



A Proposal of Eye Glance Input Interface Using Smartphone Built-in Camera

Yu Saiga^(✉), Yu Matsumoto, Kazuyuki Mito, Tota Mizuno,
and Naoaki Itakura

The University of Electro-Communications,
1-5-1 Chofugaoka, Chofu, Tokyo, Japan
yusaiga@uec.ac.jp

Abstract. For character input in smart devices with small screens, there exist problems such as erroneous input due to the small size of each button or even though the input screen is large, the information presentation screen is small. To solve these problems, in a previous research, the eye-gaze input interface was studied as an input method that did not depend on the touchscreen area.

The purpose of this study is to develop an eye-gaze input interface using the eye-glance input method, which uses a glance eye movement and method of estimating the eye-gaze movement by analyzing the image of the area near the eyeball. The motion was discriminated by taking an image using the built-in camera of a smartphone to rapidly look at the four corners of the smartphone and calculate the moving distance of the eyeball, which was divided into horizontal and vertical components using the optical flow function in Open CV library. In the experiment, we obtained data by asking the subjects to look at the reciprocal of quick gaze movements at each of the four corners and the center of the screen. As a result of data analysis, characteristics of waveforms were obtained for the movements looking at the four corners, and we could determine which of the four corners the users were looking at. Furthermore, it was suggested that an operation with eight degrees of freedom can be performed using only one input by dividing the input into two types according to the length of the time required for the input operation.

Keywords: Line-of-sight input system · Eye-gaze input · Input interface

1 Introduction

With the development of technology in recent years, information terminals carried by individuals are changing from being bulky in size, such as laptops, to small ones such as smartphones and tablets. Input to a small-screen device, such as a smartphone, is generally provided using a touch panel. However, when many options such as character input are required, the input target displayed on the small screen becomes considerably small. Therefore, an erroneous input might occur [1]. However, when the size of the input target is increased to reduce the erroneous input, the number of options that can be presented decreases. To solve these problems, various input methods and screen

designs for smart devices have been studied. In this study, we focus on line-of-sight-based input methods that do not depend on the touch area.

The line-of-sight input is generally used as a gaze position input method [2]. Although the input accuracy has been improved because of the development of research, it is still difficult to distinguish between input intentions during a gaze, and it is also difficult to implement the method on small devices such as smartphones.

However, a method that uses the eyeball movement as an input has been proposed. Ito et al. [3] proposed the eye-glance input interface that used a quick reciprocation of the eye movement as an input action. In this method, the input operation is the reciprocation of the line-of-sight movement along the oblique direction that occurs when returning after looking at any of the four corners of the screen. Therefore, using this method, it becomes easy to measure even small screens, such as smartphones, without requiring high measurement accuracy.

Akehi et al. [4] proposed a method to acquire input images in a non-contact manner by acquiring the face image of a user by employing a web camera and measuring eye movement using Open CV, which is an open source library. This suggests that gaze movement can even be measured using only a smartphone without using external devices such as electrodes.

The purpose of this study is to implement an eye-glance input interface for acquiring facial images and analyzing them using a smartphone.

2 Eye-Movement-Measurement Method

2.1 Optical Flow

Similar to a previous study, we used Open CV's optical flow calculation library for performing the eye-movement measurement. Optical flow is a method that compares two images that are temporally continuous, and it expresses the motion of the object in the image as a vector. Figure 1 shows the measurement range. Based on the coordinates and radius of the face obtained via face detection, a rectangular area near the eyeball was used as the measurement area, and the eye movement was measured by equally disposing 225 measurement points (15×15) in the measurement area. The vector obtained using the difference in the movement speed between the eyeball and other parts was divided into horizontal and vertical components, and those vectors with the largest 10% were averaged to obtain one evaluation value.

2.2 Discrimination Algorithm

Figure 2 shows an example of the waveform obtained by the method shown by Akehi et al. [4] the part enclosed by the rectangle corresponds to the eye movement. Along the horizontal direction, the right is positive, and along the vertical direction, the top is positive. The waveform obtained by the eye-glance input has the characteristic that two opposite waveforms are generated along each of the horizontal and vertical directions, and a short dwell time is observed between round trips. In addition, the number of

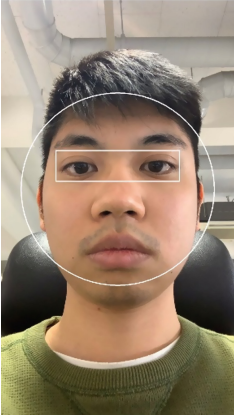


Fig. 1. Measurement area

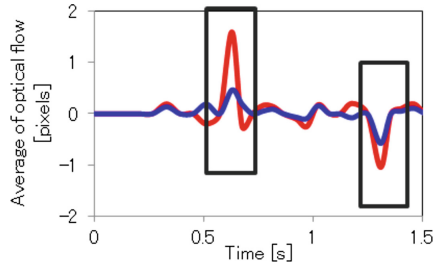


Fig. 2. Example of the upper right input waveform

options to be displayed at once can be increased to eight by dividing the input into two types according to the length of the time required for one eye-glance operation. [5]

3 Experiment

For the experiment, we used an application that performed many tasks ranging from the display of the experiment screen, acquisition of images, to the calculation of the optical flow. The smartphone used an inbuilt-camera with a time-resolution function of 30 fps; however, it was approximately 11 fps when the application was running. After the experiment, the optical-flow data calculated using the smartphone were analyzed on a PC.

3.1 Measurement Conditions

The orientation of the smartphone was set to “vertical”, and installation method was “fixed” and “gripped”. The subject was in a sitting posture, and the position of the smartphone was adjusted in advance such that the subject could easily observe the screen.

The subject performed eye-glance input operations once at each of the four corners at intervals of 2–3 s under the two conditions of fixed and gripped respectively.

3.2 Analysis Method

Based on the algorithm by Akehi et al. [4], we performed face detection for each acquired image and calculated the optical flow in the rectangular area around the eyeball. The measurement points of the optical flow were distributed as 15 points along the length and 15 along the width of the measurement area, and the gaze movement was measured by averaging the top 10% of the obtained vectors.

4 Results and Discussion

4.1 Impact on Grip Information and Solutions

Figure 3 shows an example of the optical-flow waveform during the input operation. In the fixed state, a characteristic waveform that was considered as eye movement was obtained; however, noise occurred in the gripped state. The noise might be attributed to camera shake and could hinder the eye-movement measurement.

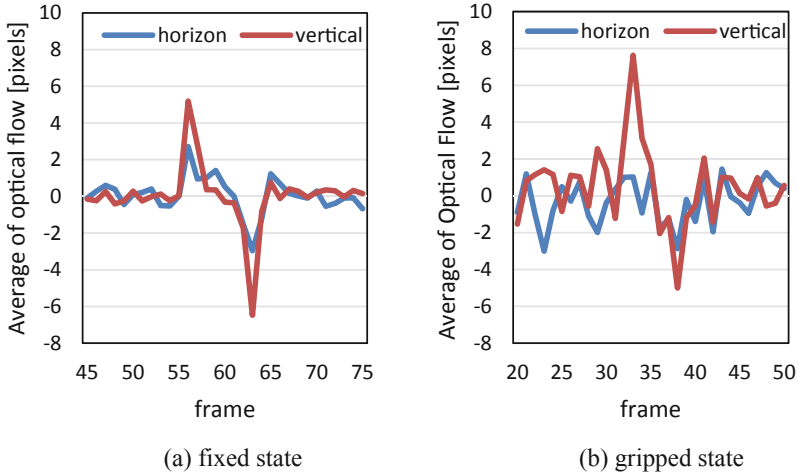


Fig. 3. Example of optical-flow waveform

Therefore, we calculated the variance by considering that only the eyeball moved. Notably, the variance becomes large when the eye moves, while the entire screen moves when the camera shakes. Figure 4 shows the variance values for the same conditions and same section. Therefore, the characteristic value of the variance along the vertical direction was observed during the movement of the line-of-sight, even in the grasping state. This suggests that eye movement can be estimated by calculating the variance value even in the grasping state.

4.2 Gaze-Movement Discrimination

The variance along the vertical direction during eye movement was greater than that during the case of non-eye movement. Therefore, the eye movement can be identified by setting a threshold along the vertical direction. However, in Fig. 4, no large waveform was observed along both the horizontal and vertical directions. Such examples are often seen during other eye movements. This might be because the smartphone had a long shape along the vertical direction, and the four corners could be seen without horizontally moving the eye. Therefore, the eye movement might be determined by setting a threshold along the vertical direction and then making a positive or negative judgment along the horizontal direction.

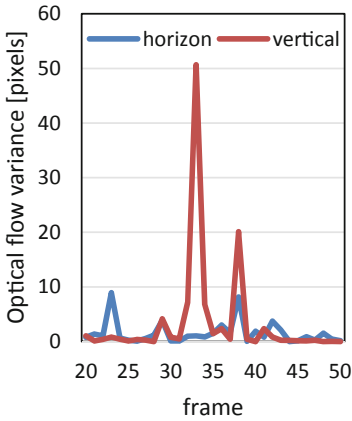


Fig. 4. Optical-flow variance

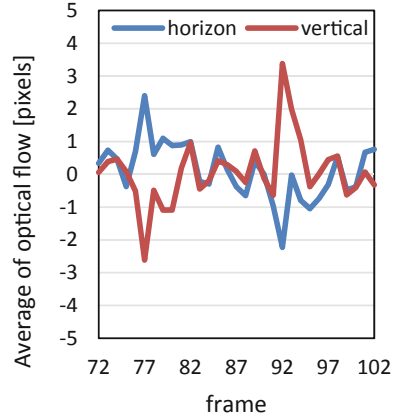


Fig. 5. Optical flow for extended eye movement

4.3 Long/Short Input

Figure 5 shows an example of the optical-flow waveform when the eye gaze is input for a long time. In Fig. 3(a), there is a difference of 7 frames in the peak value of the line-of-sight movement during reciprocation, whereas in Fig. 5, there is a difference of 15 frames. Consequently, as observed in a previous study [5], the input could be divided into two types according to the length of the time required for gaze movement, and by performing this method at the four corners, we could operate with one input and eight degrees of freedom.

5 Conclusion and Future Work

In this study, we analyzed the obtained waveforms to develop an eye-glance input interface that performed image acquisition and analysis on a smartphone. In the past, images acquired using a smartphone were analyzed on a PC; however, we showed that the eye movement could be determined using only a smartphone by calculating the optical flow within the smartphone. In future, we will conduct real-time discrimination experiments on smartphones and aim at practical applications of the line-of-sight input.

References

1. Siek, K.A., Rogers, Y., Connelly, K.H.: Fat finger worries: how older and younger users physically interact with PDAs. In: Costabile, M.F., Paternò, F. (eds.) INTERACT 2005. LNCS, vol. 3585, pp. 267–280. Springer, Heidelberg (2005). https://doi.org/10.1007/11555261_24
2. Abe, K., Nakayama, Y., Ohi, S., Ohyama, M.: A support system for mouse operations using eye-gaze input. *IEEJ Trans. Electron. Inf. Syst.* **129**(9), 11–1713 (2009)
3. Ito, Y., Kazama, Y., Itakura, N., Mizuno, T., Mito, K.: Research of eye glance input interface. *IEICE Tech. Rep.* **112**(483), 17–20 (2013)
4. Akehi, K., Matsuno, S., Itakura, N., Mizuno, T., Mito, K.: Non-contact eye-glance input interface using video camera. *J. Signal Process.* **21**(4), 207–210 (2017)
5. Matsuno, S., Ito, Y., Akehi, K., Itakura, N., Mizuno, T., Mito, K.: A multiple-choice input interface using slanting eye glance. *IEEJ Trans. Electron. Inf. Syst.* **137**(4), 621–627 (2017)