008, pp. 1–7.
[11] Z. Zheng, J. Yang, M. Wang, and Y. Wang, "A Novel Method for Eye Features Extraction," Computational and Information Science, 2005, pp. 1002–1007.
[12] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba. "Coding facial expressions with gabor wavelets". IEEE 3rd International Conference on Automatic Face and Gesture Recognition, 1998, pp 200–205.