Model-Based Eye-Tracking Method for Low-Resolution Eye-Images

Takashi FUKUDA[*1] **Kosuke MORIMOTO**[*1] **Hayato YAMANA**[*1*2]

[*1.] Dep. of Computer Science and Enginerring, Waseda Univ.
3-4-1 Okubo, Shinjuku, Tokyo 169-8555 JAPAN
{t_fukuda, morimoto, yamana}@yama.info.waseda.ac.jp

[*2.] National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku,
Tokyo 101-8430 JAPAN

ABSTRACT

Techniques for estimating gaze without restricting user movements are highly desired for their potential applications. Although commercial gaze estimation systems achieve high accuracy using infrared light, gaze estimation systems with webcams have become indispensable because of their low price. The problem using webcams is that their resolution is too low to estimate the direction of eye-gaze accurately without restricting user movements. Low-resolution eye-images have strong sources of noise that result in inaccurate estimation. In this paper, we propose a novel gaze-estimation method that uses both image processing and geometrical processing to reduce various kinds of noise in low-resolution eye-images and thereby achieve high accuracy. In our experiments, the mean horizontal error of 5 participants is 3.0° and the mean vertical error is 1.5° with calibration.

Author Keywords

Gaze, Eye tracking, Low resolution, webcam.

ACM Classification Keywords

H1.2 User/Machine Systems: Human factors

INTRODUCTION

Gaze information has been widely used for studies in neuroscience, psychology, human engineering, marketing strategy, advertising, and so on [4], because the direction of eye-gaze gives clues about a person's interest and attention. In such studies, gaze information extracted from a variety of people in natural situations is highly desired to avoid any effects of the measurement system on the testees. In other words, the system should not restrict the movement of testees and should not require testees to wear any equipment.

Moreover, in order to gather gaze information from a variety of people, low-cost systems are desirable. However, conventional commercial eye-tracking systems [2,11,13] are too expensive, costing as much as several tens of thousands of US dollars.

One way to solve the abovementioned problems is to adopt low-cost webcams that cost just a few tens of US dollars but with a resolution of only 640 × 480 pixels. Although webcams are low in cost, eye-images extracted from webcams have low resolution because it is impossible to adopt a zoom-in feature to capture eye-images without restricting the movement of users. Agustin et al. [1] used low-cost web cams. However, in their method, users have to wear some equipments to set webcams near by their eyes and to capture high resolution eye-images. This is uncomfortable for users. Therefore, eye tracking using low-resolution eye-images becomes indispensable to solve the problems.

Previous studies based on low-resolution eye-images have some drawbacks. Ono et al. [9] used machine learning correlating the appearance of low resolution eye-images to the points of gaze. Their method shows good accuracy in that the mean error was 2.4°; however, their method has one drawback—a learning process that is too complicated. A user must gaze twenty times at twenty points from various positions for each point. On the other hand, Yamazoe et al. [16] used a model matching method with low-resolution eye-images. The mean errors were 5.3° horizontally and 7.7° vertically. This low accuracy is resulted from their rough eye model.

In this paper, we propose an accurate model-based gaze-estimation method with low-resolution eye-images. In our method, eye orientation is estimated accurately without quantization. We observe the pupil as a circle and then estimate the normal direction of the pupil as the gaze direction. We approximate the observed pupil contour with an ellipse and estimate the direction normal to this ellipse.

Low-resolution eye-images have strong sources of noise distorting the pupil contours. The approximation of the pupil contour with an ellipse using hough transformation is robust against noises. However, hough transformation has two drawbacks—the huge computation and the nonunique result. Hough transformation is a kind of vote. Its voting procedure is carried out in a parameter space exhaustively. Therefore, it has huge computation in ellipse approximation that has five-dimensional parameter. Furthermore, hough

2nd Workshop on Eye Gaze in Intelligent Human Machine Interaction.
February 13, 2011, Palo Alto, California, USA.

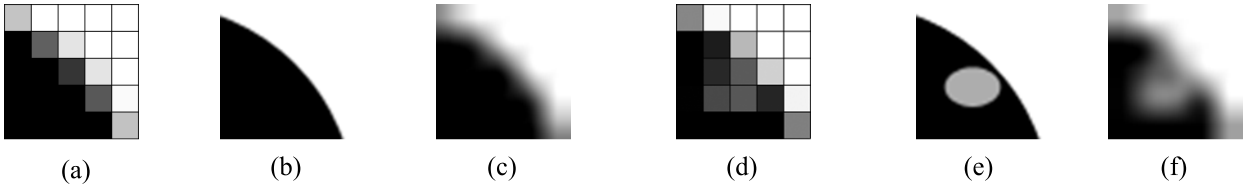


Figure 1 Examples of noise in low-resolution eye-images

transformation could have two or more results which receive the most votes. Takegami et al. [12] and Tsuji et al. [14] resolved these problems by constraining some parameters in hough transformation. However, they constrained parameters by restricting the movements of users.

We use the least square method (LSM) to allow users to move. The LSM is fast and results uniquely with no constraint. However, the LSM results wrong with the distorted ellipse contour. We adopt image processing and geometrical processing to remove the distortions from the pupil contour in low-resolution images. In our method, we expand low-resolution eye-images with the Bicubic convolution and estimate the sub-pixel pupil contour to reduce the approximation error. Furthermore, we fit the convex hull to the pupil contour to remove the distorted parts of the contour. The rest of this paper is organized as follows: In Section 2, we review related work. In Section 3, we propose our method. We then describe the experimental evaluation of the method in Section 4. Finally, we conclude the paper in Section 5.

RELATED WORK

Related work based on low-resolution eye-images is classified into two categories; appearance-based and model-based methods.

Appearance-based method

In appearance-based methods, the gaze direction is estimated using Machine Learning (ML). The classifier learns the relation between the appearance of the eye-images and gaze direction. Ono et al. [9] used N-mode Singular Value Decomposition (SVD). The resolution of the eye-images is 24 × 12 pixels, and the mean error is 2.6°. This shows a high accuracy for low-resolution eye-images. However, in this method, the calibration procedure is extensive in that each user must gaze at each point at least 20 times from various positions. This is too demanding for practical use.

Model-based method

In model-based methods, pre-generated 3D eye-models are used to estimate the gaze direction. The major model-parameters are the position, tilt, and rotation of the eye. The system calculates the parameters without any learning process.

However, most researchers assume that high-resolution eye-images are available with model-based methods [3,5,7,8,10,12,14,15]. They capture high-resolution eye-images by zooming in on movement-restricted users. To the best of our knowledge, there exist a few model-based gaze tracking methods using low-resolution eye-images.

Yamazoe et al. [16] used 30 × 15 pixel eye-images to extract the direction of eye-gaze with a model-based method. The mean errors were 5.3° horizontally and 7.7° vertically. They consider a line from the center of the eyeball to the center of the pupil as the line of gaze. They estimate the position of the center of the eyeball by matching the eye-images with pre-generated 2D eye-images using maximum-likelihood estimation. They generate 2D images by projecting variously oriented 3D eye-models that contain eyeballs, pupils, and eyelids. The 3D eye-models are rotated 5° each. However, this quantization (i.e., 5° each) is too coarse to achieve high accuracy. When the eye-image does not completely match with that of a pre-generated 3D model, the estimated center of the eye deviates considerably from the real position. This results in a low accuracy of gaze estimation.PROPOSED METHOD

Problem Formulation

Previous model-based eye-tracking methods using low-resolution eye-images face the difficulty of estimating detailed eye-model parameters.

The low-resolution eye-images are too coarse to determine the position of the pupil contour in the image. Figure 1 (a) shows an example of part of a low-resolution eye-image, especially around the pupil contour. Figure 1 (a) is generated by reducing the resolution of Figure 1 (b) to one tenth. The black pixels represent the pupil and the white pixels represent all other areas of the eye. The contour of the pupil must be placed on the gray pixels; however, we cannot determine where the true contour is in the gray pixels because of the low image resolution. The error in pupil contour estimation caused by these gray pixels results in a significant error in eye tracking. Consider that the true contour of the pupil is a circle whose diameter is 15 pixels, and that the estimated pupil contour is an ellipse whose major diameter is 15 pixels and minor diameter is 14 pixels. This 1-pixel error causes an error of 21° in the estimated gaze direction.

At first glance, it might appear appropriate to expand the image using Bicubic convolution in order to estimate the pupil contour more precisely. However, this expansion generates other noise. Figure 1 (c) shows Figure 1 (a)

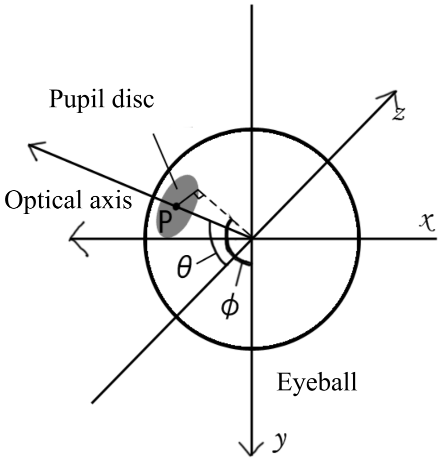


Figure 2 3D eye-model

expanded ten-fold. The wave of the contour is generated using Bicubic convolution.

In addition, the expansion emphasizes noise in the original images. Figure 1 (d) shows a low-resolution version of Figure 1 (e). The gray ellipse is included in Figure 1 (e) because of the light reflection on the pupil. In this case, we obtain Figure 1 (f) by expanding Figure 1 (d). The gray circle in the pupil in Figure 1 (e) becomes a dent in the pupil contour as shown in Figure 1 (f).

Such noise causes distortion of the pupil contour. As a result, the approximation of the pupil contour with an ellipse fails and the accuracy of eye tracking decreases. Reducing these distortions is the main problem we address in this paper.

Overview of our proposed method

We propose a method to remove the distorted portions of the pupil contour to approximate the pupil contour accurately with an ellipse. This increases the accuracy of model-based eye tracking using low-resolution eye-images.

We adopt both geometrical processing and image processing to remove distortions of the pupil contour. We expand low-resolution eye-images with Bicubic convolution and fit the convex hull to the pupil contour. Furthermore, we estimate the sub-pixel pupil contour to reduce the approximation error. After noise reduction, we approximate the pupil contour with an ellipse.

Our eye-model itself is the same as that Wang et al. [15]. Figure 2 shows the eye-model. Wang et al. modeled an eyeball as an ideal sphere, and the pupil as a disc fitted into a hole on the sclera sphere. The line normal to the pupil disc from the center of the disc represents the optical axis of the eye. In Figure 2, $\theta(0 \leq \theta < 2\pi)$ shows the angle between the optical axis of the eye and the z axis. On the other hand, $\phi(-\pi < \phi < \pi)$ shows the angle between the optical axis of the eye projected onto the x-y plane and the y axis. We refer to θ as “eye tilt angle” and call ϕ the “eye rotate angle.” We define $\phi > 0$ when the x-coordinate of the center of the pupil disc is positive.

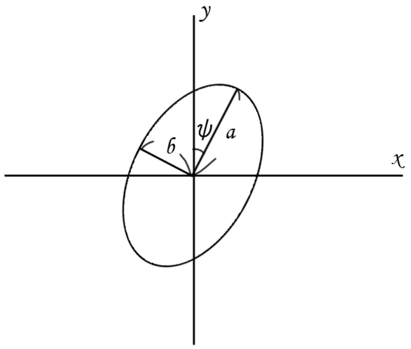


Figure 3 Example of ellipse approximation

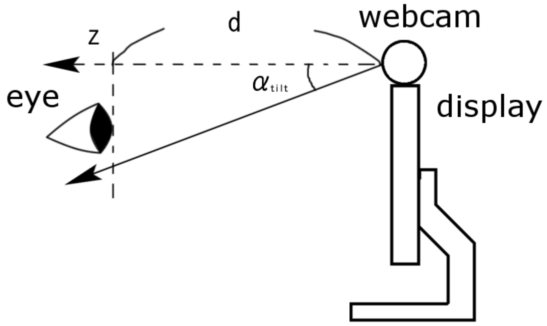


Figure 4 Tilting cameras against the display

By considering this model, we are able to approximate the pupil contour observed by cameras with an ellipse.

Figure 3 shows an example of approximating of the pupil with an ellipse. In Figure 3, ψ is the angle of rotation, *a* is the major radius, and *b* is the minor radius. We calculate θ and ϕ using these parameters as shown below.

$$\theta = a\cos\left(\frac{b}{a}\right) \tag{1}$$

$$\phi = \psi \text{ or } \pi - \psi \tag{2}$$

We must determine the value of ϕ uniquely. Thus, we set webcams tilted downward relative to the display as shown in Figure 4. The user’s eyes are then above the optical axis

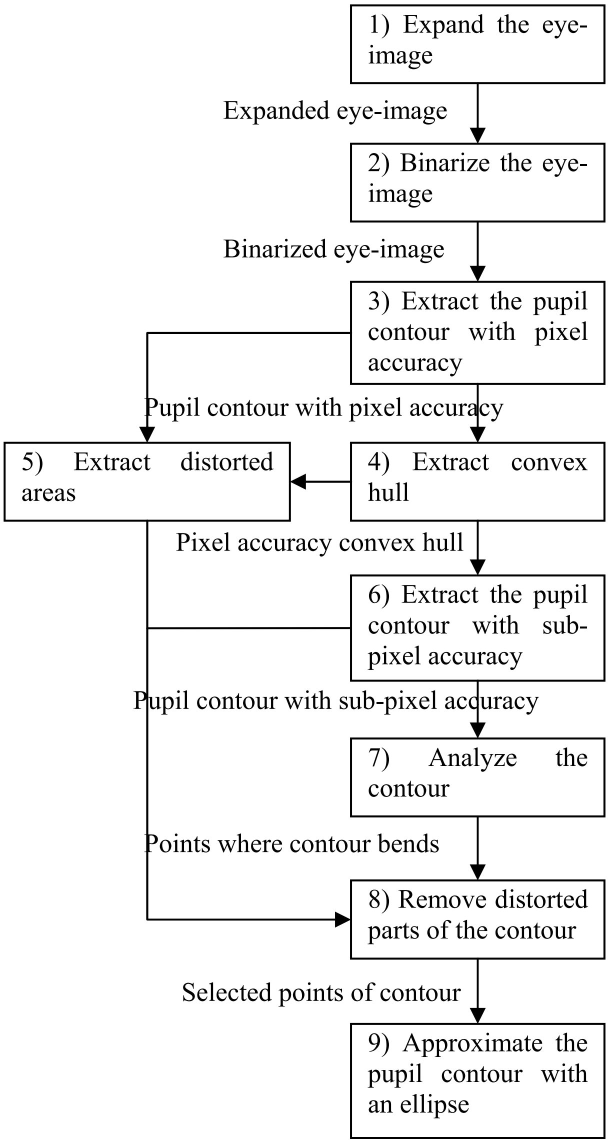

1) Expand the eye-image
Expanded eye-image
2) Binarize the eye-image
Binarized eye-image
3) Extract the pupil contour with pixel accuracy
Pupil contour with pixel accuracy
5) Extract distorted areas
4) Extract convex hull
Pixel accuracy convex hull
6) Extract the pupil contour with sub-pixel accuracy
Pupil contour with sub-pixel accuracy
7) Analyze the contour
Points where contour bends
8) Remove distorted parts of the contour
Selected points of contour
9) Approximate the pupil contour with an ellipse

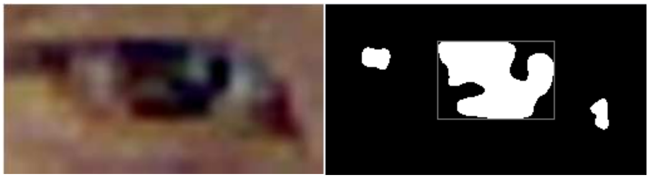
Figure 5 Image processing and geometrical processing in the proposed method

of the webcam. We can bind the range of the value, $-\frac{\pi}{2} < \phi < \frac{\pi}{2}$.

Ellipse approximation

Low-resolution images have strong sources of noise that distort the contour of the pupil. The contour distortion causes errors of ellipse approximation.

To reduce the approximation error, we expand low-resolution eye-images using Bicubic convolution and estimate the sub-pixel pupil contour to reduce the approximation error. Subsequently, we fit the convex hull to the pupil contour to remove the distorted parts of the pupil contour.



(a) (b)

Figure 6 Result of binarization

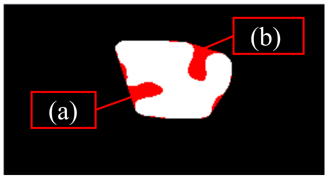


Figure 7 Examples of dents in a pupil and convex hull

In this paper, we categorize the causes of ellipse approximation error into two types: dents by light reflection, and distortions of the pupil contour by noise.

As shown in Figure 5, we apply image processing and geometrical processing to the eye-image.

Expanding the eye-image

We expand the eye-image to reduce the influence of 1-pixel error in ellipse approximation to gaze estimation. We adopt Bicubic convolution to expand the eye-image. Figure 6(a) shows an example of expanded eye image.

Binarizing the eye-image

Binarization cuts out the pupil from the eye-image using only the red channel of the eye-image. We call this R-channel. This is because the R-channel is useful for distinguishing the pupil from the other parts of the eye for Japanese people. Here, Japanese skin color has an R-channel value higher than both G and B, and the R, G, and B values are all high for the sclera because of its white color. On the other hand, the R, G, and B values are all low for the pupil because it is black. Figure 6 (b) shows the result of binarizing Figure 6 (a). In Figure 6 (b), the white areas are darker than a specified threshold. The system determines the largest white area to be the pupil. In Figure 6 (b), we treat the white area in the gray square as the pupil. In our method, we set the threshold manually.

Extracting the pupil contour with pixel accuracy

To obtain the precise location of the pupil, we need edge points. Using the information of the boundary pixels where the value of binary image changes, the pupil contour can be extracted with pixel accuracy. We assume that contiguous boundary pixels form a contour.

Fitting the convex hull to the pupil contour

In Figure 7, the white area represents the pupil area with pixel accuracy. In Figure 7, two dents, (a) and (b), represent the bright areas caused by light reflection before binarization. These distortions on the pupil contour reduce the accuracy of ellipse approximation.

Convex hulls fill dents such as (a) and (b). Figure 7 shows an example of dents filled by a convex hull. In order to ignore such dents, the system fits the convex hull to the pupil contour with pixel accuracy. However, the convex hull fills dents with line segments that cannot match the true contour of the pupil perfectly. Given that the convex hull represents the rough shape of the pupil, the system uses the convex hull of the pupil contour with pixel accuracy when analyzing the contour process.

Extracting the dent index

The dent index shows the start points and end points of the dents filled by the convex hull. This is calculated by matching the pupil contour with pixel accuracy with its convex hull. The dented portions of the contour do not match with its hull convex. The both ends of these portions of the convex hull are the dent index.

The system uses the dent index to remove the areas filled by the convex hull in the ellipse approximation phase. As the convex hull fills dents in the pupil contour with line segments, the filled areas do not match the true pupil contour completely. In the ellipse approximation process, these areas act as sources of noise. Therefore, the system should remember the dent index and remove filled areas in the ellipse approximation phase.

Estimating pupil contour with sub-pixel accuracy

In this step, pupil contour is estimated with sub-pixel accuracy. As pixel values are discrete, we apply linear convolution over the x-axis to obtain the pupil contour with sub-pixel accuracy. Figure 8 shows an example. The equations to calculate a point on the pupil contour with sub-pixel accuracy $p_{sub} = \left(x_{sub}, y_{sub}\right)$ are as follows.

$$x_{sub} = \frac{x_0\left|th - v_1\right| + x_1\left|th - v_0\right|}{\left|th - v_0\right| + \left|th - v_1\right|} \qquad (3)$$

$$y_{sub} = y_0 = y_1 \qquad (4)$$

($p_0 = \left(x_0, y_0\right), p_1 = \left(x_1, y_1\right)$)

In Equations (3) and (4), v_0 is the intensity value of R-channel of p_0 , v_1 is the value of p_1 , and *th* is the threshold value.

The system fills the area of the pupil contour dented by the convex hull with sub-pixel accuracy.

Analyzing the contour

In order to remove the distortion by the eyelids, the system analyzes the pupil contour with sub-pixel accuracy whose dents are filled by the convex hull. The eyelid covers the pupil and distorts portions of the pupil contour. These covered areas are almost horizontal, and located between two bending points where the pupil contour bends sharply.

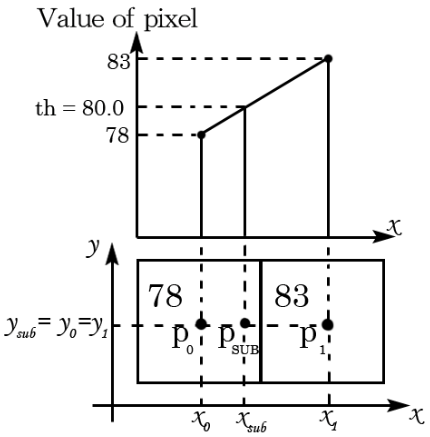


Figure 8 Pupil contour with sub-pixel accuracy

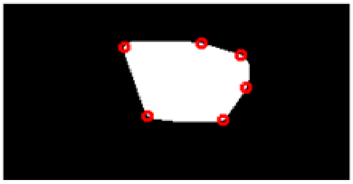


Figure 9 The bending points

To select bending points, the system calculates κ_i , the angle at point *i*. In equation (5), p_i is the i th point of the pupil contour with sub-pixel accuracy.

$$\kappa_i = \sum_k^n \frac{a\cos\left(\frac{p_i p_{i+k} \cdot p_i p_{i-k}}{\left|p_i p_{i+k}\right|\left|p_i p_{i-k}\right|}\right)}{2n} \qquad (5)$$

The point at which κ shows a local minimum becomes a candidate bending point. However, many points around the sharply bent point tend to show a local minimum. Therefore, we place two constraints. First, the bending points must have an interval longer than the threshold. Second, bending points must have a small κ . In particular, the bending point with a κ smaller than all other bending points must have the smallest κ among all contour points. Figure 9 shows an example of the result of contour analysis. Circled points represent the bending points. In this example, the length of the pupil contour with sub-pixel accuracy is 244 points where the threshold is 30 points.

Removing distorted portions of the contour

The system removes the distorted areas within the pupil contour with sub-pixel accuracy using the dent index and the bending points. First, the system simply removes areas where the dent index is evident. On the other hand, the bending points do not show distorted areas but indicate candidates of the start or end point of the area distorted by the eyelid. Therefore, the system removes areas that are nearly horizontal between two bending points. Specifically,

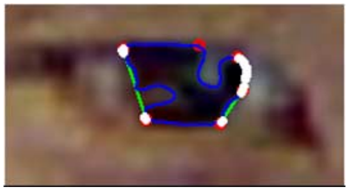


Figure 10 Selected points (White points)

User	x	y
A	0.46	0.62
B	0.65	0.79
C	0.57	0.35
D	0.57	0.61
E	0.70	0.54
Avg.	0.59	0.58

Table 1 Correlation between the estimation result and error

the system selects the areas to remove as follows. First, it selects two adjacent bending points. Second, it calculates the gradient of a line that includes the selected bending points. Finally, if the gradient is less than the threshold value, the system removes the area between the selected bending points. Figure 10 shows an example of the result of this phase. The remaining white points should comprise the portions of the true contour of the pupil.

Approximation of the pupil contour with an ellipse

In our method, the system approximates the pupil contour with an ellipse using the least square method (LSM). The sample points for the LSM are the points selected in the phase of removing distorted parts of the contour. As a result, the system obtains the major radius a, the minor radius b, and the rotation angle ψ of the approximated ellipse.

Gaze point estimation

The direction of eye-gaze is estimated in this step. Here, the eye-gaze point refers to location toward which the user is looking. In our method, the eye-gaze point for each eye is calculated using equations (1) and (2). Subsequently, the system determines the gaze point as the midpoint of the two eye-gaze points.

Calibration

The calibration process compares measurements and true values. The accuracy of measurements increases as calibration reduces sources of noise with systematic trends. In terms of eye tracking, calibration refers to the adjustment of parameters in gaze estimation. The eye-tracking system measures the difference between raw measurements and true values by measuring the gaze of the user gazing at known points. For examples, Nagamatsu et al. [6] and Ohono[8] et al. adjust the angle between the measured optical axis of an eye and the gaze direction. Chen et al. [3] adjust the distance between the center of eyeball and the center of the pupil.

User	x	y
A	9.3×10^{-10}	5.7×10^{-11}
B	1.6×10^{-11}	8.5×10^{-20}
C	1.3×10^{-7}	2.6×10^{-6}
D	2.7×10^{-9}	2.0×10^{-9}
E	4.2×10^{-9}	4.3×10^{-7}

Table 2 P-value of the t-test

	Max [cm]	Min [cm]	Range [cm]
X	11.7	-8.5	20.2
Y	5.1	-16.0	21.2
Z	89.6	36.2	53.4

Table 3 Width of users' head movements

In our method, the system adjusts gaze direction directly using a linearly approximated curve for the scatter plot of estimated direction and error in degrees.

Table 1 shows the correlation between the estimated result and error in our experiment. The estimated result is the gaze direction relative to the optical axis of the camera. The error is the angle between the estimated gaze line and the line from the midpoint of the two pupil centers to the correct point. In our experiment, we established 9 target points. We obtained 225 estimated results for the calibration. These results consist of 5 results per 9 target points for each of 5 users. Table 2 shows p-values of the t-test. Here, the null hypothesis is that no correlation exists between the estimated results and errors. As shown in Table 2, we can reject the null hypothesis at a significance level of 1%. Therefore, we expect calibration with linear approximation to raise the accuracy of gaze estimation. In fact, this is precisely what we observe in our next experiment.

EXPERIMENTAL EVALUATION

Procedure

We evaluated our method with and without calibration. The experimental procedures of both cases were almost the same. There were 5 subjects. All were males in their twenties without glasses. Only one subject wears contact lens. They sat in a chair as they would when normally using a PC. Table 3 shows the range of the users' head movements.

The subjects gazed at 9 points on the display, and the system estimated their gaze points. The coordinates of the target points were (50,50), (800,50), (1550,50), (50,600), (800,600), (1550,600), (50,1150), (800,1150), and (1550,1150) on a display with a resolution of 1600 × 1200 pixels.

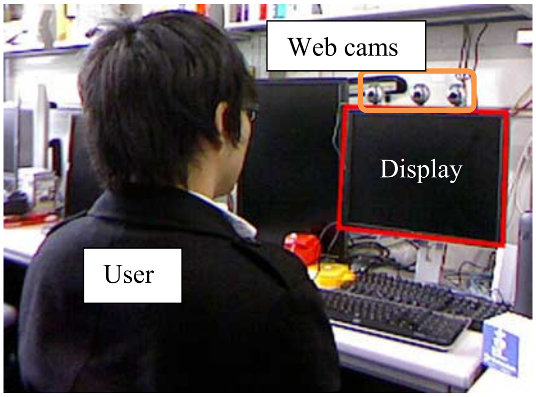


Figure 11 Environment of evaluation

User	$\overline{\lvert\theta_x\rvert}$ [deg]	σ_x [deg]	$\overline{\lvert\theta_y\rvert}$ [deg]	σ_y [deg]
A	5.4	7.3	2.0	2.5
B	3.5	4.2	2.2	2.6
C	3.6	4.0	2.0	1.9
D	3.3	4.1	1.5	2.3
E	3.3	3.9	2.0	2.2
Avg.	3.8	4.7	1.9	2.3

Table 4 Mean and standard deviation of the error degree in the case without calibration

In the case without calibration, each user observed 9 target points for 10 times. In total, we obtained 450 eye-gaze results from five users.

In the case with calibration, we divided the results into two sets: the learning and test sets. Each set included 225 results. These sets consisted of 5 results per target point. The system adjusts the result, i.e., applies calibration, using the learning set. Subsequently, another set, i.e., the test set, is used for testing after calibration.

Evaluation environment

Figure 11 shows the evaluation environment. We implemented a gaze-tracking system with three webcams (Logicool C500), one 20-inch display, and one PC (Windows 7, Core i7 950 3.07 GHz, Memory 12 GB). These cameras operated with a resolution of 640 × 480 pixels. The mean resolution of eye-images was 21.5 × 14.4 pixels. The software ran with a single thread.

Two webcams measured the 3D position of the pupil center. The other webcam captured eye-images for gaze-direction estimation.

Results

In the case without calibration

Table 4 shows the results without calibration. Subject E wears contact lens. We divide the error into two axes: horizontal and vertical. The horizontal axis is shown as x and the vertical axis is shown as y. $\overline{\lvert\theta_x\rvert}$ and $\overline{\lvert\theta_y\rvert}$ show means of the absolute error value. σ_x and σ_y show the standard deviation of the error value.

	Without calibration		With calibration	
User	$\overline{\lvert\theta_{Tx}\rvert}$ [deg]	$\overline{\lvert\theta_{Ty}\rvert}$ [deg]	$\overline{\lvert\theta_{Tx}'\rvert}$ [deg]	$\overline{\lvert\theta_{Ty}'\rvert}$ [deg]
A	6.0	2.0	4.5	1.7
B	3.3	2.1	2.4	1.3
C	3.3	1.6	2.9	1.3
D	3.1	1.7	2.3	1.6
E	3.1	1.8	3.0	1.5
Avg.	3.8	1.8	3.0	1.5

Table 5 Mean error of the test set in cases without and with calibration

	Without calibration		With calibration	
User	σ_{Tx} [deg]	σ_{Ty} [deg]	σ_{Tx}'[deg]	σ_{Ty}'[deg]
A	8.0	2.6	6.1	2.0
B	4.1	2.5	3.2	1.6
C	4.0	1.9	3.5	1.6
D	3.5	2.0	2.8	1.7
E	3.6	1.8	3.5	1.7
Avg.	4.6	2.2	3.8	1.7

Table 6 Standard deviation of the test set in cases without and with calibration

In the case with calibration

Table 5 and Table 6 show the results with calibration. According to these tables, the means and standard deviations decrease after calibration.

DISCUSSION

We compare our method to the method proposed by Yamazoe et al. [16], who used low-resolution eye-images in eye tracking. The mean resolution they used was 30 × 15 pixels. In addition, their method is a model-based method. These constraints are similar to those of our method.

In their experiment, Yamazoe et al. evaluated horizontal and vertical errors extracted from five users. The mean horizontal error was 5.3°, and the mean vertical error was 7.7°.

On the other hand, in our method, the mean horizontal error is 3.0° and the mean vertical error is 1.5°. In conclusion, our method reduced horizontal and vertical errors by 43.4 and 80.6%, respectively, compared to the method proposed by Yamazoe et al.

CONCLUSION

Gaze estimation with low-resolution eye-images is indispensable for achieving low-cost eye tracking. However, low-resolution eye-images have strong sources of noise distorting the pupil contour. This distortion results in inaccurate eye tracking. In this paper, we propose a detailed model-based eye-tracking method for low-resolution eye-images. We adopt image processing and geometrical processing to remove distortions from the pupil contour in low-resolution images. The resolution of eye-images we used was 21.4 × 14.4pixels and thus sufficiently low. The mean errors were 3.0° horizontally and 1.5° vertically. Our method reduced horizontal and vertical errors by 43.4 and 80.6%, respectively compared to a previous method.

REFERENCES

1. Agustin, J. S., Skovsgaard, H., Hansen, J. P. and Hansen, D. W. Low-Cost Gaze Interaction: Ready to Deliver the Promises. *Proc. of ACM CHI EA' 09: extended abstracts on Human Factors in Computing Systems.,* (2009) 4453-4458.

2. CRS: http://www.crsltd.com/

3. Chen, J., Tong, Y., Gray, W., and Ji, Q. A Robust 3D Eye Gaze Tracking System using Noise Reduction. *Proc. of Eye Tracking Research & Applications Symp.*, (2008) 189-196.

4. Duchowski, A.T. A Breadth-First Survey of Eye Tracking Applications. *Behavior Research Methods, Instruments, and Computers*, 34, 4 (2002), 455-470.

5. Hennessey, C., Noureddin, B., and Lawrence, P. A single Camera Eye-Gaze Tracking System with Free Head Motion. *Proc. of Eye Tracking Research & Applications Symp.*, (2006), 87-94.

6. Nagamatsu, T., Kamahara, J., Iko, T., and Tanaka, N. One-point Calibration Gaze Tracking based on Eyeball Kinematics using Stereo Cameras. *Proc. of Eye Tracking Research & Applications Symp.*, (2008) 95-98.

7. Nagamatsu, T., Kamahara, J., and Tanaka, N. Calibration-free Gaze Tracking using a Binocular 3D Eye-model," *Proc. of ACM Conf. on Human Factors in Computing Systems*, (2009) 3613-3618.

8. Ohono, T. and Mukawa, N. A Free-head, Simple Calibration, Gaze Tracking System that Enables Gaze-Based Interaction. *Proc. of ACM Eye Tracking Research & Applications Symp.*, (2004) 115-122.

9. Ono, T., Okabe, T., and Sato, Y. Gaze Estimation from Low Resolution Images. *LNCS*, 4319 (2006) 178-188.

10. Shih, S.W. and Liu, J. A Novel Approach to 3-D Gaze Tracking Using Stereo Cameras. *Proc. of IEEE Trans. on System, Man, and Cybernetics*, 34, 1 (2004) 234-245.

11. SMI: http://www.smivision.com/

12. Takegami,T., Goto, T. and Ooyama. G., An Algorithm for Model-Based Stable Pupil Detection for Eye Tracking System, *IEICE Trans. Information and Systems, vol.J86-D-II(2)* (2003) pp.252-261.

13. Tobii: http://www.tobii.co.jp/japan/home.aspx

14. Tsuji, T., et al., Iris Detection Using LMedS for Eye Tracking System, *Proc. MIRU2004* (2004) pp.I-684-689.

15. Wang, J.G., Sung, E., and Venkateswarluye, R. Eye Gaze Estimation from a Single Image of One Eye. *Proc. of IEEE Int'l. Conf. on Computer Vision*, 1 (2003) 136-143.

16. Yamazoe, H., Utsumi, A., Yonezawa, T., and Abe, S. Remote Gaze Estimation with a Single Camera Based on Facial-Feature Tracking without Special Calibration Actions. *Proc. of Symp. on Eye Tracking Research & Applications*, (2008) 245-250.