# Eye Corner Detection with Texture Image Fusion

Zhiyu Zhang[1], Yang Shen[1], Weiyao Lin[1*], Bing Zhou[2]

[1]Department of Electronic Engineering, Shanghai Jiao Tong University, China ([*]Corresponding author)
[2]School of Information Engineering, Zhengzhou University, China

***Abstract*—The localization of eye corner is of great importance since it offers crucial information in various face-related applications including face tracking, gaze estimation, and facial expression recognition. In this paper, a new approach is proposed which localizes eye corners in a precise and robust way. In our approach, we first estimate a rough location about an eye corner. Then, a set of texture images are derived from regions around the estimated eye corner location. These regions are then fused together to decide the reliability of the estimated eye corner location, and an eye corner's location will be re-estimated if the current estimated eye corner is evaluated as unreliable. Finally, a local refinement process is also applied which refines the location of the estimated eye corner to create a more satisfactory result. Experiments show that our approach can achieve better detection results than the existing methods.[1]**

## I. INTRODUCTION

Eye corner information is fundamental to many face-related applications including face tracking, gaze estimation, and facial expression recognition [1,2,3]. In many applications, the precision of eye corner location is crucial to their performances. For example, many gaze estimation approaches depend on the location of eye corners to estimate human gazes [2]. In these methods, one pixel deviation in eye corner location may make the estimated gaze far from the desired result. Besides, in facial expression cloning applications, the precision of eye corner localization is also important in creating a satisfactory facial expression transfer result.

Since many researches have been done on facial key point localization [4-6], eye corners can be directly localized by these methods as a byproduct. Shi *et al.* utilizes an active shape model (ASM) to detect key points in a face [4]. Belhumeur *et al.* [5] proposed a searching window-based detection method, which scans the whole image, to locate dozens of key points in the face. Cao *et al*. [6] introduces a framework to directly predict key points in a face using a trained vector function. However, these methods focus on the overall localization for all facial key points, while the localization accuracy of an individual key point (such as eye corner points in our paper) is not always reliable.

There are only a few works which are specifically designed to find eye corner locations. Santos *et al*. [7] introduced human eye structure to localize iris and sclera area of an eye region. Xu *et al.* [8] extracted a set of candidates using Harris' detection and utilized semantic features to decide the most appropriate one as eye corner. However, the existing methods still suffer from variations of light conditions or user changes. Besides, their eye corner localization results are also easily affected by the movement of iris. For example, when an iris moves close to an eye corner, the dark color of the iris will affect the texture around the eye corner, making the estimated eye corner location deviated.

In this paper, we propose a new approach to find eye corner locations based on texture fusion. We first estimate a rough location about an eye corner. Then, a set of texture images are derived from regions around the estimated eye corner location. These regions are then fused together to decide the reliability of the estimated eye corner location, and an eye corner's location will be re-estimated if the current estimated eye corner is evaluated as unreliable. Finally, a local refinement process is also applied which refines the location of the estimated eye corner to create a more satisfactory result.

In summary, the contributions of our approach are two folds: First, we propose to derive multiple texture information around a candidate eye corner region, and decide the reliability of an eye corner location by fusing these texture information. Second, we introduce a template-based step to refine eye corner location, so as to achieve more accurate eye corner locations.

The rest of this paper is organized as follows. Section II describes the framework of our approach. Section III describes the details of our approach. Section IV shows the experimental results and Section V concludes the paper.

## II. FRAMEWORK OF THE APPROACH

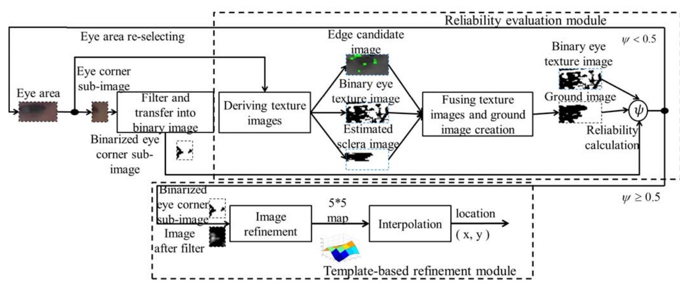



Fig. 1. The framework of eye corner detection algorithm (best viewed in color)

Fig. 1 shows the framework of the proposed approach. In Fig.1, we first estimate a rough location about an eye corner. In this paper, ASM model [9] is applied to estimate the rough eye corner location. Then, we obtain a sub-image around the estimated eye corner, and filter this sub-image to highlight information about eye corners. This filtered sub-image is further binarized to obtain a binarized eye corner sub-image.

After that, the eye corner sub-image and eye area image are input into the "reliability evaluation" model to decided the reliability of the estimated eye corner location included in the eye corner sub-image. In this module, we first derive three texture images from the eye image (i.e., edge candidate image, binary eye texture image, and estimated sclera image in Fig. 1). These three texture images reflect characteristics of a candidate eye corner sub-image from different aspects. Based on these texture images, we further fuse them into a ground image to

[1] This paper is supported in part by NSFC grants 61471235 and 61379079.

capture the overall layout around a candidate eye corner. And this ground image is correlated with the binarized eye corner sub-image and the binary eye texture image to decide the reliability of the estimated eye corner location. If the current estimated eye corner is evaluated as unreliable, an eye corner's location will be re-estimated and the above process will be repeated. In our paper, we re-estimate the eye corner location by shifting the previously estimated eye corner location to a new location in the neighborhood.

Finally, a "template-based refinement" module is also applied which first introduces a template to reduce noisy effects, and then utilizes a sub-pixel interpolation process to find the precise eye corner location.

In the following, we will describe details of our approach. We will focus on describing the process of localizing the inner corner of right eye in this paper, and the localization of other eye corners can be easily extended according to our description. Note that the "reliability evaluation" module and the "template-based refinement" module are our key contributions.

## III. The Eye Corner Localization Approach

### A. Creating binarized eye corner sub-image

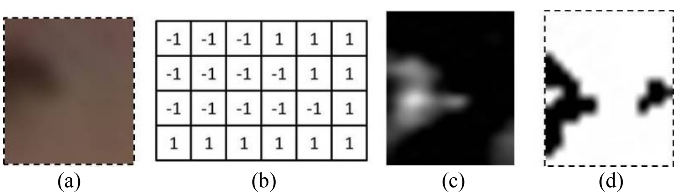



Fig. 2. (a) Eye corner sub-image. (b) The eye-corner-shaped kernel used in the filter. (c) The image after filtering. (d) Binarized eye corner sub-image.

Given an eye corner sub-image (cf. Fig. 2 (a)), we first apply an eye-corner-shaped kernel (cf. Fig. 2 (b)) [10] to filter the input sub-image. In this way, pixels related to eye corner can be properly highlighted (cf. Fig. 2 (c)). Finally, we compare the filtering result with a threshold, and achieve a binarized eye corner sub-image, as in Fig. 2 (d). We can see from Fig. 2 (d) that the pixels about an eye corner can be properly highlighted in the binarized sub-image (black pixels). However, due to the disturbance of noise, there still exists noisy regions in the binarized sub-image. Moreover, when the roughly estimated eye corner location is not reliable, the input eye corner sub-image may not include eye corner regions. Therefore, further steps are required to determine the reliability of an eye corner sub-image and to reduce the effect of noise.

### B. Deriving texture images

In order to decide reliability of an eye corner sub-image, we first derive three texture images (i.e., edge candidate image, binary eye texture image, and estimated sclera image) to depict the characteristics of a candidate eye corner sub-image. These images are derived by the following steps.

**(1) Edge candidate image.** The edge candidate image is estimated to eliminate irrelevant connected areas in binary eye texture image. Fig. 3 (a) shows an actual structure of a human eye where the red point is the eye corner constructed by two eyelids. Since the values of pixels are different inside the eyes and the outside skin, the gradient directions along the eyelids are quite special when compared with other parts of the skin. Therefore, we apply a 3×3 sliding window to scan the area of the eye area image. In each sliding window, we calculate the standard deviation of the gradient directions of nine pixels to replace the gradient value of the center, as shown in Fig. 3 (b). Since the standard deviation should become higher when the sliding window is closer to an eye corner, by applying an adaptive thresholding strategy to binary the standard deviation result, candidate eye corner regions can be achieved. The green regions in Fig. 3 (c) shows an example of the edge candidate image. Comparing Fig. 3 (c) with Fig. 2 (d), we can see that the irrelevant regions (such as the small black region in Fig. 2 (d)) can be properly eliminated in the edge candidate image.

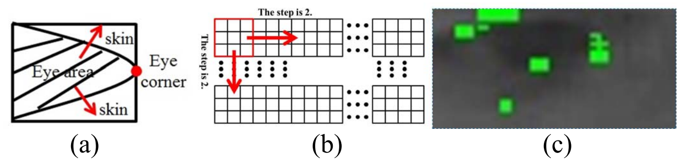



Fig. 3. Process to obtain the edge candidate image. (a) Model of inner eye corner. (b) The scanning map. (c) The edge candidate image (The green area is the black part in the binary image) . (best viewed in color)

**(2) Binary eye texture image.** The binary eye texture image can be easily obtained by binarizing grey levels of the filtered eye image using the same eye-corner-shaped kernel (as shown in Fig. 2(b)). Fig. 4 shows one example of the binary eye texture image. We can see that since the binary eye texture image is obtained by dealing with the whole eye area, it can contain more texture information about an eye region. And this information can be used to calculate the reliability in the reliability evaluation module.

**(3) Estimated sclera image.** The estimated sclera image highlights the rough sclera region in an eye image. Since sclera is highly related to an eye corner (e.g., many eye corners are located next to a sclera region), this estimated sclera image can be used to help inferring eye corner locations. In this paper, we use similar ways as [7] to achieve the estimated sclera image. More specifically, we first transfer an eye corner sub-image from RGB channel into HSV channel. Then, we extract the S channel and use adaptive-thresholding to binary the image after histogram equalization. Fig. 4 shows an example of the obtained estimated sclera image.

### C. Fusing texture images to create ground image

After obtaining three texture images, we fuse these images into a ground image to capture the overall layout around a candidate eye corner. The process of obtaining a ground image is illustrated by Fig.4.

Firstly, we use the edge candidate image to eliminate the irrelevant area. That is, regions not including candidate eye corner pixels are eliminated when creating a ground image, as the regions on the right part of the red dashed line in Fig. 4. Then, we fuse the estimated sclera image with the binary eye texture image, and create the resulting ground image.

An example ground image is shown in Fig. 4. From Fig. 4, we can see that the ground image contains eye texture layout information on one side of a possible eye corner, while textures on the other side are properly deleted. Thus, by combining the ground image with the binarized image (cf. Fig. 2d)), the reliability of the estimated eye corner location can be determined.

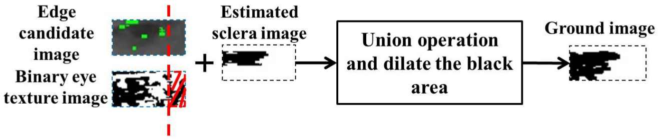



Fig.4 The process to get the ground image (best viewed in color)

### *D. Determining reliability of the estimated eye corner location*

With the obtained ground image (cf. Fig. 4), the binarized sub-image (cf. Fig. 2(d)) and the binary eye texture image, we are able to determine the reliability of the estimated eye corner. Basically, these three images can be viewed as two different descriptions of an eye corner region, where the ground image includes the eye texture layout on one side of a possible eye corner, while the binarized sub-image and the binary eye texture image include textures which have high response to a eye-corner-shaped kernel (i.e., possible eye corner regions). Therefore, the reliability of an eye corner location can be properly evaluated by combing these images.

In our approach, we first extract the two largest connected regions in the binarized sub-image and calculate their coherency with the binary eye texture image to find their related regions, and then we calculate the found related regions with the ground image, the coherency calculation equations are shown as in Eq. (1):

$$\psi(A_{area}, B_{area}) = (num(A_{area}, B_{area}) / Size(B_{area})) \tag{1}$$

where $A_{area}$ and $B_{area}$ are the sets of connected regions in the two separate images. *Size*( ) is the size of a connected region. $num(A_{area}, B_{area})=Size(A_{area} \cap B_{area})$ is the size of the intersection set between regions $A_{area}$ and $B_{area}$.

The entire process is illustrated by Fig. 5. According to Fig. 5, we calculate the coherency between two biggest connected regions in the binarized eye corner sub-image and connected regions in the binary eye texture image and between two connected regions picked in the process above and connected regions in the ground image. Since the largest connected regions in the binarized sub-image indicate the most possible eye corner regions, if its related connected region in the binary eye texture image has high coherency with the eye texture layout in the ground image (i.e., $\psi$>0.5 in our paper), we can infer that it is a reliable region for eye corner. Otherwise, it may be a noisy region since it is not coherent with the eye texture layout. Moreover, if none of the related regions in the binary eye texture image has high coherency with the ground image, we will decide the estimated eye corner location is not reliable. Thus, the eye corner location will be re-estimated to create a new eye corner sub-image whose reliability is re-determined. This process will be repeated until a reliable eye corner region is obtained.

Furthermore, there may be cases where both of the two largest connected regions are determined as reliable eye corner regions. In order to handle this ambiguity, we further propose to evaluate the eye angle of the largest connected regions (cf. Fig. 6). That is, we check the angle between two lines, the first line links the rightmost point of largest connected region and the iris center point (the iris center point can be localized by the facial key point extraction methods [11]). The other line links the iris center points of both eyes. Intuitively, the eye corner-iris center line should have similar angle to the iris-iris line. Therefore, by checking the line angle of the largest connected regions, we can determine the best eye corner region.

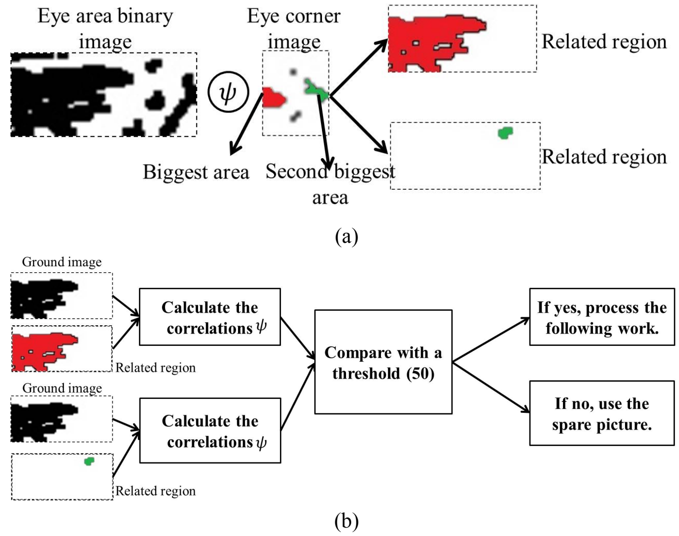



Fig.5 Reliability evaluation. (a) The coherency between two biggest connected regions in the binarized eye corner sub-image and those in the binary eye texture image. (b) The coherency between two connected regions picked in the process above and those in the ground image (best viewed in color).

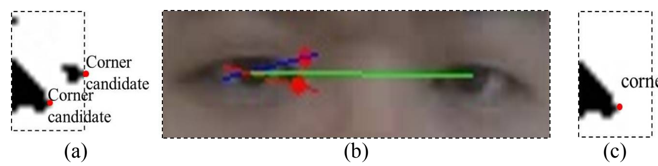



Fig. 6 (a) The original eye corner area. (b) Angle evaluation. (Green line is iris to iris, the blue line and the red line are two candidate eye corners to iris, separately). (c) The revised eye corner area after picking up the right connected area. (best viewed in color)

### *E. Refining eye corner location by template*

After obtaining a reliable eye corner region, a refinement process will be applied to further filter noises in the region. For example, in Fig. 7 (a), the obtained eye corner region includes noisy pixels. If we directly take the rightmost black pixel as the eye corner, the result will become less accurate, as in Fig. 7 (d).

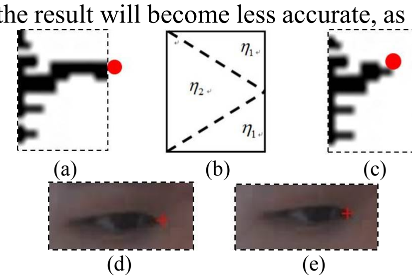



Fig. 7 (a) The binary image before revising (red point is the original output). (b) The distribution of the parameters. (c) The binary image after the revising.(red point is the output after revising). (d) The unrevised image. (e) The revised image. (best viewed in color)

In order to eliminate the above situation, we update the pixel value in the kernel-filter image (cf. Fig. 2 (c)) by multiplying it with a template. Suppose the useful information about an eye should be contained in a triangle area (cf. Fig. 7 (b)), we use a template to weight pixels into different importance, where pixels on the left and right side of the triangle boundary multiplied by a different weight as shown in Fig. 7 (b) (in our paper, we set pixels on the right-side region by $\eta_1$=0.1 and pixels on the left-side region by $\eta_2$=1.2).

After the refining with the template, the binary image is shown as Fig. 7 (c). And Fig. 7 (e) shows the eye corner localization result based on Fig. 7 (c). This is obvious that the eye corner location after refinement is improved.

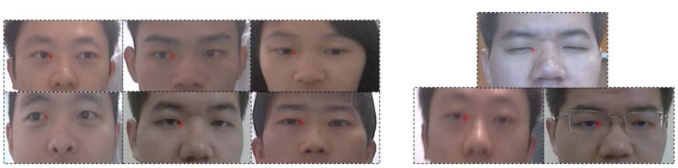
### *F. Refining eye corner location by sub-pixel interpolation*

Finally, we also apply a sub-pixel refinement process to improve the localization precision of integer pixels. Similar to [12], we extract a 5×5 window whose center is the rightmost pixel of the extracted eye corner region, then interpolate sub-pixel values among integer ones in this 5×5 window and find the refined eye corner location with the maximum value in the interpolated window.

## IV. EXPERIMENTAL RESULTS

In this section, we will show experimental results of our proposed approach. We construct a dataset which contains 7 videos of 6 subjects under different lighting conditions. Overall, there are more than 5000 frames in our dataset (some example frames are shown in Fig. 8). Besides, the subjects are asked to turn their iris or blink their eyes freely and frequently during the test, so as to test the approach's robustness to handle different eye conditions. Moreover, some challenging situations, such as users close their eyes, moving quickly, or wearing glasses, are also included in our dataset (cf. Fig. 8 (b)).

We compare our approach with two state-of-the-art facial key point detection methods (Wei's [13] and Saragih's [14]). All methods are performed on a PC with 4-core CPU and 4G RAM.



(a) (b)

Fig. 8 Our dataset. (a) Six different users. (b) Challenging frames. (best viewed in color)

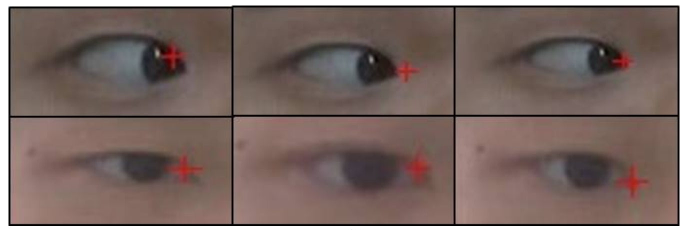
We compare the average distance between the eye corner localization results and the ground truth eye corner locations that we labeled. The results are shown in Table 1. Some eye corner detection results by our approach are shown by the red dots in Fig. 8. Furthermore, Fig. 9 compares some eye corner detection results of different methods.

Table 1 The average distance between eye corner localization results and the ground truth eye corner locations in our dataset

| Video | Frame number | Our approach | Wei's method | Saragih's method |
|---|---|---|---|---|
| 1 | 1018 | 1.63 | 3.28 | 6.21 |
| 2 | 770 | 3.24 | 4.40 | 4.08 |
| 3 | 440 | 3.19 | 3.77 | 3.49 |
| 4 | 920 | 3.88 | 4.23 | 3.85 |
| 5 | 540 | 3.53 | 3.70 | 3.89 |
| 6 | 990 | 2.96 | 4.11 | 4.95 |
| 7 | 460 | 3.81 | 5.05 | 4.38 |

From Table 1, we can see that our approach can achieve better results than the compared methods. Besides, Fig. 8 indicate that our approach can work reliably under different conditions and for different objects.

Although Wei's and the Saragih's methods can provide rough location of an eye corner, their precisions are less satisfactory. For example, in Fig. 9, when a person's iris is moved next to the eye corner, the eye corner localization results of both methods are obviously deviated from the ground truth locations. When dealing with the normal situations, (i.e., the lower three images in Fig.9), the Wei's method and Saragih's method still have error of several pixels, however, our method achieve better performance. Comparatively, by suitably fusing multiple texture information, our approach can still achieve reliable eye corner locations.



(a) (b) (c)

Fig. 9 Result comparison. (a) The result of Wei's method. (b) The result of Saragih's method. (c) The result of our method. (best viewed in color)

## V. CONCLUSION

In this paper, we propose a new approach for eye corner localization. The proposed approach derives a set of texture images and fuses information from them to decide the reliability of an estimated eye corner location. Then, a local refinement process is also applied which refines the location of the estimated eye corner to create a more satisfactory result. Experiments demonstrate the effectiveness of our approach.

## REFERENCES


[1] Mingcai Zhou, Lin Liang, Jian Sun, Yangsheng Wang. AAM based face tracking with temporal matching and face segmentation. CVPR, 2010 .

[2] Li Sun, Mingli Song, Zicheng Liu, Ming-Ting Sun. Realtime gaze estimation with online calibration. ICME, 2014.

[3] Yu Zheng-Hong, Li Cong. Research of Facial Expression Recognition Based on ASM Model and RS-SVM. ISDEA, 2014.

[4] Shi Yi-Bin, Zhang Jian-Ming, Tian Jian-Hua, Zhou Geng-Tao. An improved facial feature localization method based on ASM. CAIDCD '06.

[5] P.N. Belhumeur, D.W. Jacobs, D.J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. CVPR, 2011.

[6] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. CVPR, 2012.

[7] Santos, G. and Proenca, H. . A robust eye-corner detection method for real-world data. IJCB, 2011..

[8] C. Xu, Y. Zheng, and Z. Wang. Semantic feature extraction for accurate eye corner detection. ICPR 2008

[9] T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active shape models: Their training and application. Computer Vision and Image Understanding, 1995.

[10] Jie Zhu, Jie Yang. Subpixel eye gaze tracking. IEEE International Conference on Automatic Face and Gesture Recognition, 2002.

[11] F Timm, E Barth. Accurate eye center localisation by means of gradients. VISAPP, 2011

[12] R.G. Keys, Cubic convolution interpolation for digital image processing. IEEE Trans. Acoustics, Speech, and Signal Processing, 29(6):1153-1160, December 1981.

[13] Y. Wei, Research on Facial Expression Recognition and Synthesis. Master Thesis, Department of Computer Science and Technology, Nanjing University.

[14] J. M. Saragih, S. Lucey, and J. F. Cohn. Face Alignment through Subspace Constrained Mean-Shifts. ICCV, 2009.