Eye states detection from color facial image sequence

Hong LIU, Yuwen WU, Hongbin ZHA
National Lab. On Machine Perception, Peking University, 100871, Beijing, China
E-mail: liuhong@cis.pku.edu.cn wuyw@cis.pku.edu.cn zha@cis.pku.edu.cn

ABSTRACT

In this paper, an automatic and real time method for detecting eye states is proposed. The method is based on the fact that the regions of iris and white regions of an eyeball can be detected when it is open. In this case, color images contain more information helpful for detecting irises than intensity images. The saturation of color is used to detect whether the eye is open or closed, then the edge map of the eye image is used to detect the irises. Using the color information to detect the locations of irises is more accurate than using only the edge image. The image sequences are nearly frontal-view color facial image sequences. In the paper, some eye sequences, which are detected by this method, are shown.

Keywords: Hough Transform, Eye Tracking, Eye States, Edge Detect, Color Indexing.

1. INTRODUCTION

Eyes are the most important features of human face. There are many applications of the robust eye states extraction. For example, the eye states provide important information for recognizing facial expression and human-computer interface systems. When man is laughing, his eyes are nearly closed. And when he is surprised, his eyes are opened wide. Recently, many researches are actively conducted to build human-computer interface systems of high performance. An automatic and robust technique to extract the eye states from input images is very important in this field.

The eye states can be got from the eye features such as the inner corner of eye, the outer corner of eye, iris, eyelid, eye position and so on. There are many methods to detect eye features. Yuille, Hallinan and Cohen [1] used deformable templates to locate eye features. Deng and Lai [2] used improved deformable templates to locate eye features. The deformable templates can extract above eye features, so we can detect eye states using it. But, because the deformable templates base on minimizing energy, it will take much time to detect eye states. When Tian, Kanade and Cohn [3] adopt action units to recognized facial expression, they detected eye states by iris. Their approach could distinguish two eye states, open and closed. The iris is normally visible when eye is open and is absent when eye is closed. So they only used the edge map (without any color information) to detect eye states. Bernogger, Lujun Yin, Basu and Pinz [4] presented an approach to synthesize the eye movement by using the extracted eye features to compute the deformation of the eyes of the 3D model. They used the deformable templates to detect eye features. When they constructed the energy function the color information was used.

In this paper, a method is presented to detect the eye states in nearly frontal-view color face image sequence by using the color information of eye color images and the edge information of eye images. Our purpose is to detect eye states, so the color image sequences are nearly frontal-view color face image sequences and in the sequences the face does not move or moves little in the face plane.

This paper is organized as follows: In Section 2, the approach proposed for find the position of eye is described. In Section 3, the method of detecting eye states is described. Then some experimental results are shown in Section 4. Finally, the conclusions are given in Section 5.

2. LOCALIZING EYE

Before the eye states are detected, the eye image must be got. To detect the eye states, the eye position must be localized in a nearly frontal-view color face image firstly. There are a lot of approaches that can be used to localize the eye.

Object can be recognized on the basis of their color alone by color indexing, a technique developed by Swain and Ballard [5]. Human skin colors dose not fall randomly in color space, but cluster in a small region of color space. Human skin information is much affected in intensity than in colors. Under a certain lighting condition, a skin-color distribution can be characterized by a multivariate normal distribution in the normalized color space [6]. Human skin colors have these features, so we can use the face color to localize eye. Firstly, the color filter or the color segmentation can be use to get skin-segmented face image. After detecting the face region, two coarse regions of interest in the upper left and upper right half of face image can be defined to detect the eyes. Secondly, the holes in the skin-segmented image and the facial symmetry can be used to localize eye [7].

To detect eye states, a simple approach is used to find eye position. In the first image of a color image sequence, the eye features are marked by hand. The eye inner corners are the most stable features in a face and relatively insensitive to facial expressions. So, if the eye inner corners are localized and the length and width of the eye is known, the eye can be localized. Firstly, two inner corners and the eye's length are marked. Then in the following images, the eye inner corners are tracked by SSD [8]. We assume that intensity values of any given region (feature window size) do not change but merely shift from one position to another. Consider an intensity feature template $R_t(i,j)$ over a $n \times n$ region I in the reference image at time t. We wish to find the translation $d = (m,n)$ of this region in the following frame $R_{t+1}(i+m, j+n)$ at time $t+1$, by minimizing a cost function $SSD(i,j)$, which is defined as:

$$SSD(i,j) = \sum_{(i,j)\in I} \left[R_{t+1}(i+m, j+n) - R_t(i,j)\right]^2 . \tag{1}$$

Here, our goal is to determine the position at which the reference template best matches the image. Thus $SSD(i,j)$ must be computed over a search region of possible image locations. If both the template and the image neighborhood are suitably normalized, the SSD of a template and an image neighborhood could be replaced by cross-correlation. The most direct normalization is to divide each inner product by the square root of the energy of the template and the neighborhood. Mathematically, it can be expressed as:

$$NCC(i,j) = \frac{\sum_{m,n} R_{t+1}(i+m, j+n) R_t(m,n)}{\sqrt{\left(\sum_{m,n} R_t(m,n)\right)^2 \left(\sum_{m,n} R_{t+1}(i+m, j+n)\right)^2}} . \tag{2}$$

The region of possible eye inner corners locations is searched by minimizing the $NCC(i,j)$.

We assume the outer corners of two eyes are collinear with the inner corners. After localizing two eye inner corners, the length of eye, which was marked in first frame, is used to localize the outer corners. For the positions of inner corners and outer corners are known, the ranges of two eyes are known.

3. EYE STATES DETECTION

3.1 Eye states

There are two default eye states: open and closed. Before detecting the eye states, what open eye is must be defined. If a man can see something, it is considered that his eyes are open. But we cannot take this condition as the computer criterion. Our criterion is that if the iris and the white of eye can be saw, the eye is open otherwise is closed. Different states of eye are shown in Fig.1.

(a) (b) (c) (d)

Fig.1. Different states of an eye (a: open. b: open. c: closed. d: closed)

It is difficult to detect eye states when eye is nearly closed (c of Fig.1). The difficulty come from that the eye is actually open, but the distance between two eyelids is too small to detect. The nearly closed eye is regarded as closed and it is adoptable in many practical applications.

3.2 Using the color information

To detect eye states, the color information of eye color image is used firstly. Color is a property of enormous importance to human visual perception, but it is hardly used in eye states detection. To use the color information, a color space must be selected. The RGB color space is not less helpful than intensity space, because any part of eye has not remarkable features in red, green or blue color. For this reason, another color space of HSI color space is selected. Hue(H) refers to the perceived color, and saturation(S) measures color's dilution by white light, and intensity(I) shows the intensity of color. If the color blends more white color, its saturation is smaller. The white of eye blends lots of white color, so its saturation is smaller than other parts of eye. In some cases, light reflected by pupil becomes a high brightness point. This point's saturation is small too. So if the eye is open, the white of eye and the point (if exist) are visible. We can detect whether eye is open by the saturation of eye color image. Some eye color images and their saturation images are shown in Fig.2.

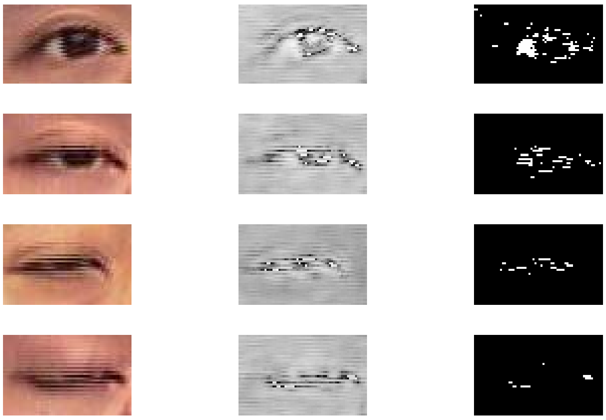
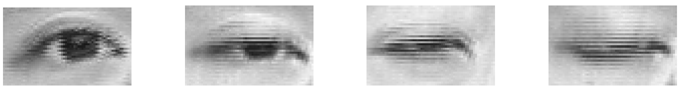


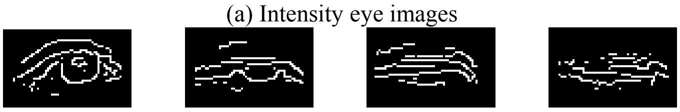
Fig.2 Column 1: eye color image. Column 2: eye saturation image.
Column 3: white-black image of saturation image.

The saturation of skin color is higher than any part of eye. From the column 2 of Fig.2, we can find it. If the saturation of the pixel in saturation image is lower than the threshold value, the pixel gray is set 255. Using this method the white-black image of saturation can be got (column 3 of Fig.2). From Fig.2, it is known that different eye states have different numbers of pixel which saturation is lower than the threshold value. There is a rule that wider the eye opens; more are pixels which saturation is lower than the threshold value. This rule is used to detect eye states. In our system, the threshold value is 40.

If the eye is closed, the iris will not be visible. But if the eye is open, part of the iris will normally be visible. So the iris can provide important information for the eye states. Generally, the pupil is the darkest area in the eye region. Some eye trackers locate the iris center by searching for the darkest area in each frame. However, the iris center often shifts to the eye corner, dark eyelash, or eyelids with eye shadow. In our system, after detecting eye states by saturation of color image, we use the edge map to detect eye states again. Using two eye states detection methods together can get better accuracy.



(a) Intensity eye images



(b) Eye edge maps.

Fig.3 Eye edge maps for eyes from wide open to closed. The edge of lower part of the iris is relative clear for an open eye. The upper eyelid often occludes the edge of upper part of the iris.

3.3 Using the edge information

To use the edge information, the edge maps must be got firstly. There are many methods to get it. Here the Canny edge operator[9] is used to get the eye edge image. The edge detection results for several different open eye states are show in Fig.3. The edge maps are very noisy even in a clear eye image, so the edge maps cannot be used directly. We observed that the iris edge is relative clear, and the upper eyelid often occludes the upper part of the iris. So, we use the Hough transform to detect part of iris circle.

To get the edge maps of eyes, the eye color image is transformed into gray image firstly. In this step, the eye states are detected by the edge of iris, so it hopes that the result of Canny operator is the iris edge only. To do this, the iris should be striking contrast with white of eye and skin. Fig.4 shows the Canny operator results in different contrasts between iris and other parts. We know from Fig.4 that the higher contrast the image has, the better result the edge image is. From the histogram of color image in Fig.5, it is true that in red image the distribution of pixels is more even than other histograms. The red image has more high contrast with the white of eye and the skin than other images show, in Fig.6. So the red image is regarded as the intensity image of color image to get edge map.

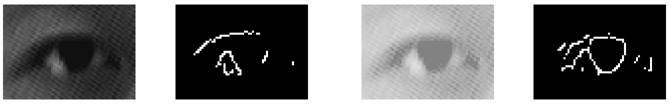
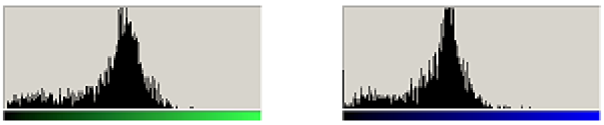


Fig.4 Eye image and its edge map in different contrast

(a) (b)



(c) (d)

Fig.5 Histogram of eye color image (Intensity image histogram, Red Image histogram, Green image histogram, Blue image histogram).

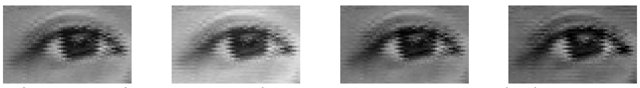


Fig.6 Intensity Image, Red Image, Green Image and Blue Image.

After getting the edge map of eye, the edge information is used to detect iris by Hough transform. Detecting iris is a problem of finding the iris within an eye image. One of many possible ways to solve this problem is to move a mask with an appropriate shape and size along the image and look for correlation between the image and the mask[10]. Detecting iris can be solved successfully by Hough transform[11]. Hough transform is a very powerful technique for curve detection. Exponential growth of the accumulator data structure with the increase of the number of curve parameters restricts its practical usability to curves with few parameters. If the prior information about radius of iris is used, computational demands can be decreased significantly. Because our image sequences are nearly front-view face sequences, the radius of iris can be taken as a constant. If the radius of iris is known, the computational complexity of Hough transform will be decreased. In the first frame, the radius of iris is not known, so the Hough transform with three parameters is used to detect iris and get the radius of it. In the follow frames, the iris radius is known and the computational complexity is decreased. There is a more effective method for getting the radius by Hough transform[12]. Fig.7 shows some results of using three parameters to detect iris and get iris radius. In the images of column 3, the white crosses are marking the center of iris. We can find from Fig.7 that the wider the eye open, the better result we can get. So in our first frame of image sequence, the face is neutral face, then the eye is in normal state as the first image showing in Fig.7.

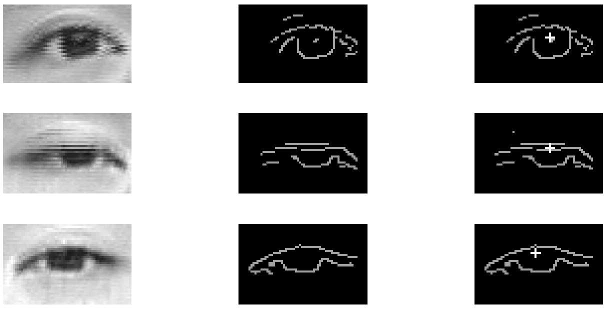


Fig.7 Column 1: the eye intensity image, Column 2: results of Canny operator on eye intensity image, Column 3: using the Hough transform with three parameters on eye edge image.

In the follow frames of an image sequence, the radius getting in the first frame is used to detect the iris. So the two parameters Hough transform can be used to detect iris.

4. EXPERIMENTAL RESULTS

To examine the algorithms, experiments with different color images of different face image sequences were collected. In the face image sequences, the images are for nearly frontal-view faces and the faces have not movement or a little movement in face plane. In Fig.8 some experiment results are shown. Because these are some images that the eyes are nearly closed state, these are some inaccuracy results. Our testing images are 300*400 facial color images. This method operates continuously at 20 images per second on a 1700 megahertz PC. Large parts of closed eyes are rejected by the color information, so this method is more accuracy than other approaches. For example, if an eye is closed and the template is used to detect eye states, for the edges of eyelids, its result is false in some case. But, if the eye is closed and it is known, this false result is not appearance. Our method's advantage is that it can detect almost all closed eyes before edge information is used.

Fig.8 Results of detecting eye states. When the eye is open, the iris is marked by red circle in the color image sequences. It can be found that if the eye is nearly closed, the eye state is detected as closed.

5. CONCLUSION

In this paper we propose an automatic and real time detecting eye states method. The particular feature of this method is using the color information and edge information together to detect eye states. Using the color information and the edge information, the results of detecting eye states are better than using edge information only, although this method has some limitations. For the color saturation has large difference in different illuminations, this method is inaccuracy when the lighting conditions are changed. It is take some time that the Hough transform with two parameters is used to get radius of iris in the first frame of an image sequences. In our feature work, we will use more information of eye to get better results.

ACKNOWLEDGEMENTS

This work was supported by National Natural Science Foundation of China (NSFC). Project No: 60175025, P. R. China.

REFERENCES

1. A. Yuille, P. Hallinan, and D. Cohen, Feature Extraction from Faces Using Deformable Templates, International Journal of Computer Vision, 8(2):99-111, 1992.
2. J. Deng and F. Lai, Region-Based Template Deformation and Masking for Eye-feature Extraction and Description, Pattern Recognition, 30(3):403-419, Mar 1997.
3. Y.L Tian, T Kanade, J.F Cohn, Recognizing Action Units for Facial Expression Analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 23(2):97-114 February 2001.
4. S. Bernogger, L. Yin, A. Basu and A. Pinz, Eye Tracking and Animation for MPEG-4 Coding, IEEE Pattern Recognition, 1998. Proceedings, Fourteenth International Conference on Volume:2:1281-1284, 1998
5. M.J. Swain and D.H. Ballard, Color indexing, International Journal of Computer Vision 7(1):11-32 1991.
6. S.A. Shafer, Optical Phenomena in Computer Vision, Proc. Canadian Soc. Computational Studies of Intelligence 572-577, 1984.
7. T.C. Chang, T.S. Huang, C. Novak, Facial feature Extraction from Color, Pattern Recognition, 1994. Vol.2-Conference B: Computer Vision & Image Processing, Proceedings of the 12^{th} IAPR International Conference on, Vol:2, 1994.
8. M.J. Crowley, J.L. Karlsruhe, Experimental Comparison of Correlation Techniques, IAS4, International Conference on Intelligent Autonomous Systems.
9. J. Canny, A Computational Approach to Edge Detection, IEEE Trans. Pattern Analysis Mach. Intell. 8(6), 1986.
10. Y. Tian, T. Kanade , and J. Cohn, Dual-State Parametric Eye Tracking, International Conference on Face and Gesture Recognition, 110-115, 1999.
11. T.H. Hong, Image Smoothing and Segmentation by Multiresolution Pixel Linking Further Experiments, IEEE Transactions on Systems, Man and Cybernetics, 12(5): 611-622,1982.
12. E. Davies, A modified Hough Scheme for General Circle Location, Pattern Recognition. 26 (12): 1739-1755, Dec. 1993.