EyeTAP: A Novel Technique using Voice Inputs to Address the Midas Touch Problem for Gaze-based Interactions

Mohsen Parisay
Concordia University
Montreal, Quebec, Canada
m_parisa@encs.concordia.ca

Charalambos Poullis
Concordia University
Montreal, Quebec, Canada
charalambos@poullis.org

Marta Kersten
Concordia University
Montreal, Quebec, Canada
marta.kersten@concordia.ca

ABSTRACT

One of the main challenges of gaze-based interactions is the ability to distinguish normal eye function from a deliberate interaction with the computer system, commonly referred to as 'Midas touch'. In this paper we propose, EyeTAP (Eye tracking point-and-select by Targeted Acoustic Pulse) a hands-free interaction method for point-and-select tasks. We evaluated the prototype in two separate user studies, each containing two experiments with 33 participants and found that EyeTAP is robust even in presence of ambient noise in the audio input signal with tolerance of up to 70 dB, results in a faster movement time, and faster task completion time, and has a lower cognitive workload than voice recognition. In addition, EyeTAP has a lower error rate than the dwell-time method in a ribbon-shaped experiment. These characteristics make it applicable for users for whom physical movements are restricted or not possible due to a disability. Furthermore, EyeTAP has no specific requirements in terms of user interface design and therefore it can be easily integrated into existing systems with minimal modifications. EyeTAP can be regarded as an acceptable alternative to address the Midas touch.

CCS CONCEPTS

• **Human-centered computing** → **Pointing devices**;

KEYWORDS

Human-computer interaction, eye tracking, Midas touch, voice recognition, dwell-time, hands-free interaction, touchless interaction

1 INTRODUCTION

Modern eye tracking sensors offer a suitable alternative to conventional input devices (i.e. keyboard and mouse) for users for whom manual interaction might be difficult or impossible. However, gaze-based interaction has well-known challenges the most important of which are (1) *Midas touch* where a system cannot distinguish the basic function of the eye (i.e. looking and perceiving) from deliberate interaction with the system, and (2) *eye jitter* which is caused by small physiological eye movements occurring during a fixation to perceive a scene visually [19]. In this paper, we propose EyeTAP (Eye tracking point-and-select by Targeted Acoustic Pulse), an effective multimodal solution to the Midas touch problem. Specifically, our method integrates the user's gaze to control the mouse with audio input captured using a microphone to trigger button-press events for real-time interaction.

The contributions of this paper are twofold. Firstly, we have designed and developed an effective, multimodal interaction technique EyeTAP. The proposed approach is low-cost and allows for a completely hands-free interaction solution between the user and the computer system using only an eye-tracker and an audio input device. Secondly, we present two independent user studies each with two experiments comparing EyeTAP with all other widely-used interaction techniques. The analysis of the results clearly shows that using EyeTAP has at least comparable performance with the mouse. Furthermore, EyeTAP reaches competitive performance with the remaining eye-based interaction methods in cases where users would have restricted physical movement, or where manual interaction with an input device is not possible, e.g. medical practitioner having both hands busy.

2 RELATED WORK

In eye-based interaction, the Midas touch problem occurs when a user accidentally activates a computer command by looking when the intention was simply to look around and perceive the scene. According to Jacob [15], this problem occurs because eye movements are natural, e.g. the eyes are used to look around an object or to scan a scene, often without any intention to activate a command or function. This phenomenon is one of the major challenges in eye interaction techniques and diverse methods have been proposed to address the Midas touch problem. The solutions can be categorized into four groups according to the interaction technique they employ: (a) dwell-time processing, (b) smooth pursuits, (c) gaze gestures, and (d) multimodal interaction. Below, we describe each of these solutions and provide example use-cases.

2.1 Dwell-time processing

Dwell-time is the amount of time that the eye gaze must remain on a specific target in order to trigger an event. Researchers have tried to detect specific thresholds to handle the Midas touch problem [25, 33]. For example, Pi *et al.* proposed a probabilistic model for text entry using eye gaze [25]. They reduced the Midas touch problem by assigning each letter a probability value based on the previously chosen letter such that a letter with lower probability requires a longer activation time to be activated and vice-versa. Velichkovsky *et al.* applied focal fixations to resolve the Midas touch problem by assigning the mean duration time (empirically set to 325 ms) of a visual search task to trigger a function [33].

Dwell time has been shown to be even faster than the mouse in certain tasks, e.g. selecting a letter given an auditory cue [30]. However, with dwell time there is a trade-off between accuracy and speed [17, 35, 37]. The method of applying focal fixations may be very subjective since searching time varies across users [3]. Moreover, increasing the threshold may increase the duration time of the entire interaction. Conversely, reducing the amount of dwell-time may lead to more errors for some users [35].

2.2 Smooth pursuits

Smooth pursuits are a form of eye movement that occurs when a moving stimulus (e.g. an object or animation) is followed with gaze [2]. The method is typically implemented by using two visual points on the interface that appear on top and below each target. Then to activate the target the user must fixate on one of these points. This technique has been used to select targets [36], control home appliances [34], to activate functions such as mouse clicks [29] or to use the music player on a smartwatch [8]. Schenk et al. proposed a framework (GazeEverywhere) which enables users to replace mouse inputs [29]. This solution includes a computer to process gaze interactions (gaze PC), a computer to show the results (unmodified PC) which are connected via a micro-controller to trigger mouse click events, and a glass pane to project gaze targets on a second screen.

Vidal et al. introduced an interaction technique (Pursuits) for large screens using moving objects to be activated by eye gaze [36]. They used a Tobii X300 eye tracker and a public display to select targets on the screen. Velloso et al. presented a framework (AmbiGaze) to control ambient devices such as TVs and stereos (each assigned with an infrared (IR) beacon) with eye gaze using a head-mounted eye tracker [34]. The system employs a server to process gaze inputs and control the devices. Esteves et al. presented a framework for a multi-touch Android smartwatch (Callisto 300) to input commands using a head-mounted eye tracker (Pupil Pro) [8]. They developed three use-cases: a music player, a notifications panel with six colored points on the smartwatch screen representing six applications (e.g. social media apps), and a missed call menu with four commands, call back, reply text, save number and clear the notification.

2.3 Gaze gestures

Gaze gestures are sequences of eye movements that follow a pre-defined pattern in a specific order [7]. Researchers have proposed techniques which can be applied to analyze eye movements to detect unique gestures (e.g. [1, 7, 13, 14]). Drewes et al. assigned up, down, left, right and diagonal directions to different characters on the keyboard thereby allowing a user to select a letter by moving the eye gaze in any direction [7]. In addition, they tried to distinguish between natural and intentional eye movements by using short fixation times during gesture detection and long fixation times to reset the gesture recognition. Istance et al. developed two-legged and three-legged gaze gestures (up, down and diagonal patterns) for command selection to play World of Warcraft for users with motor impairment disabilities [14].

In a similar work, Hyrskykari et al. studied both dwell-time and gaze gesture interactions in the context of video games and found that gaze gestures had better performance for command activation [13]. Moreover, gaze gestures produced fewer errors than the dwell-time and led to less visual distractions. Bâce et al. proposed an AR prototype, containing a head-mounted eye tracker and a smartwatch, to embed virtual messages to real-world objects to be shared with peer users [1]. The authors integrated eye gaze gestures as a pattern to encode and decode messages attached to a specific object previously tagged by another peer user, thus using gaze gestures as an authentication mechanism for secure communication.

2.4 Multimodal Interaction

Multimodal techniques apply extra inputs from another modality (e.g. touch, audio, etc.) as the trigger of a function in addition to eye tracking. They can be divided into the following sub-categories: using mechanical switches, touch interaction, or facial gestures.

2.4.1 Applying a specific (mechanical) switch. For some specific domains, such as rehabilitation, and user groups (i.e. users with motor impairments or severe disabilities), researchers have applied specific switches to activate an event or function. For instance, Rajanna et al. proposed a combined framework for users with disabilities which applies a foot pedal device to click on objects and to enter text [26]. Meena et al. applied a soft button on a wheelchair to control the movements of the wheelchair in different directions (horizontal, vertical and diagonal) [18]. Sidorakis et al. applied a switch for a gazed-controlled multimedia framework on virtual reality head-mounted displays (Oculus Rift) to resolve the Midas touch problem [31]. Biswas et al. proposed a joystick to control point-and-select tasks for combat aviation platforms to address the Midas touch problem [5].

2.4.2 Touch interaction. Some researchers have proposed the integration of using touch interaction, for a limited number of functions, to increase the accuracy of target selection. Pfeuffer et al. applied a cursor at the gaze point to be controlled by a finger holding a tablet where a finger tap on the screen leads to a click on the current location of the pointer (CursorShift method) [23]. In a similar study by Pfeuffer et al., the authors investigated the integration of finger touch and pen inputs on a tablet for zooming or annotating tasks on images [24]. Although this technique was not introduced as a solution to the Midas touch problem, it can increase the accuracy of selection which leads to reducing Midas touch.

2.4.3 Facial gestures recognition. In [27], Rozado et al. studied the potential of using live video monitoring to detect facial gestures to enhance eye tracking interaction. In their work (FaceSwitch), they associated facial gestures (opening mouth, raising eyebrows, smiling and twitching the nose up and down) to simulate left and right mouse clicks and customized some keyboard functions such as page down key press.

Using a multimodal solution that combines eye-gaze with acoustic inputs (audio or speech detection) can be regarded as an alternative to the reviewed solutions and has the advantage of not requiring either extra hardware or a specialized user interface design. For this reason, we designed EyeTAP to use audio processing for selection. Our solution: (1) provides a hands-free interaction technique for users with special needs, and (2) addresses the Midas touch problem. Although there has been some work done on audio detection to simulate system events for computer interactions (e.g. [6, 12, 22]) the focus has been on signal processing for complex interactions. Conversely, in our work we applied acoustic inputs only as a way of sending commands.

3 EYETAP PROTOTYPE

A simple mouse interaction consists of moving the pointer to a target (pointing phase), and clicking on it to trigger a function (selection phase). In the EyeTAP prototype the mouse pointer position is captured using the Tobii 4C tracker [1] and selection is done by generating an acoustic pulse by mouth (e.g. a mouth click) which is captured by a headset microphone (Logitech H370). The EyeTAP prototype was developed and the experiments were run on a commodity computer system: 64-bit Windows 10 PC with Intel i7 2.67GHz CPU, 12 GB RAM, 1 TB hard disk and NVIDIA GeForce GTX 770 graphics card. Thus, EyeTAP is a cost-effective system that can be applied at almost any work space. Figure 1 illustrates the EyeTAP system setup.

Figure 1: EyeTAP system: The eye tracker is used to move the pointer from A to B. The user makes an acoustic pulse by mouth and the signal processing module interprets the signal as an input and triggers a click event to select B. The system has an ambient noise tolerance of up to 70 dB.

3.1 Eye Tracking: Pointing Phase

The Tobii SDK (TobiiEyeXSdk−Cpp−1.8.498) supports different events related to eye tracking activities such as providing the location of the current eye gaze, positions of both eyes, fixation points and user presence in front of the eye tracker. We employed the eye gaze library (API) to obtain users' gaze locations. These locations show the current gaze position on the screen as pixels. The SDK supports eye movements in a 3D coordinate system (horizontal, vertical, depth) but we applied a 2D coordinate system (x,y) such that the mouse cursor was synchronized with the gaze positions to control the mouse pointer on the screen. Eye-tracking for the EyeTAP prototype was developed in C++ and integrated as a new plug-in into the Tobii SDK.

---

[1]https://tobiigaming.com/product/tobii-eye-tracker-4c/

3.2 Auditory Processing: Selection Phase

To simulate a click on the item to be selected a headset microphone listens to the user while suppressing the background ambient sounds/noise (conversations in office and equipment sounds) in real-time. The intensity of the mouth noise and distance of microphone is adjusted by the user before the test. A detected pulse in the real-time audio signal (a value larger than a predefined threshold) is regarded as a click. The threshold's value can be adjusted based on the environment to reduce background ambient noise. The EyeTAP prototype has an ambient noise tolerance of up to 70 dB. When a significant increase in the frequency spectrum (greater than the threshold) is detected a mouse click event is triggered. In general, recording is categorized into two phases: audible and silent periods. Any audible period with an intensity greater than the predefined threshold will be detected as an input signal to the system as the binary 1; similarly, values smaller than the threshold value are regarded as binary 0. The intuition behind the auditory processing was inspired from the simplicity of the Morse code [21], which consists of a series of ON/OFF signals triggered by tone or light. Information is interpreted using dots and dashes and therefore can be used to represent transmitted signals through a sequence of True/False variables.

4 EVALUATION

To evaluate the effectiveness of the developed EyeTAP prototype, we ran two independent user studies each with two internal experiments with 33 participants (13 female, from 22 to 35 years old, SD=2.96). All subjects partook in both experiments. Prior to running the experiments, subjects were informed about the purpose of the study, trained on each of the methods to be tested, and participated in a pre-test questionnaire probing them on their background in the fields of eye tracking, voice recognition technologies and their preferred kind of interaction in the case of hands-free alternatives. The Tobii calibration software was used to calibrate the system for each participant before starting the study. At the end of the two experiments subjects filled out a post-test questionnaire, which consisted of the NASA TLX questionnaire [10] followed by specific questions about the subjects' perceptions of the different interaction methods. The order of interaction method was randomly selected for each participant. We played an artificial ambient noise through stereo desktop speakers of 50 dB to simulate a typical work environment since EyeTAP and voice recognition rely on audio inputs.

4.1 User Study 1: Matrix-based Test

In the first experiment, the EyeTAP interaction method was compared with: (a) the mouse, (b) dwell-time, and (c) eye tracking with voice-recognition. In this experiment, a matrix of buttons (targets), were randomly distributed across the screen. The task of the subjects was to point and click on buttons shown on the screen in increasing numerical order for various levels of difficulty from 1 (easy) to 5 (hard), described in detail below. The order of interaction methods seen by each subject was randomly selected for each participant however, the level of difficultly was presented in ascending order.

4.1.1 Stimulus. The stimulus consisted of 77 buttons (11 columns × 7 rows) some labeled with numbers and others not, which covered the entire screen at a resolution of 1920 × 1080 pixels on a Dell P2411Hb monitor. Two marginal columns (far left, far right) and two rows (top, bottom) were removed from the active selection due to the high difficulty to be selected by users during the pilot-test. Buttons that were not labeled are considered as *barriers* or *distractions*. To provide feedback to the subject, labeled buttons change color after the user has successfully pointed and selected on the correct button. Wrongly selected barriers (buttons with no label) are highlighted in red. The level of difficulty of the stimulus was also increased across subject trials. This was done by increasing the number of targets that had to be selected by the subject. Five levels of difficulty were used for each interaction method: level 1 (4 targets), level 2 (6 targets), level 3 (8 targets), level 4 (10 targets) and level 5 (12 targets). Targets were randomly distributed over the entire screen for each level. Figure 2 shows the matrix-based test during difficulty level 5. The cursor that was used was a black circle because it is easier for users to keep it on the target's boundary rather than a pointer.

Figure 2: The matrix-based test for difficulty level 5. Target buttons are distributed randomly across the screen. The red button illustrates an error. The black circle on number 12 shows the current eye gaze location. Labels were enlarged for higher visibility.

4.1.2 Mouse. For the mouse method (our baseline method for comparison), subjects simply used a mouse to move to targets and select them in numerical order.

4.1.3 Dwell-time. For the dwell-time method, where an internal timer is used to determine if a target was selected. The range of dwell-time is in (300-1100) milliseconds for target selection [32]. Then we defined the target activation threshold to 500 milliseconds, since it showed best performance in [16] and participants preferred a dwell-time around 500 ms in a user study [32]. In other words, a target was selected when a subject focused on a target for 0.5 seconds, and if the subject moved their gaze away from the target prior to 0.5 seconds the target selection process would restart.

4.1.4 Eye Tracking with Voice recognition. For voice recognition, eye tracking was used for pointing and voice for selection. The method was developed using the built-in Windows 10 speech recognition capabilities available in the .NET framework. We implemented a C# application to respond to the activation keyword

'select' to trigger a mouse click. The same microphone was used as for the EyeTAP test.

4.1.5 Measures. The following variables were recorded: *completion time*, *path cost of selecting targets*, *error locations*, and *cognitive load* (based on the NASA TLX scores). An internal logging module recorded subjects' actions, selection times, as well as the number of correct and wrong selections. For the path cost measure the shortest path between targets and the produced path by each interaction method was processed. To compare the shapes of the generated paths, we used the dynamic time warping (DTW) algorithm [4, 20, 28]. Since DTW works on a time-value domain the paths produced by the eye tracker were decomposed into their horizontal and vertical values and compared with their associated shortest path models' X and Y values. We applied the built-in *DTW* function in the Python DTW 1.3.3 module [2] to measure the deviations of each path from the shortest path model.

4.2 User Study 1: Dart-based Test

The purpose of this experiment was to measure the accuracy of EyeTAP in comparison to the previously proposed eye-based interaction methods. The task of the subject was to select, as accurately as possible, the bull's-eye of a dart target using each interaction method. In this experiment, the eye tracker was used for the pointing phase for each of the interaction methods, however selection of the target was triggered by different methods, i.e. dwell-time, voice command or EyeTAP acoustic signal. In order to take into consideration the fact that eye tracking has different accuracy in different regions of the monitor, we computed an average value based on five trials for each interaction method where the stimulus was shown at different areas of the screen near the center of the screen randomly. Each new randomly chosen trial began two seconds after selection of the previous target, allowing users time to change their gaze and to focus on the new target. For the dwell-time method, a countdown (5 to 0) representing remaining 100 milliseconds was displayed during the selection phase and users needed to focus on the dart shape before this time was up.

4.2.1 Stimulus. The stimulus for this experiment consisted of a dart-like target with three circles, green (0 to 30 pixels radius), blue (30 to 60 pixels radius) and red (60 to 90 pixels radius) as in Figure 3. Points within the center area i.e. green have the lowest range of distances to the bulls-eye; each other co-centric circle has a larger range of distance values. Any point lying outside the three co-centric circular areas is considered as having a fixed maximum distance of 90 pixels. For this experiment, a cross-hair icon was used.

4.2.2 Measures. The purpose of this test was to measure the selected point's distance on the dart target to the center of the core circle (in green), thus the accuracy is measured in pixels. Since the measured trials are chosen randomly, the average is calculated to compare different methods based on accurate selection.

Figure 3: Dart-based test stimuli: the accuracy is highest in the green area. The cross-hair icon indicates the correct eye gaze location.

4.3 User Study 2: Ribbon-shaped Test

In order to compare our method to other studies, we performed the FittsStudy [38]. This study is used to analyze pointing interaction methods in accordance to well-established academic standards. As part of this study, we measured three metrics to compare the performance of all interaction techniques for point-and-select tasks, (1) *throughput*, (2) *movement time* and (3) *error rates* for ribbon-shaped targets (see figure 4). We applied the FittsStudy application [3] by Wobbrock *et al.* [38]. The test session includes three distances (256, 384, 512) and two widths (96, 128) pixels.

4.4 User Study 2: Circle-shaped Test

This test is similar to the Ribbon-shaped test, however, contains different target shapes. Figure 4 illustrates the screenshots of both test applications. This experiment contains uni-variate endpoint deviation (SD_x) through one axis and bi-variate endpoint deviation ($SD_{x,y}$) through both axes for throughput calculations which results in better Fitts' law model [38].

Figure 4: Screenshots of the 'FittsStudy' application [38]. Top figure illustrates the ribbon-shaped stimuli and the bottom figure shows the circle-shaped stimuli. The highlighted targets are shown in blue to represent the active target to be selected.

5 RESULTS

To determine the effectiveness of the EyeTAP method, we analyzed the results of our experiments using an analysis of variance

[3]http://depts.washington.edu/acelab/proj/fittsstudy/index.html

(ANOVA) followed by Bonferroni posthoc tests with the IBM SPSS software [4].

5.1 User Study 1: Matrix-based User Study

A two-way repeated measures ANOVA (methods × difficulty levels) was performed to examine the effect of interaction type on: (1) *completion time* and (2) *path costs of target selection* for each method and difficulty levels.

5.1.1 Completion time. We found a significant effect of interaction method on completion time (F(12,384)=8.51, $p < .001$). A posthoc Bonferroni comparison test showed a significant difference between mouse ($M = 8017.955$ *ms*, $SE = 645.433$ *ms*) and all other eye tracking methods (see figure 5). In addition, EyeTAP ($M = 19998.812$ *ms*, $SE = 2122.329$ *ms*), dwell-time ($M = 11154.830$ *ms*, $SE = 788.395$ *ms*) and voice recognition ($M = 26904.333$ *ms*, $SE = 2467.576$ *ms*) are significantly different ($p < .05$). Figure 5 illustrates the average completion time per method for 8 targets per level ($\frac{40\ targets}{5\ levels}$).

Figure 5: Average completion time of point-and-select tasks for all participants obtained from the matrix-based user study for 8 targets per level ($\frac{40\ targets}{5\ levels}$). Completion time was significantly different for all techniques ($p < .001$).

5.1.2 Path costs of target selections. To examine the paths produced by selecting targets we compared the original locations of the targets and the shortest path (ideal path model), as described in Section 4. For each method, we had a $\frac{distance}{cost}$ measure to the shortest path. This metric can be regarded as the *footprint* of each interaction technique on the display. A two-way repeated measures ANOVA (methods × difficulty levels) showed that there was a significant effect of interaction type on path cost (F(12,384)=2.57, $p < .05$). A Bonferroni posthoc test showed that dwell-time ($M = 76.73$ *pixels*, $SE = 5.09$ *pixels*) produced the shortest path among all other interaction techniques, even better than the mouse interaction

[4]https://www.ibm.com/analytics/spss-statistics-software

($M = 109.25\ pixels$, $SE = 3.82\ pixels$) with $p < .05$. However, there is no significant difference between dwell-time ($M = 76.73\ pixels$, $SE = 5.09\ pixels$), EyeTAP ($M = 84.80\ pixels$, $SE = 3.59\ pixels$) and voice recognition ($M = 82.03\ pixels$, $SE = 4.41\ pixels$). Figure 6, which shows the path costs for all interaction methods, reveals that eye tracking movements produce significantly lower movements than mouse on a large screen.

Figure 6: Mean path cost comparison calculated using the dynamic time warping (DTW) algorithm. All eye tracking techniques have shorter path lengths than mouse interaction for traversing items on a screen ($p < .05$).

5.1.3 Errors in target selections. To measure the effectiveness of each Midas touch solution we need to consider a penalty for wrongly selected neighboring targets. Those targets are shown in red on the screen (see figure 2). We projected the locations of errors per each interaction method, since difficulty level 5 has the highest number of targets (12 targets) on the screen, we illustrate the locations for this difficulty level in Figure 7. EyeTAP has the highest number of errors, however the figure reveals the potential regions of the screen which are more error prone. As shown in the figure, most errors occurred from the center towards the right side of the screen. In fact, the right side of the screen produces more errors than the left side. Moreover, the lower side produces more errors than the top side. Feit *et al.* showed that the same bottom and right regions of the screen have lower accuracy [9]. We confirm their results and also demonstrate that the same regions are also more error prone.

5.2 User Study 2: Dart-based User Study

We performed a one-way repeated measures ANOVA to compare the effect of the different interaction methods on accuracy. The results of the ANOVA showed all eye tracking methods have statistical difference (F(3,96)=104.92, $p < 0.001$) on selection accuracy. In fact, the mouse interaction has the lowest distance to target (higher accuracy) compared to eye tracking techniques. EyeTAP ($M = 45.11\ pixels$, $SE = 2.28\ pixels$) achieved the highest mean pixel accuracy compared to dwell-time ($M = 35.30\ pixels$, $SE = 2.11\ pixels$)

Figure 7: The locations of errors during the matrix-based user study (figure 2) for difficulty level 5. The right side of the screen as well as bottom side are more error prone than the left and top sides.

and voice recognition ($M = 29.27\ pixels$, $SE = 2.07\ pixels$). Figure 8 depicts the results of the accuracy test.

5.3 User Study 2: Ribbon-shaped Test

A one-way repeated measures ANOVA was performed to examine the effect of interaction type on: (1) *movement time*, (2) *throughput* and (3) *error rates* for each interaction method.

5.3.1 Movement time. We found a significant effect of the interaction method on movement time (F(3,96)=69.42, $p < .001$). A posthoc Bonferroni comparison test showed a significant difference between mouse ($M = 684.15\ ms$, $SE = 16.80\ ms$) and all other eye tracking methods (figure 9). In addition, among all eye tracking methods, dwell-time ($M = 599.39\ ms$, $SE = 18.76\ ms$) achieved significantly lower movement time than EyeTAP ($M = 1794.89\ ms$,

Figure 8: The mean distance to target in pixels for dart-based experiment ($p < .001$).

$SE = 170.90\ ms$) and voice recognition ($M = 2014.20\ ms$, $SE = 89.28\ ms$) techniques. However, there is no statistical significance between EyeTAP and voice recognition. The lower movement time of dwell-time method compared to the mouse interaction is associated with the low activation time (500 ms).

Figure 9: The calculated movement time per method for the ribbon-shaped test ($p < .001$).

5.3.2 Throughput. We found a significant effect of the interaction method on throughput (F(3,96)=75.13, $p < .001$). A posthoc Bonferroni comparison test showed a significant difference between dwell-time ($M = 3.30\ bits/sec$, $SE = 0.36\ bits/sec$) and all eye tracking methods (figure 10). The mouse ($M = 4.81\ bits/sec$, $SE = 0.11\ bits/sec$) achieved higher throughput than the eye tracking methods. However, there is no statistical difference between

voice recognition ($M = 1.15\ bits/sec$, $SE = 0.09\ bits/sec$) and Eye-TAP ($M = 1.34\ bits/sec$, $SE = 0.12\ bits/sec$).

Figure 10: The calculated throughput per method for the ribbon-shaped test ($p < .001$).

5.3.3 Error rates. We found a significant effect of the interaction method on error rates (F(3,96)=27.15, $p < .001$). A posthoc Bonferroni comparison test showed a significant difference between mouse ($M = 0.01\ errors$, $SE = 0.005\ errors$) and all eye tracking interactions (see Figure 11). In addition, dwell-time ($M = 0.28\ errors$, $SE = 0.03\ errors$) reached a higher error rate than EyeTAP ($M = 0.18\ errors$, $SE = 0.02\ errors$) and voice recognition ($M = 0.10\ errors$, $SE = 0.02\ errors$).

Figure 11: The calculated error rates per method for the ribbon-shaped test ($p < .001$).

5.4 User Study 2: Circle-shaped Test

A one-way repeated measures ANOVA was performed to examine the effect of interaction type on: (1) *movement time*, (2) *throughput* and (3) *error rates* for each interaction method. This experiment is similar to ribbon-shaped test but contains an extra metric to measure throughput of each method.

5.4.1 Movement time. We found a significant effect of the interaction method on movement time ($F(3,96)=67.48$, $p < .001$). A posthoc Bonferroni comparison test showed a significant difference between EyeTAP ($M = 1578.95\ ms$, $SE = 95.34\ ms$), dwell-time ($M = 638.80\ ms$, $SE = 24.35\ ms$), voice recognition ($M = 2123.35\ ms$, $SE = 132.42\ ms$) and mouse ($M = 727.91\ ms$, $SE = 46.12\ ms$). However, there is no statistical difference between mouse ($M = 727.91\ ms$, $SE = 46.12\ ms$) and dwell-time ($M = 638.80\ ms$, $SE = 24.35\ ms$). Figure 12 illustrates the mean movement time per method for the circle-shaped test.

Figure 12: The calculated movement time per method for the circle-shaped test ($p < .001$).

5.4.2 Throughput. Since the circle-shaped test contains two variations (uni-variate, bi-variate) to measure throughput [38], we ran a two-way repeated measures ANOVA (throughput × variation) and found a significant effect of the interaction method on throughput ($F(3,96)=19.75$, $p < .001$). A posthoc Bonferroni comparison test showed a significant difference between mouse ($M = 4.16\ bits/sec$, $SE = 0.18\ bits/sec$), dwell-time ($M = 3.20\ bits/sec$, $SE = 0.25\ bits/sec$), voice-recognition ($M = 1.24\ bits/sec$, $SE = 0.07\ bits/sec$) and EyeTAP ($M = 1.04\ bits/sec$, $SE = 0.13\ bits/sec$). However, there is no statistical difference between voice-recognition ($M = 1.24\ bits/sec$, $SE = 0.07\ bits/sec$) and EyeTAP ($M = 1.04\ bits/sec$, $SE = 0.13\ bits/sec$). Figure 13 shows both variations of throughput per interaction method.

5.4.3 Error rates. We found a significant effect of the interaction method on error rates ($F(3,96)=18.25$, $p < .001$). A posthoc Bonferroni comparison test showed a significant difference between

mouse ($M = 0.02\ errors$, $SE = 0.01\ errors$), dwell-time ($M = 0.23\ errors$, $SE = 0.03\ errors$), voice recognition ($M = 0.13\ errors$, $SE = 0.02\ errors$) and EyeTAP ($M = 0.28\ errors$, $SE = 0.02\ errors$). Voice recognition ($M = 0.13\ errors$, $SE = 0.02\ errors$) reached the lowest error rate among eye tracking methods, however, there is no statistical difference between dwell-time ($M = 0.23\ errors$, $SE = 0.03\ errors$) and EyeTAP ($M = 0.28\ errors$, $SE = 0.02\ errors$). Figure 14 illustrates the calculated error rates for the circle-shaped test.

5.5 EyeTAP rating by users

We asked participants to evaluate the overall performance of Eye-TAP in the post-test questionnaire on a scale from 1 (worst) to 5 (best). EyeTAP reached the average rate of 3.64 ($SD = 0.99$) by 33 users. In addition, Users were asked to select multiple interaction techniques. Figure 15 illustrates the popular interaction techniques by users obtained from the post-test questionnaire. EyeTAP reached the second desired eye tracking technique.

5.6 NASA TLX scores

Figure 16 shows the NASA TLX scores for all interaction methods obtained during the user study. The overall workload is the average of scale values since we assume all scales equally important and therefore eliminated the weighting calculations to apply a simplified version [11] of the basic NASA TLX ratings [10]. According to our findings, the dwell-time method has the lowest workload among other eye tracking techniques. However, EyeTAP shows relatively lower workload compared to voice recognition technique.

6 DISCUSSION

Regarding the experiments with the reviewed Midas touch solutions, we found several benefits and disadvantages of each method. We discuss each method individually.

6.1 Voice Recognition

This interaction method showed relatively acceptable results but suffers from some limitations. In general, a voice recognition engine depends on the user's voice, gender, language, and accent. Additionally, it is not applicable to users with speech impediments. Another drawback is the need of prior training samples to detect words correctly. Furthermore, similar words may lead to false recognition as we experienced during our user study. The quality of the microphone and its distance to the user is also another factor to be considered for this kind of interaction. Regarding the accuracy of recognition, the choice of recognition software plays an important role. Finally, speaking out loud may not be suitable in certain working environments.

In general, voice recognition presented some challenges for the users in terms of wrongly recognized words, need for action word repetition, and delay between input and feedback. The subjects' rating of this technique was very low (9.1%) in our user study. Voice recognition showed the highest completion time in the matrix-based test and highest movement time in the circle-shaped test and reached the highest cognitive workload among all interaction techniques. However, voice recognition showed the lowest error rates in both Fitts' study experiments and reached the lowest distance

Figure 13: The calculated throughput for both uni-, and bi-variations per method for the circle-shaped test ($p < .001$).

Figure 14: The calculated error rates per method for the circle-shaped test ($p < .001$).

to target (highest selection accuracy) among other eye tracking techniques.

6.2 Dwell-Time processing

Dwell-time method showed the fastest completion time in the matrix-based test, and fastest movement time and highest throughput in both Fitts' experiments due to the low amount of activation time (500 ms). In addition, it reached the lowest amount of cognitive workload. However, it showed the highest error rates in the ribbon-shaped test and with EyeTAP in the circle-shaped test. Moreover, some users complained about eye fatigue after a while during test sessions.

Figure 15: The recorded users' multiple choice of interaction techniques among 33 participants.

6.3 EyeTAP

We found several benefits of using EyeTAP in comparison to the other interaction techniques. First of all, it has no dependent features, rather it requires only an acoustic pulse (making sound with mouth) near a microphone to send a signal. In fact, the output of EyeTAP in a noisy environment (up to 70dB) can appear deterministic after a number of repetitions. According to the results of our study, it achieved faster completion time in the matrix-based test, and faster movement time in the circle-shaped experiment than voice recognition. In addition, it showed a similar path cost (pointer footprint on display) with the other eye tracking techniques. It also achieved lower cognitive workload in comparison to the voice recognition technique. Furthermore, EyeTAP was the popular choice of interaction (36.4%) compared to voice recognition (9.1%). However, EyeTAP showed relatively lower accuracy and higher error rates than voice recognition, since most users had no

Figure 16: The NASA TLX scores for the interaction methods. (Left) Comparison of each method based on different scales. (Right) The overall mean workload of tested interaction methods.

prior experiences with this kind of interaction. The performance of EyeTAP can be improved with more training.

In general, EyeTAP is simple, integrates well into existing user interfaces, and allows for easy and accurate point-and-select interaction because it separates the actions of *pointing* and *selecting* to two different modalities while relaxing the requirement for accurate voice recognition. The results of our user study demonstrate that EyeTAP is a feasible alternative interaction technique. Moreover, it is a robust and effective solution to the Midas touch problem for eye tracking platforms and can be regarded as an alternative to voice recognition technique.

7 CONCLUSION AND FUTURE WORK

In this paper, we proposed EyeTAP (Eye tracking point-and-select by Targeted Acoustic Pulse), an eye-tracking interface that addresses the Midas touch problem with acoustic input detection capabilities. EyeTAP allows for accurate and effective interaction without the need for extra equipment or user interface design for gaze-based interactions. The performance of the prototype was measured in two independent user studies with 33 participants based on eight criteria: (1) *completion time*, (2) *path cost of target selection*, (3) *error rate*, (4) *error locations on screen*, (5) *accuracy of target selection*, (6) *movement time*, (7) *throughput*, and (8) *cognitive workload.*

The results of our user studies showed that the dwell-time method outperformed other eye tracking techniques, including EyeTAP on most criteria. At the same time we found that EyeTAP, in comparison to to the other tested methods is a competitive and a promising solution and provides a faster task completion time, faster movement time and lower workload than voice recognition. In addition, EyeTAP showed similar performance compared to the dwell-time method and lower error rate in the ribbon-shaped experiment.

Moreover, our study showed that eye tracking has a lower footprint on the screen compared to a mouse pointer in time scale. Additionally, we confirmed that center regions towards the right and bottom side of the screen are more error prone than the left and top sides. Additionally, we developed two user tests that would be effective in studying different target selection for gaze-based interaction techniques.

Although we only developed the left mouse click event, EyeTAP demonstrates a completely hands-free or touchless alternative to mouse interaction for users with disabilities and users who need to avoid physical contact with input devices considering their workplace or situation. Thus, we believe EyeTAP can be regarded as a competitive technique to both dwell-time and voice recognition. In future work, we will apply the EyeTAP technique on AR/VR headsets to measure its usability in different case scenarios.

REFERENCES

[1] Mihai Bâce, Teemu Leppänen, David Gil de Gomez, and Argenis Ramirez Gomez. ubigaze: Ubiquitous augmented reality messaging using gaze gestures. In *SIGGRAPH ASIA 2016 Mobile Graphics and Interactive Applications*, SA '16, pages 11:1–11:5, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4551-4. doi: 10.1145/2999508.2999530. URL http://doi.acm.org/10.1145/2999508.2999530.

[2] Graham R. Barnes. Rapid learning of pursuit target motion trajectories revealed by responses to randomized transient sinusoids. *Journal of Eye Movement Research*, 5(3), 2012. ISSN 1995-8692. URL https://bop.unibe.ch/JEMR/article/view/2337.

[3] Roman Bednarik, Tersia Gowases, and Markku Tukiainen. Gaze interaction enhances problem solving: Effects of dwell-time based, gaze-augmented, and mouse interaction on problem-solving strategies and user experience. *Journal of Eye Movement Research*, 3(1), Aug. 2009. doi: 10.16910/jemr.3.1.3. URL https://bop.unibe.ch/JEMR/article/view/2287.

[4] R. Bellman and R. Kalaba. On adaptive control processes. *IRE Transactions on Automatic Control*, 4(2):1–9, November 1959. ISSN 0096-199X. doi: 10.1109/TAC. 1959.1104847.

[5] Pradipta Biswas and Pat Langdon. Multimodal intelligent eye-gaze tracking system. *International Journal of Human-Computer Interaction*, 31(4):277–294, 2015.

[6] Anind K. Dey, Raffay Hamid, Chris Beckmann, Ian Li, and Daniel Hsu. A cappella: Programming by demonstration of context-aware applications. In *Proceedings of*

the *SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 33–40, New York, NY, USA, 2004. ACM. ISBN 1-58113-702-8. doi: 10.1145/985692.985697. URL http://doi.acm.org/10.1145/985692.985697.

[7] Heiko Drewes and Albrecht Schmidt. Interacting with the computer using gaze gestures. In Cécilia Baranauskas, Philippe Palanque, Julio Abascal, and Simone Diniz Junqueira Barbosa, editors, *Human-Computer Interaction – INTERACT 2007*, pages 475–488, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. ISBN 978-3-540-74800-7.

[8] Augusto Esteves, Eduardo Velloso, Andreas Bulling, and Hans Gellersen. Orbits: Gaze interaction for smart watches using smooth pursuit eye movements. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, pages 457–466, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3779-3. doi: 10.1145/2807442.2807499. URL http://doi.acm.org/10.1145/2807442.2807499.

[9] Anna Maria Feit, Shane Williams, Arturo Toledo, Ann Paradiso, Harish Kulkarni, Shaun Kane, and Meredith Ringel Morris. Toward everyday gaze input: Accuracy and precision of eye tracking and implications for design. In *Proceedings of the 2017 Chi conference on human factors in computing systems*, pages 1118–1130. ACM, 2017.

[10] NASA Human Performance Research Group. Nasa task load index (tlx) paper and pencil package, 1986. URL https://humansystems.arc.nasa.gov/groups/TLX/downloads/TLX.pdf.

[11] Sandra G. Hart. Nasa-task load index (nasa-tlx); 20 years later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 50(9):904–908, 2006. doi: 10.1177/154193120605000909. URL https://doi.org/10.1177/154193120605000909.

[12] Björn Hartmann, Leith Abdulla, Manas Mittal, and Scott R. Klemmer. Authoring sensor-based interactions by demonstration with direct manipulation and pattern recognition. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 145–154, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-593-9. doi: 10.1145/1240624.1240646. URL http://doi.acm.org/10.1145/1240624.1240646.

[13] Aulikki Hyrskykari, Howell Istance, and Stephen Vickers. Gaze gestures or dwell-based interaction? In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, pages 229–232, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1221-9. doi: 10.1145/2168556.2168602. URL http://doi.acm.org/10.1145/2168556.2168602.

[14] Howell Istance, Aulikki Hyrskykari, Lauri Immonen, Santtu Mansikkamaa, and Stephen Vickers. Designing gaze gestures for gaming: An investigation of performance. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, ETRA '10, pages 323–330, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-994-7. doi: 10.1145/1743666.1743740. URL http://doi.acm.org/10.1145/1743666.1743740.

[15] Robert J. K. Jacob. What you look at is what you get: Eye movement-based interaction techniques. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '90, pages 11–18, New York, NY, USA, 1990. ACM. ISBN 0-201-50932-6. doi: 10.1145/97243.97246. URL http://doi.acm.org/10.1145/97243.97246.

[16] I Scott MacKenzie. Evaluating eye tracking systems for computer input. In *Gaze interaction and applications of eye tracking: Advances in assistive technologies*, pages 205–225. IGI Global, 2012.

[17] Päivi Majaranta, I Scott MacKenzie, Anne Aula, and Kari-Jouko Räihä. Effects of feedback and dwell time on eye typing speed and accuracy. *Universal Access in the Information Society*, 5(2):199–208, 2006.

[18] Y. K. Meena, H. Cecotti, K. Wong-Lin, and G. Prasad. A multimodal interface to resolve the midas-touch problem in gaze controlled wheelchair. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 905–908. IEEE, July 2017. doi: 10.1109/EMBC.2017.8036971.

[19] Darius Miniotas, Oleg Špakov, and I. Scott MacKenzie. Eye gaze interaction with expanding targets. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '04, pages 1255–1258, New York, NY, USA, 2004. ACM. ISBN 1-58113-703-6. doi: 10.1145/985921.986037. URL http://doi.acm.org/10.1145/985921.986037.

[20] C. Myers, L. Rabiner, and A. Rosenberg. Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(6):623–635, December 1980. ISSN 0096-3518. doi: 10.1109/TASSP.1980.1163491.

[21] The Editors of Encyclopaedia Britannica. Morse code, March 2018. URL https://www.britannica.com/topic/Morse-Code. [Online; accessed September 14, 2018].

[22] Shwetak N. Patel and Gregory D. Abowd. Blui: Low-cost localized blowable user interfaces. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*, UIST '07, pages 217–220, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-679-0. doi: 10.1145/1294211.1294250. URL http://doi.acm.org/10.1145/1294211.1294250.

[23] Ken Pfeuffer and Hans Gellersen. Gaze and touch interaction on tablets. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, pages 301–311, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4189-9. doi: 10.1145/2984511.2984514. URL http://doi.acm.org/10.1145/2984511.2984514.

[24] Ken Pfeuffer, Jason Alexander, and Hans Gellersen. Partially-indirect bimanual input with gaze, pen, and touch for pan, zoom, and ink interaction. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 2845–2856, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3362-7. doi: 10.1145/2858036.2858201. URL http://doi.acm.org/10.1145/2858036.2858201.

[25] J. Pi and B. E. Shi. Probabilistic adjustment of dwell time for eye typing. In *2017 10th International Conference on Human System Interactions (HSI)*, pages 251–257. IEEE, July 2017. doi: 10.1109/HSI.2017.8005041.

[26] Vijay Rajanna and Tracy Hammond. A gaze-assisted multimodal approach to rich and accessible human-computer interaction. *CoRR*, abs/1803.04713, 2018. URL http://arxiv.org/abs/1803.04713.

[27] David Rozado, Jason Niu, and Martin Lochner. Fast human-computer interaction by combining gaze pointing and face gestures. *ACM Transactions on Accessible Computing (TACCESS)*, 10(3):10, 2017.

[28] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49, February 1978. ISSN 0096-3518. doi: 10.1109/TASSP.1978.1163055.

[29] Simon Schenk, Marc Dreiser, Gerhard Rigoll, and Michael Dorr. Gazeeverywhere: Enabling gaze-only user interaction on an unmodified desktop pc in everyday scenarios. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 3034–3044, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4655-9. doi: 10.1145/3025453.3025455. URL http://doi.acm.org/10.1145/3025453.3025455.

[30] Linda E. Sibert and Robert J. K. Jacob. Evaluation of eye gaze interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI âĂŹ00, page 281âĂŞ288, New York, NY, USA, 2000. Association for Computing Machinery. ISBN 1581132166. doi: 10.1145/332040.332445. URL https://doi.org/10.1145/332040.332445.

[31] N. Sidorakis, G. A. Koulieris, and K. Mania. Binocular eye-tracking for the control of a 3d immersive multimedia user interface. In *2015 IEEE 1st Workshop on Everyday Virtual Reality (WEVR)*, pages 15–18. IEEE, March 2015. doi: 10.1109/WEVR.2015.7151689.

[32] Oleg Špakov and Darius Miniotas. On-line adjustment of dwell time for target selection by gaze. In *Proceedings of the third Nordic conference on Human-computer interaction*, pages 203–206. ACM, 2004.

[33] Boris B Velichkovsky, Mikhail A Rumyantsev, and Mikhail A Morozov. New solution to the midas touch problem: Identification of visual commands via extraction of focal fixations. *Procedia Computer Science*, 39:75–82, 2014.

[34] Eduardo Velloso, Markus Wirth, Christian Weichel, Augusto Esteves, and Hans Gellersen. Ambigaze: Direct control of ambient devices by gaze. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems*, DIS '16, pages 812–817, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4031-1. doi: 10.1145/2901790.2901867. URL http://doi.acm.org/10.1145/2901790.2901867.

[35] Roel Vertegaal. A fitts law comparison of eye tracking and manual input in the selection of visual targets. In *Proceedings of the 10th International Conference on Multimodal Interfaces*, ICMI âĂŹ08, page 241âĂŞ248, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605581989. doi: 10.1145/1452392.1452443. URL https://doi.org/10.1145/1452392.1452443.

[36] Mélodie Vidal, Andreas Bulling, and Hans Gellersen. Pursuits: Spontaneous interaction with displays based on smooth pursuit eye movement and moving targets. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, pages 439–448, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1770-2. doi: 10.1145/2493432.2493477. URL http://doi.acm.org/10.1145/2493432.2493477.

[37] Oleg Špakov and Darius Miniotas. On-line adjustment of dwell time for target selection by gaze. In *Proceedings of the Third Nordic Conference on Human-Computer Interaction*, NordiCHI âĂŹ04, page 203âĂŞ206, New York, NY, USA, 2004. Association for Computing Machinery. ISBN 1581138571. doi: 10.1145/1028014.1028045. URL https://doi.org/10.1145/1028014.1028045.

[38] Jacob O Wobbrock, Kristen Shinohara, and Alex Jansen. The effects of task dimensionality, endpoint deviation, throughput calculation, and experiment design on pointing measures and models. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1639–1648. ACM, 2011.