Tracing Shapes with Eyes: Design and Evaluation of an Eye Tracking Based Approach

Mohammad Rakib Hasan
rakib.hasan@usask.ca
University of Saskatchewan
Saskatoon, Saskatchewan, Canada

Debajyoti Mondal
d.mondal@usask.ca
University of Saskatchewan
Saskatoon, Saskatchewan, Canada

Carl Gutwin
carl.gutwin@usask.ca
University of Saskatchewan
Saskatoon, Saskatchewan, Canada

ABSTRACT

Eye tracking systems can provide people with severe motor impairments a way to communicate through gaze-based interactions. Such systems transform a user's gaze input into mouse pointer coordinates that can trigger keystrokes on an on-screen keyboard. However, typing using this approach requires large back-and-forth eye movements, and the required effort depends both on the length of the text and the keyboard layout. Motivated by the idea of sketch-based image search, we explore a gaze-based approach where users draw a shape on a sketchpad using gaze input, and the shape is used to search for similar letters, words, and other predefined controls. The sketch-based approach is area efficient (compared to an on-screen keyboard), allows users to create custom commands, and creates opportunities for gaze-based authentication. Since variation in the drawn shapes makes the search difficult, the system can show a guide (e.g., a 14-segment digital display) on the sketchpad so that users can trace their desired shape. In this paper, we take a first step that investigates the feasibility of the sketch-based approach, by examining how well users can trace a given shape using gaze input. We designed an interface where participants traced a set of given shapes. We then compared the similarity of the drawn and traced shapes. Our study results show the potential of the sketch-based approach: users were able to trace shapes reasonably well using gaze input, even for complex shapes involving three letters; shape tracing accuracy for gaze was better than 'free-form' hand drawing. We also report on how different shape complexities influence the time and accuracy of the shape tracing tasks.

CCS CONCEPTS

• **Human-centered computing → Accessibility systems and tools**; **Usability testing**; *Interactive systems and tools.*

KEYWORDS

Eye Tracking, Gaze-based Interaction, Shape Tracing, Similarity Coefficient

ACM Reference Format:

Mohammad Rakib Hasan, Debajyoti Mondal, and Carl Gutwin. 2020. Tracing Shapes with Eyes: Design and Evaluation of an Eye Tracking Based Approach. In *11th Augmented Human International Conference (AH '20), May 27–29, 2020, Winnipeg, MB, Canada.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3396339.3396390

AH '20, May 27–29, 2020, Winnipeg, MB, Canada

ACM ISBN 978-1-4503-7728-7/20/05…$15.00
https://doi.org/10.1145/3396339.3396390

1 INTRODUCTION

The idea of gaze-based interactions has inspired the development of a number of eye tracking systems [2, 7, 16] that provide motor-impaired people with an alternative way to communicate. Such eye-tracking based systems usually integrate hardware and software solutions, where an eye-tracker samples the eye movement, and an algorithm transforms the data into gaze input to trigger other events. A rich body of research explores algorithms for transforming users' eye movement and head position data into gaze points on a computer screen [8, 9, 19]. In a traditional keyboard layout, gaze typing requires eye movements on keys that are not necessarily within close proximities to each other, and thus increases the required typing effort. This inspired research on designing a better keyboard layout for gaze-typing, where users type by producing gaze inputs on an on-screen keyboard [10, 12, 13].

We propose a sketch-based typing approach, which is motivated by the idea of sketch-based image search [1]. In sketch-based search, the user draws a shape in a sketchpad area, and the shape is searched against an image database using image recognition algorithms [1, 11] to construct a list of potential matches. The sketch-based search approach can potentially help gaze-based typing in a number of ways. First, users can draw a shape on the sketchpad and find a list of similar letters and words to choose from (Figure 1(a)). Chinese handwriting recognizers, e.g., YellowBridge [21], resemble this scenario for searching characters, but with hand-drawn shapes. Second, an on-screen keyboard can be used as a sketchpad to enable gesture-based typing (Figure 1(c)), which is a common method of typing on an on-screen keyboard in smartphones and other touch devices. Third, users can create custom shapes for authentication purposes, and shape-based commands for custom controls (Figure 1(e)).

Implementation of sketch-based typing can be challenging due to the wide variation in the position, size and curvature of the drawn shapes (i.e., shape/image recognition algorithms need to identify these shapes). A natural solution to this problem is to provide people with some 'universal shape set' (e.g., segment display for drawing characters, as in Figure 1(b)) to guide their drawing process. A concrete example is shown in Figure 1(d), where the character R is drawn on a 14-segment display. Once the user draws the shape, the corresponding segments light up and act as a feedback mechanism. An on-screen keyboard itself can act as a 'grid'-like guide for gesture-based gaze-typing (Figure 1(c)).

Our Contribution: In this paper we examine the feasibility of sketch-based gaze typing. Since tracing given shapes is fundamental to this idea, we investigated whether users can trace a given shape with their gaze input, and if so, how accurately. We designed an eye-tracking interface where users were given various shapes to trace with their gaze input. Our study results reveal that users were able to trace shapes more accurately than 'free-form' touch-based drawing (although not as accurately as touch-based tracing). We also report on how different shape complexities may influence the time and accuracy of the shape tracing tasks.

2 RELATED WORK

Here we briefly review the research on transforming eye movement data into gaze-based input to help people with motor impairments to interact with computers.

Using Eye Movements and Gaze as Input: In 1996, Gips and Oliviera developed *EagleEyes* that allows people with severe disabilities to control the computer using eye or head movements [5]. This system places electrodes around the eyes to track the movement of the eyes and head. In 2002, Majaranta and Raiha [16] compiled various challenges of designing eye-tracker based systems for text typing using gaze input. In most eye-tracking based typing systems, users locate and focus on a letter to trigger selection action based on dwell time [13, 15, 18]. Studies have explored various feedback mechanisms [14], e.g., audio and animated visuals, to provide users with an idea about the progress of the dwell time. Zheng et al. [23] explored a keyboard layout that orders letters and frequently used words on nested rings. They observed that novice users can achieve a 7 wpm typing speed using this layout, whereas experienced users may achieve a typing speed of 10 wpm.

Drawing and Authentication with Gaze Input: Early systems that attempt to draw with gaze input, directly color screen pixels based on users' gaze [4, 5, 20], which produces scribbled drawing. Hornhof et al. [7] developed an improved system, *EyeDraw*, that categorizes user's behavior into 'just looking', 'ready to draw' and 'drawing' modes. This allows users to have better control over their drawing actions. Many authentication systems have been proposed based on gaze input [3, 17]. Zakaria et al. [22] explored various ways to draw graphical passwords on a given 5 × 5 grid using gestures. Luca et al. [3] explored an eye-tracking based password authentication, where users who used gaze input on numeric keypads started to remember PINs as a shape instead of the number sequence.

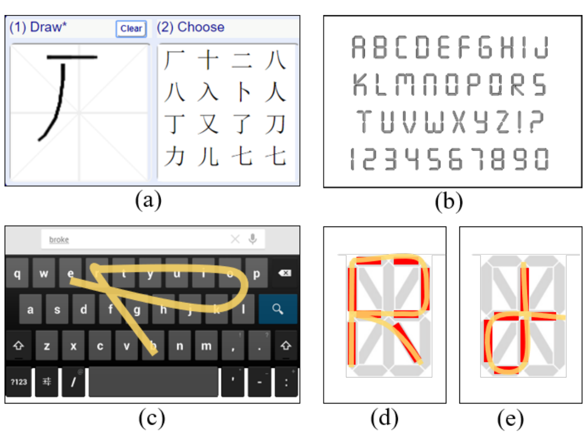

Figure 1: Potential applications for sketch-based typing.

Previous research shows a great promise in enabling users to use gaze as an alternative form of computer input. But the potential of improving these systems by leveraging the idea of tracing given shapes has not yet been explored using gaze-based interaction.

3 SYSTEM DESIGN

We developed a gaze-based interaction interface using Tobii EyeX 4C (90 Hz) and an MSI - GL727QF laptop with 17.3" FHD (1920x1080) display. We captured run-time gaze-point data using *gazePoint-DataStream*. To draw a continuous shape avoiding fluctuation in gaze points, each gaze point was continuously moved towards the subsequent gaze point. In particular, let x and y be the current and next gaze positions, respectively. We considered a 5×5 neighborhood matrix around x and extended the drawn shape from x to a pixel within this neighborhood following direction $\vec{xy}$ (Figure 2).

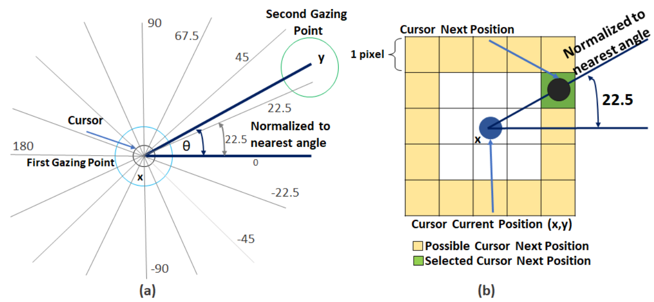

Figure 2: (a) Computation of the direction $\vec{xy}$. (b) Extension of the current drawing.

We tested the interface in a pilot study with two male participants, where each participant was asked to trace various shapes (lines and curves) using the interface. Based on the observation and participants' feedback, we added an option to redo a task after discarding a previous attempt. We also added a custom calibration panel into the system to reduce the gaze point offset error and to recalibrate quickly whenever needed. After clicking the start button for calibration, participants were asked to look directly at a given red point inside the sketch area on the screen, where the real-time gaze point appeared as a black circle. Participants were requested to adjust the horizontal and vertical offset values such that the gaze point (black circle) matches with the red point. To draw two or more shapes, one could stop the drawing of a continuous shape by blocking the line of sight between the eyes and the eye-tracker.

4 USER STUDY

We were interested in examining the following two questions.

Q1: Is it feasible to use gaze input to trace complex shapes?
Q2: How does accuracy vary with shape complexity?

4.1 Condition 1 (Gaze input)

Our first condition consisted of 7 participants (age 25-34, 6 males). Each participant was requested to finish six tasks. Before completing these tasks, participants undertook a practice session, where they traced two lines and two curves, i.e., the 4 shapes of Figure 3(a).

Each of the 6 shapes in Figure 3(b) was used as a task for the main experiment. Participants were asked to calibrate the application interface, if needed, before starting any task or after moving their head. The participant started a task by pressing the start button, which displayed the shape to trace and started the timer. Participants were asked to trace the shapes using gaze input as accurately as possible, and press done when they felt the shape was traced satisfactorily. They were told that they could redo the tasks as many times as they wanted (if they are not satisfied with their drawing). Participants were allowed to take as much time as needed, and to rest if their eyes became tired. Participation lasted 30 min (including instruction, practice trials, test trials, and breaks).

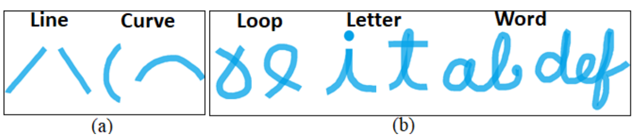

Figure 3: Shapes for (a) training and (b) main experiment.

4.1.1 *Similarity Metric.*

We used the Jaccard similarity coefficient (JSC) to compute how accurately a given shape has been traced. JSC is a commonly used metric to calculate object overlaps on images [6]. If B is a set of pixels of the drawn shapes using gaze input and G is the set of pixels of given shapes, then the Jaccard similarity coefficient, JSC = $\frac{|B \cap G|}{|B \cup G|}$.

In Figure 4, the dark-green region corresponds to pixels that are common to both given and traced drawings. The red region represents pixels that belong to the given shape, but not to the traced drawing. Finally, the gray region corresponds to pixels that do not belong to the given shape, but to the traced drawing. JSC values range from 0 to 1, where 0 denotes that the traced drawing is disjoint from the given shape and 1 denotes a perfect match.

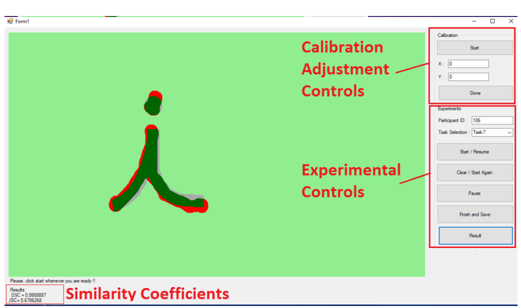

Figure 4: Computing the JSC from a traced shape.

4.2 Condition 2 (Touch input)

Our second condition asked participants to draw shapes on the touch screen using the tip of their finger. The study consisted of 6 participants (one in the 19-24 age group and five in 25-34; 5 males).

Each task consisted of tracing a given shape, and then drawing the shape three times without any guide ('free-form'). While tracing a shape, the participants were told to trace the shape as accurately as possible. The drawing phase was *free-form*, i.e., while drawing, no shape was shown on the screen, and the participants were asked to draw a shape three times making them as similar as possible. Once the participant drew a shape, we cleared the screen and the participant tried to replicate the previously drawn shape from memory. The tasks were based on the shapes used in Study 1.

The reason for this condition was to obtain reference points, against which the accuracy of gaze tracing could be compared. Since people have better control while tracing shapes with their hands, we expected touch-based tracing to provide an upper bound for the tracing accuracy. On the other hand, since producing consistent drawings is difficult without any guiding shape, we expected the free-form drawing to give a lower bound on the tracing accuracy.

4.3 Results

To investigate the feasibility of shape tracing using gaze input (Question Q1), we examined the mean JSC scores for the tasks. The mean of JSC scores was 0.49 for gaze-based tracing, which indicates around 50% similarity between the given and traced drawings. The mean JSC score for touch-based tracing was much higher 0.69, whereas the free-form drawing had a lower mean JSC score[1] (0.21). The unpaired t-test showed significant difference between means when comparing gaze-tracing with touch-based tracing ($t(78) = 10.7, p < 0.001$) and free-form drawing ($t(78) = 15.5, p < 0.001$). We also see a similar relative ordering of the mean JSC scores for each shape complexity category (Figure 5).

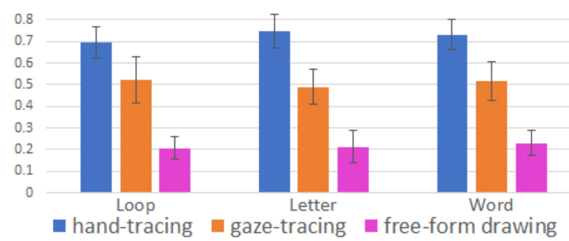

Figure 5: JSC scores for various shapes and input methods.

We categorized the tasks into three categories: `loop`, `letter`, `word` (Figure 3(b)). While examining how the shape tracing accuracy varies across various shape complexities for gaze-based tracing (Question Q2), we did not observe any major difference in the mean JSC scores: `loop` (0.52), `letter` (0.48), `word` (0.51). This indicates a consistent effort from the participants to accurately trace the given shapes. However, the mean number of attempts for drawing different shape categories was quite high for the `loop` and `word` categories (2.4 and 2.5) compared to the `letter` category (1.3)

[1] To compute JSC score for a free-form drawing task, we took the mean of the pairwise JSC scores of the three drawings.

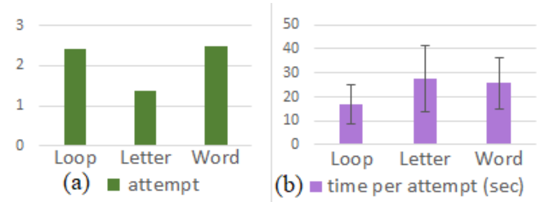

Figure 6: Mean (a) number of attempts and (b) time per attempt for each shape category for gaze-based tracing.

(Figure 6(a)). One possible explanation is that both `loop` and `word` categories include shapes consisting of at least one complete circle, which is considerably difficult to trace compared to the shapes in the `letter` category.

We also examined the mean time taken per attempt for gaze-based tracing (Figure 6(b)). The `loop` category had a much lower mean (16.9) than the `word` category (25.7). Note that both of these categories had a similar mean for the number of attempts, which suggests that the shapes in the `word` category were harder to trace than the shapes in the `loop` category. Although the `letter` category had a low mean number of attempts compared to the `word` category, the mean time per attempt was as high as the `word` category. A possible reason is that each shape in the `letter` category consisted of two separate curves: ('ℓ' and a dot), or ('ℓ' and a horizontal line). Hence drawing a letter shape includes an additional timespan between drawing two curves (i.e., to stop drawing the first curve and then initiate the next curve). Therefore, the time required per attempt increased while compared to the `loop` or `word` category.

5 LIMITATIONS AND FUTURE WORK

In this paper we have examined how well people can trace a given shape using gaze-based input. We observed that people can trace a shape fairly accurately, even when the shape is complex (involving three letters). This suggests that a sketch-based approach to typing could be an interesting avenue to explore. This also opens the opportunity of designing custom user controls and authentication approaches using gaze input.

An eye-tracker with high sampling rate could improve the users' performance in our study. A larger number of participants, as well as participants with motor impairments, could give us more insights into the gaze-based tracing approach. A longitudinal study would be interesting to determine how long people need to use such a gaze-based tracing system to become an expert user. Although our approach seems promising, gaze-based tracing seems to be tiring for eyes. Hence future work may attempt to integrate shape prediction while tracing, in the same spirit of suggesting words while typing.

We plan to expand our work using a more detailed and structured categorization of shapes. Another direction for future investigation will be to compare the sketch-based typing approach against the existing gaze-typing methods using an on-screen keyboard.

ACKNOWLEDGMENTS

This work is supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) and by a Canada First Research Excellence Fund (CFREF) grant.

REFERENCES

[1] Yang Cao, Changhu Wang, Liqing Zhang, and Lei Zhang. 2011. Edgel index for large-scale sketch-based image search. In *CVPR 2011*. IEEE, 761–768.
[2] Fulvio Corno, Laura Farinetti, and Isabella Signorile. 2002. A cost-effective solution for eye-gaze assistive technology. In *Proceedings. IEEE International Conference on Multimedia and Expo*, Vol. 2. IEEE, 433–436.
[3] Alexander De Luca, Roman Weiss, and Heiko Drewes. 2007. Evaluation of eye-gaze interaction methods for security enhanced PIN-entry. In *Proceedings of the 19th australasian conference on computer-human interaction: Entertaining user interfaces*. ACM, 199–202.
[4] Theresa H Escobedo and Ambika Bhargaua. 1991. A study of children's computer-generated graphics. *Journal of Computing in Childhood Education* 2, 4 (1991), 3–25.
[5] James Gips and Peter Olivieri. 1996. EagleEyes: An eye control system for persons with disabilities. In *The Eleventh International Conference on Technology and Persons with Disabilities*. 1–15.
[6] Jeffrey W Godden, Ling Xue, and Jürgen Bajorath. 2000. Combinatorial preferences affect molecular similarity/diversity calculations using binary fingerprints and Tanimoto coefficients. *Journal of Chemical Information and Computer Sciences* 40, 1 (2000), 163–166.
[7] Anthony J Hornof and Anna Cavender. 2005. EyeDraw: enabling children with severe motor impairments to draw with their eyes. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 161–170.
[8] Kyung-Nam Kim and RS Ramakrishna. 1999. Vision-based eye-gaze tracking for human computer interface. In *IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 99CH37028)*, Vol. 2. IEEE, 324–329.
[9] Tomasz Kocejko, Adam Bujnowski, and Jerzy Wtorek. 2009. Eye-mouse for disabled. In *Human-computer systems interaction*. Springer, 109–122.
[10] Per Ola Kristensson and Keith Vertanen. 2012. The potential of dwell-free eye-typing for fast assistive gaze communication. In *Proceedings of the symposium on eye tracking research and applications*. 241–244.
[11] Li Liu, Fumin Shen, Yuming Shen, Xianglong Liu, and Ling Shao. 2017. Deep sketch hashing: Fast free-hand sketch-based image retrieval. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2862–2871.
[12] I Scott MacKenzie and Xuang Zhang. 2008. Eye typing using word and letter prediction and a fixation algorithm. In *Proceedings of the 2008 symposium on Eye tracking research & applications*. 55–58.
[13] Päivi Majaranta, Ulla-Kaija Ahola, and Oleg Špakov. 2009. Fast gaze typing with an adjustable dwell time. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 357–360.
[14] Päivi Majaranta, I Scott MacKenzie, Anne Aula, and Kari-Jouko Räihä. 2003. Auditory and visual feedback during eye typing. In *CHI'03 Extended Abstracts on Human Factors in Computing Systems*. 766–767.
[15] Päivi Majaranta, I Scott MacKenzie, Anne Aula, and Kari-Jouko Räihä. 2006. Effects of feedback and dwell time on eye typing speed and accuracy. *Universal Access in the Information Society* 5, 2 (2006), 199–208.
[16] Päivi Majaranta and Kari-Jouko Räihä. 2002. Twenty years of eye typing: systems and design issues. In *Proceedings of the 2002 symposium on Eye tracking research & applications*. ACM, 15–22.
[17] Kenrick Mock, Bogdan Hoanca, Justin Weaver, and Mikal Milton. 2012. Real-time continuous iris recognition for authentication using an eye tracker. In *Proceedings of the 2012 ACM conference on Computer and communications security*. 1007–1009.
[18] Martez E Mott, Shane Williams, Jacob O Wobbrock, and Meredith Ringel Morris. 2017. Improving dwell-based gaze typing with dynamic, cascading dwell times. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 2558–2570.
[19] Zhor Ramdane-Cherif and Amine NaÏt-AliNait-Ali. 2008. An adaptive algorithm for eye-gaze-tracking-device calibration. *IEEE Transactions on Instrumentation and Measurement* 57, 4 (2008), 716–723.
[20] J Tchalenko. 2001. Free-eye drawing. *Point: Art and Design Research Journal* 11 (2001), 36–41.
[21] YellowBridge. Handwriting Recognizer. https://www.yellowbridge.com/chinese/character-stroke-order-rules.php, Accessed: March 12, 2020.
[22] Nur Haryani Zakaria, David Griffiths, Sacha Brostoff, and Jeff Yan. 2011. Shoulder surfing defence for recall-based graphical passwords. In *Proceedings of the Seventh Symposium on Usable Privacy and Security*. 1–12.
[23] Xianjun Sam Zheng, Stuart Goose, Joeri Kiekebosch, and James Jeng-Weei Lin. 2011. AVIN (Assisted visual interactive notepad): a novel interface design to expedite the eye writing experience. In *International Conference on Universal Access in Human-Computer Interaction*. Springer, 635–644.