Gaze-based Command Activation Technique using Two-level Stroke

Toshiya Isomoto
University of Tsukuba
Ibaraki, Japan
isomoto@iplab.cs.tsukuba.ac.jp

Shota Yamanaka
Yahoo! Japan Corporation
Tokyo, Japan
syamanaka@yahoo-corp.jp

Buntarou Shizuki
University of Tsukuba
Ibaraki, Japan
shizuki@iplab.cs.tsukuba.ac.jp

ABSTRACT

We show a gaze-based command activation technique which uses a two-level stroke as a gesture. A two-level stroke is a simple gesture such as a horizontal then a vertical stroke or a vertical then a horizontal stroke. An object on which users want to activate a command is selected with dwell-based target selection. Thus, the command is activated by dwelling on an object and then moving the gaze to form a two-level stroke. As a result of an experiment, the success rate of command activation is 85.8% and the time taken for a command activation is 956 ms.

Author Keywords

Gaze-based interaction; gaze gesture; eye tracking; gaze movement; GUI.

CCS Concepts

•**Human-centered computing** → **Human computer interaction (HCI);** *Gestural input;* User studies;

INTRODUCTION

In gaze-based interaction, gesture-based manipulation is preferred to dwell-based manipulation for a command activation [3, 4]. Using dwell-based manipulation that users can select a target by dwelling on it (i.e., looking at it for a certain time), they can activate a command by performing dwell-based target selection at least twice; dwelling on an object they want to activate and then dwelling on a command label displayed after selecting the object. This process seems to be easy because of similarity to a command activation using mouse-based interaction; a command is activated by right-clicking on an object and then left-clicking on a command label displayed after right-clicking on the object. However, researchers reported that gesture-based manipulation surpasses dwell-based manipulation in the time taken for a command activation (the activation time) and the success rate [3, 4]. These are because dwell-based target selection faces the Midas-touch problem [6], which is a user's unintentional selection caused by mainly short dwell time (the time users need to look). On the basis of these results, gesture-based manipulation has been preferred as a command activation technique.

In gesture-based manipulation, a user can activate a command by performing a gesture defined beforehand. Drewes and Schmidt [2] use a gesture that passes a corner of the display to activate a command, for example, moving right and left alternately three times or moving four corners in the anti/clockwise direction. Rajanna and Hammond [9] proposed nine gestures: a one-level stroke (e.g., moving from the left to the right) or a 'U'-shaped stroke starting from the top left corner of the screen. Istance et al. [5] showed a technique that displays a semi-transparent region to help users to easily activate a command. Gestures used in this research are a two-level stroke and a three-level stroke. Møllenbach et al. [8] showed a one-level stroke, which is a gesture such as a right to a left gaze movement.

Compared with other manipulations such as one using a mouse, these manipulations do not allow users to select an object on which to activate the command. To allow users to select an object, a combination of dwell-based target selection and gesture-based manipulation is used. Urbina et al. [10] used a pie menu in a gaze-based interaction to activate a command. The pie menu was displayed after fixation (short time dwell) on an object, and a command was activated when the gaze crossed the edge of the menu. In this technique, the menu may be unintentionally displayed because of a short dwell time. In contrast, Delamare et al. [1] used a long dwell time, such as 2 s, to prevent the Midas-touch problem. The gestures used in this technique were complex such as a curved gesture that is difficult to activate a command without displaying visual guidance that helps users move their gaze. The problem of these techniques which use the combination of dwell-based target selection and gesture-based manipulation is the necessity of displaying visual guidance and a long dwell time to prevent the unintentional display of them.

In this paper, we show a gesture-based command activation technique; the users select a target by using dwell-based target selection and then activate a command by using a two-level stroke, which consists of two continuous one-level strokes, a horizontal one then a vertical one, or a vertical one then a horizontal one. The two-level stroke is simple and easy to perform; thus, visual guidance for helping users activate command is not necessary. As shown in Figure 1, our technique allows users to activate two types of command when users perform a two-level stroke:

CHI '20, April 25–30, 2020, Honolulu, HI, USA

ISBN 978-1-4503-6708-0/20/04. . . $15.00

DOI: https://doi.org/10.1145/3313831.XXXXXXX

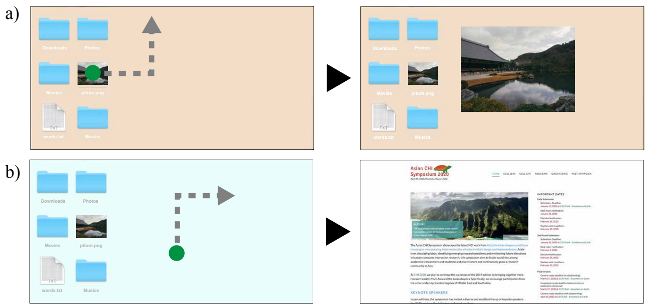

Figure 1. Applications of our technique. a) A right then up stroke starting from a file icon opens it. b) An up then right stroke on a window or the desktop switches the work-space.

a) a command for an object (e.g., opening a file or copying an icon) when the user's dwell is detected on the object.

b) a command for the desktop or a window (e.g., switching the workspace or showing the previous web page) when the user's dwell is detected outside an object.

The gestures onto which commands are mapped are eight: a gesture that user moves the gaze upward then right ($U \rightarrow R$), upward then left ($U \rightarrow L$), downward then right ($D \rightarrow R$), downward then left ($D \rightarrow L$), right then upward ($R \rightarrow U$), right then downward ($R \rightarrow D$), left then upward ($L \rightarrow U$), and left then downward ($L \rightarrow D$).

GESTURE DETECTION SYSTEM

The detection system of a two-level stroke works as follows (Figure 2).

a) Once the point/object on which a command should be activated is selected by dwelling, an invisible straight path for detecting the first-level stroke is generated, with the center at a point on which a user dwelled (dwell point) and a width W_{path}. In our technique, a dwell was detected when the gaze stays in 5 mm for the dwell time (T_{dwell}).

b) The first-level stroke is detected when the gaze moves horizontally/vertically in the path longer than a threshold D_{thld} from the dwell point.

c) The path for detecting the second-level stroke is generated at the point P where the gaze deviates from the first path, with the center at P and the width W_{path}.

d) The second-level stroke is recognized when the gaze moves vertically/horizontally in the path longer than D_{thld} from P. After detecting the second-level stroke, the command is activated on the point/object.

The two-level stroke is a simple gesture. Using a simple gesture, unintentional detection of a gesture may occur than using a complex gesture because the simple gaze movements are likely to occur in daily life. For example, while reading a sentence, the gaze is likely to move left to right and then, if the gaze un/intentionally moves upward or downward, the command is activated although the user just reads the sentence (i.e., $R \rightarrow U$ or $R \rightarrow D$ might occur). Therefore, the T_{dwell}, W_{path}, and D_{thld} should be carefully determined.

To make the detection system robust to unintentional two-level stroke, we determined the T_{dwell}, W_{path}, and D_{thld} using the gaze trajectories collected through user studies. The gaze trajectories were collected when users move their gaze diagonally, when users move their gaze to form two-level stroke, when users manipulate applications with a dwell-based target selection technique, when users manipulate applications with a gesture-based command activation technique. As the results, we adopted 506 ms as T_{dwell}, 34.6 mm as W_{path}, 116.0 mm as D_h (D_{thld} for a horizontal gaze movement), and 66.9 mm as D_v (D_{thld} for a vertical gaze movement). Moreover, we limited the time that users should finish to move their gaze to 0–733 ms; the time taken for more than 773 ms is not recognized as a gaze movement for a command activation. Because of the non-necessary of visual guidance and robustness to unintentional detection of two-level stroke, we can adopt a short dwell time without the effect of the Midas-touch problem; no unintentional displaying visual guidance and no unintentional command is activated on an object unintentionally selected.

PERFORMANCE OF OUR TECHNIQUE

We conducted an experiment to investigate the success rate and activation time when the gesture detection system is used.

Experimental Conditions and Tasks

Ten volunteers (all male, including students in our laboratory) aged 21–25 participated. We used a Tobii Eye Tracker 4C[1] as an eye tracker. A 24-inch non-glare display with a resolution of 1,920 × 1,080 pixels was used to prevent reflections. The participant's head was positioned approximately 60 cm from the display.

The task was to perform a two-level stroke from the center of the display (the circle that letter 'A' is written in the left of Figure 3). We mapped eight gestures onto the letter 'A'–'H' as shown in the right of Figure 3; the participants could refer to this mapping freely during this experiment. We asked the participants to push the 'Enter' key before starting to move their gaze in the instructed gesture and re-push the key after finishing to move their gaze. After re-pushing the key, the participants were notified of the detection result (success or failure) by a sound; if the detection was successful, the next letter appears; otherwise, the same letter appears again. A trial was to perform two-level stroke until the stroke is correctly recognized and the session consisted of eight trials (i.e., eight gestures). We asked them to perform six sessions; the first session is a practice session and the last five sessions are test sessions. The order of the gestures was randomized in a session.

The participants calibrated the eye tracker at the beginning of the experiment. After that, they practiced each gesture of two-level strokes at least five minutes and performed the practice session. After the practice session, they conducted five test sessions with at least one minute rest between each session. In total, we collected data of 400 two-level strokes (8 gestures × 5 sessions × 10 participants). The experiment took approximately 25 minutes per participant.

[1]https://gaming.tobii.com/product/tobii-eye-tracker-4c/ (accessed on January 17, 2020)

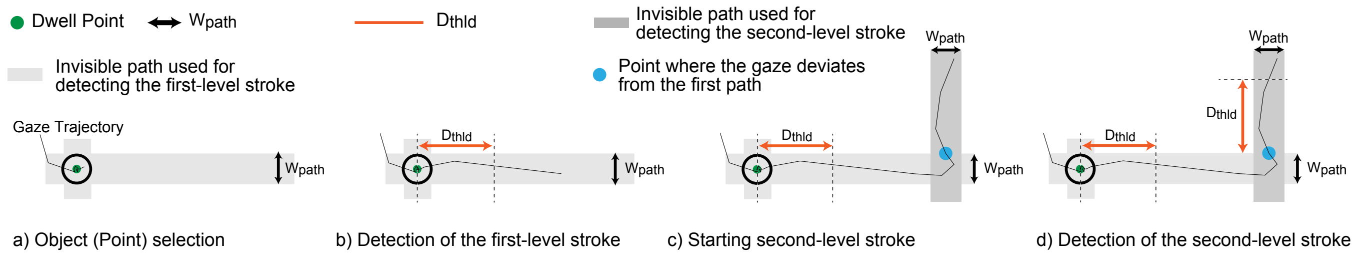

Figure 2. The process of detection of two-level stroke.

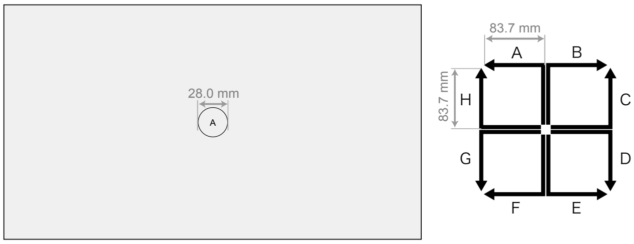

Figure 3. The displays used in the experiment. The left figure shows the position of the circle and instruction of the gesture. The right figure shows the mapping between gestures and ‘A’–‘H’.

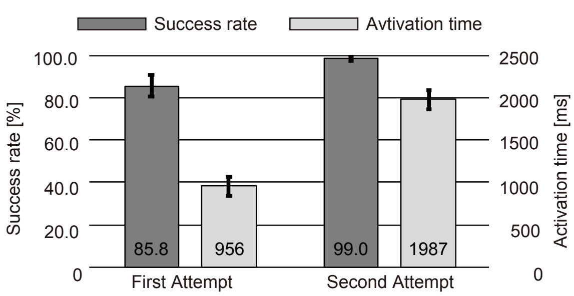

Figure 4. Performance of our technique when the participants attempt to activate a command once (left) and twice (right). Error bars indicate *SD*s.

Results

The success rate (Figure 4, dark gray graphs) averaged in all gestures and all participants was 85.8% (SD = 5.1) when the participants attempted to perform a two-level stroke once. The highest rate was 90.0%, and the lowest rate was 72.5% across participants. The average success rate when the participants attempted to perform a two-level stroke twice was 99.0%. Surprisingly, a participant whose rate was 72.5% in the first attempts had a rate of 97.5% in the second attempt. These results suggest that even when users of our system fail to activate a command, they can activate a command by attempting to do so again without being affected by unintentional manipulation. Moreover, the average activation time (Figure 4, light gray graphs) was 956 ms ($SD = 119$) for the first attempt and 1,031 ms ($SD = 112$) for the second attempt. Therefore, the participants activated a command with a success rate of 99.0% with 1,987 (1,031 + 956) ms of activation time.

ADVANCED INTERACTION: DISPLAYING VISUAL GUIDANCE

Although our technique allows users to easily activate a command without displaying visual guidance (Figure 5a–c) because of the simplicity of the gesture, users who do not remember a mapping between commands and gestures of two-level stroke can not activate a command. To support such users, visual guidance for notifying the users of the mapping is necessary (this usage of visual guidance has been also adopted in previous research in gaze-based interaction [1]). However, the techniques which adopt a combination of dwell-based target selection and gesture-based manipulation like ours face unintentional displaying of visual guidance due to the Midas-touch problem. Therefore, displaying visual guidance should be done by dwell-based target selection which never causes the Midas-touch problem.

At present, dwell time for dwell-based target selection in our technique is 506 ms. Basically, in our technique, the Midas-touch problem does not have negative effects because of robustness to unintentional detection of two-level stroke. However, if we adopt displaying of visual guidance to our technique, the Midas-touch problem has a negative effect (i.e., unintentional displaying of visual guidance). Therefore, we plan to use a long dwell time, e.g., 1.3 s (the zero dwell detection time, which we revealed from gaze trajectories collected when users manipulate applications with dwell-based target selection) to display visual guidance. The process of a command activation with visual guidance is shown in Figure 5d–g. First, users dwell on an object for 506 ms (Figure 5d). Then, to display visual guidance, the users dwell on it for more 794 ms (Figure 5e). After displaying visual guidance, the users can refer to the mapping and then perform two-level stroke to activate a command (Figure 5f and g). Note that no command is activated when the users refer to the mapping because of the advantage of our detection system.

This usage, that is, users can activate a command with/without displaying visual guidance, is similar to a marking menu [7]. Using a marking menu, users can activate a command fast without waiting for displaying the menu (Figure 5a–c in our technique) and activate a command by referring to the mapping after displaying the menu (Figure 5d–g in our technique). Although the marking menu has been established as a basic interface in other interactions such as a mouse or pen, in gaze-based interaction, it has not been established. Therefore, showing this usage is also the huge contribution of this paper.

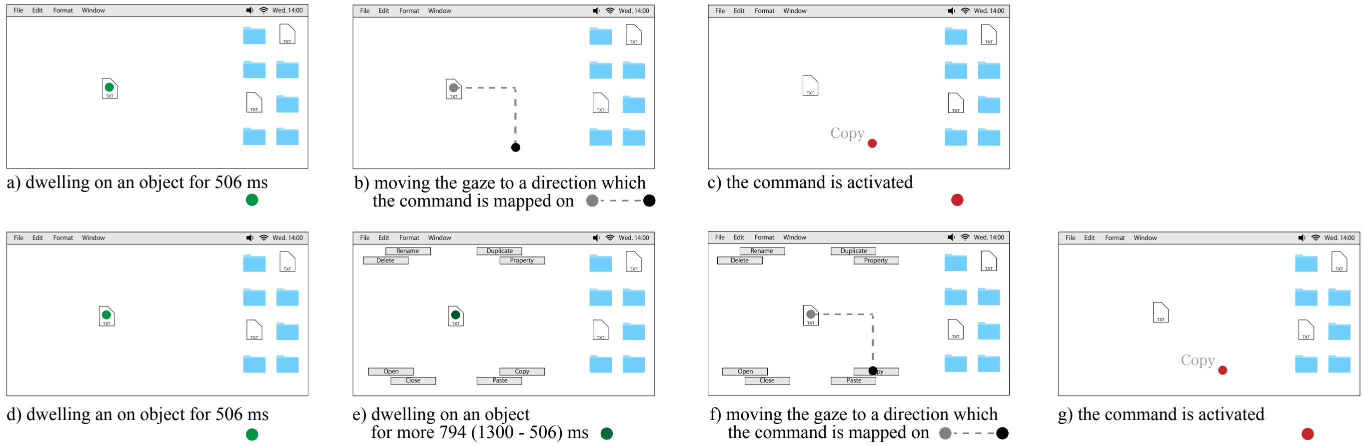

Figure 5. The usage of our technique without visual guidance (a–c) and with visual guidance (d–g).

CONCLUSION

We showed a gaze-based command activation technique which uses a two-level stroke as a gesture. A two-level stroke is a simple gesture such as a horizontal then a vertical stroke or a vertical then a horizontal stroke. An object on which users want to activate a command is selected with dwell-based target selection. Therefore, the command is activated by dwelling on an object and then moving the gaze to form a two-level stroke. As a result of an experiment, the success rate of command activation is 85.8% and the time taken for a command activation is 956 ms; users can activate a command with a 99.0% success rate and 1987 m of the time if they perform a two-level stroke twice.

REFERENCES

[1] William Delamare, Teng Han, and Pourang Irani. 2017. Designing a Gaze Gesture Guiding System. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '17)*. ACM, New York, NY, USA, Article 26, 13 pages. DOI: http://dx.doi.org/10.1145/3098279.3098561

[2] Heiko Drewes and Albrecht Schmidt. 2007. Interacting with the Computer Using Gaze Gestures. In *Proceedings of the 11th IFIP TC 13 International Conference on Human-computer Interaction - Volume Part II (INTERACT '07)*. Springer-Verlag, Berlin, Heidelberg, 475–488. http://dl.acm.org/citation.cfm?id=1778331.1778385

[3] Morten Lund Dybdal, Javier San Agustin, and John Paulin Hansen. 2012. Gaze Input for Mobile Devices by Dwell and Gestures. In *Proceedings of the 2012 ACM Symposium on Eye Tracking Research & Applications (ETRA '12)*. ACM, New York, NY, USA, 225–228. DOI: http://dx.doi.org/10.1145/2168556.2168601

[4] Aulikki Hyrskykari, Howell Istance, and Stephen Vickers. 2012. Gaze Gestures or Dwell-based Interaction?. In *Proceedings of the 2012 ACM Symposium on Eye Tracking Research & Applications (ETRA '12)*. ACM, New York, NY, USA, 229–232. DOI: http://dx.doi.org/10.1145/2168556.2168602

[5] Howell Istance, Aulikki Hyrskykari, Lauri Immonen, Santtu Mansikkamaa, and Stephen Vickers. 2010. Designing Gaze Gestures for Gaming: An Investigation of Performance. In *Proceedings of the 2010 ACM Symposium on Eye-Tracking Research & Applications (ETRA '10)*. ACM, New York, NY, USA, 323–330. DOI: http://dx.doi.org/10.1145/1743666.1743740

[6] Robert J. K. Jacob. 1990. What You Look at is What You Get: Eey Movement-based Interaction Techniques. In *Proceedings of the 8th SIGCHI Conference on Human Factors in Computing Systems (CHI '90)*. ACM, New York, NY, USA, 11–18. DOI: http://dx.doi.org/10.1145/97243.97246

[7] Gordon Kurtenbach. 1993. *The Design and Evaluation of Marking Menus*. Ph.D. Dissertation. Toronto, Ont., Canada, Canada. UMI Order No. GAXNN-82896.

[8] Emilie Møllenbach, John Paulin Hansen, and Martin Lillholm. 2013. Eye Movements in Gaze Interaction. *Journal of Eye Movement Research* 6, 2 (2013), 1–15. https://bop.unibe.ch/JEMR/article/view/2354

[9] Vijay Rajanna and Tracy Hammond. 2018. A Gaze Gesture-based Paradigm for Situational Impairments, Accessibility, and Rich Interactions. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications (ETRA '18)*. ACM, New York, NY, USA, Article 102, 3 pages. DOI: http://dx.doi.org/10.1145/3204493.3208344

[10] Mario H. Urbina, Maike Lorenz, and Anke Huckauf. 2010. Pies with EYEs: The Limits of Hierarchical Pie Menus in Gaze Control. In *Proceedings of the 2010 ACM Symposium on Eye-Tracking Research & Applications (ETRA '10)*. ACM, New York, NY, USA, 93–96. DOI: http://dx.doi.org/10.1145/1743666.1743689