Interaction methods for eye tracking in games for cognitive disabled people

Paritosh Sharma
Université Grenoble Alpes
Grenoble, France
paritosh.sharma@etu.univ-grenoble-alpes.fr

Abstract

This work aims to evaluate multiple techniques of interaction for eye tracking in games for cognitively disabled people. The traditional kernel of **GazePlay** utilizes a technique know as **dwell-time** based activation. Although dwell-time solves many aspects of the **Midas touch problem**, the problem is still observed in various games existing in GazePlay. Dwell-time based interaction is also slower for various applications. We address these issues by adding a new interaction method known as **crossing interaction**. Crossing interaction triggers an activation when the target is crossed using gaze. For first time users where precision of selection is more important than speed, we notice a considerably low error rate compared to the dwell-time approach. For experienced users, where higher speed is required, crossing interaction rendered faster results than dwell-time for most of our selected games.

Keywords: eye tracking, gaze tracking, dwell time, crossing interaction.

Links:

1 Introduction

Gaze based interaction provides a simple yet practical method for input in a variety of situations, including games, mixed reality applications and even non-verbal communication. However, programming a gaze based interaction is often complex compared to other simple means like a keyboard or a controller. The activation for the gaze input is often based on a trigger. The most popular method for such an activation is the dwell-time technique where the user looks at the target for a particular time period in order to select it. The dwell-time technique is however inefficient since it doesn't completely solve the Midas touch problem and requires the user to wait while doing consecutive interactions.

In this work, we seek to understand the shortcomings of the dwell-time based approach for a small subset of our games. We also consider adding another interaction technique known as crossing interaction for these games. Since the two interaction techniques are very different, not all games can be ported to use crossing interaction. Thus, we choose a small subset of games in GazePlay. We observed no Midas touch problem with these games while using crossing interaction.

The Midas touch problem is a major problem in development of gaze based interfaces. The situation occurs when the gaze activation happens without user's intention since the system treats every fixation as an interaction request. This problem is named after King Midas, a popular figure in Greek mythology who had the power to turn everything he touched into gold.

Although the problem can be generalized to almost any kind of audience. We focus only on people with disabilities. This, leads to several abnormalities in design since the interaction has to consider people from the ones who have never done any computer interaction, to the ones who have but cannot now.

2 Related Work

Gaze based interaction systems are conceptually complex to design than more physically consistent models which utilize other means of input. It is a popular means of interaction which is used in video games, medical science, mixed reality and many other fields. The model of gaze interaction used in the application highly depends on its use and target audience. As a result, we have a wide variety of interaction techniques developed over the years. Some platforms incorporate the use of gaze tracking along with other modes of input like voice, controllers, etcetra. This has been popularly observed in the recent development of virtual reality headsets. However, few platforms like GazePlay utilize gaze as the only source of input.

There are some very popular state of the art techniques used currently, one of them is dwell-time. Since the gaze is never fixed, the dwell-time technique simply enforces a timer for activation. The rapid movement of eye is known as a saccade. There have been some popular anti-saccade based gaze interaction techniques like the one shown by [Huckauf and Urbina 2011] and [Kurauchi 2018]. The other popular techniques utilized are the use of blinking or winking. However, these could be exhausting for long play sessions especially for disabled people. They also may result in false activations due to involuntary eye movements [Majaranta and Räihä 2002].

A variety of techniques have been implemented to enable quick typing using gaze. Previous works [Bee and André 2008; Kristensson and Vertanen 2012; Isokoski 2000] have shown that the dwell-free based approaches have resulted in faster activation speeds simply because of the fact that there was no dwell time associated before every activation.

Crossing based implementations such as Eye swipe [Kurauchi et al. 2016] provide predictive text input for gaze based interaction while some work have shown a generalized comparison between point based and crossing based interactions [Wobbrock and Gajos 2007]. PEye [Huckauf and Urbina 2007] is another example of such work which gives deeper insights into gaze typing.

Lastly, consistent efforts have been put into gaze interaction as a means of communication between individuals with disabilities. [Townend et al. 2015] shows some work specific to people with Rett syndrome. [Maurer et al. 2018] have taken this one step further by presenting a simple design which could utilize gaze interaction in online multiplayer gaming.

This work focuses on making gaze interactions simpler for our target audience i.e. children with multiple disabilities. For this, we use a limited set of games.

3 Background and Notation

This section focuses on deriving notations for performance benefits when using dwell-time or crossing interaction.

For dwell-time interactions, the moving time stays variable based on the object position and timer wait.

$$MT = A + \Delta$$

Where MT is the moving time for the saccade, A is distance moved in the saccade and Δ is the time limit for the dwell.

For crossing based interaction, [FITTS 1954] showed a formal relation to model speed and accuracy in aimed movements. Later, [MacKenzie 1992] reformulated the moving time required to point to a target at some distance in a rapid movement(a saccade in this case). Thus, we get the following expression.

$$MT = a + b \log\left(\frac{A}{W} + c\right)$$

Where MT is the moving time for the saccade, W is the width of the target object in the game while A is distance moved in the saccade. a and b are empirically determined regression coefficients while c can be 0, 0.5 or 1. The logarithmic expression determines the Index of Difficulty(ID) while constant b is inversely proportional to the performance.

Thus, with careful game designing, we focus on minimization of the ratio A/W.

4 Method

This section reviews the methods utilized for various interactions being used in GazePlay.

4.1 Games

A total of eight types of games were considered. These include the where-is-it games, memory games, magic cards, math games, cups and balls, odd one out, fun with flags and identify the letters. These set of games provide different layout and arrangements which help us in evaluating the performance.

4.2 Design and Implementation

For implementing a game into crossing interaction, we have to mainly take care of the layout and the activation trigger.

For the layout, there's a certain loss of screen real estate when using designing a game for crossing interaction. This is mainly because we need some area for the pointer to retract. Consider the following simple expression for the number of objects in a layout based on dwell-time interaction.

$$n = \frac{W_{screen} \times H_{screen}}{W_{object} \times H_{object}}$$

Where n is the number of objects while W and H represent widths and heights of the screen and object. In most cases, a factor is used to create a m x n layout such as follows.

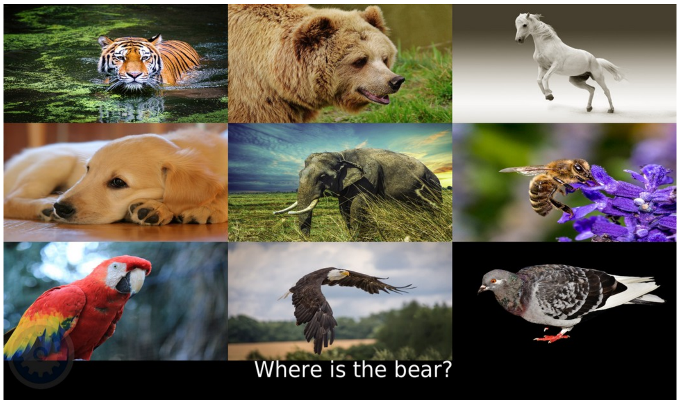

Figure 1: Where-is-it layout - dwell-time

Here, we show an example of where-is-it games with n = 3 x 3.

For crossing interaction however, we areange a boundary of tiles to leave some space for the crossing initialization and post activation pointer reset. Thus, we derive the following expression for the layout.

$$n=\frac{W_{screen}\times H_{screen}}{((W_{object}-W_{margin})\times(H_{object}-H_{margin})-area_{center})}$$

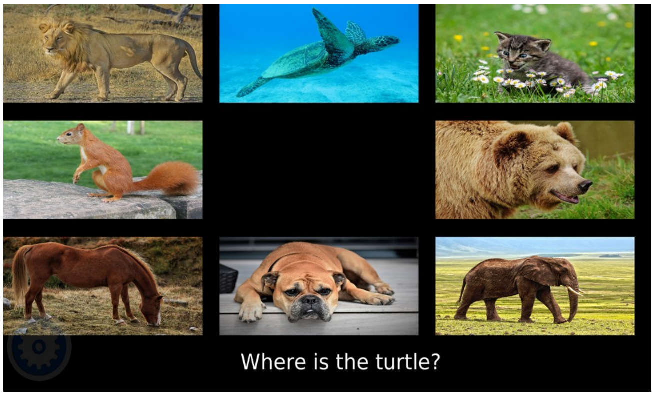

Figure 2: Where-is-it layout - crossing

Now for implementing the activation triggers, we simply use mouse over just like dwell-time approach except we check for the crossing trigger by measuring the distance from center of our pointer. As soon as it crosses an object, we send an activation trigger and reset the pointer's position to the center of the screen. For fairness towards each object, the speed of crossing could be variable based on the distance of the object. This can be simply be achieved by multiplying the saccade update with a factor.

4.3 Hardware

Due to certain constraints, we had to limit ourselves to use a traditional mouse instead of an eye tracker throughout development. The work however could be extended to be used with other eye trackers(like Tobii, EyeTribe) as well. The deployment was done on a test bench with a 15 inch 1920 x 1080 display, i5 8300H processor, 8 Gigabytes of RAM and a GTX 1050Ti GPU.

4.4 Procedure

All participants for testing played each of these games once using dwell-time interaction and then using crossing interaction. The maximum and minimum number of interactions per round for each of these games can be summarized as follows.

Game Type	minimum	maximum
Memory	n	∞
Magic cards	1	n
Math	1	3
Cups and Balls	1	n
Odd One Out	1	n
Fun with Flags	1	4
Identify the Letters	Random(≤n)	∞
Where is It	1	n

Table 1: Number of interactions for each game per round

Here n represents the maximum number of objects on the screen.

5 Results

For testing, we asked 7 people to try each of the selected GazePlay games. Since we are not testing with people with disabilities for now, we focus mainly on the time required for each interaction using either of the schemes. This will help us get a general idea when conducting tests on our actual target audience, children with multiple disabilities.

We get the following results for each of our selected games. The results do not account use of an eye tracker but provide some approximation. When using dwell-time, a constant Δ of 1 second was necessary to make the games playable. Also, we used similar layouts for comparisons.

Game Type	Average movement time - dwell-time(seconds)	Average movement time - crossing(seconds)
Memory	4.19	4.81
Magic cards	1.33(per selection)	0.43(per selection)
Math	1.1	0.65
Cups and Balls	1.07	0.46
Odd One Out	0.96	0.39
Fun with Flags	0.79	0.33
Identify the Letters	Not applicable	Not applicable
Where is It	1.27	0.44

Table 2: Average movement time for round completion

We observe that for single selections, crossing is considerably faster than dwell-time. Some randomized algorithm games like Identify the Letters are not applicable for speed tests due to randomized algorithm. However, we observed that the Midas touch problem was a big issue for this game.

The Memory based games render slower results when used with crossing interaction. This is simply because of the fact that dwell-time enables consecutive selection while crossing undergoes a pointer reset after every selection.

The following table summarizes the Midas touch problem for these games when using these two interaction techniques.

Game Type	Midas Touch Problem - dwell-time	Midas Touch Problem - crossing
Memory	no	no
Magic cards	low	no
Math	low	no
Cups and Balls	low	no
Odd One Out	high	no
Fun with Flags	high	no
Identify the Letters	high	no
Where is It	high	no

Table 3: Midas Touch Problem for each game

Note that not all games can be ported to crossing interaction and many of them simply won't fetch better results than dwell-time. When we have to work with successive activations if the next object is closer to the previous one, dwell-time will always be faster for the long run.

A major part of this project focuses on minimizing the error rate which is substantially high for people with multiple disabilities. No tests were conducted to observe that. [Wobbrock and Gajos 2007] has shown this to some extent using multiple physical input devices like trackball and mouse.

6 Limitations and Future Work

The area of gaze based interaction is farely unexplored and only a small number of succesful state of the art interaction schemes have been developed in the past few years.

Recent modifications in dwell-time such as variable dwell-time [Chen and Shi 2018] have shown promising solutions to the Midas Touch Problem. Modifications in similar direction for crossing interaction could be an instrumental upgrade over the existing model. The visual feedback of the current crossing interaction implementation in GazePlay could be improved a lot. One of the major problems in virtual reality right now is gaze based typing. Projects like PEye [Huckauf and Urbina 2007] seem to solve that problem but more robust game design principles can be applied to make it more user friendly.

Lastly, there have been major developments in gaze based communication [Maurer et al. 2017]. These could be instrumental in creating a state of the art for poly-disabled people to enable communication using gaze. This can even be further extended to gaze based multiplayer games which is a fairly unexplored area.

7 Conclusions

In this work, we shed light on different eye tracking interaction schemes for GazePlay. We presented some possible performance improvements which could be obtained by using crossing interaction in GazePlay. We removed the possibility of Midas touch problem in some of our games by porting them to support crossing interaction. Lastly, we conducted a bunch of tests to compare the two means of interaction and shed light on some possible improvements. Even though we focus on a specific audience, developments in eye tracking could be instrumental for developing a state of the art to improve existing cumbersome development pipelines in the fields of simulation and mixed reality.

Acknowledgments

I would like to express my sincere gratitude to my supervisors Didier Schwab and Sébastien Riou for many insightful discussions and assistance during the internship as well as letting me participate in GazePlay's development. I am thankful to François Bérard, Gaëlle Calvary and Mina Alipour for giving us a good insight about Human Computer Interaction this semester. I am also thankful to the MoSIG supervisors Renaud Lachaize and Denis Trystram for their important provisions during the semester. I express heartfelt gratitude to my colleagues and my family who provided me with support and heartening.

References

BEE, N. AND ANDRÉ, E. 2008. Writing with Your Eye: A Dwell Time Free Writing System Adapted to the Nature of Human Eye Gaze. *Proceedings of the 4th IEEE Tutorial and Research Workshop on Perception and Interactive Technologies for Speech-Based Systems: Perception in Multimodal Dialogue Systems*, Springer-Verlag, 111–122.

CHEN, Z. AND SHI, B. 2018. Using Variable Dwell Time to Accelerate Gaze-Based Web Browsing with Two-Step Selection. *International Journal of Human-Computer Interaction*, 1–16.

FITTS, P. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology 47*, 6, 381—391.

HUCKAUF, A. AND URBINA, M. 2007. Gazing with PEYE: New Concepts in Eye Typing. *Proceedings of the 4th Symposium on Applied Perception in Graphics and Visualization*, Association for Computing Machinery, 141.

HUCKAUF, A. AND URBINA, M.H. 2011. Object Selection in Gaze Controlled Systems: What You Don't Look at is What You Get. *ACM Trans. Appl. Percept. 8*, 2.

ISOKOSKI, P. 2000. Text Input Methods for Eye Trackers Using Off-Screen Targets. *Proceedings of the 2000*

Symposium on Eye Tracking Research & Applications, Association for Computing Machinery, 15–21.

KRISTENSSON, P.O. AND VERTANEN, K. 2012. The Potential of Dwell-Free Eye-Typing for Fast Assistive Gaze Communication. *Proceedings of the Symposium on Eye Tracking Research and Applications*, Association for Computing Machinery, 241–244.

KURAUCHI, A., FENG, W., JOSHI, A., MORIMOTO, C., AND BETKE, M. 2016. EyeSwipe: Dwell-Free Text Entry Using Gaze Paths. *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, Association for Computing Machinery, 1952–1956.

KURAUCHI, A.T.N. 2018. EyeSwipe: text entry using gaze paths. .

MACKENZIE, I.S. 1992. Fitts' Law as a Research and Design Tool in Human-Computer Interaction. *Hum.-Comput. Interact. 7*, 1, 91–139.

MAJARANTA, P. AND RÄIHÄ, K.-J. 2002. Twenty Years of Eye Typing: Systems and Design Issues. *Proceedings of the 2002 Symposium on Eye Tracking Research & Applications*, Association for Computing Machinery, 15–22.

MAURER, B., KRISCHKOWSKY, A., AND TSCHELIGI, M. 2017. Exploring Gaze and Hand Gestures for Non-Verbal In-Game Communication. *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play*, Association for Computing Machinery, 315–322.

MAURER, B., LANKES, M., AND TSCHELIGI, M. 2018. Where the eyes meet: Lessons learned from shared gaze-based interactions in cooperative and competitive online games. *Entertainment Computing 27*, 47–59.

TOWNEND, G., MARSCHIK, P., SMEETS, E., BERG, R., BERG, M., AND CURFS, L. 2015. Eye Gaze Technology as a Form of Augmentative and Alternative Communication for Individuals with Rett Syndrome: Experiences of Families in The Netherlands. *Journal of Developmental and Physical Disabilities 28*.

WOBBROCK, J.O. AND GAJOS, K.Z. 2007. A Comparison of Area Pointing and Goal Crossing for People with and without Motor Impairments. *Proceedings of the 9th International ACM SIGACCESS Conference on Computers and Accessibility*, Association for Computing Machinery, 3–10.