Detection of Face and Facial Features in digital Images and Video Frames

M. Beigzadeh¹, M. Vafadoost²

¹²Department of Biomedical Engineering, Amir Kabir University (Poly technique), Tehran, Iran
E-mail: mbeigzadeh@bme.aut.ac.ir, vmansur@cic.aut.ac.ir

Abstract - In the recent few decades, automatic detection and tracking of face and facial features such as eyes and mouth, in image and video sequences has become an active research area in machine vision applications such as Human-Computer Interaction (HCI). In this paper, a new algorithm for detection of face and facial features is proposed that can localize eyes and mouth very accurately in images. In this method, a combination of luminance, color and edge properties of image is used. This method is compared to the method introduced by Rein Lien Hsu, in which color and luminance information is used, and it is shown that the new algorithm is more robust and accurate in locating eyes and mouth in facial images with maximum 30 degrees of lateral rotation. Both methods are implemented and tested on a database containing 103 different images of face, and it is shown that the proposed method increases the accuracy by 4 percent and reaches to 91.26% of accuracy.

Keywords - Face Detection, Facial Features Detection, eye tracking, Eye localization.

I. INTRODUCTION

Face plays an essential role for human communication. It is the main source of information to discriminate and identify people, to interpret what has been said by lip reading, and to understand one’s emotion and intention based on the emotional facial expressions. Therefore, face and facial feature detection plays an important role in numerous applications including video surveillance, face tracking, face recognition, virtual reality, and intelligent human machine interface and security systems.

As the need for better ways of interaction with computers increases everyday, computer vision researchers are becoming more and more interested in finding face and facial features in images and video sequences. For people like patients with neuro-muscular disorders, who are not able to use their hands, the only way to communicate with the world is using their facial expression. So, designers of HCI devices try to use facial features to facilitate the use of computers for these people and help them to do tasks like typing by the use of their eyes. The most important facial components that are used for these purposes are eyebrows, eyes, nose, and mouth. They encode critical information about facial expression and head movement.

A vast amount of research is performed in the field of face and facial components detection, localization and tracking. Generally, we can classify these methods into two main groups: local and global methods. In the local methods, each face component such as eyes, mouth and nose is detected separately. But in the global methods, all facial parts are detected jointly and a model based on relative locations of theses parts, is introduced for face [1].

In another point of view, these surveys can be categorized into five classes. (1) Geometry-based methods: These methods utilize geometrical information. Each feature is demonstrated as a geometrical shape [2,3]. They can accurately detect face and facial features, but cannot handle large variations of the face images such as images with some occluded facial features and images with noise. (2) Color based approaches: These approaches have problems in robustly detecting skin colors in the presence of complex background and different illuminations [4]. These algorithms are applicable only for color images. (3) Appearance-based methods: These methods use the models learned from a set of training images. Intensity is the most important parameter for the detection. These approaches are not able to perfectly detect face images with poor quality in intensity, some occlusions, and unnatural intensities. (4) Motion-based methods: Face and facial features are detected from the image sequence. Using such methods, facial features cannot be detected using only a single still image from one view. (5) Edge-based methods: In the last class of methods, faces are detected without information due to intensity and motion. The edge information is used as input. These methods can handle large variations of the face images [4].

In this paper a method is introduced in which color, intensity and edge information are combined to accurately detect face and facial components, i.e. eyes and mouth, in digital images and video frames. Using edge information in this method, makes some refinement on the method introduced by Lien Hsu, and makes the results more robust and reliable. This method is not robust to wearing glasses, but different expressions do not affect it’s performance. In this research facial features detection is performed as a part of a system that helps disabled people to perform eye typing. The paper is configured as follows: First, in the details of proposed method is illustrated in section II. Then experimental results on a database containing 103 images is reported and compared to the reference method, in section III. Finally, we will have conclusion in section IV.

II. METHODOLOGY

In this paper a method for extracting face, eyes and mouth from digital images is introduced. An overview of the proposed face detection method is shown in the block diagram of Fig. 1. The method is composed of five main steps: 1) finding face candidates in the image based on color information, 2) detecting eye candidates based on color and intensity values, 3) verification of eye regions based on edge information, 4) extraction of mouth areas based on color and geometry relations, 5) finding best joint candidates for two eye segments and one mouth segment based on distances and geometrical locations. In this section, each step is illustrated briefly and the main ideas are introduced.

1) Extracting face candidates based on color information: as the first step, we need to know facial regions.
Some researchers have used edge information combined with template matching to find face candidates. But color is a rich source of useful information to find face and skin like regions. However, as mentioned in the introduction, color is always influenced by luminance and the brightness of the ambient. So we have to do some preprocessing to reduce the effect of luminance on color information. In this way, we can use our color model more reliably. For this purpose, we just have to find the reference white color of the image, and normalize all pixel values to the reference white color. To do so, we did the same operation as what was reported in [5]: pixels that have the most 5% of luma values are recognized, and their color is known as the reference white color.

Then, the RGB values are transformed to YCbCr and a nonlinear model for the skin tone color in this color space is used. In this nonlinear model, skin tone color values, are more compact and so clustering the pixels as skin/non-skin is more accurate and reliable. This skin tone color model has some advantages. For example because color information is separated from luminance in it, the model can distinguish a large variety of skin color tones, from light to dark skins. Results of luminance correction and skin model application to detect skin pixels are shown in Fig. 2.

2) Detection of eye candidates based on color and intensity values: The next step is to find eye candidates. Eyes are the most important features of face that have rich information useful for applications like gaze detection, face pose determination and etc. so we first try to find eye components based on color and intensity information in each facial candidate. As indicated in [5], we used two facts to determine eye candidates. 1) Cb component of color has a value greater than Cr in eye regions. This is the basis of defining Eye Color Map as shown in (1), in which $C_r$, $C_b$, $C_c$ values are normalized to $[0, 255]$. 2) Variation of gray level values is high in eye regions, i.e. The darkest (pupil) and the brightest (sclera) values exist in eyes. So we use luminance information to define a gray level map of eye, based on morphological operations of erosion and dilation as is seen in (2).

$$\text{EyeMap}C = \frac{1}{3} \left[ C_r^2 + C_b^2 + \left( C_c / C_r \right) \right]$$

$$Y = 0.299R + 0.587G + 0.114B$$

$$\text{EyeMap}Y = 10 \frac{Y_{\text{Dilation}}(x, y)}{1 + Y_{\text{Erosion}}(x, y)}$$

Based on these two formulas color and intensity eye maps are extracted. The logical AND operation is applied on two maps to reach final eye map. Then, some verification is applied on new eye map to omit eye regions that are in the lower part of the ellipse. Results are shown in Fig. 3.

Fig. 1. Block diagram of the proposed method.

Fig. 2. Results of face candidate detection. a) original image, b) luminance corrected, c) face candidate regions, d) ellipse fitting to face candidates.

Fig. 3. Eye Map extraction. a) Color Eye Map, b) Intensity Eye Map, c) logical AND operation on both maps.
3) **Verification of eye candidates based on edge information:** The key difference of proposed method to the approach introduced by Lien Hsu, is the use of edge information in verifying eye candidates. As shown in Fig. 4, we can see that edge value is a very informative quantity in localization of eyes. Intensity value changes repeatedly in eyes. In fact, because of going from bright (cornea) to dark (Iris and Pupil) regions and dark to bright ones, vertical edges have great values near the eyes. So, if we integrate the vertical edge values in the horizontal direction, a kind of edge histogram is produced. As shown in Fig. 4, this histogram has a distinguishable peak, around the eye regions. We use this fact to verify eye candidates that has been extracted from previous steps. We try to fit the peak of edge histogram to a function like Gaussian, and then we give a score to the amount of this fitness. These scores are used in the last step to choose the best eye candidates among all.

In this paper a very simple approach to calculate edge values is used. Other researchers calculate edge values using operators such as canny and sobel. But here, we just use a very simple differential operation to extract vertical edge values, more simply and rapidly.

4) **Extraction of mouth candidates based on color and geometry relations:** To detect face areas more reliably, we also use color information to locate mouth regions. To find mouth candidates, we use the fact that in pixels belonging to mouth, red color is dominant. So we see that Cr component of color has greater value than Cb. Based on this reality, Mouth Map is defined as (3), in which \( F_g \) is the face mask with n points, and \( C_r/C_b \), \( C_r^2 \) values are normalized to [0,255].

\[
MouthMap = C_r^2/C_b - \eta C_r/C_b,
\]

\[
\eta = \frac{1}{n \sum_{(x,y) \in F_g} C_r^2(x,y) / C_b(x,y)}
\]

5) **Giving suitable scores to eyes and mouth candidates jointly to select best ones:** Up to here, we have found some candidates for face regions, eyes and mouth. In the last step, we have to choose the best joint components mainly based on geometrical locations and their relative sizes and distances.
VI. CONCLUSION

Detection and tracking of face and facial components such as eyes, play an important role in many computer vision applications like Human Computer Interaction (HCI). In this paper a new approach is introduced which uses edge information, combined with color and intensity data, to locate eyes accurately in facial images. The idea of the method comes from the fact that near the eyes, variation in intensity values is high, and therefore edges are stronger there. We used this reality to help eye detection methods, and it showed to have reasonable performance on the database containing 103 different images of face.

REFERENCES


