Journal of Eye Movement Research
8(1):2, 1-11

SMOOVS: Towards calibration-free text entry by gaze using smooth pursuit movements

Otto Hans-Martin Lutz
Chair of Human-Machine Systems
Technische Universität Berlin

Antje Christine Venjakob
Chair of Human-Machine Systems
Technische Universität Berlin

Stefan Ruff
Chair of Human-Machine Systems
Technische Universität Berlin

Gaze-based text spellers have proved useful for people with severe motor diseases, but lack acceptance in general human-computer interaction. In order to use gaze spellers for public displays, they need to be robust and provide an intuitive interaction concept. However, traditional dwell- and blink-based systems need accurate calibration which contradicts fast and intuitive interaction. We developed the first gaze speller explicitly utilizing smooth pursuit eye movements and their particular characteristics. The speller achieves sufficient accuracy with a one-point calibration and does not require extensive training. Its interface consists of character elements which move apart from each other in two stages. As each element has a unique track, gaze following this track can be detected by an algorithm that does not rely on the exact gaze coordinates and compensates latency-based artefacts. In a user study, 24 participants tested four speed-levels of moving elements to determine an optimal interaction speed. At 300 px/s users showed highest overall performance of 3.34 WPM (without training). Subjective ratings support the finding that this pace is superior.

Keywords: gaze interaction, eye movements, text entry, smooth pursuit, calibration

Introduction

Gaze interaction was used initially to provide a modality of communication for physically impaired users (Majaranta & Räihä, 2002). Current research and commercial products move towards applying gaze interaction to every-day human computer interaction (Drewes & Schmidt, 2007). The often cited advantages of gaze interaction are freeing hands for other tasks and increasing hygiene due to contactless interaction. In general, gaze interaction can be based on dwell time (fixations), eye blinks, saccades and smooth pursuit movements (Mollenbach, Hansen, & Lillholm, 2013). However, there are drawbacks associated with gaze interaction, particularly involuntary interaction and the need for individual calibration.

If the same modality is used for perception and interaction, involuntary input activation is likely to happen. In gaze interaction, this is known as *Midas Touch Problem* and refers to the legendary king Midas, who wished that everything he touched would be turned into gold, but then finds himself trapped in his wish (Jacob, 1991). As gaze is used permanently to gather information, interaction via the same modality holds a high risk of inadvertent user input. Therefore, strategies to avoid the Midas Touch Problem need to be considered in interaction design, as well as the implementation of differentiated and easy-to-comprehend feedback (Majaranta, 2011).

Interaction based on gaze positions on the screen demands good accuracy. This is achieved by calibration, i.e. mapping gaze directions to several positions on the screen (Holmqvist et al., 2011). Dwell- and blink-based interaction in particular highly depend on accurate calibration. As every user has unique physiological properties, individual calibration on the user is indispensable. At the same time, user acceptance of calibration procedures is low, particularly in cases where a recalibration is needed (Villanueva, Cabeza, & Porta, 2004; Pfeuffer, Vidal, & Turner, 2013). Furthermore, for usage with public displays, where spontaneous and fast interaction is needed, individual calibration is inefficient due to the temporal demands.

Integration of gaze interaction in public displays holds additional challenges as users perform natural movements and do not necessarily have an understanding of the limitations of the eye-tracking system. Hence, an intuitive user interface, ideally with implicit

DOI 10.16910/jemr.8.1.2
This article is licensed under a
Creative Commons Attribution 4.0 International license.
ISSN 1995-8692

calibration[1] and a robust underlying algorithm that does not require accurate gaze positions, are necessary for user-friendly gaze interaction with public displays. Gaze gestures are a possible solution to this problem. As they do not rely on accurate positions, but are defined by shape, spatial accuracy is less critical. Usage of smooth pursuit movements to stimulate gaze gestures is a promising solution as it was demonstrated in interactive games, basic selection tasks (Vidal, Bulling, & Gellersen, 2013) and PIN-pads (Cymek et al., 2014). It has even been suggested that gesture-based gaze interaction can be realised without calibrating the system to the individual user at all (Drewes & Schmidt, 2007). A gaze speller represents a complex selection task. This has not been realised yet explicitly using the benefits of smooth pursuit movements for intuitive interaction. Aim of this work is the implementation of a gaze speller which does not rely on accurate calibration utilizing smooth pursuit movements.

Current Concepts of Gaze Spellers

Majaranta (2011) defines several categories of gaze spellers, differentiated by their interaction concepts. To compare gaze spellers, a commonly used benchmark is text entry rate in words per minute (WPM), defined as the number of characters[2] per minute, divided by the average word length of five characters (Arif & Stuerzlinger, 2009). *Direct gaze pointing* is the most commonly used method, selecting and confirming a character on an on-screen-keyboard by long fixations. All purely fixation-based methods are relatively slow (5 - 10 WPM). To improve typing speed, *(dynamic) context switching* was introduced additionally (Morimoto & Amir, 2010; Tula, de Campos, & Morimoto, 2012). A second keyboard is displayed at the cost of more screen space needed. Character selection is performed by a fixation, but confirmed by a saccade to the second keyboard.

Entering text by saccades between dynamic display objects is a hybrid form between position- and gesture-based methods. A multi-level selection process is implemented, where a group of characters is selected first, with the selection of the desired character to follow in a second step. Text entry rates of 5 WPM (Bee & André, 2008) and 7.9 WPM (Huckauf & Urbina, 2008) are reported for this method.

In gesture-based interaction, Majaranta (2011) distinguishes between *discrete gaze gestures* and *continuous pointing gestures*. Discrete gaze gestures are gestures in the classical sense, consisting of saccades between several points. These interfaces need little screen space, but the user needs to learn a particular gesture alphabet. In the study of Wobbrock, Rubinstein, Sawyer, and Duchowsky (2008), a text entry rate of 4.9 WPM was achieved with their speller *EyeWrite*. The gaze spellers *Dasher* (Ward, Blackwell, & MacKay, 2000) and *StarGazer* (Hansen, Skovsgaard, Hansen, & Mollenbach, 2008) are examples of continuous pointing gestures, where moving display elements are used to guide attention. As *Dasher* uses a dictionary-based auto completion and participants conducted 10 training units before examination, the text input rate rose to 17.3 WPM. In spellers based on continuous pointing gestures, the user's gaze follows moving display elements, so the eyes perform pursuit movements. However, the described position-based classifications of these spellers do not explicitly utilise the properties and benefits of smooth pursuit movements. Summarised, one property of all gaze spellers presented so far is the need for accurate gaze positions, hence proper calibration and limited head movement of the user are mandatory. Designing a gaze speller based on smooth pursuit movements overcomes the dependence on accurate positions.

Smooth Pursuit Movements in Gaze Interaction

Smooth pursuit eye movements are relatively slow (10 - 30 °/s) and regular ('smooth') movements of the eye that occur when a moving object is followed by gaze (Holmqvist et al., 2011). During the first 100 ms the eye is accelerated towards the anticipated stimulus position. This results in an offset between gaze and stimulus position, the so-called *open-loop-pursuit*. Within less than 300 ms, the pursuit movement converges on the true stimulus motion (Wallace, Stone, Masson, & Julian, 2005; Burke & Barnes, 2006). This so-called *closed-loop-pursuit* is more precise and continous, as long as the stimulus motion is predictable (Bahill & McDonald, 1983). Horizontal smooth pursuit movements or horizontal components of diagonal movements can be performed faster and more precisely than vertical ones (Collewijn & Tamminga, 1984; Rottach et al., 1996).

Vidal and colleagues were the first to show the feasibility of identifying smooth pursuit eye movements in real time and matching them to the course of a moving object (Vidal et al., 2013; Vidal & Pfeuffer, 2013). Drewes and Schmidt (2007) suggest that interaction via gaze gestures could be performed without calibration to the individual subject. Cymek et al. (2014) used smooth pursuit movements to enter numbers on a PIN pad. Even without individual calibration, direction-based classification proved to be a robust approach. At the same time gaze interaction based on pursuing objects' movements was accompanied by high user acceptance ratings.

[1] A calibration process the user does not explicitly identify as such.

[2] More precisely the number of characters -1, as the measurement starts as soon as the first character is typed

DOI 10.16910/jemr.8.1.2
This article is licensed under a
Creative Commons Attribution 4.0 International license.
ISSN 1995-8692

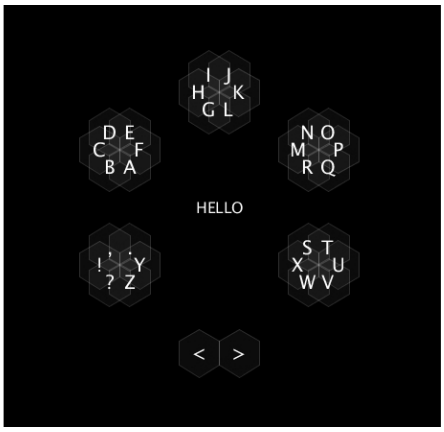


Figure 1. Cluster and character layout of SMOOVS in Phase 0 with current word in the central area

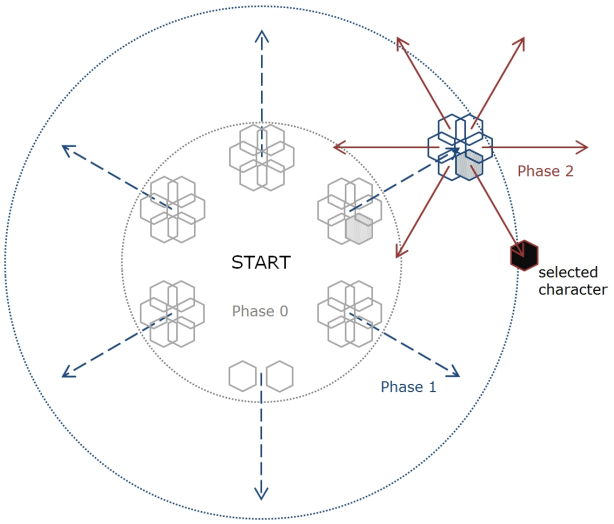


Figure 2. Interaction: display dynamics

SMOOVS

Our smooth pursuit gaze speller called SMOOVS, is an approach towards robust, calibration-free gaze spellers for public displays. It combines a two-stage interface concept (Huckauf & Urbina, 2008; Bee & André, 2008) with interaction designed specifically for smooth pursuit movements. The detection algorithm is related to the approach of Cymek et al. (2014). Layout and interaction are designed to provide intuitive interaction and facilitate closed-loop-pursuit as early as possible.

Layout and Interaction Design

Similar to the layout of the EEG[3]-based Hex-O-Spell (Blankertz et al., 2006), a hexagonal layout with hexagonal tiles is used. This approach is supported by a study on smooth pursuit-based interaction, where the detection rate of four and six objects was similar, but dropped when presenting more than six objects (Vidal & Pfeuffer, 2013). We use six interactive objects in two interaction stages: Six clusters of characters, each consisting of at most six character tiles (see Figure 1). Each cluster comprises six neighboring letters of the alphabet, respectively the letters Y, Z and special characters. Within each cluster, the first and last character of the cluster appear closest to the center of the screen. By looking at these two closest tiles, the user can determine the range of characters covered by the cluster.

To achieve best possible discrimination between the interactive objects, the clusters are arranged in a circular layout around a central area, an idle area where the currently typed word is displayed. As long as the user's gaze remains within that area, the objects do not move. Returning the gaze position there at any time interrupts the interaction and sets the system back into idle phase. Selected characters are appended to the current word shown in the central area. Text size is adaptive to word length in order to prevent involuntary initiation of the interaction by reading the word. After the current word is confirmed, it is appended to the sentence (i.e. to the list of previously typed words) at the bottom of the screen.

Display Dynamics. The stimulus movement consists of two parts. First, the character clusters move outwards. If a valid pursuit movement is detected, the individual character tiles of the detected cluster start moving away from each other (see Figure 2). The clusters' movement directions are distinct from the direction of the tiles within clusters. This minimizes the variability in difficulty to follow objects' trajectories as a change in movement orientation occurs for all objects. None of the character clusters moves solely horizontally to avoid involuntary interaction when reading the current word. All characters of the English alphabet and four special characters (. , ? !) are available. The bottom character cluster consists of only two tiles for correction and confirmation of the current word.

We define four phases of interaction. As discriminable feedback is very important whenever the same modality is used for perception and control (Majaranta, 2011), differentiated visual feedback is provided to the user. In order to avoid distraction, we designed subtle, but distinct feedback of the interaction phase and system state.

Phase 0 is an idle phase, where the tiles do not move. It is divided into an inactive and an active condition. During inactive condition, interaction cannot be initiated. The system waits for the user to return the gaze to the central area. This avoids involuntary initiations of the tiles' movements. In active condition, the system waits for the user to start the interaction by looking at the desired character. When the gaze position is moved from the center towards one of the character clusters,

[3] Electroencephalography, the measurement of electrical activity in different parts of the brain

DOI 10.16910/jemr.8.1.2
This article is licensed under a
Creative Commons Attribution 4.0 International license.
ISSN 1995-8692

interaction (phase 1) is initiated. The aim is to design the interaction to facilitate closed-loop-pursuit as early as possible. Ideally, it should directly follow the initial saccade towards the character cluster, which triggers the start of the movement. To the user, the switch from inactive to active condition is disclosed by a subtle change in saliency of the character clusters.

Phase 1 is the first movement, where character clusters move apart from each other. At the end of Phase 1, a character cluster is selected if the user's gaze path matches the movement path of a cluster (see section on the classification algorithm for details).

Phase 2 represents the second stage of movement, where individual tiles move away from each other. Only character tiles of the selected cluster are explicitly visible. After 200 ms subtle visual feedback (slightly higher saturation of the tile) is presented continuously to indicate which character tile would be selected according to the currently detected gaze path. At the end of Phase 2, a character is selected if the user's gaze path matches the movement path of a character tile. The tiles of other character clusters move as well, but are presented as barely visible tile shadows not showing characters to provide a dynamic impression without distracting the user.

Phase 3 is the final movement where all tiles move back to their initial positions. If the detection of a character was successful, salient visual feedback of the selected character is given by high saturation and broader edges of the tile. Additionally, a short faint sound is presented as low-key auditory feedback. All other tiles are presented as barely visible shadows without characters, but as all move back towards their initial position, an impression of optical flow is created which guides the user back to the center. With completion of Phase 3, the system state changes to phase 0, inactive condition. If no valid gaze path was detected in Phase 1, Phase 2 is skipped and the character tiles move back to their initial positions as well.

Our approach to provide an idle phase from which the user initiates interaction requires appropriate timing for the start of the character clusters' movement and sufficient accuracy of the eye-tracking system in the central area. Therefore, prior to interaction, a one-point-calibration is performed at the center of the screen. In the following section we specify the technical environment in which SMOOVS was realised. Subsequently, we describe our one-point-calibration, pre-test and detection algorithm, before the empirical evaluation is reported.

Technical Environment

For development and evaluation of SMOOVS, a SMI RED-oem eye-tracker running at a sampling rate of 60 Hz was used as its specifications are close to mass-market available models and it supports a factory-default calibration. A 24" monitor (pixel pitch 0.27 mm) operating at 60 Hz was connected to the experimenter's computer behind a screen. At a distance of 60 cm from user to screen, 1 ° visual angle corresponds to 38.8 pixels (px). Using the gaze interaction software *Mousey* (Lutz, 2013), mouse movements were emulated based on gaze position. The gaze speller itself uses cursor position as a substitute for gaze position, hence it is compatible with any eye-tracking-system providing mouse cursor positioning. Additionally, this allows developing the speller without an actual eye-tracker. With the eye-tracker present, the mouse cursor is hidden from the participant to avoid visual distraction. The speller was realised in Processing, a Java-based programming language convenient for designing dynamic graphical user interfaces. Additional functionality for experimentation (audio stimulus presentation, event logging and keyboard controls for the investigator) was included as well.

One-Point Calibration

As the interaction design specifies the central area as an idle spot from where the user initiates the interaction, gaze positions in this area have to be sufficiently accurate. For this reason, a one-point calibration is performed prior to interaction. To achieve independence of the eye-tracker in use we created our own one-point calibration method. A fixation cross is displayed at the center of the screen for 800 ms. To avoid sampling the orientation reaction and saccade to the stimulus, only the gaze positions of the final 300 ms are used for calibration. Means of the x- and y- distance between stimulus and gaze positions are calculated, as well as the standard deviation of total distance. If standard deviation is less than five pixels, the calibration is accepted and all subsequent gaze positions are corrected by the x- and y-means. If the standard deviation is higher, the calibration procedure is repeated as we assume that the participant did not precisely focus the fixation cross.

Pre-Test

For self-paced interaction, we defined the central idle area, where sufficient eye-tracker accuracy is needed. We conducted an exploratory pre-test to determine accuracies of the eye-tracker's factory default calibration and our one-point-calibration. These accuracies set constraints which have to be considered in software, algorithm and interaction design.

In the second part of the pre-test, we used an early development version of SMOOVS to determine the optimal distance between the center and the point at which the interaction is initiated. Because of latencies, movement of the character clusters has to be initiated before the measured gaze position is on the clusters. The goal is to provide a 'natural' or 'gliding' feeling to the user when starting the movement of the clusters with an initial saccade. Six participants (50 % female) attended the pre-test, all had previous experience with

DOI 10.16910/jemr.8.1.2
This article is licensed under a
Creative Commons Attribution 4.0 International license.
ISSN 1995-8692

Table 1
Pre-Test: Calibration results

		outer group	inner group	center
distance [px] using default calibration	*M*	92.51	84.43	81.01
	SD	23.44	33.74	24.36
distance [px] using 1-point-calibration	*M*	64.37	25.83	11.18
	SD	24.37	12.46	7.38

eye tracking. Participants were asked to sit steady, but no chin rest or other artificial support was used.

Design and Procedure

Calibration Accuracy. On a black background, nine white fixation crosses were displayed one after the other in a randomized order. To determine calibration accuracy at different positions at the screen, the stimuli were arranged in three groups: An inner and an outer group of four crosses in rectangular alignment and one central cross. The distance from the center to the inner group was about 5 ° viewing angle, and about 10 ° to the outer group. Each cross was presented twice during the trial. Participants were asked to fixate the cross with their eyes, then press a key and keep fixating until the cross disappeared. Hereby, we ensured that the sampling of gaze data happened only when the cross was fixated. Related to the sampling of the one-point calibration, we used 300 ms of each stimulus presentation for our analysis. The independent variable of the within-subjects design was the stimulus group (outer group, inner group, center). One dependent variable was the calibration accuracy, given by the euclidean distance *d* between gaze and stimulus position (cf. equation 1). For statistical analysis, we used the mean of all measured distances, segmented by stimulus groups.

$$d = \sqrt{(x_{gaze} - x_{stimulus})^2 + (y_{gaze} - y_{stimulus})^2} \qquad (1)$$

Subsequently, we simulated a one-point calibration as described above. The distance between the gaze position corrected by one-point calibration and stimulus position is the second dependent variable of the pretest. For all statements regarding calibration accuracy, the values given represent a combination of several possible sources of errors. The measurements combine the accuracy of the eye-tracker with the participants' level of preciseness in fixating the targets. Hence, the values are suitable for practical design and parametrisation considerations.

Start Distance. For determination of the optimal distance from the center at which the interaction is initiated, we used a modified development version of

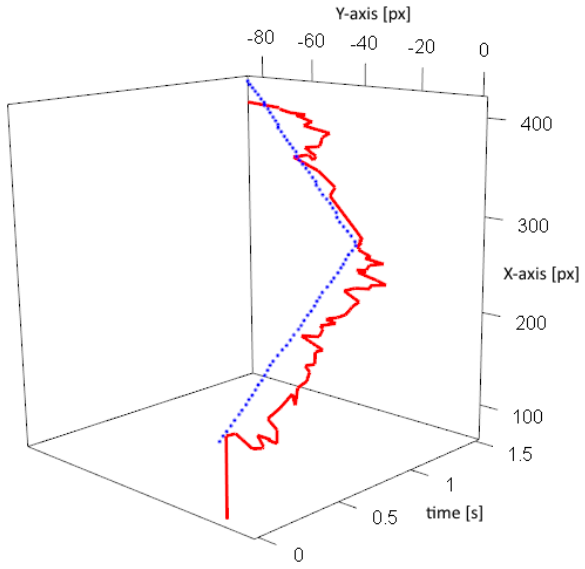


Figure 3. Exemplary stimulus movement (blue, dotted) and gaze path (red) over time

SMOOVS, where the start distance is adjustable to three levels (50, 65, 80 px distance from the center). Independent variable of the within-subjects design was the start distance, dependent variable was subjective feedback. After each condition, participants rated effort and comfort of starting the interaction. Concluding the experiment, they were asked to state their favourite condition. The sequences of start distances were randomised across participants.

Results and Discussion

Calibration Accuracy. The mean values (*M*) and standard deviations (*SD*) of the distances are shown in Table 1. Using repeated measures ANOVA at a significance level of $\alpha = 0.05$, the effect of stimulus group on factory default calibration accuracy was not statistically significant. The effect of stimulus group regarding one-point calibration accuracy was significant, $F_{0.05}(2,10) = 34.18,\ p < 0.001$. The accuracy of the factory default calibration was not sufficient to distinguish the central idle area as the distance was above 80 px, which exceeds the radius of the idle area. The one-point calibration delivers high accuracy in the central area, but deviation increases significantly with distance to the center. For software and interaction design, this allows the use of relatively conservative criteria in the center. As the accuracy decreases with increasing distance, the detection criteria for the distant parts of the interface (i.e. for Phase 2) need to be liberal.

Start Distance. Subjective user feedback given directly after the individual start distance conditions did not reveal any significant results. Nonetheless, a dis-

DOI 10.16910/jemr.8.1.2
This article is licensed under a
Creative Commons Attribution 4.0 International license.
ISSN 1995-8692

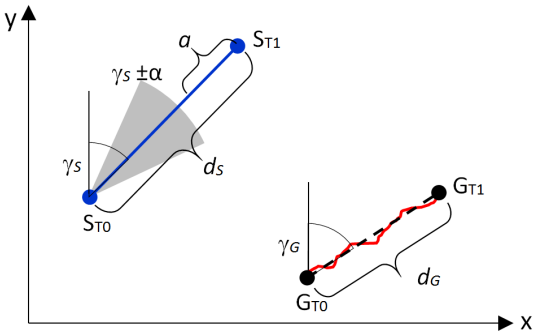


Figure 4. Algorithm: Mode of operation

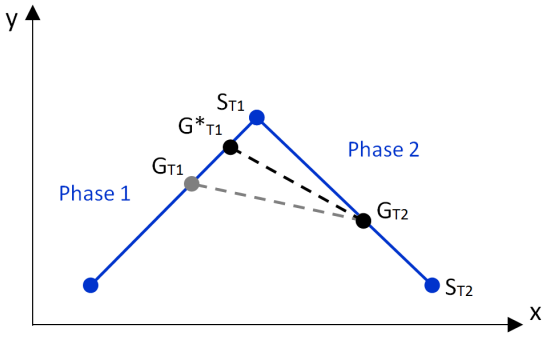


Figure 5. Algorithm: Compensating the latency error

tance of 65 px from the center of the screen was rated as favourite distance by four of six participants. In the final implementation of SMOOVS, this distance is used.

Latency Issues. Using gaze data collected in the pretest, we performed an exploratory in-depth analysis to calculate the delay of the technical system. This delay occurs from asynchronous screen refreshing, cursor positioning and sampling of the eye-tracker, as well as data processing time and communication lags of the eye-tracker and gaze speller software. In cases where stimulus movement changes its direction in a sharp angle, the participants' orientation reaction and subsequent smooth pursuit movement can be easily identified by visual inspection of the gaze data. Using markers sent to the eye-tracking data stream by SMOOVS, the temporal relations between events in software and the corresponding time in gaze data were analysed. The combination of delay originating from technical sources on one hand and from cognitive, perceptual and physiological processes on the other hand (excluding open-loop pursuit, as this can be identified in the data), was observed to range around 200 ms. At high object movement speeds, this represents a delay of more than half the available data points of a movement. Hence, this delay has to be taken into consideration in the design of the classification algorithm.

Classification Algorithm

In context of the desired real-time detection of smooth pursuit movements, we define a basic algorithm suitable for a robust prototype of the gaze-speller, which does not rely on absolute or precise gaze positions. Therefore, we analysed tracks of smooth pursuit versus stimulus movements. Figure 3 shows an exemplary gaze path (red) and the corresponding stimulus path (blue, dotted) over time, X- and Y-axis. There is both an offset due to calibration inaccuracy and a temporal delay. The initial saccade at the beginning of the movement (t = 0 s) is clearly visible as well.

Our algorithm is based on the vectors between start and end positions of both stimulus and smooth pursuit movements. On the screen, the stimulus moves from position S_{T0} to S_{T1}. The stimulus vector is defined by its angle γ_S and distance d_S. The gaze vector is calculated using the gaze positions G_{T0} and G_{T1} detected at the beginning and end of the stimulus movement, and defined by γ_G and d_G, accordingly. These parameters are shown in Figure 4, a hypothetical gaze path is drawn in red colour, the corresponding gaze vector as a dashed line. By comparing stimulus and gaze vectors, the classification is performed independent of absolute positions. It is based on an angular criterion allowing an angular error $\pm\alpha$ and a distance criterion allowing a distance error a, which are defined in advance. Classification is successful if the detected gaze vector is within these criteria. Therefore, equations 2 and 3 have to be met.

$$\gamma_S - \alpha \leq \gamma_G \leq \gamma_S + \alpha \tag{2}$$

$$d_G \geq d_S - a \tag{3}$$

For Phase 1 (the user is following a cluster of characters), this approach works fine as the stimulus movement is started by an eye movement in the same direction. Phase 2 starts based on a point in time, namely when Phase 1 is finished and a valid pursuit movement was detected. For the detection of the second smooth pursuit movement, the system delay has to be considered. To compensate for artefacts arising from that delay, the algorithm for detecting the second movement has been adapted as follows. Instead of using the detected gaze position at the beginning of movement G_{T1}, a sample recorded after 200 ms delay (G^*_{T1}) is used. By compensating for system latency, the angular error is reduced (compare Figure 5). On the other hand, the distance error increases as the calculated distance will decrease as a result of the geometrical relations. This can be compensated by defining more liberal distance criteria. The possible tolerance of the angular criterion is limited to $\pm 30^\circ$ maximum to avoid overlap between movements. The distance criterion though can be set

DOI 10.16910/jemr.8.1.2
This article is licensed under a
Creative Commons Attribution 4.0 International license.
ISSN 1995-8692

liberal, as long as it is above zero. Including these considerations, we used $\alpha = 29^\circ$ and $a = 0.8\,d_S$ as parameters for experimentation.

The classification algorithm is independent of absolute positions. To realise the central idle area, allowing the user to initiate the interaction, a one-point calibration at the center of the screen is necessary. In the detection of the central idle area, a hysteresis is included to eliminate unintended object movement initiations due to inaccuracies of the eye-tracker. The character clusters start moving as soon as the distance between gaze position and center exceeds 65 px. To get back into the active idle phase, the distance has to fall below two thirds of that value (43 px).

Empirical Evaluation

The empirical evaluation aimed at validating the interaction concept and algorithm in a controlled, but realistic setting. We investigated the influence of object movement speed on text entry rate, error rate and perceived subjective quality of the interaction.

Experimental Design

Object movement speed was varied over four different speed conditions (200, 260, 300, 340 px/s) in a within-subjects design. The slowest speed level corresponds to the smooth pursuit pace rated most pleasant in the study of Cymek et al. (2014). The other speed levels increase by 40 px/s (approx. 1° visual angle per second) each to maximize text entry rate. In order to balance training effects, the sequence of speed conditions was fully randomised. Effects on the performance measures *words per minute*, *number of completed gaze paths per minute*, *number of corrections per sentence* and *number of discontinuations per sentence* were studied. The number of discontinuations is the amount of cases where a pursuit movement was aborted and the user's gaze returned to the central area. We formulated the following hypotheses regarding the performance measures: We postulated that the number of completed gaze paths per minute as well as the number of corrections and discontinuations per sentence rise with increasing object movement speed. The combination of these effects leads to the assumption that at higher speeds, the benefit of an increased number of gaze paths per minute is mitigated by higher error rates originating from more corrections and discontinuations. Supplementary to performance data, subjective data was acquired by asking participants to rate ease, effort and comfort of pursuing characters with the eyes. Rating was conducted using a semantic differential. In an electronic questionnaire, participants set a mark on an unmarked line between two semantic poles, resulting in a value between zero and 100. Additionally, we asked for feedback on the perceived character movement speed between the poles *too slow* and *too fast*, where the optimal speed corresponds to a value of 50.

Task and Procedure

Participants were asked to enter the holoalphabetic German sentence *'Zwei Boxkaempfer jagen Eva quer durch Sylt. Nein, oder? Ja!'* (Pommerening, 2013). This sentence includes all supported characters to ensure that each implemented character movement path is performed at least once at each speed level. The same sentence was used for all speed conditions and dictated automatically word by word via the gaze speller software. A one-point calibration was performed at the beginning of each condition, then the dictation started. A short training session was conducted prior to the investigation. The design of the training session was similar to the experiment, but used a shorter sentence and steadily increased the speed from lowest to highest for each participant. After completing each condition, participants were asked to fill out an electronic questionnaire on subjective ratings. In order to gain data from a realistic setting, participants were asked not to move their heads extensively, but no chin rest or other artificial support was used.

Participants

To allow complete permutation of all speed condition sequences, data of 24 participants was collected. As we experienced irregularities in audio output and frame rate control of Processing, five sequences were repeated with additional participants. In the analysis, the proper datasets of 24 participants (age: $M = 25.4$, $SD = 3.41$), 50 % women, were used. Eight people wore soft contact lenses. We purposefully excluded participants wearing glasses, as we wanted to validate our interaction concept and algorithm rather than the robustness of the eye-tracking hard- and software. A quarter of them had previous experience with gaze interaction. Participants received a financial compensation of EUR 10 or partial course credit for attendance.

Results

Statistical analysis was performed using repeated measures ANOVA at a significance level of $\alpha = 0.05$. Mauchly's test for sphericity was performed prior to analysis, no correction was needed. To determine differences between levels of object movement speed, post-hoc paired t-tests with Bonferroni adjustment were conducted.

Performance Measures. Mean values (M) and standard deviations (SD) of the dependent variables are shown in Table 2. Object movement speed has a significant effect on the number of completed gaze paths per minute, $F_{0.05}(3,69) = 24.83$, $p < 0.001$, supporting our hypothesis. The number of completed gaze paths per minute increases significantly between 220 and 260 px/s ($p = 0.03$) and between 260 and 300 px/s ($p = 0.008$). The difference between 300 and 340 px/s is

DOI 10.16910/jemr.8.1.2
This article is licensed under a
Creative Commons Attribution 4.0 International license.
ISSN 1995-8692

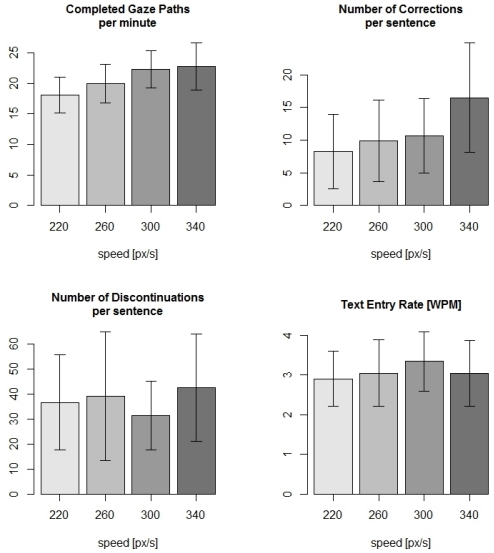


Figure 6. Performance results

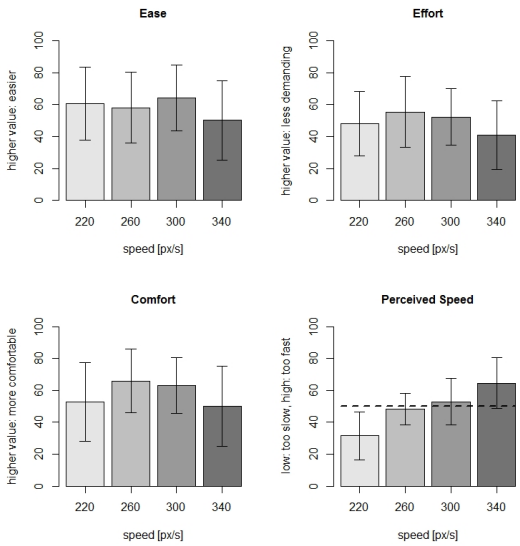


Figure 7. Subjective measures

Table 2
Performance measures

speed [px/s]		220	260	300	340
Completed gaze paths per minute	*M*	18.12	19.95	22.31	22.79
	SD	2.96	3.18	3.07	3.78
Corrections per sentence	*M*	8.21	9.88	10.63	16.50
	SD	5.72	6.25	5.72	8.40
Discontinuations per sentence	*M*	36.50	39.08	31.33	42.42
	SD	18.99	25.77	13.85	21.37
Words per minute	*M*	2.90	3.05	3.34	3.04
	SD	0.69	0.84	0.75	0.83

Table 3
Subjective measures (on a score from 0 to 100)

speed [px/s]		220	260	300	340
ease	*M*	60.50	58.13	64.33	50.13
	SD	22.89	22.12	20.55	24.74
effort	*M*	52.04	44.67	47.75	59.12
	SD	20.28	22.11	17.77	21.62
comfort	*M*	52.67	65.83	63.08	50.04
	SD	24.56	19.91	17.29	25.23
perceived speed	*M*	31.54	48.17	52.92	64.63
	SD	15.09	9.86	14.64	15.76

not significant.The effect of object movement speed on the number of corrections per sentence is significant, $F_{0.05}(3,69) = 10.74$, $p < 0.001$. A post-hoc test revealed a significant increase ($p = 0.012$) of the number of corrections per sentence between 300 and 340 px/s. Even though there is no significant effect of object movement speed on the number of discontinuations per sentence, there is still a notable pattern in the data. Looking at it descriptively, both the mean and the standard deviation of the number of discontinuations are lowest at an object movement speed of 300 px/s. On text entry rate, the main effect is significant, $F_{0.05}(3,69) = 2.97$, $p = 0.037$, but the generalized η^2-measure of effect size is notably low (0.042). The statistical results in combination with the descriptive analysis of the data on text entry rate support our assumption that at high speed the benefit of increased number of gaze paths per minute is mitigated by higher error rates. In our experiment, the highest text entry rate is not achieved at the highest object movement speed of 340 px/s, but at 300 px/s (compare Figure 6).

Perceived Quality of Interaction. In general, there were no significant effects of object movement speed on subjective ratings. Nevertheless, pursuing the character tiles was rated less strenuous and more comfortable at the two medium speed levels in comparison to the slowest or fastest condition. Furthermore, the perceived speed was rated close to optimal - a value around 50 - for the two medium speed levels (compare Table 3 and Figure 7). When asked after the experiment, users frequently gave the feedback, that the character layout was not easy to get used to. Even after the short training session, it occasionally happened that participants needed to search for the right character cluster, involuntarily initiating an interaction by their visual search. Participants rated the perceived input speed as relatively fast. The dynamics of the interface, the smooth, flowing movement of the character tiles was frequently mentioned to be a very pleasant, positive way of interaction.

DOI 10.16910/jemr.8.1.2
This article is licensed under a
Creative Commons Attribution 4.0 International license.
ISSN 1995-8692

Discussion

This paper shows that the concept of smooth pursuit-based gaze interaction can be applied to complex selection tasks like a gaze speller. Utilizing characteristics of these movements, we achieve sufficient accuracy with a one-point calibration.

Effects of object movement Speed

With increasing object movement speed, the number of completed gaze paths and the number of corrections rise as well. At 340 px/s, the drawbacks of corrections outweigh the benefits of the higher number of gaze paths. The text entry rate cannot be boosted by further increasing object movement speed. In our experiment, an average text entry rate of 3.34 WPM was reached at 300 px/s. More intensive training is likely to result in less corrections, providing a higher text entry rate. Looking at mean and standard deviation of the number of discontinuations being lowest at an object movement speed of 300 px/s, we assume that this pace might allow interaction in a rhythm which is comfortable for the users, as both the mean value is lowest and the low standard deviation indicates less variance between participants. Subjective ratings like ease, effort and comfort of following the characters with the eyes as well as the perceived efficency, speed and performance of the system indicate users' preference for both 260 px/s and 300 px/s object movement speed. Hence, we conclude that a speed of 300 px/s is superior compared to the 220, 260 and 340 px/s.

Comparison with other gaze spellers

In general, the text entry rate of SMOOVS is lower than that of other gaze spellers which use fully calibrated eye-trackers. However, the reported text entry rate depends on the calculation method. It should base on the number of final characters (excluding correction gestures), but could be computed using the number of all gaze paths per minute (including correction gestures). In other publications, the method of computation is not always indicated precisely. We employed the more conservative measure (3.34 WPM). Using the number of all gaze paths, a text entry rate of 4.5 WPM is achieved. This number is close to other gesture-based gaze spellers, e.g. 4.9 WPM with *EyeWrite* (Wobbrock et al., 2008).

More intensive training, comparable to the amount of training used in other studies, e.g. *Dasher* in Ward et al. (2000), is likely to result in less corrections, resulting in a higher text entry rate. We purposefully refrained from long training sessions to obtain a result compatible with the concept of spontaneous gaze interaction on public displays. In this context of application, high text entry rate is less important than intuitiveness and robustness, as users are typically not required to enter long text. A unique feature of SMOOVS is the explicit use of smooth pursuit movements under realistic experimental conditions using merely a one-point-calibration.

Limitations

User feedback showed that the character layout was not as easy to comprehend as intended. Suggestions for improvement included a horizontal arrangement of the characters or a layout comparable to mobile phones, which consists of nine clusters of three to four characters each. Both approaches are not suitable for gaze interaction using only a one-point-calibration, as the horizontal arrangement needs accurate gaze positions and using nine clusters contradicts the results of Vidal and Pfeuffer (2013), where the detection rate dropped significantly with more than six interaction objects. One possible solution is placing a hint around the center of the screen. Hints in the parafoveal field of view showing the range of each cluster (A-F, G-L etc.) could help choosing the right cluster while the user's gaze is still within the central idle area.

An important limitation of the current implementation is the lack of timing accuracy in the frame rate regulation of the programming language Processing. Data analysis showed that the audio output plug-in caused irregularities in the frame rate at times. Due to this, we had to repeat five trials with different participants. Although Processing is well suitable for developing interactive prototypes, precise timing requirements for scientific research are not fully met.

Strictly speaking, the proposed algorithm does not detect the whole smooth pursuit movement as it is based on a vector defined by only two gaze points. On the other hand, it is a simple, fast, robust, real-time algorithm which proved to be sufficient.

The controlled laboratory conditions used are not accordant to real world conditions of public displays. Additionally, as the majority of participants were students, this sample is not a representative selection. Further research on gaze interaction with a representative sample of the intended user group, including elderly people, is needed.

Outlook

With the prevalence of smartphones, users are accustomed to automatic word completion as a component of any modern text entry system. Based on a language database and probabilistic approaches, such features could be implemented as well. If the saliency of character tiles depended on their probability to occur as the next letter, visual search is simplified. In addition to the visual appearance, tolerance criteria for the algorithm could be changed adaptively as well. The algorithm itself could be compared to more complex approaches to real-time smooth pursuit detection which so far have been used on calibrated systems. Vidal and Pfeuffer

DOI 10.16910/jemr.8.1.2
This article is licensed under a
Creative Commons Attribution 4.0 International license.
ISSN 1995-8692

(2013) used product-moment-correlations to match object and gaze path. Comparing this method, a machine learning based approach like hierarchical temporal memory (Rozado, Rodriguez, & Varona, 2010; Rozado, Agustin, Rodriguez, & Varona, 2012) and our algorithm in a real-time, one-point-calibrated smooth pursuit interaction paradigm is a logical next step. In addition to technical refinements, the interaction design could be improved as well. Implicit one-point-calibration utilising appropriate stimuli to catch attention or pursuit movement-based calibration (Pfeuffer et al., 2013) might further enhance user experience.

Conclusion

We developed the first gaze speller explicitly utilizing smooth-pursuit eye movements and their particular characteristics. It achieves sufficient accuracy with a one-point calibration. In the development, we followed a holistic approach that accounts for both technical and human limitations and inaccuracies. For interaction with dynamic interfaces, high accuracy and precision of the eye-tracker and calibration are not important. But as the trajectory of a moving stimulus is used, low system latency for detection of the gaze position is critical. In an empirical evaluation, users showed highest overall performance at 300 px/s object movement speed. Subjective ratings support the finding that this pace is superior.

References

Arif, A. S., & Stuerzlinger, W. (2009, September). Analysis of text entry performance metrics. *2009 IEEE Toronto International Conference Science and Technology for Humanity (TIC-STH)*, 100–105.

Bahill, A. T., & McDonald, J. D. (1983, January). Smooth pursuit eye movements in response to predictable target motions. *Vision research*, *23*(12), 1573–83.

Bee, N., & André, E. (2008). Writing with your eye: A dwell time free writing system adapted to the nature of human eye gaze. *Perception in Multimodal Dialogue Systems*, 111–122.

Blankertz, B., Dornhege, G., Krauledat, M., Schr, M., Williamson, J., & Murray-smith, R. (2006). The Berlin Brain-Computer Interface presents the novel mental typewriter HEX-O-SPELL. In *Proceedings of the 3rd international brain-computer interface workshop and training course* (pp. 108–109). Graz.

Burke, M. R., & Barnes, G. R. (2006, November). Quantitative differences in smooth pursuit and saccadic eye movements. *Experimental brain research*, *175*(4), 596–608.

Collewijn, H., & Tamminga, E. (1984). Human smooth and saccadic eye movements during voluntary pursuit of different target motions on different backgrounds. *The Journal of physiology*, 217–250.

Cymek, D., Venjakob, A., Ruff, S., Lutz, O. H.-M., Hofmann, S., & Rötting, M. (2014). Entering PIN Codes by Smooth Pursuit Eye Movements. *Journal of Eye Movement Research*, *7*(4), 1–11.

Drewes, H., & Schmidt, A. (2007). Interacting with the computer using gaze gestures. In *Human-computer interaction-interact 2007*.

Hansen, D. W., Skovsgaard, H. H. T., Hansen, J. P., & Mollenbach, E. (2008). Noise tolerant selection by gaze-controlled pan and zoom in 3D. *Proceedings of the 2008 symposium on Eye tracking research & applications - ETRA '08*, 205.

Holmqvist, K., Nyström, M., Andersson, R., Dewhurst, R., Halszka, J., & van de Weijer, J. (2011). *Eye Tracking : A Comprehensive Guide to Methods and Measures*. New York: Oxford University Press.

Huckauf, A., & Urbina, M. (2008). Gazing with pEYEs: towards a universal input for various applications. In *Proceedings of the 2008 symposium on eye-tracking research & applications* (pp. 51–54).

Jacob, R. J. K. (1991, April). The use of eye movements in human-computer interaction techniques: what you look at is what you get. *ACM Transactions on Information Systems*, *9*(2), 152–169.

Lutz, O. H.-M. (2013). *Mousey: A multi-purpose eye-tracking and gaze-interaction interface* (Tech. Rep.). Technical University Berlin, Fachgebiet Mensch-Maschine-Systeme. Retrieved from `http://www.mms.tu-berlin.de/fileadmin/fg268/Mitarbeiter/Mousey2_Documentation.pdf`

Majaranta, P. (2011). Communication and text entry by gaze. In P. Majaranta, H. Aoki, & M. Donegan (Eds.), *Gaze interaction and applications of eye tracking - advances in assistive technologies.* Hershey, PA: IGI Global.

Majaranta, P., & Räihä, K. (2002). Twenty years of eye typing: systems and design issues. In *Proceedings of the 2002 symposium on eye tracking research & applications* (pp. 15–22).

Mollenbach, E., Hansen, J., & Lillholm, M. (2013). Eye Movements in Gaze Interaction. *Journal of Eye Movement Research*, *6*, 1–15.

Morimoto, C., & Amir, A. (2010). Context switching for fast key selection in text entry applications. In *Proceedings of the 2010 symposium on eye-tracking research & applications* (Vol. 1, pp. 271–274).

Pfeuffer, K., Vidal, M., & Turner, J. (2013). Pursuit calibration: making gaze calibration less tedious and more flexible. *Proceedings of the 26th annual ACM symposium on User interface software and technology*, 261–269.

Pommerening, K. (2013). *Pommerenings Pangramm-Sammlung.* Retrieved from `http://www.staff.uni-mainz.de/pommeren/Miszellen/Pangramme.html`

Rottach, K. G., Zivotofsky, a. Z., Das, V. E., Averbuch-Heller, L., Discenna, a. O., Poonyathalang, a., & Leigh, R. J. (1996, July). Comparison of horizontal, vertical and diagonal smooth pursuit eye movements in normal human subjects. *Vision research*, *36*(14), 2189–95.

Rozado, D., Agustin, J. S., Rodriguez, F. B., & Varona, P. (2012, January). Gliding and saccadic gaze gesture recognition in real time. *ACM Transactions on Interactive Intelligent Systems*, *1*(2), 1–27.

Rozado, D., Rodriguez, F., & Varona, P. (2010). Optimizing hierarchical temporal memory for multivariable time series. In *Proceedings of the 20th international conference on artificial neural networks icann 2010* (pp. 506–518).

Tula, A., de Campos, F., & Morimoto, C. (2012). Dynamic context switching for gaze based interaction. *Proceedings of the 2012 Symposium on Eye Tracking Research and Applications*, *1*(212), 353–356.

DOI 10.16910/jemr.8.1.2
This article is licensed under a
Creative Commons Attribution 4.0 International license.
ISSN 1995-8692

Vidal, M., Bulling, A., & Gellersen, H. (2013). 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* (pp. 439–448).

Vidal, M., & Pfeuffer, K. (2013). Pursuits: Eye-based interaction with moving targets. In *Chi '13 extended abstracts on human factors in computing systems* (pp. 3147–3150).

Villanueva, A., Cabeza, R., & Porta, S. (2004). Eye tracking system model with easy calibration. In *Proceedings of the 2004 symposium on eye tracking research & applications* (Vol. 1, p. 58113).

Wallace, J. M., Stone, L. S., Masson, G. S., & Julian, M. (2005). Object Motion Computation for the Initiation of Smooth Pursuit Eye Movements in Humans. *Journal of Neurophysiology*, 2279–2293.

Ward, D. J., Blackwell, A. F., & MacKay, D. J. C. (2000). Dasher—a data entry interface using continuous gestures and language models. *Proceedings of the 13th annual ACM symposium on User interface software and technology - UIST '00*, 2, 129–137.

Wobbrock, J., Rubinstein, J., Sawyer, M., & Duchowsky, A. (2008). Longitudinal evaluation of discrete consecutive gaze gestures for text entry. In *Proceedings of the 2008 symposium on eye tracking research & applications* (pp. 11–19).

DOI 10.16910/jemr.8.1.2
This article is licensed under a
Creative Commons Attribution 4.0 International license.
ISSN 1995-8692