EyeGrip: Detecting Targets in a Series of Uni-directional Moving Objects Using Optokinetic Nystagmus Eye Movements

Shahram Jalaliniya
IT University of Copenhagen
Copenhagen, Denmark
jsha@itu.dk

Diako Mardanbegi
IT University of Copenhagen
Copenhagen, Denmark
dima@itu.dk

ABSTRACT

EyeGrip proposes a novel and yet simple technique of analysing eye movements for automatically detecting the users objects of interest in a sequence of visual stimuli moving horizontally or vertically in front of the user's view. We assess the viability of this technique in a scenario where the user looks at a sequence of images moving horizontally on the display while the user's eye movements are tracked by an eye tracker. We conducted an experiment that shows the performance of the proposed approach. We also investigated the influence of the speed and maximum number of visible images in the screen, on the accuracy of EyeGrip. Based on the experiment results, we propose guidelines for designing EyeGrip-based interfaces. EyeGrip can be considered as an implicit gaze interaction technique with potential use in broad range of applications such as large screens, mobile devices and eyewear computers. In this paper, we demonstrate the rich capabilities of EyeGrip with two example applications: 1) a mind reading game, and 2) a picture selection system. Our study shows that by selecting an appropriate speed and maximum number of visible images in the screen the proposed method can be used in a fast scrolling task where the system accurately (87%) detects the moving images that are visually appealing to the user, stops the scrolling and brings the item(s) of interest back to the screen.

Author Keywords
Gaze tracking, Optokinetic Nystagmus (OKN) eye movements, Implicit interaction, Scrolling

ACM Classification Keywords
H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

INTRODUCTION

We are living in the digital information age where companies, organizations, and even end users are producing an enormous and rapidly growing flow of digital information. Users of Internet applications such as social networks have already been overloaded by tremendous amount of digital information ranged from textual to graphical contents. This has resulted in us to make our browsing more efficient by quickly moving our eyes across the contents and picking the contents that seem more interesting to us. The fact that our brain processes images significantly faster than text [4] might be one of the reasons of why we are often more engaged with images than textual information and why viewing pictures is among the most popular functions in social networks such as Facebook [19].

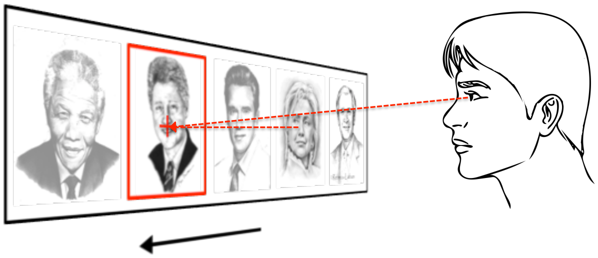


Figure 1. EyGrip technique to detect an object of interest among horizontally moving images

When people are browsing their Facebook [1] page on their mobile device, it's often that they quickly scan the Newsfeed by scrolling down or up the Facebook page until they find some interesting information. However, scrolling for navigation on small-screen devices has its own usability and inefficiency problems [9]. The three steps of a) scrolling, b) stopping the page, and c) bringing the desired content back to the display by scrolling back up are the main parts of browsing the contents. We go through the same steps when we search for a particular image in our photo gallery. Our ability to rapidly scan and process the visual cues that are quickly moving across our eyes, enables us to speed up the scrolling task. However, the third step (bringing the desired content back) can be a cumbersome task for users in a fast scrolling task since it requires a very high coordination between eyes, brain and our motor control system (e.g. touching the display with our fingers). Finding the target image that has gone out of the screen during a fast scrolling is not always easy and it sets a limitation to how fast the scrolling can be done.

[1] www.facebook.com

CHI'16, May 07-12, 2016, San Jose, CA, USA.
© 2016 ACM. ISBN 978-1-4503-3362-7/16/05...$15.00
DOI: http://dx.doi.org/10.1145/2858036.2858584

This paper proposes the EyeGrip method that enables computer systems to automatically detect the moving content that seems to be interesting for a user by monitoring and analysing the user's eye movements. Depending on the application, such systems can for example tag the content of interest in a series of scrolling contents or they can immediately react by stopping the content of interest in front of the user's view. EyeGrip provides an attentive scrolling mechanism which analyses the user's natural eye movements (Optokinetic Nystagmus) subtly in the background and it does not require any explicit command from the user or any change in their gaze behavior. Optokinetic Nystagmus (OKN) is a type of eye movement that occurs when a person tracks a moving field. OKN stabilizes images on the retina while viewing a sequence of moving objects. OKN has a sawtooth-like pattern that consists of alternating pursuit movements made in the direction of stimulus (slow phase) followed by saccacdes (fast phases). Generally, two forms of OKN have been described in the literature [26]. One is called Stare OKN which is a reflexive response that occurs when a viewer passively follows a moving visual field [16] and the other one is called Look OKN when a viewer voluntarily tracks moving stimulus in the visual field.

The principle behind the EyeGrip method is to analyze the combination of the saccades and smooth pursuits in the OKN eye movements to detect deviations in the OKN signal which is related to the long smooth pursuits or slow phase in the OKN eye movements (peaks in Figure 2). We used a machine learning approach to detect these peaks by feeding a window of the horizontal eye movement signal as a feature into the WEKA classifier. As we discuss this further in the paper, implementing EyeGrip does not necessarily require gaze estimation or any gaze calibration between the eye tracker and the display. However, depending on what approach is used for detecting a peak in the signal, we might need some algorithm calibration (not gaze calibration) or a learning phase to build a classifier as we did in our implementation.

In this paper, we show the feasibility of the EyeGrip method by detecting the images of interest in an image scrolling application. We further investigate the effect of two independent variables on the accuracy of the classification through a lab experiment. The first independent variable is *speed of scrolling*, and the second one is *maximum number of visible images in a single frame*. We manipulated the latter variable by changing image width. Based on our findings from the experiment, we propose some design guidelines for implementing EyeGrip. Finally, to demonstrate the utility of the EyeGrip technique in interactive systems, two follow up usability studies have been presented: 1) a picture selection application and 2) a mind reading game.

RELATED WORK

Gaze-based interaction

Using gaze as an input modality for computing devices has long been a topic of interest in HCI community, and it is due to the fact that humans naturally tend to direct eyes toward the target of interest. Gaze can be used both as an explicit and implicit input modality. Implicit input are actions and behaviors of humans, which are done to achieve a goal and are not primarily regarded as interaction with a computer, but captured, recognized, and interpreted by a computer system as input [23]. While explicit input are our intended commands to the system through mouse, keyboard, voice commands, body gestures, and etc.

Gaze for explicit input

One of the most explored explicit ways of using gaze to interact with computers is to use gaze as a direct pointing modality instead of mouse in a target acquisition task [12]. The target can be selected either by fixating the gaze for a while on a particular area (dwell-time) [25] or using a mouse click [14]. However, controlling cursor with eye movements is limited to pointing towards big targets due to the inaccuracy of gaze tracking methods and subconscious jittery motions of the eyes [29]. Eye-gesture is another explicit approach for gaze-based interaction where user performs predefined eye-strokes [8]. Previous studies [3, 14] have shown that using gaze as an explicit input modality is not always a convenient method for users. In fact, overloading eyes as humans' perceptual channel with a motor control task is not convenient [29].

Gaze for implicit input

In implicit method of using gaze in user interface design, natural movements of the eyes can be used to detect context, for example looking at certain objects in an environment can reveal interest of humans to those objects [17]. Gaze can also be used to infer about user's behaviour, for instance which objects attracts user attention during an everyday activity like cooking [20]. Another example of using gaze as an implicit input is to detect user's attention point and react to the users eye contact [24], or adapt user interface behavior [11] accordingly. The gaze data can also be used indirectly for interaction purposes [29, 13, 18, 28]. For instance, in the MAGIC pointing technique [29, 13], gaze data is used to move the cursor as close as possible to the target. Mardanbegi et al. [18] proposed a gaze-based interaction technique where the gaze data is used indirectly for head-gesture recognition. The other relevant work to our study is Pursuits interaction technique [28] which enables users to select an object on the screen by correlating eye pursuit movements with objects moving on the screen. The accuracy of their proposed technique depends on the difference of trajectories which means it fails to detect uni-directional moving objects due to the similarity of the trajectories in a uni-directional movement. On the contrary, our proposed EyeGrip method enables computer devices to detect the object of interest among uni-directional moving objects. EyeGrip is an implicit way of using gaze since we do not ask users to perform any kind of predefined eye-strokes or fixating on a particular target. The EyeGrip technique is based on analyzing natural eye movements for automatically detecting object of interest in a user interface.

Smooth pursuit recognition

The main part of our proposed approach is to automatically detect a deviation in the OKN eye movements when a particular object grabs user's attention. This deviation is related

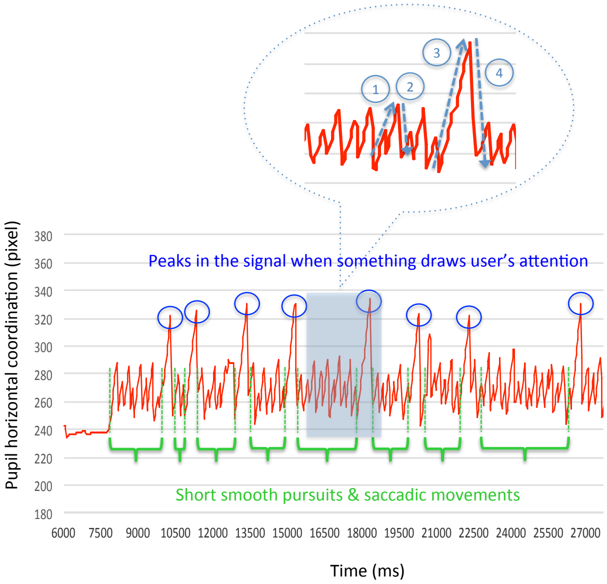


Figure 2. OKN signal generated from horizontal eye movements in a visual search task among uni-directional moving objects that move from the right side of the screen to the left. 1- Short smooth pursuit movements when eyes are scanning pictures, 2- Short saccade after a short pursuit when eyes are about to scan the next picture, 3- Long smooth pursuit (which may be supplemented by saccadic movements for fast moving objects) when an object draws user's attention, 4- Long saccade that takes the gaze back to the right area of the screen

to the slow phase of the OKN which is basically a combination of long smooth pursuits and saccadic movements. To the best of our knowledge this is the first study on using OKN in HCI. In the earlier studies, Kalman filters was used to process smooth pursuits [5, 1] while more recent works have analyzed both dispersion and velocity of the signal to classify smooth pursuits [21, 15]. Vidal and et al. [27] used a machine learning-based approach to detect pursuits by analyzing a combination of different features. In this study, we also use a machine learning algorithm to recognize patterns in the eye movement data. However, in contrast to the Pursuits [21, 15] method, we just use a single feature for classification. Our approach is explained in the next section.

THE EYEGRIP METHOD

When an object catches our visual attention, the eyes try to follow that moving object closely. This type of eye movements is called smooth pursuit. In contrast to other types of eye movements such as saccades and micro-saccades and also fixations that occur between saccades, pursuit parameters are generally more difficult to measure and are not as stereotyped as saccades [15]. Smooth pursuit consists of two phases: initiation and maintenance. Measures of initiation parameters can reveal information about the visual motion processing that is necessary for pursuit. Maintenance involves the construction of an internal, mental, representation of target motion which is used to update and enhance pursuit performance.

When we look at a series of linearly moving images, and we search for a particular image, our eyes perform a combination of saccadic and smooth pursuit movements (OKN). The smooth pursuit movements are relatively short when our eyes do not see an interesting image. As soon as an image draws user's attention, the maintenance phase of the smooth pursuit movement gets longer. In the EyeGrip technique, we exploit the difference between smooth pursuit lengths when the eyes are looking for an interesting object and when an object catches user's attention. In a visual search task among a series of uni-directional moving images, the viewer's eyes mainly move in the same direction as the moving contents. If we record the amplitude of the user's horizontal eye movements while looking at a series of moving images in the horizontal direction on the display, the generated signal looks like Figure 2 that illustrates a sawtooth like OKN signal. This figure shows the short saccadic and smooth pursuit eye movements that happen in a visual search task. The longer smooth pursuit movements occur when an object draws users' attention. In this phase of the visual search task (slow phase in OKN), eyes follow the object of interest for a longer time which generates a peak (deviation) in the signal. By detecting the moment and location of this peak (deviation), we are able to detect the object of interest among other moving objects.

We used a machine learning approach (Multilayer perceptron classifier) to detect these peaks by feeding a window of the horizontal eye movement signal as a feature into the WEKA classifier. To generate the OKN signal, we only need to detect eye movements which means there is no need for any gaze estimation or gaze calibration. In our experiment, we used a camera-based eye tracker to detect eye movements; however, to generate the OKN signal it is also possible to use other eye tracking methods such as Electrooculography (EOG) [6]. In our implementation, the classifier needs to be trained first. We collected training data from 15 participants in the experiment, and we used the same trained classifier for new participants in the follow up usability studies without adding any new training data. Since the accuracy of EyeGrip in the both usability studies for unseen data remained in the same range as the accuracy of EyeGrip in the experiment, we can conclude that the EyeGrip does not necessarily need any training phase for new users.

EXPERIMENTAL DESIGN

To characterise the eye movements in different conditions and investigate the accuracy of different algorithms, we conducted an experiment with two independent variables: 1) the speed of scrolling, and 2) the maximum number of images visible in the view-port (visible part of the sequence on the screen). To manipulate the maximum number of visible images, we can change either the size of the view-port, offset between images, or image width. Assuming that the offset between images and the width of the view-port are fixed, we changed the image width to manipulate the maximum number of images visible in the view-port. This means in some conditions the images are squeezed (Figure 4 (a) and (b)) however, since humans are extremely good in detecting faces even when they are deformed, we believe slightly squeezing images by only changing the image width has not a considerable effect on the recognition rate.

Con	Speed	Image width
1	Slow (1400 pixel/s ≃ $26.5°/s$)	Small (480 pixels)
2	Slow (1400 pixel/s ≃ $26.5°/s$)	Big (960 pixels)
3	Med (2000 pixel/s ≃ $37.5°/s$)	Small (480 pixels)
4	Med (2000 pixel/s ≃ $37.5°/s$)	Big (960 pixels)
5	Fast (2600 pixel/s ≃ $49°/s$)	Small (480 pixels)
6	Fast (2600 pixel/s ≃ $49°/s$)	Big (960 pixels)

Table 1. 6 different conditions used in the experiment

The dependent variables in our experiment are: 1) accuracy of the classification for detecting the moment when an image draws users' visual attention and 2) the error rate which is defined as number of target images missed by the participants divided by total number of target images.

Method

Participants

20 participants (mean age = 28, ranged from 20 to 56 years old, and 2 females) were recruited among local university staff and students to participate in the experiment. After preprocessing the data we removed the data of 5 participants which was not usable due to the inaccuracy of the eye detection for them. All of the participants had perfect visual acuity or wearing contact lens.

Apparatus

We used a home-made wearable monocular gaze tracker and the open-source Haytham gaze tracking software [2] to record the eye movement data (see Figure 4 (c)). The eye tracker was set to track the left eye for all the participants without considering the eye dominance. We assume that any possible difference between the movements of the left and right eyes will not be significant for our study. However, investigation of whether left and right eyes move differently in OKN due to the eye-dominance could be an interesting subject for the future research. The accuracy of our eye tracker is about 1 degree, and the frequency of the sampling eye data is 20 Hz. Although, in our experiment, we have not used the gaze data provided by the software. In fact, we did not calibrate the gaze tracker to calculate the gaze coordinate on the screen. We developed an application to display a series of horizontally moving images at a certain speed. The speed, direction, and the size of the images in the screen can be adjusted in the application. Both our gaze tracker software and the picture display application run on a HP laptop with a 8G RAM, Corei7 processor with the speed of 2.6 GHz, and a display with 1600 × 900 pixels resolution and 34.5 × 19.5 cm dimensions. The viewing distance form the display is about 60cm.

Procedure

The experiment started with a short introduction to the purpose of the experiment and the use of the apparatus. Then participants were asked to wear the gaze tracker, and we controlled if the gaze tracker is positioned appropriately in front of the participant's eye. Then each participant was asked to complete the task in six different conditions. The task was to look at a series of horizontally moving images of famous people's face (e.g. politicians, athletes, actors/actresses) on the screen and find the Bill Clinton's picture as target image. As soon as the participant recognizes Bill Clinton's face among other faces he/she should press space bar on the keyboard. Before starting the task, the participants were asked if they are familiar with Bill Clintons face or not. All of the participants mentioned that they know Bill Clinton, and they are able to recognize his face.

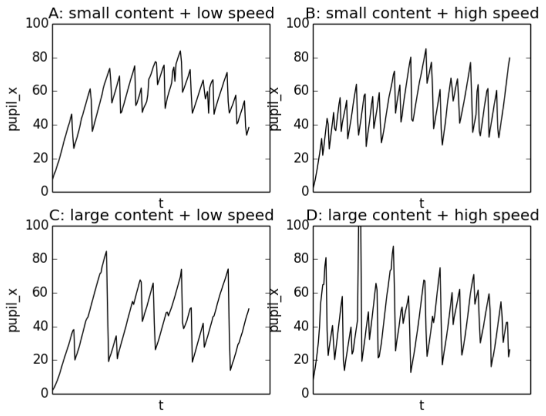


Figure 3. Optokinetik Nystagmus pattern sampled for 4 extreme conditions while viewing a set of scrolling images. W is defined as the size of the image divided by the size of the display and S is defined as the scrolling speed measured in degrees of visual field per second. The four images show the OKN pattern for conditions A) $\{W = 0.2, S = 19°/s\}$ **B)** $\{W = 0.2, S = 50°/s\}$ **C)** $\{W = 0.8, S = 19°/s\}$ **D)** $\{W = 0.8, S = 50°/s\}$

During each condition 40 pictures were displayed where 7 of them were target images. We recorded the eye movement data in the horizontal direction and the moment participants pressed the space key. In our study, the eye movement data is defined as pupil horizontal position in the eye image. The conditions were counterbalanced to avoid any learning effect. Also the position of the target images were counterbalanced in each condition.

Design

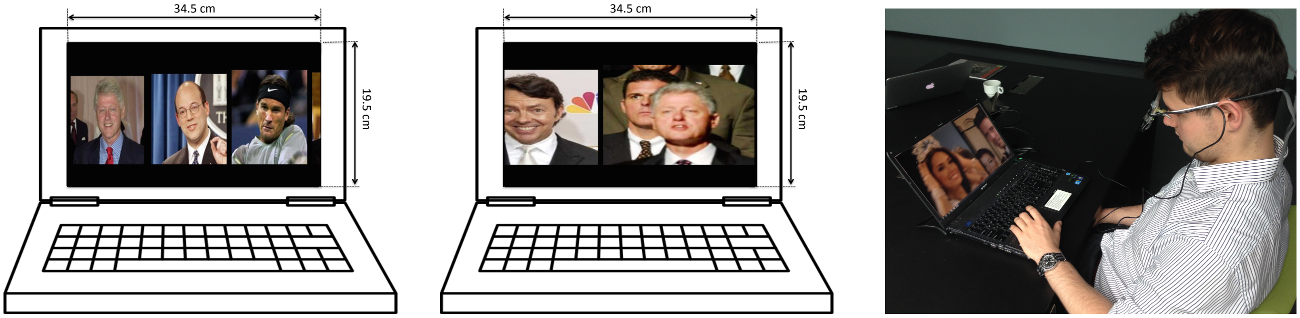
The experiment was an 3×2 within-subjects design with 15 participants, and each participant completed all conditions in one experimental session that lasted for approximately 10 minutes. In each condition, participants completed the task with three different speeds (1400, 2000, 2600 pixel/s which are respectively equal to 26.5, 37.5, and 49 degrees/s) and two different image widths (480, 960 pixels equal to 9.1 and 18.2 degrees). To change the image width we kept the height of the image fixed and just rescaled the image width. All combinations of speed and image width parameters generated 6 different conditions (see Table 1.)

Event detection algorithm

In order to recognize the moment and the location of the peak in the eye movement signal when a user performs a visual search task, we used machine learning algorithms in WEKA software [3]. A comparative analysis between different classifiers in WEKA showed that Multilayer perceptron algorithm

[2] http://eyeinfo.itu.dk

[3] http://www.cs.waikato.ac.nz/ml/weka/



Small width conditions: 1, 3,and 5
(a)

Big width condition: 2, 4, and 6
(b)

(c)

Figure 4. a) a screen-shot of the system in small width conditions: 1, 3, and 5, b) a screen-shot of the system in big width conditions: 2, 4, and 6, c) a participant wearing the home-made mobile eye tracker performing the task

is the most accurate and reliable classifier among other available classifiers in WEKA. We used the default setting for the Multilayer perceptron algorithm in the WEKA with a single hidden layer. The eye movement data is used as the only feature in our classification. We used a sliding window to detect the moment when something draws users' visual attention. Since the experiment included three different speeds and two different widths of moving images, the best window size needed to be found for each condition. In the following sections, we briefly explain the data preparation and classification steps.

Pre-processing data

Removing outliers: as mentioned in the participant section, 20 participants were recruited for the experiment. First of all, the eye movement data of each participant is reviewed to investigate whether the eye tracker detected the pupil of the user appropriately or not. If pupil of the participant is not detected more than 25% of times, we removed the data of the participant from the experiment. After analyzing data from 20 participants, 5 participants were removed from the experiment. For the remaining participants, the missing values of pupil coordination are calculated based on the linear regression method.

Data cleaning & normalization: Before starting the experiment and after performing the task, participants were asked to look at the center of two red circles on the left and right sides of the screen. Each circle was displayed for 3 seconds. These two targets were later used for determining the lower and upper bounds of the eye movement signal. In order to prepare the data for aggregated data sets for each condition, the lower and upper values were used to normalize the eye movement data for all participants using the min-max method. The process of removing noise and outliers from the eye signal was easier after data normalization. We also used these two target points and the corresponding pupil positions while looking at each target, for roughly estimating the gaze area in the screen and locating the image of interest in small size image conditions.

Data aggregation: To calculate the performance of the classifier for each condition, we aggregated the normalized data from all participants in 6 data sets.

Sliding window & classification

To detect the event when an image draws user's attention, we used a sliding window with 50% overlap between two neighbor windows. To find the best window size for each condition, we used 4 different window sizes (10, 16, 20, 30). These window sizes have been chosen to cover the minimum and maximum duration that takes for an image to appear on the screen and disappear from the screen. This time period depends on the speed of the moving images (ranged from 1400 to 2600 pixels/s), the image widths (ranged from 480 to 960 pixels), the screen width (1600 pixels), and the sampling rate (20 Hz). The time needed for appearing an image on the screen and disappearing from the screen can be calculated using this equation: $time = (screenwidth + imagewidth)/speed$. Using the above values for screen width, image width, and speed the maximum time duration can be calculated as $time_{max} = (1600 + 960)/1400 = 1.8 seconds$, and the minimum time is equal to $time_{min} = (1600 + 480)/2600 = 0.8 seconds$. Since the sampling rate is 20 Hz, the minimum window size is equal to $0.8 \times 20 = 16$ and the maximum window size is equal to $1.8 \times 20 = 36$.

The performance of the classification is calculated for each condition using the aggregated data sets. The data sets are labeled based on the moment of pressing space key by participants as the center of each window with "Event" label. The accuracy and precision of the classification is measured using 10 folds cross-validation method. Figure 5-a illustrates the performance of classification for each condition with 4 different window sizes. We tried to find the highest classification performance where precision of classification is high for both classes: 1) "Event" and 2) "No event". We finally selected window size 30 for the first, second, and fourth conditions, window size 20 for the third and sixth conditions, and window size 16 for the fifth condition.

Results

To analyze the effect of speed and maximum number of visible images in the view-port on the classifier, the performance of classification is calculated for each participant in different conditions using 10 folds cross-validation method. Figure 6 (a) shows the mean and standard deviation of the performance of the classifier in each condition. A repeated measure

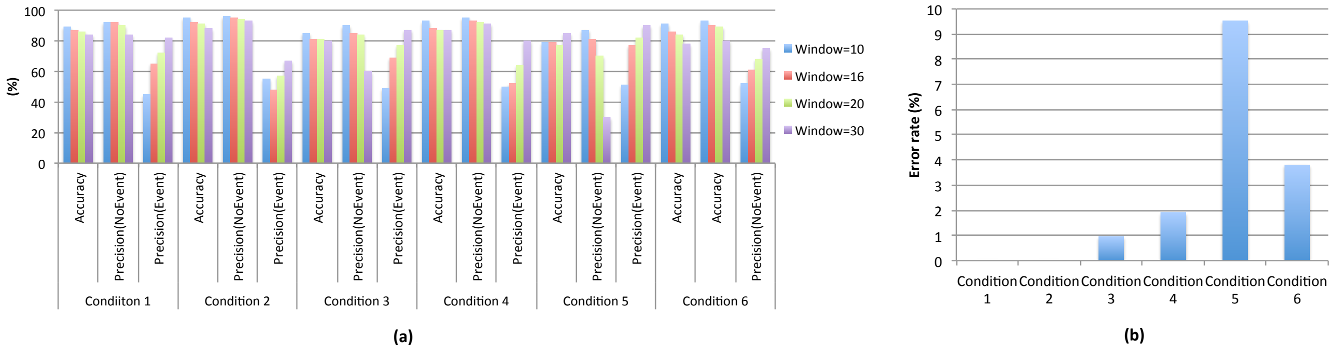


Figure 5. a) Accuracy and precision of the classification for each condition with 4 different window sizes, b) Error rate (percentage of missing targets)

ANOVA is used to investigate the differences in performance of the classifier. Post-hoc paired samples analysis with a Bonferroni correction is used for pairwise comparisons ($\alpha = .05$).

In order to measure the robustness of the classifier against unseen data, the performance of the classifier is also evaluated for the condition 4 taking a leave-one-out approach where the data of each participant was removed from the training data and used as test data. The leave-one-out evaluation for condition 4 reported an average accuracy of 87.2% (σ = 11.5) for the classification. The results of leave-one-out evaluations is illustrated in a box plot diagram (Figure 7).

The error rate (total missing target images by participants divided by total number of targets) is represented in Figure 5-b.

Effect of image width

The result of statistical analysis showed that the classification performance significantly varied with the image width: F(1, 14) = 34.9, $p < .0001$. The post-hoc pairwise comparisons revealed that the accuracy of the classifier is significantly higher when the image width is bigger. Figure 6 (c) illustrates changing the average accuracy of the classification when the image width changes.

Effect of speed

The statistical analysis indicated no significant effect of speed on the classification performance (see Figure 6 (b)). However, the medium speed shows a higher performance specially for small images. Moreover, participants missed more target images in the high speed conditions. Moreover, some of the participants mentioned after the experiment that it was difficult for them to complete them the task in high speed conditions specially in condition 5 where the speed was maximum and the image width was minimum.

Discussion

The results of the experiment indicates that our EyeGrip technique is more accurate for the lower number of visible images in the view-port where the image width is 960 pixels (equal to 60% of the screen width) moving with the medium speed (2000 pixel/s). As it is visible in Figure 3, increasing number of visible images in the view-port makes the sawtooth shapes in the OKN signal more homogeneous which decreases the accuracy of the classifier. Increasing the speed of moving images has a similar effect on the OKN signal (Figure 3). When images move faster on the screen, even the smooth pursuit component of the OKN eye movements have a sacadic characteristics. This makes harder for the classifier to detect slow phase of the OKN. Apart from the classification challenges, the high number of missing images in the fast conditions (see Figure 5-b) shows that following and processing fast-moving objects is harder for humans particularly when they need to see more complex images. On the other hand, there is also a lower limitation for the speed. Very low speeds let users follow all images one by one which means the shape of the smooth pursuit component of the OKN becomes more homogeneous and harder to detect for the classifier.

DESIGN GUIDELINES

We believe that the EyeGrip method is applicable to different application areas. To increase the usability of the EyeGrip technique and minimize the limitations of using the EyeGrip method, we propose the following guidelines for user interface designers.

Uni-directional moving objects

In the EyeGrip method, there is no limitation for the number of detectable objects. The important assumption is that objects need to move next to each other in the same direction at a certain speed. The objects might be placed dynamically in the queue but the system needs to know the position of each object within the sequence. In some applications, we may not be interested in detecting which content has grabbed the user's attention, and we only want to know which part of the sequence was visible in the display at the time the system has detected a long slow phase. In this case, the system can bring that part of the sequence back to the display even though there might be multiple contents visible in that moment.

Balance between moving speed & number of images

As we mentioned in the discussion section before, there are upper and lower limits for the speed of moving objects. If the objects move at lower speed the accuracy of the event detection classifier decreases. Also higher speeds increases the human error rate and risk of missing objects by user. Since the effect of increasing number of image in the view-port is similar to the increasing the speed, to maximize the accuracy of EyeGrip we need to find a balance between speed of the objects and the number of images in the view-port. Actually

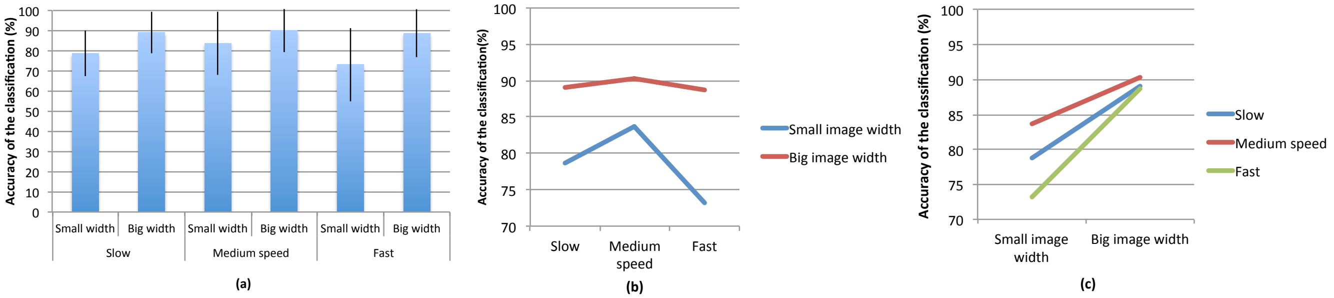


Figure 6. a) Performance of classification for each participants in different conditions, b) no significant effect of speed is observed, c) the effect of image width on the accuracy of the classifier is significant.

the EyeGrip technique works when there is a temporal tension in the visual search task. We need to be sure that we generate enough temporal tension by adjusting an appropriate speed and number of images in the view-port. At the same time, the speed should not exceed the upper limit to let users easily follow images on the screen.

Complexity of the visual search task

One of the limitations of using EyeGrip in user interface design is the fact that when a lot of images draw users' visual attention, the number of false positives increases. In other words, if users spend equal visual attention on each image in the line, the classifier cannot differentiate between target images and other objects. This limitation might be important for some applications where there is a need for high accurate recognition such as a visual inspection task. In such occasions, we can implement a two-stage algorithm where the system filters out some irrelevant objects at the first stage, and in the next step the user reviews the remaining objects to control the false positive detections.

POTENTIAL APPLICATIONS

Most of the existing gaze-based interfaces use gaze location as input. Which means for a graceful interaction, they need a very accurate gaze tracker with a cumbersome calibration procedure. In contrast, EyeGrip uses just one dimension eye movement which is much easier to achieve specially in mobile and wearable settings. This opens up a wide range of application areas that can use EyeGrip. In the following sections, we explain some of the applications that can use the EyeGrip technique for interaction.

Mind reading game

The EyeGrip technique helps the system know what attracts users' attention. This can be used in a mind reading game where the user is asked to select a person among some faces displayed on the screen. Then the user is asked to count the number of repetitions in displaying the face of that particular person among other faces while all images move horizontally in one direction with a fixed speed. The main purpose of asking users to count the number of repetitions is to draw their visual attention to a particular object. At the end the system predicts the identity of the selected person. Since EyeGrip does not need calibration and an accurate gaze tracker, the mind reading game can be installed on public displays to entertain passers-by in public places such as train stations, airports, or waiting halls.

Picture explorer on head-mounted display

One of the main challenges of interaction with eyewear computers such as Google Glass is providing input to the device. There are many situations where the hands of the user are busy with real-world task and providing a hands-free input channel can be a big advantage. The EyeGrip technique helps users with a fast and hands-free method for browsing graphical contents in eyewear computers. If we assume that in mobile scenarios, interaction with an eyewear computer should not take so much time [2], the EyeGrip method seems to be a promising technique for fast scanning visual contents on the HMD. For instance, users of the social network applications such as Facebook will be able to scan many graphical contents in a short time without any explicit input to the eyewear computer. In this case, to start scrolling the user can perform a head gesture to the left or right side or use voice commands. The graphical or textual content will start scrolling in one direction at a fixed speed. As soon as something draws the user's visual attention, the system stops scrolling and lets the user to look at that particular image or Facebook post. The user can continue scanning other contents by performing head gestures or voice commands. The beauty of using EyeGrip technique for implicitly finding users' interests is the fact that it does not need to work 100% accurate since users will always have an explicit way such as voice commands, hitting the touch-pad in Google Glass, or performing head gestures to stop scrolling. In this application, if the EyeGrip method detects object of interest in 80% of cases, it means EyeGrip has reduced the need for providing an explicit command in 80% of the times which can be a big success.

Public displays

Public displays have long been used for advertisement purposes. However, they have always been in a one-direction communication with passers-by. The EyeGrip method can help the public displays get feedback from users. One example could be to show a series of mono-directional moving images of different products on a public display where the scrolling stops whenever an image attracts attention of a user who is standing in front of the display and his/her eye is being tracked by a stationary eye tracker. EyeGrip method can

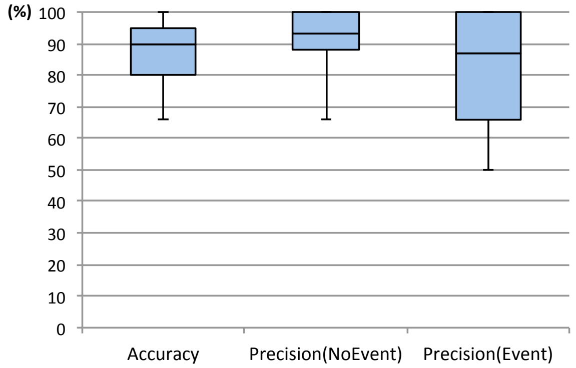


Figure 7. The performance of the classifier for condition 4 taking a leave-one-out approach.

for example be implemented using the Pupil-canthi-ratio approach [30] which is an interesting calibration-free approach for interaction with public displays. Because the relative movement between the user and the display may change the range of the horizontal movements of the eye, such a system requires the users to only move their eyes and to keep their head direction towards the center of the display. This challenge can be solved by placing a stationary infrared light source and using pupil-corneal reflection method. It is also possible that within a few seconds of recording the eye movement data while the user is looking at the moving (scrolling) contents on the display, the system figures out the lower and upper range of the eye movement signal. This can be an implicit way of calibrating the gaze direction and makes it possible to detect the images attracted users' visual attention after the scrolling has stopped.

Text reading assistant for small displays

Reading large amount of texts on small displays such as mobile devices, smart glasses, or smart watches is still challenging. One of the common approaches to facilitate reading in small displays is to enlarge the text and move it based on reader's eye movement [22]. The EyeGrip technique can be applied to such applications in order to give feedback to the system about the words which are harder to read or understand for the reader. When a user follows a word for a longer time the system can slow down the speed of moving text on the screen and provide some help, e.g. synonyms, to the user to better understand the challenging part of the text.

Assistant for visual inspection in production lines

Visual inspection is still part of quality control process in many production lines. In many cases, one or more workers control the appearance properties of the products while product move on a conveyor belt with a fixed speed. In a visual inspection task, quality controllers detect the potentially unqualified products and manually separate them from the others. If we use a gaze tracker to capture the eye movements of the quality controllers, the EyGrip technique can automatize the detection of unqualified products. If the system detects the target objects, a robot or other machines can separate them automatically. EyeGrip can potentially increases the speed of inspection by removing the manual part of the task.

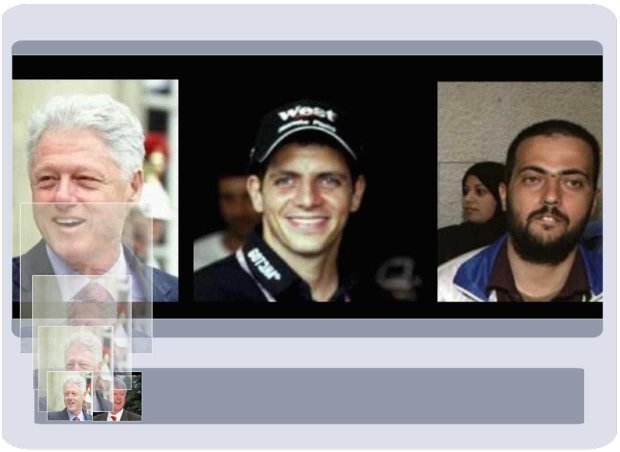


Figure 8. A screen-shot of the picture selection system (study1).

USABILITY STUDIES

To evaluate the usability of the EyeGrip method from users' point of view, we conducted two user studies: 1) a picture selection system and 2) a mind reading game. The picture selection system utilizes EyeGrip in a live interaction scenario, while the mind reading game uses EyeGrip as a context recognition method to detect what draws users' visual attention. In the following sections, we report the results of the usability evaluation in each study.

Study1: A picture selection system

We designed a desktop application to select a predefined set of images among scrolling pictures on the screen. A screenshot of the system user interface is illustrated in Figure 8. The system starts scrolling by pressing the space bar on the keyboard. To use the picture selection system in a mobile scenario, the start mechanism can change to head gestures, voice commands, etc.

Participants

8 participants (mean age = 25, ranged from 20 to 37 years old, and 1 female) were recruited among local university students to try the system. All of the participants had perfect visual acuity.

Procedure

The session started with a short introduction to the purpose of the experiment and the instruction of using the system. After preparing participants for the experiment, they were asked to wear the eye tracker apparatus and perform the task. To maximize the accuracy of the EyeGrip, we adjusted the scrolling speed equal to 2000 pixel/s and the number of visible images in the view-port (image width = 960 pixels) based on the results of our experiment. The task was similar to our experiment. The participants were asked to look at a series of moving images in the upper rectangle (see Figure 8) and count the number of Bill Clinton's pictures. Whenever the participant pays extra attention to an image, the image is selected and moved to the thumbnail panel at the bottom of the page. To give users a visual feedback about the selection mechanism, the moving procedure is animated in the user interface.

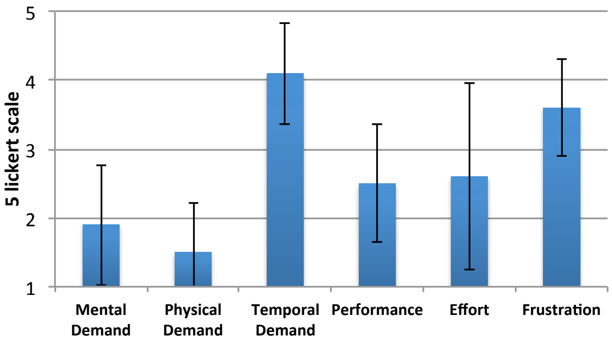


Figure 9. The result of usability questionnaire (NASA-TLX) for the picture selection system (study 1).

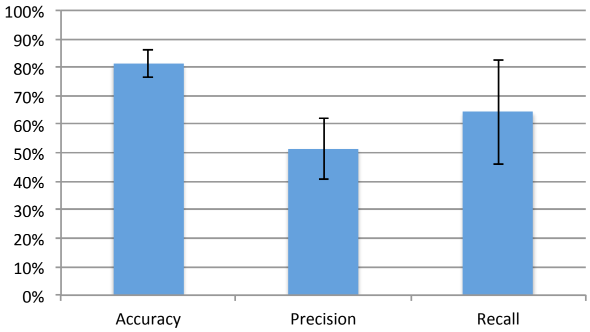


Figure 10. Performance of the EyeGrip classifier in picture selection application (study 1).

Evaluation results

To calculate the accuracy of the system, we recorded the number of correct selections, missed pictures, and wrong selections. The average accuracy, precision, and recall of the classification for all 8 participant is illustrated in Figure 10. After performing the task, the participants were asked to complete a usability questionnaire designed based on NASA-TLX [10] to reflect their experience. The result of the questionnaire is illustrated in Figure 9. The participants' general impression was also asked in an open question. They found the EyeGrip interaction technique *different* and *fun*. However, some of the participants found the EyeGrip method a bit confusing since they do not exactly know how the system selects images. Moreover, animating the image selection procedure was distracting for some users.

As it is illustrated in Figure 10, the performance of EyeGrip in the picture selection task (mean = 81%, σ = 5) is relatively close to the performance of EyeGrip in our controlled experiment (87% in Condition 4). This shows the robustness of the classifier to detect the object of interest even for the unseen data which means EyeGrip can be trained only once.

The result of the NASA-TLX questionnaire, indicates that using EyeGrip for picture selection puts time pressure on the users. This might be the reason why they felt a relatively high amount of frustration while performing the task and the accuracy of EyeGrip was slightly lower than what we observed in the experiment. Nevertheless, the task has not been physically and mentally demanding for users because the picture selection happens automatically based on their natural OKN eye movements without providing any explicit input.

Study2: A mind reading game

We also developed a mind reading game based on the software and hardware platform that we used as apparatus in the experiment. We adjusted the speed and the maximum visible number of images in the view-port similar to the condition 4 in the experiment and the picture selection application in the first usability study.

Participants

10 participants (mean age = 29, ranged from 21 to 44 years old, no female) among local university students and staff participated in the study. All of the participants had perfect visual acuity or wearing contact lens.

Procedure

We asked the participants to select a person among 4 faces printed on an A4 paper without telling us who has been chosen. Next we asked participants to wear the eye tracker hardware and sit in front of the laptop screen. They were asked to count the number of images of the selected person among other moving images on the screen. After finishing the task, the name of the selected person is displayed to the participants. All of the target images are repeated 4 times in the queue among 50 images of other people.

Evaluation results

The mind reading game was 100% accurate, and users got exited when they saw the result. Some of the participants even asked to repeat the game. Since the mind reading game has the chance to guess the selected person in 4 different occasions the probability of guessing the right person increases significantly.

Top-down & bottom up attention mechanisms

The two above-mentioned applications for EyeGrip utilize a top-down attention mechanisms in the brain. In both applications, the user knows what s/he is looking for; therefore, the visual attention is directed based on the user's longer-term cognitive strategies which is more like a top-down mechanism [7]. The EyeGrip technique can also be useful in applications where the user does not have any predefined plan for the visual search such as the Facebook Newsfeed reader explained earlier in the paper. In such applications the user's attention can be directed based on raw sensory input such as an attractive colour or fast movements (bottom-up mechanism).

DISCUSSION & CONCLUSIONS

In this paper, we introduced EyeGrip which is a novel interaction technique to support users in a visual search task in desktop, mobile, and wearable settings. EyeGrip analyzes Optokinetic Nystagmus eye movements to detect the object or area of interest among a sequence of uni-directional moving objects. This information enables users to potentially select an object without providing any explicit input to the computer.

Since OKN is a natural reaction of the eyes to the moving visual field, EyeGrip opens room for designing more intuitive methods of eye-based interaction.

We also tried to characterise the EyeGrip technique by empirically investigating the effect of scrolling speed and maximum number of visible images in the view-port (manipulated by changing image width) on the accuracy of the system and users' performance. The results of our experiment indicated a significant effect from number of image in the view-port on the performance of the classification while the effect of speed on the classification accuracy was not statistically significant. However, increasing the speed of moving images indicated a significant effect on the users' performance. But there is also a lower limit for the speed of moving objects. If the objects move very slow the user has enough time to pay equal visual attention to all of the objects. This makes the sawtooth shapes of the OKN signal more homogeneous which means it will be difficult to detect a deviation in the OKN signal when something draws user's attention.

EyeGrip utilizes the limitation of humans visual perception system in temporally intensive visual tasks where the user's visual perception mechanism needs to prioritize the time spent on following visual cues. To use the EyeGrip technique in user interface design we need to find an optimum speed and number of images in view-port to create a temporal intention, but we need to keep the speed low enough in order to minimize users' error. The temporal intention might seem to be a limitation for EyeGrip, but considering the increasing pace of producing visual contents in the Internet, we will need such mechanisms in the future to support users in quickly scanning a lot of visual contents.

In this paper, we used a home-made eye tracker, a very simple eye movement feature and classification algorithm to demonstrate the concept of EyeGrip. Using this setting we reached the accuracy of 87% where the scrolling speed is equal to 2000 pixels/s and the maximum number of visible images in the view-port is 3 (image width = 960 pixels). We believe by using more advanced features and classification models the accuracy of EyeGrip can be improved even more than what we reached in this study.

The results of the usability studies and the leave-one-out evaluation indicated an acceptable level of classification performance for the unseen data. The leave-one-out evaluation for condition 4 reported an average accuracy of 87.2% (σ = 11.5) for the classification. Furthermore, in the picture selection study, as a real-time interactive application, the average accuracy was 81% (σ = 5) and in the mind reading game the EyeGrip technique was 100% accurate. This shows that the EyeGrip technique can be used pretty accurate without any additional training for new users.

In the future work, we will implement the EyeGrip method by capturing the eye movement data from stationary eye trackers and other sensing technologies such as EOG for wearable systems. In that case, the OKN signal will be generated based on only eye movement data, and other peak detection algorithms can be used for finding local peaks in the OKN signal.

ACKNOWLEDGEMENTS

We would like to thank our fellow PIT Lab members for their valuable input at various stages of the work.

REFERENCES

1. W. Abd-Almageed, M.S. Fadali, and G. Bebis. 2002. A non-intrusive Kalman filter-based tracker for pursuit eye movement. In *American Control Conference, 2002. Proceedings of the 2002*, Vol. 2. 1443–1447 vol.2. DOI: http://dx.doi.org/10.1109/ACC.2002.1023224
2. Daniel L. Ashbrook. 2010. *Enabling Mobile Microinteractions*. Ph.D. Dissertation. Atlanta, GA, USA. Advisor(s) Starner, Thad E. AAI3414437.
3. Richard Bates and Howell O Istance. 2003. Why are eye mice unpopular? A detailed comparison of head and eye controlled assistive technology pointing devices. *UAIS* 2, 3 (2003), 280–290.
4. L Burmark. 2004. Visual Presentations That Prompt, Flash & Transform Here are some great ways to have more visually interesting class sessions. *Media and methods* 40 (2004), 4–5.
5. T. Cecchin, D. Sauter, D. Brie, and B. Martin. 1990. On-line Separation Of Smooth Pursuit And Saccadic Eye Movements. In *Engineering in Medicine and Biology Society, 1990., Proceedings of the Twelfth Annual International Conference of the IEEE*. 777–778. DOI: http://dx.doi.org/10.1109/IEMBS.1990.691327
6. M. Cheng and J.S. Outerbridge. 1975. Optokinetic nystagmus during selective retinal stimulation. *Experimental Brain Research* 23, 2 (1975), 129–139. DOI: http://dx.doi.org/10.1007/BF00235455
7. Charles E Connor, Howard E Egeth, and Steven Yantis. 2004. Visual attention: bottom-up versus top-down. *Current Biology* 14, 19 (2004), R850–R852. DOI: http://dx.doi.org/doi:10.1016/j.cub.2004.09.041
8. Heiko Drewes and Albrecht Schmidt. 2007. Interacting with the Computer Using Gaze Gestures. In *Human-Computer Interaction INTERACT 2007*. Lecture Notes in Computer Science, Vol. 4663. Springer Berlin Heidelberg, 475–488. DOI: http://dx.doi.org/10.1007/978-3-540-74800-7_43
9. Johannes Harms, Martina Kratky, Christoph Wimmer, Karin Kappel, and Thomas Grechenig. 2015. Navigation in Long Forms on Smartphones: Scrolling Worse than Tabs, Menus, and Collapsible Fieldsets. In *INTERACT 2015*. Lecture Notes in Computer Science, Vol. 9298. Springer, 333–340. DOI: http://dx.doi.org/10.1007/978-3-319-22698-9_21
10. Sandra G Hart and Lowell E Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. *Advances in psychology* 52 (1988), 139–183. DOI: http://dx.doi.org/doi:10.1016/S0166-4115(08)62386-9

11. Aulikki Hyrskykari, Päivi Majaranta, and Kari-Jouko Räihä. 2005. From gaze control to attentive interfaces. In *Proceedings of HCII*, Vol. 2.

12. Robert J. K. Jacob. 1990. 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)*. ACM, New York, NY, USA, 11–18. DOI: http://dx.doi.org/10.1145/97243.97246

13. Shahram Jalaliniya, Diako Mardanbegi, and Thomas Pederson. 2015. MAGIC Pointing for Eyewear Computers. In *Proceedings of the 2015 ACM International Symposium on Wearable Computers (ISWC '15)*. ACM, New York, NY, USA, 155–158. DOI:http://dx.doi.org/10.1145/2802083.2802094

14. Shahram Jalaliniya, Diako Mardanbegi, Thomas Pederson, and Dan Witzner. 2014. Head and Eye Movement as Pointing Modalities for Eyewear Computers. In *Wearable and Implantable Body Sensor Networks Workshops (BSN Workshops), 2014 11th International Conference on*. 50–53. DOI: http://dx.doi.org/10.1109/BSN.Workshops.2014.14

15. Do Hyong Koh, Sandeep Munikrishne Gowda, and Oleg V. Komogortsev. 2010. Real Time Eye Movement Identification Protocol. In *CHI '10 Extended Abstracts on Human Factors in Computing Systems (CHI EA '10)*. ACM, New York, NY, USA, 3499–3504. DOI: http://dx.doi.org/10.1145/1753846.1754008

16. Lori A Lott and Robert B Post. 1993. Up-down asymmetry in vertical induced motion. *PERCEPTION-LONDON-* 22 (1993), 527–527.

17. PaulP. Maglio, Teenie Matlock, ChristopherS. Campbell, Shumin Zhai, and BartonA. Smith. 2000. Gaze and Speech in Attentive User Interfaces. In *Advances in Multimodal Interfaces ICMI 2000*. Lecture Notes in Computer Science, Vol. 1948. Springer Berlin Heidelberg, 1–7. DOI: http://dx.doi.org/10.1007/3-540-40063-X_1

18. Diako Mardanbegi, Dan Witzner Hansen, and Thomas Pederson. 2012. Eye-based Head Gestures. In *Proceedings of the Symposium on Eye Tracking Research and Applications (ETRA '12)*. ACM, New York, NY, USA, 139–146. DOI: http://dx.doi.org/10.1145/2168556.2168578

19. Thomas Paul, Daniel Puscher, and Thorsten Strufe. 2015. The User Behavior in Facebook and its Development from 2009 until 2014. *arXiv preprint arXiv:1505.04943* (2015).

20. Ravikrishna Ruddarraju, Antonio Haro, Kris Nagel, Quan T. Tran, Irfan A. Essa, Gregory Abowd, and Elizabeth D. Mynatt. 2003. Perceptual User Interfaces Using Vision-based Eye Tracking. In *Proceedings of ICMI '03*. ACM, New York, NY, USA, 227–233. DOI: http://dx.doi.org/10.1145/958432.958475

21. Javier San Agustin. 2009. *Off-the-shelf gaze interaction*. Ph.D. Dissertation. IT-Universitetet i KøbenhavnIT University of Copenhagen, DirektionenManagement, InstituttetThe Department, Innovative CommunicationInnovative Communication.

22. Christopher A Sanchez and James Z Goolsbee. 2010. Character size and reading to remember from small displays. *Computers & Education* 55, 3 (2010), 1056–1062. DOI:http://dx.doi.org/doi:10.1016/j.compedu.2010.05.001

23. Albrecht Schmidt. 2005. Interactive context-aware systems interacting with ambient intelligence. *Ambient intelligence* 159 (2005).

24. Jeffrey S Shell, Roel Vertegaal, and Alexander W Skaburskis. 2003. EyePliances: attention-seeking devices that respond to visual attention. In *CHI'03 extended abstracts on Human factors in computing systems*. ACM, 770–771.

25. Linda E. Sibert and Robert J. K. Jacob. 2000. Evaluation of Eye Gaze Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '00)*. ACM, New York, NY, USA, 281–288. DOI: http://dx.doi.org/10.1145/332040.332445

26. JWG Ter Braak. 1936. Untersuchungen über optokinetischen Nystagmus. *Arch Neerl Physiol* 21 (1936), 309–376.

27. Mélodie Vidal, Andreas Bulling, and Hans Gellersen. 2012. Detection of Smooth Pursuits Using Eye Movement Shape Features. In *Proceedings of the Symposium on Eye Tracking Research and Applications (ETRA '12)*. ACM, New York, NY, USA, 177–180. DOI:http://dx.doi.org/10.1145/2168556.2168586

28. Mélodie Vidal, Andreas Bulling, and Hans Gellersen. 2013. Pursuits: Spontaneous Interaction with Displays Based on Smooth Pursuit Eye Movement and Moving Targets. In *Proceedings of UbiComp '13*. ACM, New York, NY, USA, 439–448. DOI: http://dx.doi.org/10.1145/2493432.2493477

29. Shumin Zhai, Carlos Morimoto, and Steven Ihde. 1999. Manual and Gaze Input Cascaded (MAGIC) Pointing. In *Proc. of the CHI '99*. ACM, 246–253. DOI: http://dx.doi.org/10.1145/302979.303053

30. Yanxia Zhang, Andreas Bulling, and Hans Gellersen. 2014. Pupil-canthi-ratio: A Calibration-free Method for Tracking Horizontal Gaze Direction. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces (AVI '14)*. ACM, New York, NY, USA, 129–132. DOI: http://dx.doi.org/10.1145/2598153.2598186