Eye-Gaze Tracking System By Haar Cascade Classifier

Yunyang Li [1], Xin Xu [1,2,], Nan Mu[1], and Li Chen [1,2]*

[1]School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan, China, 430081.
[2]Hubei Province Key Laboratory of Intelligent Information Processing and Real-time Industrial System, Wuhan University of Science and Technology, Wuhan, China, 430081.

Abstract—**Human can quickly and effortlessly focus on a few most interesting points in an image. Different observers tend to have the same fixations towards the same scene. In order to predict observer's fixations, eye gaze information can be used to reveal human attention and interest. This paper presents a real-time eye gaze tracking system. Haar cascade classifier is used to calculate the position of eye gaze based on the rectangular features of human eye. Then this position is adopted to match the space coordinates of screen representing where an observer is looking. The experimental results from different kinds of scenes validate the effectiveness of our system.**

Keywords—visual attention; regions of interest; eye gaze tracking; Haar cascade classifier;

I. INTRODUCTION

Based on the principle of *human visual system* (HVS), when people look at an image, they will naturally focus on the *region of interest* (ROI). In this process, human eyes will movement according to the gaze point, which contains valuable information to describe the essence of the focus.

The eye-gaze tracking method has been increasingly used for studying human visual attention over the last few years. Understanding how eye tracking impacts the humans' gazing behavior will contribute to the researches in computer vision field [1]. The aim of this work is to study how to acquire effective information by eye tracking system. Vu *et al.* [2] conducted two experiments to explore the relationship between eye-movements and cognitive behavior and decision. Andersen *et al.* [3] utilized the eye tracking technology to quantify sex differences in different virtual environments. Elsner *et al.* [4] employed eye tracking systems to investigate the relationship between infants' visual attention and action demonstration.

The indispensable procedure to realize the eye-gaze tracking is to locate pupil precisely, base on the monocular head-mounted gaze estimation, Barz *et al.* [5] put forward a support vector regression model to calculate eye tracking error in real-time. Yu *et al.* [6] proposed a point-of-gaze estimation method to extract features from a video to detect the position of human head. Then the eye position data can be generated from the position of human head. Itoh and Klinker [7] introduced a novel method to estimate the eye gaze by utilizing the head-mounted displays, this operation can avoid frequent calibration. Such immobilization techniques only need to match the parameters of screen coordinates and pupil coordinates in one-time, which can be very efficient to achieve the parameter calibration. But these models are not convenient and comfortable in real applications.

Recently, most of the researchers explored the tracking algorithm under nature head movement. Pires *et al.* [8] applied gaze estimation method to motor-sports application according to visible-spectrum. Zhu and Ji [9] built the eye tracking system under the condition of nature head movement by using conventional pupil center corneal refection technique. Lu *et al.* [10] proposed an *Binocular Vision Method* (BVM), which are based on space coordinates and *Local-Binary-Pattern Texture Feature* (LBP). They mainly utilized the *Support Vector Regression* (SVR) to match the gaze mapping function. Compared with the immobilization techniques, these models can avail to the flexibility of parameter calibration and the real-time tracking under nature head movement.

Although the above methods can effectively construct the eye tracking, they generally ignored the user's needs and preferences [11]. In order to improve the visualization performance, user adaptive visualization system has been constructed to locate the eye tracking. Steichen *et al.* [12] described the eye gaze system by adapting user's personal habits, which achieves encouraging results on simple eye tracking measure. Courtemanche *et al.* [13] proposed an application-independent model to recognize the users interacting, which has the ability to relate users' interactions for various monitoring purposes.

To make the eye-gaze tracking system more humanized, this paper utilize the haar cascade classifier to fulfills the requirements of eye-gaze tracking. Our approach can improve the recognition accuracy and efficiency.

The rest of this paper is organized as follows. In Section II the rectangular features, pupil-glint vector and haar cascade classifier are introduced. The eye-gaze tracking system is described in Section III. Finally, conclusions are provided in Section IV.

II. EYE CHARACTERISTICS

A. Rectangular Features

Eye characteristics can be used to tracking the eye gaze, and some of them can achieve idea results. The proposed method utilizes the rectangular features [14] for pupil detection. These features can be applied to haar cascade classifier. The rectangular features are shown in Figure 1.
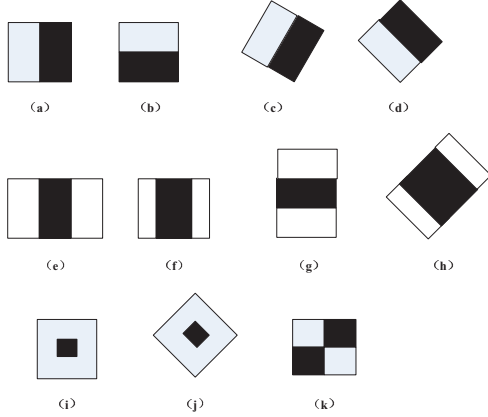

Figure 1. Rectangular features applied to haar cascade classifier. (a-d) are the edge features, (e-h) are the line features, (i) and (j) are the center-surround features, (k) is the special diagonal line feature.

As shown in Figure 1, the pixel value of pupil area is clearly distinguish from its surrounding. From our experiment, the rectangular features are suitable for detecting human eyes in an image. The eigenvalues of the object area are measured by calculating the pixel difference between white areas and black areas of reference rectangle features. When the detection region near the eyes, the eigenvalues are approximately constant. The eigenvalues (denoted as V) of the test results are obtained as

$$V = R_1 - R_2 \tag{1}$$

where R_1 and R_2 denote the sum value of pixels in the white areas and the black areas, respectively, which are computed as

$$R_i = \sum_{m,n \in R_i, i=\{1,2\}} r(m,n) \tag{2}$$

where $r(m,n)$ is the detection target of the white areas or the black areas, m and n denote the horizontal coordinate and vertical coordinate of the object region, respectively.

B. Pupil-Glint Vector

The pupil-glint vector can be defined by pupil center and the corneal reflection. Based on the calibration operation, this vector is mapped to screen coordinates. To make the pupil tracking match the screen coordinates, Morimoto [15] proposed a calibration method by utilizing two polynomials and nine points.

$$x = k_0 + k_1 v_x + k_2 v_y + k_3 v_x v_y + k_4 v_x^2 + k_5 v_y^2 \tag{3}$$

$$y = l_0 + l_1 v_x + l_2 v_y + l_3 v_x v_y + l_4 v_x^2 + l_5 v_y^2 \tag{4}$$

where v_x and v_y denote the pupil-glint vector in horizontal direction and vertical direction, respectively. (x, y) is the screen coordinates. Formula (3) and (4) is the calibration function.

C. Haar Feature-based Cascade Classifier for Eye Detection

The classifier is trained from about one hundred simple positive examples and negative examples, which are of same size. We mark the region with 1 or 0 by judging whether it is similar to the object or not. After the training, classifier can locate all the object regions by traversing the whole images. In order to detect the target region more accurately, the size of the scanning window is changed adaptively by the classifier. In the process of classification, the optimal rectangular feature model are selected according to the object and scanning window.

To speed up the testing, summed-area table [16] is used to record the pixels in the object region. The summed area table is an two-dimensional array, where each element stores the sum of all pixels between the entry location $(0,0)$ and the lower right corner (i, j). This processing procedure can save a lot of time when obtain the sum of pixels in an image region. Only need through the corners of the window coordinates and find the pixels in the integral figure when we change the line search window, thus can save a lot of computing time. The summed area table (denoted as $E(i, j)$) is calculated as

$$E(i, j) = \sum_{0 \le m \le i, 0 \le n \le j} p(m,n) \tag{5}$$

where $p(m,n)$ is the pixel in the detection object of the image.

Let $E\_line[m][n]$ denote the sum of all pixels between the entry location $(m, 0)$ and the lower right corner (m, n). The summed area tables process contains one algorithm shown in Algo 1.

```
Begin:
1:Function get_E( i, j )
2:       m set 0, n set 0, E_line[m][n]=0 and E[m][n]=0;
3:       while( m=0 to i, n=0 to j)
4:            E_line[m][n]=E_line[m][n-1]+p[m][n];
5:            E[m][n]=E[m-1][n]+E_line[m][n];
6:       return E;
END
```

Algo 1. The summed area tables algorithm.

In this module, it consists of four steps, all of steps are performed with update data. A description to compute a sum in the summed area table algorithm is shown in Figure 2.
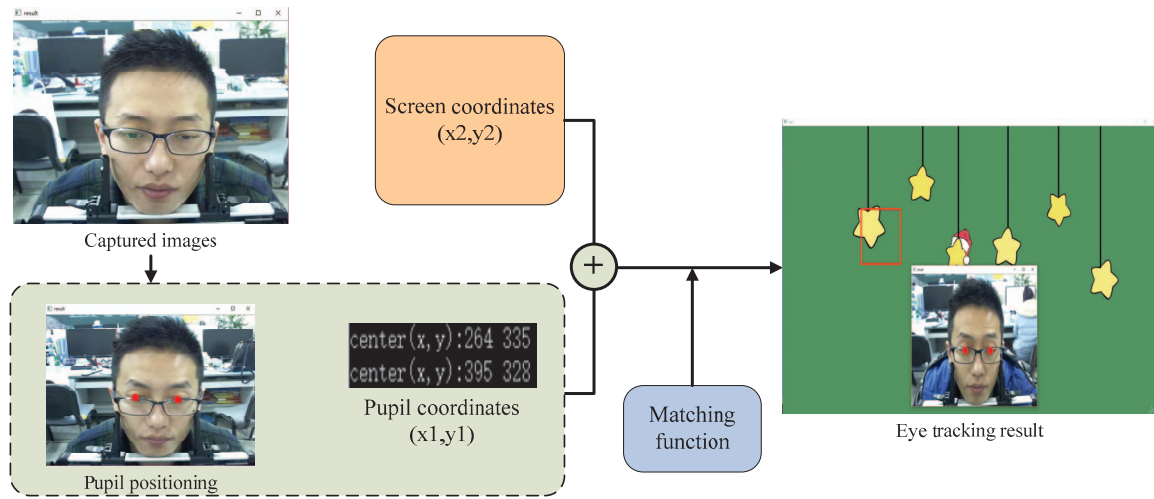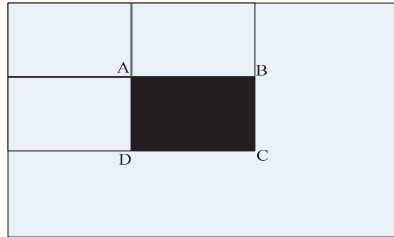
Figure 3. Flow Chart of the eye-gaze tracking system.

Figure 2. A description of computing a sum in the Summed Area Table data structure.

As shown in Figure 2, the processing procedure to calculate the black area only need to query the value of the four vertices from summed area tables, which is computed by

$$S_{ABCD} = E(A) + E(C) - E(B) - E(D) \qquad (6)$$

where S_{ABCD} is the sum of pixels in the black area, $E(A)$ is the element value of point A in the summed area tables.

III.   EXPERIMENT SYSTEM

In this system, the position of support is adjusted according to different people. Using the camera bracket can make the camera extract the head image clearly, which can contribute to focus the face area. Our system detects the pupil position accurately of images acquired by camera, the key process used haar cascade classifier. Then this position is adopted to match the space coordinates of the screen and the real time eye tracking is realized. The flowchart of our eye-gaze tracking system is shown in Figure 3.
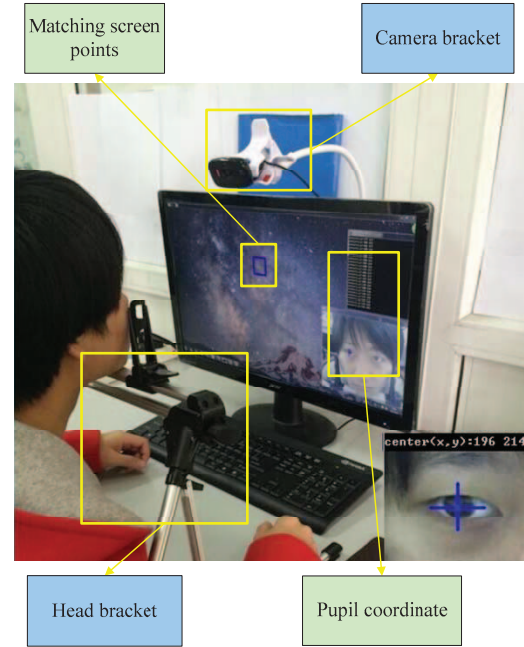

Figure 4. Eye-gaze Tracking System.

Figure 4 is general picture of our eye-gaze tracking system. In the system, the hardware includes head support, camera bracket, high definition camera and person computer. The whole system is constructed by using the eye tracking techniques base on haar cascade classifier. In the process of detection, we compress the image to the size of 30×30 to speed up the testing. To confirm whether the detection region

is the target object, they should be matched at least three times. The matching process is defined as

$$\frac{S(x,y)}{P(x,y)} = \frac{S(x,y)}{S_w} * \frac{S_r - S_l}{P(x,y) - S_l} \qquad (7)$$

where $S(x,y)$ is the screen coordinates, $P(x,y)$ is the pupil coordinates, S_w is the width of the screen, S_r and S_l are the coordinates when observer is looking at the upper left corner and the upper right corner of the screen respectively.

In our experiment, different point gazes on the screen are selected for testing coordinate error. We present nine random points in different directions observed by various subjects, which are shown in Figure 5.
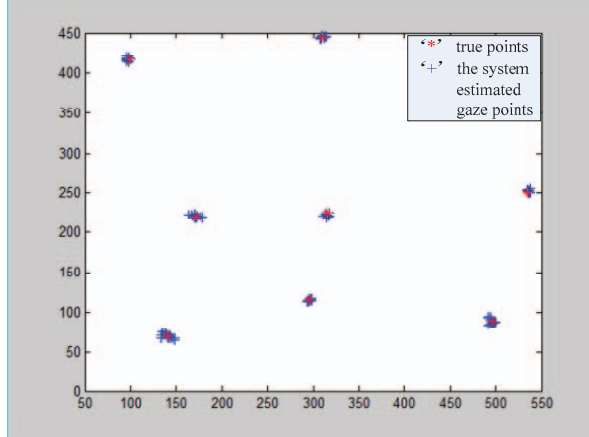

Figure 5. The system estimated gaze points and true gaze points, '+' represents the system estimated gaze points and '*' represents true points.

The System error estimation is shown in Table 1 by computing the average Euclidean distance between the red point and the blue points, which shows that our algorithm has a better performance.

Table 1. Comparison of various methods

Algorithm	Euclidean Errors(mm)
Zhu's	8.29
Lu's	14.22
Our's	5.99

IV. CONCLUSION

This paper proposes a Haar cascade classifier based eye gaze tracking system. The position of eye gaze can be calculated using the Haar-like rectangular features, next this position is adopted to match the space coordinates of screen, and then the real-time eye gaze tracking system is realized to locate where an observer is looking at. Experimental results show that our pupil localization technique can achieve better predictive accuracy.

REFERENCES

[1] J. Zhang, M. Wang, S. Zhang, X. Li, and X. Wu, "Spatiochromatic context modeling for color saliency analysis," *IEEE Transactions on Neural Networks and Learning Systems*, no. 99, August 2015.

[2] T. M. H. Vu, V. P. Tu, and K. Duerrschmid, "Design factors influence consumers' gazing behaviour and decision time in an eye-tracking test: A study on food images," *Food Quality and Preference*, vol 47, pp. 130–138, January 2016.

[3] N. E. Andersen, L. Dahmani, K. Konishi, and V. D. Bohbot, "Eye tracking, strategies, and sex differences in virtual navigation," *Neurobiology of Learning and Memory*, vol. 97, no. 1, pp. 81–89, January 2012.

[4] B. Elsner, C. Pfeifer, C. Parker, and P. Hauf, "Infants' perception of actions and situational constraints: An eye-tracking study," *Journal of Experimental Child Psychology*, vol. 16, no. 2, pp. 428–442, October 2013.

[5] M. Barz, A. Bulling, and F. Daiber, "Computational modelling and prediction of gaze estimation error for head-mounted eye trackers," *DFKI Research Reports*, vol. 1, no. 1, no. 3, 2015.

[6] L. H. Yu, and Moshe. Eizenman, "A new methodology for determining point-of-gaze in head-mounted eye tracking systems," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 10, pp. 1765–1773, 2014.

[7] Y. Itoh, and G. Klinker, "Interaction-free calibration for optical see-through head-mounted displays based on 3D Eye localization," *IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 75–82, March 2014.

[8] B.R. Pires, M. Hwangbo, M. Devyver, and T. Kanade, "Visible-spectrum gaze tracking for sports," *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1005–1010, June 2013.

[9] Z. Zhu, and Q. Ji, "Nonlinear eye gaze mapping function estimation via support vector regression," *International Conference on Pattern Recognition (ICPR)*, pp. 1132–1135, 2006.

[10] H. Lu, C. Wang, and Y. Chen, "Gaze tracking by binocular vision and LBP features," *International Conference on Pattern Recognition*, pp.1–4, December 2008.

[11] H Zhang, P Wu, A. Beck, Z Zhang, and X Gao, "Adaptive incremental learning of image semantics with application to social robot," *Neurocomputing*, vol. 173, pp. 93–101, 2016.

[12] B. Steichen, G. Carenini, and C. Conati, "User-adaptive information visualization - using eye gaze data to infer visualization tasks and user cognitive abilities," *Conference Website*, pp. 317–328, 2013.

[13] F. Courtemanche, E. Aimeur, A. Dufresne, M. Najjar, and F. Mpondo, "Activity recognition using eye-gaze movements and traditional interactions," *in Proceedings of Interacting with Computers*, pp. 202–213, 2011.

[14] P.Viola, "Robust real-time face detection," *International Journal of Computer*, vol. 57, no. 2, pp. 137–154, 2004.

[15] C. Morimoto, D. Koons, A. Amir, and M. Flickner, "Pupil detection and tracking using multiple light sources," *Image Vis. Comput*, vol. 18, no. 4, pp. 331–336, 2008.

[16] F. C. Crow, "Summed-area tables for texture mapping," *in Proceedings of SIGGRAPH*, vol. 18, no. 3, pp. 207–212, 1984.