Gaze Tracking By Binocular Vision and LBP Features

Hu-chuan Lu[1], Chao Wang[1]
[1]*Department of Electronic Engineering*
Dalian University of Technology
China 116023
email: lhchuan@dlut.edu.cn

Yen-wei Chen[1,2]
[2]*College of Information Science and Engineering*
Ritsumeikan University
Japan 5258577
email: chen@is.ritsumei.ac.jp

Abstract

In this paper, a new method for eye gaze tracking is proposed under natural head movement. In this method, Local-Binary-Pattern Texture Feature (LBP) is adopted to calculate the eye features according to the characteristic of the eye, and a precise Binocular Vision approach is used to detect the space coordinate of the eye. The combined features of space coordinates and LBP features of the eyes are fed into Support Vector Regression (SVR) to match the gaze mapping function, in the hope of tracking gaze direction under natural head movement. The experimental results prove that the proposed method can determine the gaze direction accurately.

1. Introduction

Eye gaze, representing where a person is looking, reveals a person's focus of attention and interest. Nowadays non-intrusive techniques are developed, most of them are vision-based, and they use cameras to capture images of the eyes, but some camera-based techniques might be intrusive if they require head fixing.

Yu et al. [1] developed a head-mounted methodology, he uses features extracted from the video of the scene camera to determine the position of the head which is relative to the objects in the scene. Since the eye position data is provided with respect to the head, these two data sets are readily combined to determine the fixation behavior on objects in the scene. K. Tan [2] uses an Appearance-based Eye Gaze Estimation system, S. Baluja [3] uses ANN to track gaze direction, Zhiwei Zhu [9], [10], [11] uses Pupil Center Corneal Reflection (PCCR) to determine the gaze direction, and uses some geometry to compensate for the effects of head movement. These methods make tracking technology in actual sight of a great step forward, but the accuracy and reliability also need to be constantly improved.

This paper proposes a new method for eye gaze tracking under natural head movement. A precise method of binocular vision is adopted to consider the space coordinates of the eye, and the coordinates combining the LBP [5] features as the input features are fed into Support Vector Regression(SVR) to predict the mapping function of gaze direction. For PCCR, many people try various methods to accurately calculate the vector [6], but pupil fuzzy borders, shape changes and other factors may make the pupil center drifting, additionally, the reflection point sometimes appears to be so large that it offsets the actual position, which causes an inaccurate pupil-glint vector and affects experimental results. The LBP features of the eye images can represent the "pupil-glint" vector information by obtaining the texture changes information of different eye images. This method can avoid the disadvantage of the calculation of pupil-glint vector, thus estimating the gaze more accurately under natural head movement.

In Section II the binocular vision method is briefly introduced, Section III the LBP is described then follows the discussion of its applications in gaze tracking compared with the classical PCCR method. Section IV presents experiment results and Section V the conclusion.

2. Binocular Vision Method

The goal of the binocular vision method is to determine the space coordinates of the eye. When the location of eyes in the two camera images is

978-1-4244-2175-6/08/$25.00 ©2008 IEEE

determined, the purpose of image matching can be achieved. Here is a brief introduction to the eye detection method, and how to obtain the Space Coordinates of the eye is also introduced.

2.1. Eye Detection

Many eyes detection methods have been proposed; in this paper the conception of rectangular features proposed by Paul Viola [7] is used. Some rectangular features suitable for the human eye characteristics (Figure 1) are adopted to train cascade classifiers for eye detection.

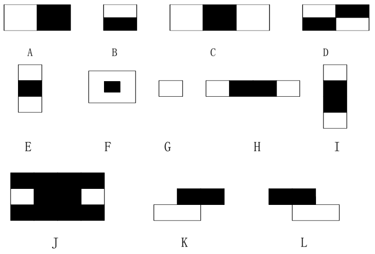

Figure 1. **Rectangular features used for eye detection**

2.2. Space Coordinates of the Eye

A camera calibration matlab [12] toolbox is used to calculate the space coordinates (Xc, Yc, Zc) of the eyes according to the camera coordinate system. In order to get a high measuring accuracy, a pair of high-resolution camera is located in a suitable position which is also calibrated, so that the eyes in the gaze tracking system are approximately positioned in the area near the center of the images in order to reduce the impact of tangential distortion and radial distortion. Another key point we must focus on is that a synchronous pair of images should be captured from its individual camera to reduce the following calibration errors. For that purpose, our application uses the cvcam class of OpenCV [13].

Once the image projections of the eye in each captured image are available, the space coordinates can be calculated by the calibrated stereo system. In our application the distance error on average between the calculated result and the corresponding real one is less than 2mm.

3. Features of the Eyes

Similar to the foregoing methods, the eye images are used as training sets, then the eyes can be detected from the new captured image, finally the LBP features of the eye image could be obtained. Before discussing the LBP method, the classical pupil-glint vector algorithm is described as a contrast to our method.

3.1. Classical Pupil-Glint Vector Algorithm

The dark pupil effect eye image is shown in Figure 2, from which the corneal reflection can be clearly seen. Assuming that the eye is a sphere that only rotates around its center, and that the camera and light source are fixed, the position of the Corneal Reflection (CR) does not move with the eye rotation, and therefore can be used as a reference point. The center of the pupil and the CR defines a vector in the image. This vector can be easily mapped to screen coordinates on a computer monitor after a calibration procedure. The calibration procedure is required to compute the mapping from the pupil-glint vector to monitor screen coordinates. In general, the user is asked to look at several points on the computer screen, one point at a time, pressing a button. Morimoto [8] uses 9 points for calibration and a second order polynomial calibration function.

The polynomial is defined as:

$$x_g = a_0 + a_1 v_x + a_2 v_y + a_3 v_x v_y + a_4 v_x^2 + a_5 v_y^2 \qquad (1)$$

$$y_g = b_0 + b_1 v_x + b_2 v_y + b_3 v_x v_y + b_4 v_x^2 + b_5 v_y^2 \qquad (2)$$

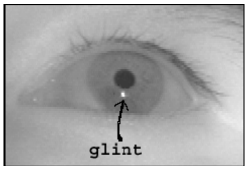

Figure 2. **Eye image with glint**

The pupil-glint vector is $\vec{v}$, and the mapping coordinates in x and y axis are v_x and v_y separately. (x_g, y_g) are the corresponding screen coordinates. Since each calibration point defines 2 equations, the system is defined with 12 variables and 18 equations, and can be solved using least squares method. Actually, because the set of parameters a and b are independent, it can be solved as 2 sets of 6 variables and 9 equations.

In the classical algorithm, the head is mounted, which is not convenient and comfortable in real applications, thus many people have put forward various methods to improve gaze tracking system for the purpose of natural head movement. Zhiwei Zhu [9] uses simple geometry relationship, [10] combined the space coordinate of the eye and pupil-glint vector to achieve the tracking process, [11] afterwards try to estimate the 3D eye gaze directly or the 3D gaze direction of the users. The methods above are all based on PCCR. Although they try to accurately calculate the vector, pupil fuzzy borders, shape changes and other factors affect the pupil center migration, in addition, the reflection point sometimes appears to be too large that offsets the actual position, which caused the pupil-glint vector to be inaccurate and finally affect experimental results. The LBP features of the eye images can represent the “pupil-glint” vector information by obtaining the texture changes information, so this method can avoid the disadvantage of the calculation of pupil-glint vector, thus estimating the gaze more accurately under natural head movement.

3.2. LBP Features

The original LBP operator was introduced by Ojala et al [5]. The operator labels the pixels of an image by thresholding a 3×3 neighborhood of each pixel with the center value and considering the results as a binary number. The 256-bin histogram of the labels computed over a region can be used as a texture descriptor. Each bin (LBP code) can be regarded as a micro-texton. Later the operator was extended to use neighborhood of different sizes using circular neighborhoods [4].

The LBP $LBP_{P,R}$ operator produces 2^P different output values, corresponding to the different binary patterns that can be formed by the P pixels in the neighbor set. It has been shown that certain bins contain more information than others [4]. Ojala et al called these fundamental patterns uniform patterns.

However, a LBP histogram computed over the whole eye image encodes only the occurrences of the micro patterns without any indication about their locations. To also consider shape information of eyes, which can represent the gaze direction the same as the PCCR method, eye images are divided into small regions $R_0, R_1, \cdots R_m$ to extract LBP features.

In this way, the classical pupil-glint vector, which is two-dimension is replaced by the high-dimension LBP vector (in our method $59 \times 9 = 531$). As mentioned above, the LBP vector not only can represent the gaze direction (divide the eye image into small regions), but also can provide the image texture information, which is its intrinsic advantage. The experiment in Section IV proves the assumption.

4. Experiment

The experiment in this paper adopts $\varepsilon-$ SVR method, the penalties parameter C equals to 1, using the Gaussian function as kernel function, whose sigma is 100. The training parameters are $(x, y, z, a_1, a_2 ... a_{531})$, x, y, z is space coordinates of the eye, $(a_1, a_2 ... a_{531})$ is LBP features. (X, Y) are the regression parameters, each pair of x and y have done two training SVR classification hyperplane to predict the test sample’s mapping result.

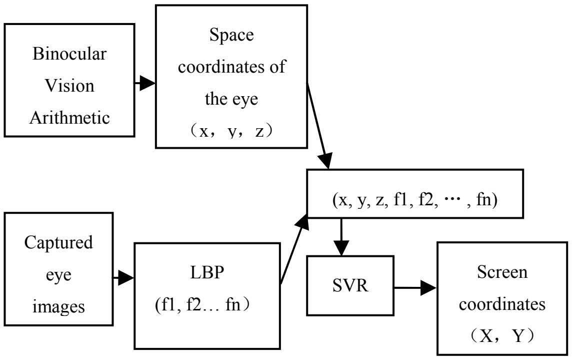

Figure 3. **The flow chart of the gaze tracking algorithm**

Here a group of 1014 samples are used as a training set, with 126 group parameters for testing, and the results are shown in table 1.

Table 1. **Tracking results of space coordinates combine with LBP vector method**

	X direction(mm)	Y direction(mm)
Algorithm	Errors $(\mu \pm \sigma)$	Errors $(\mu \pm \sigma)$
Ours	4.24 ± 4.54	3.24 ± 2.81
Zhu’s	5.02 ± 2.03	6.40 ± 2.35

The experiment results show that the method based on Space Coordinates and LBP features of the eyes can approximate the gaze mapping function very well. The average horizontal and vertical errors are approximately 4.24 mm and 3.24 mm respectively as show in Table 1, which performs better than Zhiwei Zhu’s [11] (5.02 mm, 6.40 mm respectively). Another interesting fact is that the error in horizontal direction is larger than the error in vertical direction, while Zhiwei Zhu’s the opposite. This is mainly because the LBP features not only represent the shape information of different region of the eye the same use of the PCCR vector, but also consider the micro texture patterns of the whole eye. So it can draw the conclusion that the difference of the micro texture patterns in vertical direction is larger that in horizontal direction when the eyes gaze on the different positions of the monitor screen. That is to say, when a person gazes at the point, for example, (170,128) and the point (511,384), the distance of the X-Coordinate is usually larger than that of the Y-Coordinate, which is reflected in PCCR, while in our method, the distance in high-dimensional (LBP+ Space Coordinates) is the opposite.

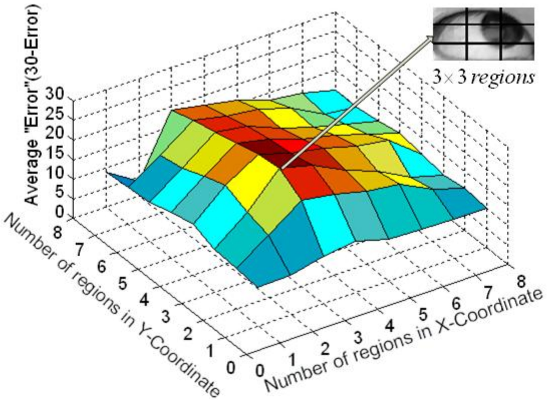

Figure 4. **The aver age errors with respect to different regions in both X-Coordinate and Y-Coordinate**

Figure 4 shows us the change tendency of errors with respect to different regions in both X-Coordinate and Y-Coordinate. From the picture, it is concluded that when the number of region of X-Coordinate is 3, and the number of Y-Coordinate region is 3, the minimum "error" (in the picture 30-error for display) is gained, while the average error of (3*4) regions and (4*3) regions are 4.63 mm and 4.51 mm respectively, which are very close to the (3*3) regions.

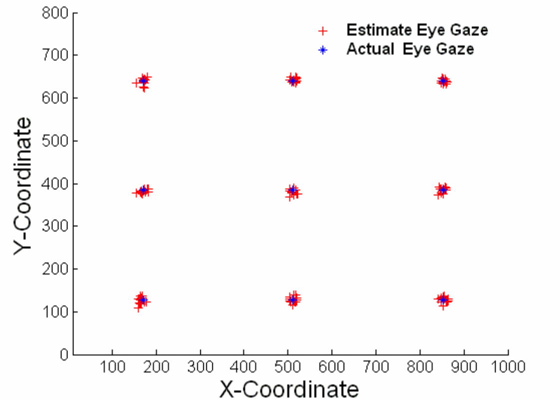

Figure 5. **The plot of the estimated gaze points and the true gaze points, where "+" represents the estimated gaze point and "*" represents the actual gaze point**

Figure 5 displays the error between the estimated gaze points and the actual gaze points. The average horizontal error is around 4.24 mm in the screen. The average vertical error is around 3.24 mm.

5. Conclusion

This paper uses rectangular features for eyes detection, and a precise binocular vision method is adopted to compute the space coordinates of the eye, and then the LBP is proposed to describe the features of the captured eyes. Combining the space coordinates with LBP features as the input features of SVR, the mapping function of gaze direction and screen coordinates can be forecasted. The experimental result proved that the proposed method is effective in gaze tracking. In future work the algorithm should be further improved to increase the accuracy, and make it a better application in human computer interaction.

References

[1] Lawrence H. Yu, Moshe Eizenman. A New Methodology for Determining Point-of-Gaze in Head-Mounted Eye Tracking Systems[J]. *IEEE Transactions on Biomedical Engineering*, 2004, 51(10): 1765-1773.

[2] K. Tan, D. Kriegman, H. Ahuja. Appearance based eye gaze estimation[C], *Proc. of the IEEE Workshop on Applications of Computer Vision-WACV02*, 2002, 191–195.

[3] S. Baluja, D. Pomerleau. Non-intrusive gaze tracking using artificial neural networks[J]. *Advances in Neural Information Processing systems*, 1994, 6:753-760.

[4] T. Ojala, M. Pietikinen, and T. Menp. Multiresolution grayscale and rotation invariant texture classification with local binary patterns[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, 24(7):971－987.

[5] T. Ojala, M Pietikinen, and D. Harwood. A comparative study of texture measures with classification based on featured distribution[J]. *Pattern Recognition*, 1996, 29(1):51－59.

[6] Jian-Gang Wang, E. Sung, Ronda Venkateswarlu. Eye gaze estimation from a single image of one eye[C], *IEEE International Conference on Computer Vision*, 2003, 136-143.

[7] Paul Viola. Robust Real-Time Face Detection[J]. *International Journal of Computer*, 2004, 57(2): 137–154.

[8] C. Morimoto, D. Koons, A. Amir, M. Flickner. Pupil detection and tracking using multiple light sources[J], *Image Vis. Comput*, 2000, 18 (4): 331–336.

[9] Zhiwei Zhu, Qiang Ji. Eye gaze tracking under natural head movements[C], *IEEE International Conference on Computer Vision and Pattern. Recognition (CVPR)*, 2005, 918–923.

[10] Zhiwei Zhu and Qiang Ji. Nonlinear Eye Gaze Mapping Function Estimation via Support Vector Regression[C], *the 18th International Conference on Pattern Recognition (ICPR)*, 2006, 1132-1135

[11] Zhiwei Zhu and Qiang Ji. Novel Eye Gaze Tracking Techniques Under Natural Head Movement[J]. *IEEE Transactions on Biomedical Engineering*, 2007, 54(12):2246-2260.

[12] J. Boughet, "Camera calibration toolbox for Matlab," http://www.vision.caltech.edu/bouguetj/calib_doc/ October, 2004.

[13] http://www.opencv.org.cn/index.php/