Pursuits: Spontaneous Interaction with Displays based on Smooth Pursuit Eye Movement and Moving Targets

Mélodie Vidal
Lancaster University
m.vidal@lancaster.ac.uk

Andreas Bulling
Max Planck Institute for Informatics
andreas.bulling@acm.org

Hans Gellersen
Lancaster University
hwg@comp.lancs.ac.uk

ABSTRACT
Although gaze is an attractive modality for pervasive interactions, the real-world implementation of eye-based interfaces poses significant challenges, such as calibration. We present *Pursuits*, an innovative interaction technique that enables truly spontaneous interaction with eye-based interfaces. A user can simply walk up to the screen and readily interact with moving targets. Instead of being based on gaze location, Pursuits correlates eye pursuit movements with objects dynamically moving on the interface. We evaluate the influence of target speed, number and trajectory and develop guidelines for designing Pursuits-based interfaces. We then describe six realistic usage scenarios and implement three of them to evaluate the method in a usability study and a field study. Our results show that Pursuits is a versatile and robust technique and that users can interact with Pursuits-based interfaces without prior knowledge or preparation phase.

Author Keywords
Spontaneous interaction; Eye-based interfaces; Eye movement; Correlation; Smooth pursuits

ACM Classification Keywords
H.5.2 Information Interfaces and Presentation: User Interfaces — *Input devices and strategies*.

INTRODUCTION
Gaze holds great promise as an input modality because it indicates what our attention is directed at and is faster than a mouse as a pointer [24]. Gaze-based interaction is well established under controlled conditions and for person-dependent use. Applications range from desktop control [12, 29], eye typing [9] and target selection [25] to password entry [2] and videogame control [10]. But gaze also make a very attractive input modality for ubiquitous applications. It is a particularly promising modality to interact with the increasing number of out-of-reach displays as our eyes naturally point at what we are interested in and indicates our spontaneous attention [27]. Gaze interaction also offers an alternative to touch input in environments where hygiene is critical, such as during surgeries [13].

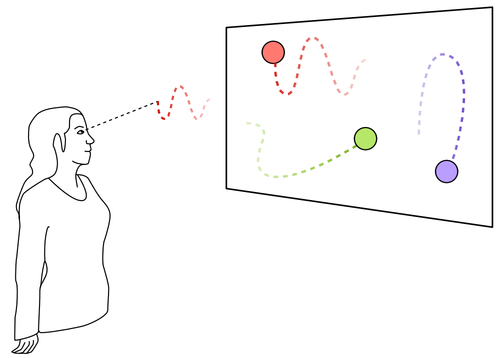


Figure 1: Pursuits matches the user's eye movement with the movement of on-screen objects.

Although gaze is an attractive modality for pervasive interactions, the real-world implementation of eye-based interfaces poses significant challenges. Current interfaces typically require an additional modality to confirm input commands, such as touch [25], a calibration phase from the user [2] or previous knowledge on how to operate the system [4]. This makes spontaneous eye-based interaction in everyday environments particularly challenging. Previous efforts focused on detecting whether a user is attending a device [23] or on using gaze to select areas on a large screen [30].

In this work we present *Pursuits*, a method that enables spontaneous eye-based interaction with pervasive screens that display moving targets. Our method enables users to walk up to a display and immediately interact with it in an engaging way. The key idea of Pursuits is that the eyes perform the same trajectory as the object they are following (see Figure 1). Thus, by correlating a moving object's trajectory with the eyes' trajectory in real time, it is possible to detect which object is being looked at. Because it is based on eye trajectory rather than direct point-of-gaze, it does not matter how that data is translated on the screen or in space: the trajectory of the eyes still resembles that of their moving target. Thus, this approach does not require any calibration phase from the user.

Pursuits is named after the movement our eyes perform when they latch onto a moving object, called smooth pursuits. Although the dynamics of smooth pursuits are well understood [21] they have not yet been leveraged for human-computer interaction. In Pursuits, smooth pursuit movements are embraced for the selection of objects for interaction, contrast-

UbiComp'13, September 8–12, 2013, Zurich, Switzerland.

http://dx.doi.org/10.1145/2493432.2493477

ing conventional gaze-based interaction where objects are selected by fixation using dwell-time. Dwell-time requires the user to keep their gaze on a fixed location, which is highly unnatural for our eyes since we usually perform three to four fixations per second [20]. We naturally dwell on an object that moves, which makes Pursuits a much more natural interaction technique.

This natural behaviour, combined with the fact that Pursuits does not need calibration, means any user can simply walk up to the display and readily interact without needing instructions. The use of correlation also implies that Pursuits can be used with any eye-tracker or any size of screen. This is because the correlation formula inherently normalises the data: the gaze coordinates do not need to be in the same range as the objects coordinates. Finally, because the core of the interaction is based on moving targets, Pursuits opens up new possibilities for interface design. While HCI research has attended to target movement as a problem [17], Pursuits embraces it and allows for dynamic interfaces which naturally attract the eyes.

This works makes the following contributions. We first introduce the concept of correlating pursuit eye movements with the trajectory of moving objects on the screen. We then evaluate the influence of several target parameters on the performance of Pursuits and deduce guidelines for future interface design. Finally, we identify six example real-life applications of Pursuits-based interfaces and evaluate three of them in a user study and an uncontrolled field study.

RELATED WORK

Gaze-based interfaces traditionally aim to use the eyes as a direct replacement for the mouse for tasks such as pointing or object selection [6, 12]. This is usually achieved by using dwell-time selection, which means that a selection is triggered once the eyes have been looking at a target for a certain amount of time [12, 24, 16]. These works exploit eye fixations, which means the eye needs to stay still on the target for a longer amount of time than natural [12, 20]. In contrast, the eyes naturally dwell on moving objects. As a result, Pursuits uses dwell time but does not require the users' eyes to perform any unusual tasks. This approach has also been used by two eye-based text entry systems: Dasher [26] and StarGazer [9].

These works, however, all require calibration, in order to map the user's gaze to the screen coordinates. This means users generally need to keep their head still and look at several calibration points on a screen, which is time-consuming and prevents spontaneous interaction. Eye gestures have been proposed as a way to overcome this problem. Eye gestures are sequences of pre-learned eye movements that trigger a predefined reaction from the system [4], for instance to control a video game [10]. Our work is similar to eye gestures in that it is based on raw eye movement rather than direct gaze coordinates. However, Pursuits does not require prior knowledge from the user, as it exploits the natural occurence of pursuits when a moving object is being followed. This means that Pursuits is destined for spontaneous interactions, for example in public spaces.

Interaction with pervasive screens altogether is a challenge, and is commonly based on using intermediary devices such as mobile phones [1, 3] or on the necessity that the screen is within reach, through the use of multi-touch [19, 28]. More specifically, the use of eye-based interfaces for realistic ubiquitous interactions has not been explored much. A reason is that passer-bys do not have time to learn how to use a system or to calibrate, which makes eye-based interaction particularly challenging. So far, the use of eye gaze has been limited to the awareness that a user is attending a device [23] or to the selection of large areas on a screen [30]. Pursuits offers a new range of possibilities for pervasive screen interaction in public spaces as it does not require any additional device from the user and can be used even out of reach. However, it requires a dynamic interface.

Many applications, such as games, involve moving targets [18]. Selecting moving target has previously been addressed as a problem [17]. Gunn *et al.* pointed out the difficulty of moving target selection with a mouse and developed different ways to cope with it [8]. In daily life, the eyes frequently lock on moving objects, such as a person or car passing by. They behave as a closed-loop system in that adjusts over time depending on the perceived velocity of the object being followed and are thus not a ballistic movement like a saccade [14, 21]. This means our eyes are fast and extremely good at judging speed of moving stimuli [7]. Pursuits is therefore not only natural for interaction with dynamic interfaces and content that moves, but also highly effective for fast and accurate selection of moving targets.

Although the eyes are not meant for controlling an interface, they still indicate a lot of the context a person is in and where their attention is focussed, which can be used to create seamless interfaces. Because the eye is a perceptual organ, attentive user interfaces (AUIs) aim at using eye gaze in a subtle way, e.g. to adapt interface behaviour to the user [11, 27]. For example, Zhai *et al.* have explored the context information held by gaze to help users performing tasks quicker without using the eye as a direct means of control [29]. Others have aimed at using eye gaze to infer about user behaviour [22], infer context in ambiguous situations [15], and augment devices with visual attention sensors to act depending on where the user's attention is directed [23]. While our work is not an AUI by definition, it shares conceptual similarities with AUIs in that it is aimed at providing a non-disruptive, natural interaction for the user.

THE PURSUITS METHOD

The key idea of Pursuits is to match in real time the movements of the eyes with the trajectories of objects moving on the screen. The matching is performed using Pearson's product-moment correlation method. The input to this method are the synchronised time series of horizontal and vertical gaze coordinates and the coordinates of all objects on the screen. For each object on the screen, the method evaluates the similarity between the corresponding time series and that of the eyes'. The method outputs the single object whose trajectory is most similar to that of the eyes or nothing if no object is detected as being followed. In the following we explain the calculation for horizontal dimension x only for readability purposes, but it is computed the same way for vertical dimension y.

The similarity between the gaze and object trajectories is calculated using the Pearson's product-moment correlation coefficient $corr_x$ of the horizontal coordinates of an object Obj_x and of the eyes Eye_x, defined as

$$corr_x = \frac{E[(Eye_x - \bar{Eye_x})(Obj_x - \bar{Obj_x})]}{\sigma_{Eye_x}\sigma_{Obj_x}}$$

where $\bar{Eye_x}$ and σ_{Eye_x} are the mean and the standard deviation of the horizontal gaze position. The coefficient is calculated for both horizontal and vertical positions. The closer the coefficient is to 1, the more correlated the two time series are and thus the more similar the eyes' trajectory is to the object's trajectory. For real-time interaction we use this measure in the following way: For every new data point we calculate $corr_x$ and $corr_y$ for all eye-object couples $(object(i), eye), i = 1..n$, with n the number of objects, on a window covering the last w ms of data. If both the $corr_x$ and $corr_y$ of the object with the highest $corr_x$ and $corr_y$ are above a threshold th_{corr} then the object is detected as being followed with the eyes; otherwise no object is selected. This means Pursuits has two parameters: the size of the time window w over which the correlation coefficients are calculated and the threshold of the correlation coefficient th_{corr}.

Implications for design

The use of trajectory correlation in Pursuits has a number of implications and properties for interface design.

Independence of target size. Because Pursuits matches trajectories, the size of the target does not matter, only its movement does. Pursuits allows the selection of very small targets that would be difficult to select accurately by using gaze position or any other position-based selection technique.

Use of dynamic interfaces. Using trajectories also means that the technique requires interfaces that contain moving targets. The correlation formula relies on standard deviations of both the eyes and the objects: if the latter are static, the correlation coefficients could not be computed. Pursuits is thus only applicable to dynamic interfaces.

Difference of trajectories. To accurately distinguish between objects, Pursuits requires the correlation coefficients to be significantly different for different object trajectories. The difference between two linear trajectories can be characterised by the angular difference of their vector and is thus tightly linked to the maximum number of objects on the screen.

Alignment with line-of-sight. Finally, the use of correlation requires that the eye data follows the same evolution as the screen graphics, i.e. if the object's horizontal coordinate value gets bigger when moving to the right of the screen, so should the eye's horizontal coordinate value. The use of correlation also has underlying implications about the setup: the screen should be positioned orthogonal to the line of sight.

ALGORITHM EVALUATION

The accuracy of detecting whether a specific object is being followed with the eyes depends on the interface design as well as the method and parameters of the correlation. We conducted a controlled laboratory study to analyse the influence of the number, speed and trajectory of objects and the correlation parameters on the detection performance. We aimed to characterise the key parameters of Pursuits and to provide insights that may guide the design and implementation of Pursuits-enabled interfaces.

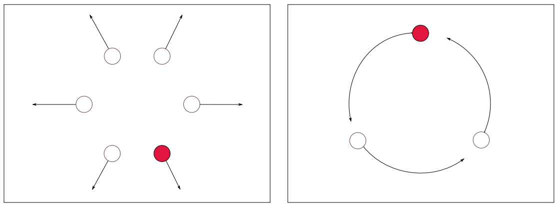
Experimental procedure

Participants performed a series of interactions with different interfaces (each called a trial). To collect synchronised eye and object trajectory data in a controlled manner, in each trial participants were instructed to look at a single red object moving across the screen. Additional objects were present and moved at the same time, so that we could study the effects of the number of objects on the method, but they had the same colour as the background of the interface so that participants could not see them (see Figure 2a, additional objects only shown here for illustration purposes). This was to ensure that participants followed a single object throughout without being distracted by other objects moving on the screen. Each trial was repeated three times. In each trial either the number, speed and trajectory of all objects varied, always one at a time while the other two stayed constant (see Table 1). These variables varied as follows:

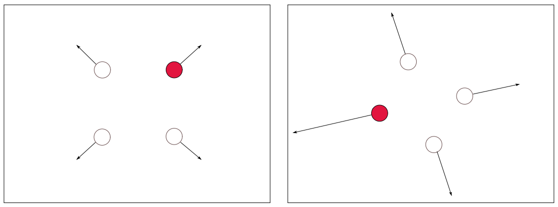
- Number of objects: We always maximised the distance between two objects, so that when four objects moved simultaneously the vectors they were moving along had a $\pi/2$ difference, when there were eight objects, the vectors had a $\pi/4$ difference, etc (see Figure 2a). In other words, the more objects on the screen, the more similar their trajectories were, which means we expected the correct detection rate to drop as the number of objects increases.
- Trajectory of objects: We assumed that the objects' trajectories in future applications will be a combination of a linear and a circular movement component. We thus ran the test once for objects moving in a linear fashion, starting from the centre of the screen towards its borders, and once for objects moving along a circle.
- Speed of objects: We tested both when objects move at the same speed, and when speed between objects differs (see Figure 2b). We expected this last case to potentially increase the detection rate in cases where the number of objects would otherwise have a negative influence on the detection performance. When there were only two objects going at a different speed we had them move in parallel along the same line.

The specific parameter values that we evaluated are summarised in Table 1, where each line represents a different set of trials. For example, the first line describes a series where the number of objects varies, but all objects move at the same speed and in a linear fashion.

We further evaluated Pursuits for different values of its two parameters – window size w and correlation threshold th_{corr}



(a) Left: Example for $nbObj = 6$ and linear trajectory. Right: Example for $nbObj = 3$ and circular trajectory.



(b) Left: Example for speed=200px/s for all objects. Right: Example for $speed_{red}$=650px/s and $speed_{others}$=450px/s.

Figure 2: Example stimuli consisting of a red dot moving on the screen. Only the red object is visible to the participant, additional dots are shown here only for illustration purposes.

Number of objects	Speed of objects (px/s)		Trajectory
$\{2, 3, 4, 6, 8, 10, 15, 20\}$	450		linear
$\{2, 3, 4, 6, 8, 10, 15, 20\}$	450		circular
4	$\{100, 200, 450, 650, 850\}$		linear
	red object	other objects	
$\{2, 8, 10, 15, 20\}$	200	450	linear
$\{2, 8, 10, 15, 20\}$	650	450	linear
$\{2, 8, 10, 15, 20\}$	850	450	linear

Table 1: Summary of the parameter values evaluated in the different experimental trials.

– by comparing the prediction of which object was being followed with the recorded ground truth. We evaluated whether the red object was detected as being followed (correct detection), detected as another object (wrong detection) or not detected as being followed at all (missed detection). All calculations were based solely on the first object being detected given that the first detection is the only important one for real-world applications. We evaluated the following values:

- Window size: Because Pursuits aims for instant interaction and is based on dwell-time, the window size needs to be quite short. Smooth eye pursuits last for a minimum of 100ms [14], so this is the smallest window size we tested. We also ran Pursuits on $w = 500ms$ as this is similar to conventional dwell-time based gaze interactions.

- Correlation threshold: We processed the data for values of $th_{corr} \in \{0.2, 0.3 \ldots 0.8, 0.9\}$. For small values of th_{corr} we expected a high detection rate, while for large values of th_{corr} we expected a lower detection rate but less wrong detections.

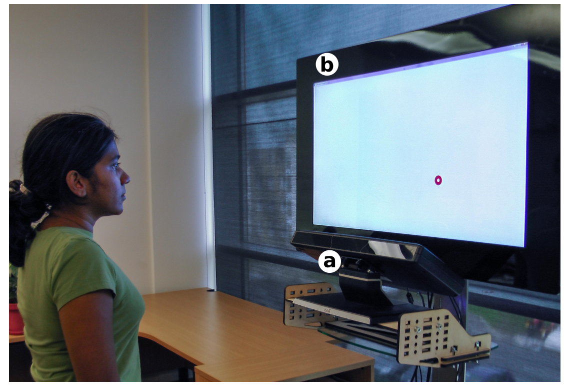


Figure 3: Experimental setup consisting of a Tobii X300 remote eye tracker (a) and a public display (b).

Apparatus and participants

For recording gaze data we used an uncalibrated Tobii X300 remote eye tracker. The horizontal angle of the tracker was adjusted depending on the participant's height to ensure it properly tracked their eyes. The stimulus was presented on a 40" 1920x1080px screen mounted perpendicular to the participant's line of sight. The eye tracker was placed on a platform under the screen (see Figure 3). The participant's gaze data and the trajectories of all visible and hidden objects were logged at 40Hz, which corresponds to the refresh rate of the screen. On average across all participants, the gaze estimation error of the eye tracker was 12.5° visual angle.

Eight participants (three female, five male) aged between 24 and 32 years (*mean=26.3, std=2.8*) took part in the study. None of the participants wore glasses during the study. Upon arrival they were first introduced to the goals of the study as well as the system and the stimulus. They were asked to stand 85 cm away from the screen.The total experiment time was about 10 minutes for each participant. To minimise fatigue, they was allowed a break in the middle of the experiment.

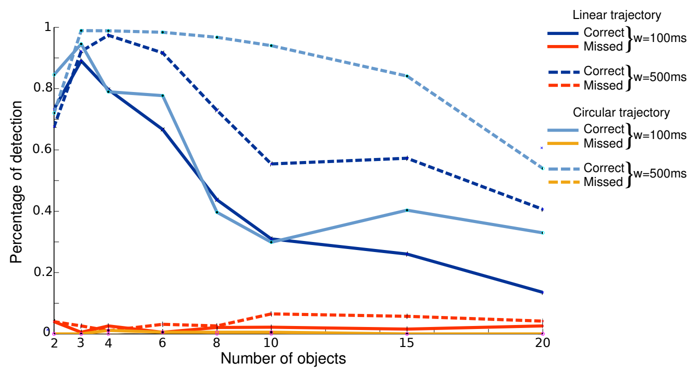
Results

Effects of the Number and Trajectory of Objects

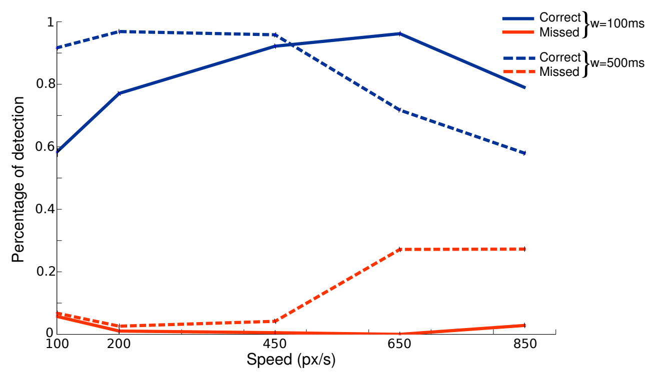
Figure 4a shows that the detection rate highly depends on the number of objects. The percentage of detection starts to drop with eight or more objects on the screen, i.e. the lower the number of objects on the screen the higher the performance. Circular movements allow for more robust detection performance for up to 15 objects. The style of trajectory does not seem to considerably impact performance up to four objects given that the correct detection rate stays above 80%. For both trajectory types, we note that when the correct detection rate decreases, the missing rate stays stable. This means that the algorithm still detected objects, but detected the wrong ones.

Effects of Speed

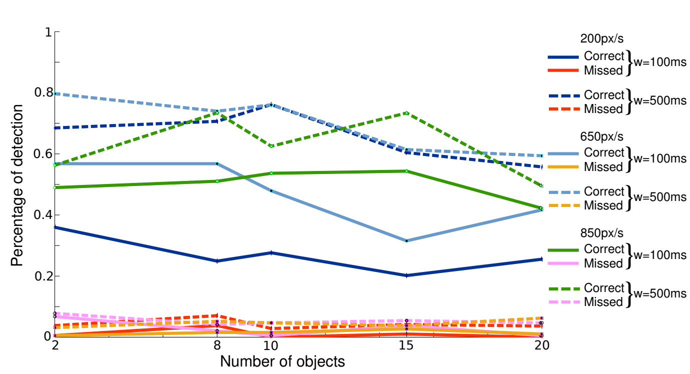
Figure 4b shows the correct and missed detection versus the speed of objects. The figure indicates that the speed of objects, when moving at the same speed, does not have a strong influence on detection performance. When objects move fast, from 650px/s we can see a significant increase in the missed



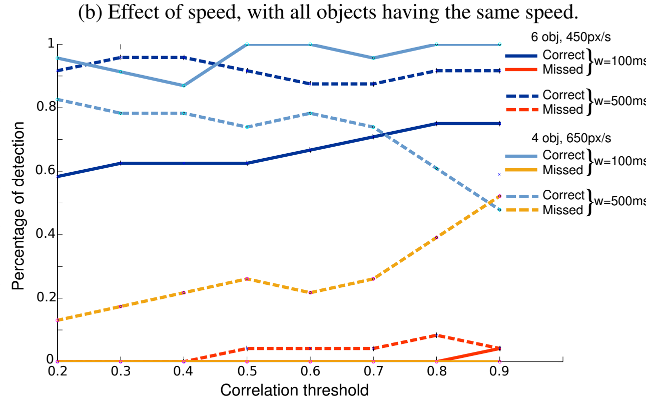
(a) Effect of the number of objects for both linear and circular trajectories.



(b) Effect of speed, with all objects having the same speed.



(c) Effects of speed, with the red object moving at a different speed than the other objects.



(d) Effects of the correlation coefficient for two cases : 6 objects moving linearly at 450px/s and 4 objects moving linearly at 650px/s.

Figure 4: Results of the algorithm evaluation study. For figures a, b and c, the datapoints are generated from the average value given by correlation thresholds set between 0.2 and 0.9.

detections for the larger window size, while performance remains high for the shorter window size. This can be explained by the fact that the objects moved so quickly that participants did not manage to follow them long enough before they reached the border of the screen, as the high missing rate shows. This can be avoided with a circular trajectory.

Figure 4c shows the detection performance when object speed varies as a function of the number of objects. Each colour corresponds to a specific speed of the red object, while other objects move at a different, constant speed. As we can see from the figure, if two objects have different speed but similar trajectories Pursuits can still differentiate them especially for larger window size ($w = 500ms$). A speed difference of 200px/s seems enough to increase the correct detection rate. A variation in speed also appears to help considerably to identify two objects that have exactly the same angle.

Effects of Correlation Threshold

Figure 4d shows a sweep over the correlation threshold th_{corr} for six objects moving linearly at 450px/s and four objects moving linearly at 650px/s. The correlation threshold does not seem to have a significant impact on detection performance. However, if accuracy is crucial, preference should be given to higher thresholds (beyond 0.5). Indeed, a higher thresholds results in a higher missing detection rate but also in less errors. This also implies that the time for a detection should be longer, as it is possible that Pursuits will miss a detection if the threshold is too high.

Effects of Window Size

Figure 4a shows that the detection performance is better for a larger window size w. This means that Pursuits performs better if the correlation is based on a longer amount of time, which implies the data contains a better excerpt of what the trajectories look like. Circular movement is significantly easier to detect with a larger window size, especially when there are eight objects or more. In Figure 4b, for lower speeds the accuracy of the smaller window size decreases while larger window size still yields a high detection performance. This was to be expected given that a slow pursuit resembles a fixation if considered only for a small period of time. The correlation on a fixation is not computable because the deviation of the data points will be too small. In contrast, even if the pursuit is slow but analysed for a longer period of time ($w = 500ms$), it can be robustly detected by Pursuits. Figure 4d shows the case where a smaller window size seems to be better than a bigger one, however it corresponds to the case where objects were too quick to be followed by the eyes for long enough.

Summary

This controlled laboratory study provides a number of important insights for choosing parameters that also have implications for interface design. First, detection performance decreases for eight objects and more if the objects move in a linear fashion and at the same speed. This can be avoided by using circular trajectories and a larger window size, or by varying the speed of objects moving on the screen. This is also the case if two objects move in parallel along the same trajectory. If objects on the screen have to move slowly a larger window size is required for Pursuits to properly differentiate between fixations and pursuits. The window size is a particularly important parameter and should in general be set around 500ms. It is still possible to perform a high detection with a shorter window size but it is more error-prone. This should only be favoured for high speeds and for systems whose reactivity is essential.

APPLICATIONS

Current gaze-based interfaces typically use on-screen gaze location as the sole input. In contrast, Pursuits is based on movements and thus opens up a whole new range of dynamic user interfaces. The algorithm evaluation highlighted which features are desirable and are likely to produce better accuracy when designing interfaces for Pursuits; to illustrate the specific characteristics and potential of Pursuits, in the following we describe six example real-world applications.

Public information display. A Pursuits-enabled public display on a university campus provides information to visitors of the university. The interface shows three large boxes each containing snapshots of information that a visitor might be interested in: bus times, a map, or upcoming events. These boxes smoothly "float" on the screen and attract the attention of bypassers. Once the interactive display detects that a box is being followed with the eyes, the screen fades out and displays the corresponding information. The content is left available for the visitor to read, and the only moving object on the screen is a small "back" box drifting slowly in a corner for the user to go back to the main menu by looking at it. This application demonstrates the possibility of creating harmonious interactions that exploit slow motion and large-sized targets.

Music shop display. A Pursuits-enabled display in a music store shows the covers of the latest music albums. A customer standing in front of the screen can follow the trajectory of one of the covers with their eyes to listen to a music extract from that album. Playback will fade out after 30 seconds or if the customer decides to look at another album cover. This application demonstrates how Pursuits can be used as a natural interaction trigger. It is important to note that in this example the feedback is not visual but auditive to prevent from a possible visual field saturation. This application also demonstrates a way to use Pursuits with a greater number of targets.

Game for waiting areas. People spend a lot of time in waiting areas, such as at bus stops or airports. A Pursuits-enabled display is placed on a wall to entertain passengers while waiting. It offers to play a simple game with a frog in the center and flies that randomly fly around. As the player follows a fly with their eyes, the frog unfolds its tongue to eat the target fly. The more flies are eaten, the higher the score of the player. Flies gradually become faster and smaller to increase the challenge. This application demonstrates that the target

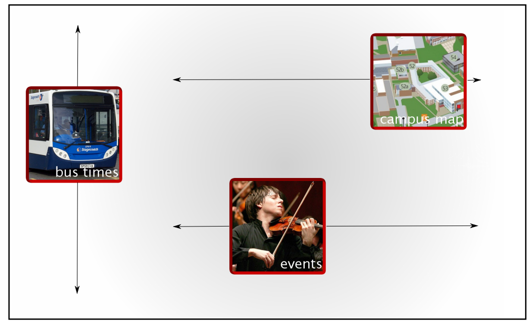


Figure 5: The public information display. This is an example of a harmonious, slow interface with large targets.

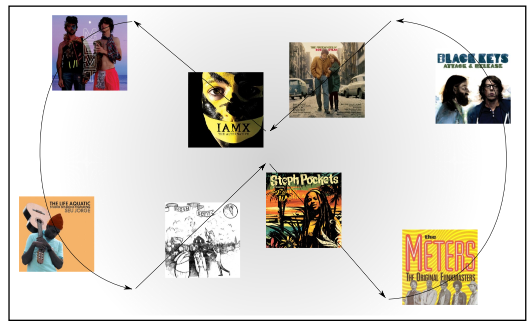


Figure 6: The music shop display. This presents an interface with a greater number of targets and audio feedback.

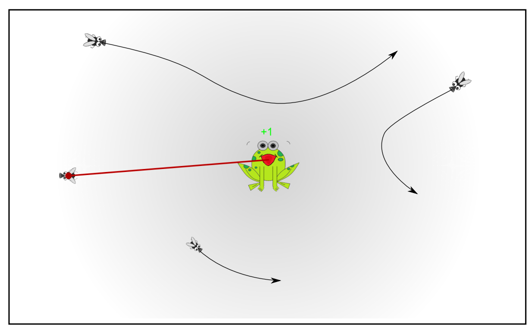


Figure 7: The frog game. This demonstrates several capabilities: the accurate selection of very small and quick targets, and the possibility of targets with more complex trajectories.

size does not matter for Pursuits. This is important for eye-based interfaces whose accuracy on the screen is often limited. This application would also be difficult to implement using other modalities. The flies are too quick and random to allow mouse or finger selection.

Gaze-aware museum display. A Pursuits-enabled screen in a museum shows the solar system with planets rotating around the sun along different trajectories and at different speeds. A child interested in the interface walks up to the display and looks at the planets. If a specific planet is followed the screen smoothly displays additional information such as the planet's rotation time, temperature and size. While the previous scenarios offer gaze control, this use of Pursuits develops gaze-

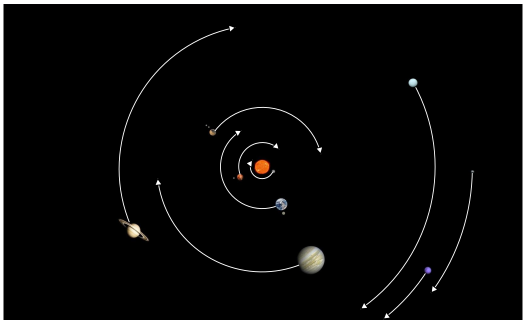


Figure 8: The gaze-aware museum display. This is an example of a more subtle, gaze-aware attentive display.

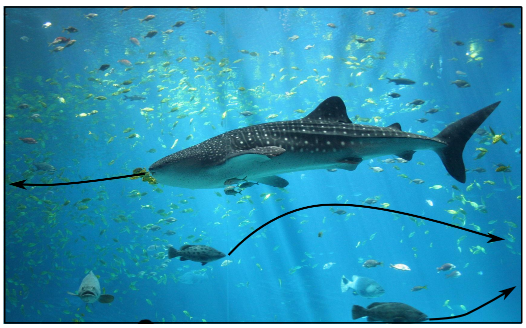


Figure 9: The fish password screensaver. This shows Pursuits could be used for discrete password entry.

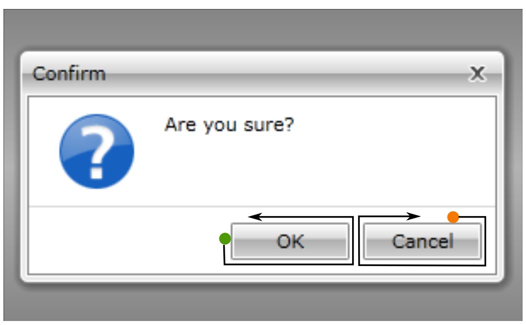


Figure 10: The hospital screen control. This is an example of how Pursuits could be used to control static desktop interfaces in touch-sensitive environments.

awareness of displays. This is different in that the display is attentive and enhances the user experience without explicit control.

User authentication. Secure user authentication is particularly challenging in public spaces and in situations where traditional authentication devices, such as a keypad or keyboard, are not available. For example in an office environment, if a user has been away from their computer, a Pursuits-enabled screen displays a screensaver-type animation, such as fish swimming in a fish tank in a cyclic fashion. As the person walks back up to the display, they look at four specific fish in a precise sequence. This is actually their own password for this computer, as they previously defined which fish they were going to look at, in which order. The user looks at the first fish out of the five on the screen, then the second one as it appears, and so on. The fish do not give feedback that they are being looked at. After the four-fish password is entered, the screen unlocks, allowing the user to interact. This application demonstrates how to use Pursuits for gaze-based graphical passwords [2].

Hospital screen control. Pursuits could also be used as an alternative input method for static desktop environments. In a hospital, nurses need to interact with a screen that contains a patient's information. However, touch is a problem in this environment as it can carry germs [13]. The screen interface has added stimulus around buttons, for example "yes" or "no" buttons have a small stimulus moving along their edge clockwise and anti-clockwise. As the nurses moves closer to the display, they can immediately select options by following the added stimulus, without needing to calibrate or to touch an input device. This interface is similar to Fekete *et al.* which raised the idea of driver movement mimicking for interface control [5] and that of Drewes *et. al* which introduced eye-based gestures for desktop control [4].

USABILITY EVALUATION

To assess the usability, accuracy and speed of Pursuits we evaluated three of the aforementioned applications in a constrained laboratory study. Findings from the parameters evaluation informed the choice of parameters for their implementation.

Public information display. Because the interaction needed to be robust and rapid to avoid user frustration we used a mid-long selection time $w = 400ms$ and a high threshold $th_{corr} = 0.9$ with three content boxes. The boxes moved at a speed of 300px/s, which is low enough to be harmonious but high enough to avoid a high error rate. The boxes moved linearly along one axis, either horizontal or vertical (see Figure 5). Because the correlation formula divides by the standard deviation it is not possible to calculate the correlation coefficient on both axis when an object moves on one axis only. We thus adapted Pursuits to compute the correlation coefficient on the relevant axis only.

Music shop display. The algorithm evaluation showed that performance dropped for more than eight objects. To produce individual motions that are different but still harmonious the trajectories of the album covers follows a figure-of-eight, with circular and linear sections alternating (see Figure 6). To make the selection robust and not flicker constantly we chose a very long window size, $w = 2s$ and a rather high correlation threshold, $th_{corr} = 0.7$.

Game. For the frog game robustness is not as important as speed of action, since time is controlled. The algorithm thus operates with a $w = 300ms$ reaction time and a correlation threshold $th_{corr} = 0.4$, which ensures quick selection instead of waiting for a perfect correlation fit. Flies move along suites of randomly generated bezier curves, adding to the difficulty to follow their trajectories. They start with a peak speed of 650px/s and for each eaten fly, a new fly reappears with an increased speed and decreased size.

Procedure

15 participants (5 female, 10 male) aged between 21 and 47 years (*mean=26.4*, *std=7.8*) participated in this study, none of which wore glasses, and three of which had already participated in the algorithm evaluation. Participants were asked to follow a scripted sequence of the previous tasks. For the public information display, they had to select a box randomly chosen by the researcher ten times. For the music shop display, they were asked to try to listen to different albums ten times, again randomly chosen by the researcher. In the case of the game they were asked to play it twice and to make the frog eat as many flies as possible. For the frog game, participants were instructed to tell the researcher when the frog was catching a different fly than the one they were looking at. The setup was the same as in the algorithm evaluation (see Figure 3). We recorded participants' success rate for each task as well as the time it took to perform a selection.

Results

Figure 11 shows the target selection times and detection rate for each task averaged over all participants. Participants were able to successfully use all three example applications, with the lowest success rate being 80% for the shop display application and the highest 98% for the public information display. This demonstrates that Pursuits can be robustly used for different tasks, independently of the nature of the application. For the public information display the selection time is almost seven times longer than the window size ($w = 400ms$). This can be explained by the high correlation threshold ($th_{corr} = 0.9$): most of the time, Pursuits did not detect any object was being looked at. However, when it did detect that an object was being followed it was almost always accurate.

The selection time of the shop display application is twice as long as the window size ($w = 2s$), which may result from the rather high correlation threshold ($th_{corr} = 0.7$) but also from the difficulty of following the complex trajectory of the albums. Moreover, we found that in most cases an album wrongly selected had a trajectory similar to the target, but was circular if the target was linear and vice-versa. This highlights the importance of selecting significantly different trajectories in future applications.

Finally, for the frog game, participants could select the flies 89% of the time, which shows that a quicker system or complex trajectories can be successfully used with Pursuits. The average highest score was 15.5 flies in 30 seconds, which is one fly every 1.94 seconds. The average completion time is thus six times as long as the correlation time $w = 300ms$, which can be explained by the trajectory of the flies. When updating the trajectory from a curve to another, the fly would slow down before accelerating again. During that slow part, the eye pattern resembled that of a fixation and was thus too slow to be detected. This again highlights that Pursuits is only relevant on targets that are constantly moving.

FIELD STUDY

To demonstrate the spontaneity and simplicity of the interaction, we conducted a field study with real environment settings. During the university open days, we placed the system

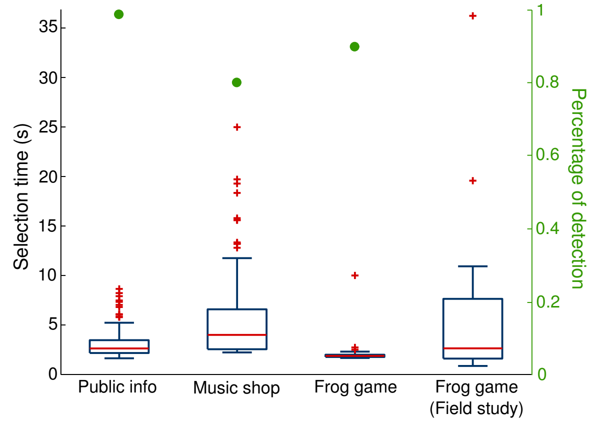


Figure 11: Selection time (boxes in blue, median as red line and outliers as red crosses) and percentage of accurate detection (green dots) for the three applications and the field study. Public information display: median = 2.63s, detection = 98%. Music shop display: median = 3.99s, detection = 80%. Frog game: median = 1.88s, detection = 89%. Frog game in field study: median = 2.64s.

in the department's lobby area, next to the main door. We displayed the frog game ready to be used. We placed fliers in the lunch area on the tables, to entice people to come and try to catch as many flies as possible with their eyes. The flier indicated that they had to step on footmarks on the floor depending on their height – this is to palliate to the narrow angle of view of the remote eye tracker, whose angle we adapted to the height of the user during the controlled studies. It also asked them to place themselves so as to turn an indicator light green (the color, green or red, depended on whether the eye-tracker could see their eyes properly).On the screen frame, a notice read "Try me! I work with your eyes!". The setup is presented in Figure 12. The study was filmed from the back so that participants are not identifiable. No researchers were present during the study: the system was left for visiting prospective families to discover on their own.

The experiment lasted 2.5 hours. During this time, 16 unique persons interacted with the display, totalling 21 interactions (four persons came back at least once). Two persons were unable to interact with the system. This means 87.5% of people were able to successfully interact with the display, with no help from the researchers and no other instructions than these on the flier. The median time between the time participants placed themselves on the footmarks and a first successful interaction was 2.64 seconds (see figure 11). This is slightly longer than the same game during the controlled user study, which comes from the fact that a researcher was not there to place them in the ideal position or to adjust the eye tracker's angle to their height. The percentage of accurate detection could not be computed, since we could not ask participants to indicate when the wrong fly was selected.

This field study shows that the system is simple enough to be used by naïve users. It also shows that, even though users had never seen the system before, they were able to interact with it quickly, without having to adjust themselves other than by

Figure 12: The setup of the field study. (a) Screen with frog game. (b) Eye tracker. (c) Placement footmarks with colours depending on the user's height. The field study showed that users were able to interact without given assistance.

placement. This is to palliate to the narrow angle of view of the eye tracker: The use of an eye-tracker with a wider angle of view, or a mobile eye-tracker could solve this problem and further speed interaction.

DISCUSSION

Results from our user studies suggest that Pursuits is a versatile and robust technique for interaction with moving objects. It opens up new perspectives on the design and implementation of a new class of gaze-based interfaces that rely on smooth pursuits as input. A key difference to interfaces that use absolute point of gaze is that smooth pursuit movements require the interface to be dynamic, i.e. with the objects of interest moving with different trajectories and speeds. While this requirement might be problematic for interfaces that follow the traditional WIMP (windows, icons, menus, pointer) paradigm or use static interface elements, Pursuits is predestined for highly dynamic interfaces, such as interactive multimedia installations or games. Beyond such special-purpose interfaces, Pursuits also encourages to break out of conventional thinking with respect to how future gaze-based interfaces might be designed and look like.

Pursuits' inherent necessity of a dynamic interface has several implications for the design of such interfaces. First, the constant movement might be a source of confusion or fatigue for users if used for longer periods of time. This should be investigated in more detail in order to establish how straining for the eyes the technique might be. This dynamism is also potentially unsuitable for objects that contain more than a short segment of text as it may be difficult to read and follow the moving text at the same time. Objects that move too slowly may also cause bad performance, which can be difficult to assess when designing an application. Another problem to consider is its vaddiulnerability to head movements: If a user moves their head as the same time as they move their eyes, for instance if the target performs a very large movement, then the eye trajectory will not look similar to that of the object.

Because the eye tracker we used is remote, its field of view was limited which is why we had to adjust its angle to view the eyes of users that were particularly tall or short. This could be avoided by using a mobile eye tracker. It would also be interesting to use a device that can detect the eye movements of several persons at the same time, thus enabling multi-user interactions. This could potentially be done together with a comparison of the effect of algorithms other than Pearson's product-moment correlation.

The current work presents several challenges that would be interesting to explore in future work. While it offers a solution to the problem of selection of very small targets, additional researcher would be beneficial to understand the impact of large moving targets on gaze scanning behaviour and user distraction. The effect of non-smooth pursuits on the technique's performance would also be an interesting area to explore. Additionally, a qualitative study comparing different gaze interaction techniques would certainly provide valuable insights for further interface and interaction design. A quantitative comparison of selection speed and accuracy against touch interaction would also be highly valuable in the case of the frog game, for example.

CONCLUSION

We presented Pursuits, a novel method for eye-based interaction that we believe is compelling in its simplicity – simply matching eye movement against the trajectories of on-screen objects. Yet it facilitates robust selection of moving objects, in ways that are spontaneous and pervasively deployable. Our studies have shown that users can simply walk up to a display and readily interact. As Pursuits embraces movement, it opens up a compelling design space in which moving targets can be used to create engaging user experiences.

REFERENCES

1. R. Ballagas, M. Rohs, and J. G. Sheridan. Sweep and point and shoot: phonecam-based interactions for large public displays. In CHI EA '05, pages 1200–1203, 2005. ACM.
2. A. Bulling, F. Alt, and A. Schmidt. Increasing the security of gaze-based cued-recall graphical passwords using saliency masks. In CHI '12, pages 3011–3020, 2012. ACM.
3. K. Cheverst, A. Dix, D. Fitton, C. Kray, M. Rouncefield, C. Sas, G. Saslis-Lagoudakis, and J. G. Sheridan. Exploring bluetooth based mobile phone interaction with the hermes photo display. In MobileHCI '05, pages 47–54, 2005. ACM.
4. H. Drewes and A. Schmidt. Interacting with the computer using gaze gestures. In INTERACT '07, pages 475–488. Springer, 2007.

5. J.-D. Fekete, N. Elmqvist, and Y. Guiard. Motion-pointing: target selection using elliptical motions. In CHI '09, pages 289–298. ACM, 2009.
6. D. Fono and R. Vertegaal. Eyewindows: evaluation of eye-controlled zooming windows for focus selection. In CHI '05, pages 151–160. ACM, 2005.
7. K. R. Gegenfurtner, D. Xing, B. H. Scott, and M. J. Hawken. A comparison of pursuit eye movement and perceptual performance in speed discrimination. *Journal of Vision*, 3:865–876, 2003.
8. T. J. Gunn, P. Irani, and J. Anderson. An evaluation of techniques for selecting moving targets. In CHI EA '09, pages 3329–3334, 2009. ACM.
9. D. W. Hansen, H. H. T. Skovsgaard, J. P. Hansen, and E. Møllenbach. Noise tolerant selection by gaze-controlled pan and zoom in 3d. In *Proc. of the Symposium on Eye Tracking Research and Applications,* ETRA '08, pages 205–212. ACM, 2008.
10. A. Hyrskykari, H. Istance, and S. Vickers. Gaze gestures or dwell-based interaction? In *Proc. of the Symposium on Eye Tracking Research and Applications*, ETRA '12, pages 229–232, 2012. ACM.
11. M. P. Hyrskykari, A. and K.-J. Rih. From gaze control to attentive interfaces. In *Proc. of HCII*. Lawrence Erlbaum Associates, 2005.
12. R. J. K. Jacob. What you look at is what you get: eye movement-based interaction techniques. In CHI '90, pages 11–18. ACM, 1990.
13. R. Johnson, K. O'Hara, A. Sellen, C. Cousins, and A. Criminisi. Exploring the potential for touchless interaction in image-guided interventional radiology. In CHI'11, pages 3323–3332. ACM.
14. R. J. Leigh and D. S. Zee. *The neurology of eye movements*, chapter 3. Number 55 in Contemporary neurology series. FA Davis, 2 edition, 1991.
15. P. Maglio, T. Matlock, C. Campbell, S. Zhai, and B. Smith. Gaze and speech in attentive user interfaces. In T. Tan, Y. Shi, and W. Gao, editors, *Advances in Multimodal Interfaces ICMI 2000*, volume 1948 of *Lecture Notes in Computer Science*, pages 1–7. Springer Berlin Heidelberg, 2000.
16. D. Miniotas, O. Špakov, and I. S. MacKenzie. Eye gaze interaction with expanding targets. In CHI EA '04, pages 1255–1258, 2004. ACM.
17. D. Mould and C. Gutwin. The effects of feedback on targeting with multiple moving targets. In GI '04, pages 25–32, 2004. Canadian Human-Computer Communications Society.
18. A. Pavlovych and W. Stuerzlinger. Target following performance in the presence of latency, jitter, and signal dropouts. In GI '11, pages 33–40, 2011. Canadian Human-Computer Communications Society.
19. P. Peltonen, E. Kurvinen, A. Salovaara, G. Jacucci, T. Ilmonen, J. Evans, A. Oulasvirta, and P. Saarikko. It's mine, don't touch!: interactions at a large multi-touch display in a city centre. In CHI '08, pages 1285–1294, 2008. ACM.
20. K. Rayner. Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin*, pages 372–422, 1998.
21. D. Robinson. The mechanics of human smooth pursuit eye movement. *The Journal of Physiology*, 180(3):569–591, Oct. 1965.
22. R. Ruddarraju, A. Haro, K. Nagel, Q. T. Tran, I. A. Essa, G. Abowd, and E. D. Mynatt. Perceptual user interfaces using vision-based eye tracking. In *Proc. of the 5th international conference on Multimodal interfaces*, ICMI '03, pages 227–233, 2003. ACM.
23. J. S. Shell, R. Vertegaal, and A. W. Skaburskis. Eyepliances: attention-seeking devices that respond to visual attention. In CHI EA '03, pages 770–771, 2003. ACM.
24. L. E. Sibert and R. J. K. Jacob. Evaluation of eye gaze interaction. In CHI '00, pages 281–288, 2000. ACM.
25. S. Stellmach and R. Dachselt. Look & touch: gaze-supported target acquisition. In CHI '12, pages 2981–2990. ACM, 2012.
26. O. Tuisku, P. Majaranta, P. Isokoski, and K.-J. Räihä. Now dasher! dash away!: longitudinal study of fast text entry by eye gaze. In *Proc. of the Symposium on Eye tracking Research and Applications*, ETRA '08, pages 19–26, 2008. ACM.
27. R. Vertegaal. Attentive user interfaces. *Commun. ACM*, 46(3):30–33, Mar. 2003.
28. D. Vogel and R. Balakrishnan. Interactive public ambient displays: transitioning from implicit to explicit, public to personal, interaction with multiple users. In UIST '04, pages 137–146, 2004. ACM.
29. S. Zhai, C. Morimoto, and S. Ihde. Manual and gaze input cascaded (magic) pointing. In CHI '99, pages 246–253, 1999. ACM.
30. Y. Zhang, A. Bulling, and H. Gellersen. Sideways: A gaze interface for spontaneous interaction with situated displays. In CHI'13, 2013. ACM.