

Exploration of Smooth Pursuit Eye Movements for Gaze Calibration in Games



Argenis Ramirez Gomez
a.ramirezgomez@lancaster.ac.uk

Supervisor: Professor Hans Gellersen

MSc in Computer Science

School of Computing and Communications

May 2017

Declaration

This thesis has not been submitted in support of an application for another degree at this or any other university. It is the result of my own work and includes nothing that is the outcome of work done in collaboration except where specifically indicated. Many of the ideas in this thesis were the product of discussion with my supervisor Hans Gellersen.

Argenis Ramirez Gomez

Lancaster University, UK

Abstract

Eye tracking offers opportunities to extend novel interfaces and promises new ways of interaction for gameplay. However, gaze has been found challenging to use in dynamic interfaces involving motion. Moving targets are hard to select with state of the art gaze input methods and gaze estimation requires calibration in order to be accurate when offering a successful interaction experience. Smooth pursuit eye movements have been used to solve this new paradigm, but there is not enough information on the behavior of the eyes when performing such eye movement. In this work, we tried to understand the relationship between gaze and motion when performing smooth pursuit movements through the integration of calibration within a videogame. In our first study, we propose to leverage the attentive gaze behavior of the eyes during gameplay for implicit and continuous re-calibration. We demonstrated this with *GazeBall*, a retro-inspired version of Atari's BreakOut game in which we continually calibrate gaze based on the ball's movement and the player's inevitable ocular pursuit on the ball. Continuous calibration enabled the extension of the game with a gaze-based 'power-up'. In the evaluation of GazeBall, we show that our approach is effective in maintaining highly accurate gaze input over time, while re-calibration remains invisible to the player. GazeBall raised awareness on the lack of information about smooth pursuit for interaction. Therefore, in our second study, we focused on gaining more understanding on the behavior of the eyes. By testing different motion directions and speeds we found anticipatory predictions during gaze trajectory that indicates that the common reaction of the eyes when a moving target is present is not only following but trying to predict and advance the displayed movement.

Contents

1. Introduction	1
1.1 Problem Analysis	1
1.2 Investigating Smooth Pursuit for Continuous Re-calibration	2
1.3 Contributions	3
2. Related Work	4
2.1 Gaze for Interaction	4
2.1.1 Gaze as Pointer	4
2.1.2 Gaze Gestures	5
2.1.3 Implicit gaze	5
2.1.4 Multimodal interaction	6
2.2 Gaze interaction for Gameplay	6
2.2.1 Gaze in Games	7
2.3 Gaze Calibration	8
2.3.1 Calibration Games	9
2.4 Gaze and Motion	10
2.4.1 Smooth Pursuit eye movements	11
2.4.2 Smooth Pursuit calibration	11
3. Smooth Pursuit re-calibration	14
3.1 Game Design	15
3.1.1 Gaze Interaction	15
3.2 Behavioral Continuous Re-calibration	17
3.2.1 Implementation of the calibration	19
3.3 Evaluation	21
3.3.1 Participants	21
3.3.2 Apparatus	22
3.3.3 Procedure	22
3.4 Results	23
3.4.1 Calibration Accuracy	23
3.4.2 Gameplay Experience	25
3.5 Discussion	26
3.6 Future Work	28
3.7 Conclusion	28
4. A Further Insight on Smooth Pursuit eye movements	30
4.1 Motivation	30
4.2 Evaluation	30
4.2.1 Participants	31
4.2.2 Apparatus	32
4.2.3 Procedure	32
4.3 Results	33

4.3.1	Calibration Accuracy	33
4.3.2	Smooth Pursuit Performance	35
4.3.3	Smooth Pursuit Motion Analysis	38
4.4	Discussion	42
4.5	Conclusion	48
5.	Discussion	50
5.1	Guidelines for Smooth Pursuit Design	52
5.2	Direction, Generalization and Limitations	52
5.3	Future work	54
6.	Conclusion	56
	Bibliography	57

1. Introduction

Eye tracking technology provides information about where our eyes are looking and has been used as a novel ubiquitous user input for Human-Computer Interaction (HCI) since the 80s. The estimation of gaze position on the screen as the point of regard or interest allowed the creation of new ways of hands-free interaction and more accessible interfaces, by being able to select with just staring at an object on the screen. Such accessibility and the evolution of the price of eye tracking technology led towards the interest on the creation of new interaction techniques based on different eye movements, turning gaze interaction into a compelling field to research.

Human-Computer Interaction literature was originally grounded in the use of gaze fixations on static content, considering the dwell time performed when the eyes focus on an object (or item). Further research used saccadic eye movements, encoding eye gestures when the eyes change their focus from one point to another deliberately. Against saccadic movement, smooth pursuit eye movements are represented by continuous motion of the eyes performed spontaneously when the eyes follow a target in motion. All these approaches proposed gaze as an interesting input method. The eyes give information of what interests us and are involved in every action we make. Thus, interfaces integrating eye tracking technologies can use such information to either substitute the mouse for selection, enhance its performance by indicating what we want to select. All in all, broadening the opportunity to make interfaces more accessible for people with reduced mobility.

Gaze trackers have become cheaper during the last few years allowing them to get out from research labs and be introduced as an everyday domestic device for more creative uses. The gaming industry has benefited from that evolution and is starting to introduce gaze interaction in big franchises' titles. Gaze interaction offers accessibility to extend the market, and the potential of gaze as input allowed game developers to introduce an enhanced and novel gameplay. New techniques combining gaze pointing and game controllers are emerging to offer players an improved and extended game experience able to upgrade the gaming immersion.

However, the user's experience when using gaze interaction as input, that fosters accessibility and enhances interfaces, is dependent on the accuracy of the eye tracker estimation.

1.1 Problem Analysis

To guarantee an accurate gaze interaction, careful calibration becomes compulsory and needs to be performed to the individual user. Eye tracking calibration is needed in order to accurately map gaze pointing onto the screen. Without calibration, gaze interaction could lead to a bad experience by triggering unwanted outcomes, by selecting items unintentionally. Conventional methods for gaze calibration require users to look for a period of time (fixate) on several calibration points spread around the target display.

Paired gaze and screen points are collected and matched to create a transformation function used for the later estimation (mapping) of the calibrated gaze point.

The calibration task is alienated from the application experience [15] and is of poor usability [43]. It requires repetition due to its lack of robustness to compensate head movements or the user’s position changes. When we work in front a computer we are not always in the same position, we take breaks and move away from the workspace. Only by performing again the calibration task (re-calibration) can we obtain a sufficiently accurate mapping of the gaze point to the display space [52, 72].

However, re-calibration becomes a challenge when using calibration tasks based on fixations. The calibration process remains separated from further interaction and is not considered enjoyable. Every time users need to re-calibrate they need to stop the end-use application to correct gaze accuracy. Hence, the calibration task can potentially affect negatively the user’s perception of the target application experience (using gaze interaction). This liability could be able to blur the compelling future leveraged by eye movements for novel accessible interfaces.

On the other hand, in order to make the calibration task less tedious *Pfeuffer et al.* [49] proposed smooth pursuit eye movements for calibration. Their approach is based on the unconscious attention of the eyes to moving cues by following them. Matched gaze and on-screen target points are paired and collected to be transformed into an accurate calibration profile. Smooth pursuit eye movements allow the design of a user-friendly calibration task that could be integrated and customized according to the context of the target application enabling gaze interaction. Nevertheless, their approach to making calibration tasks more user-friendly remains a separate task, again making the user break the end application experience to re-calibrate the system.

The appearance of novel interfaces introducing more dynamic content-in-motion provides richer environments to design novel interactive frameworks using smooth pursuits. Movement is reported as a problem in novel interfaces, as it is harder to hit a moving target than a static one [26], gaze pointing becomes insufficient and a significant problem when the content is not static. Modern interfaces start involving moving content; examples including games [15, 32]; simulations [10, 63] or video streams. Our eyes movement are good in attending to motion [38, 49], and following moving objects; thus providing new potential for techniques that leverage the behavior and use of gaze [18, 70]. Smooth pursuit eye movement demonstrated to be a good asset for calibration and show compelling potential to be integrated into different interface dynamics, suggesting that re-calibration can be done implicitly.

1.2 Investigating Smooth Pursuit for Continuous Re-calibration

We studied the integration of smooth pursuit calibration within applications featuring dynamic elements and content in motion. Smooth pursuit is reported in state of the art literature as a robust predictor for where the user looks and can be invisibly detected when moving content is presented, motivating our approach to using this eye movement

for implicit calibration. We used games as a case study to integrate the gaze estimation calibration process in order to enhance it with innovative interaction.

We developed *GazeBall*, a videogame inspired on *Breakout* by *Atari* [1] and extended with a gaze-based game mechanic. It embedded calibration process on the game dynamics, to be performed implicitly and invisibly, rather than treating calibration as a separate task. Our case study does not represent a calibration game, but a game with embedded calibration. Moreover, we designed calibration inside *GazeBall* thinking of the natural behavior (the trend in behavior) that players exhibit in gameplay as they follow moving elements in the scene with their eyes rather than requiring them to fixate on calibration targets. Finally, we proposed to use continuous re-calibration to continually calibrate during gameplay and maintain high accuracy of gaze input over time rather than calibrating once at the start only.

During the development of *GazeBall* we assumed that using a videogame as experiment context would make users tend to follow the moving targets with their gaze unconsciously, performing smooth pursuit eye movements while playing. By embedding the gaze re-calibration inside gameplay we aimed to confirm that invisible integration is possible when dynamic content is presented and that the quality of the obtained accuracy level can be maintained. However, there is no evidence of whether smooth pursuits are accurate predictors of where the gaze is pointing, even though they are used for calibration. The aim of our follow-up research is to understand the response of gaze when moving content on-display is presented. This knowledge will help in determining the capacities of eye movement to assess when the quality of the calibrated gaze point is deteriorating.

With a second experiment, we propose to expose smooth pursuit movements to different moving targets' directions and speeds to understand their precision and the characteristics of their performance when they follow motion. Learning about pursuits would provide us with new tools to design interfaces that consider the spontaneous behavior of the eyes as a method of interaction.

1.3 Contributions

Our work with smooth pursuit eye movements made three contributions. First, we designed *GazeBall*, a game that uses gaze data for both enhancing interaction and gameplay. It integrates the attentive behavior of following moving objects with gaze to perform a continuous and implicit calibration based on smooth pursuit eye movements. Second, we propose a behavioral analysis in order to integrate smart continuous and unaware gaze re-calibration methods in ongoing game dynamics rather than having a separate calibration task that would affect immersion. Not compromising gaze estimation accuracy, not consuming time or affecting game performance. Third, we identified gaze trends while it follows content in motion confirming an anticipatory behavior depending on the displayed moving features that will help improving state of the art calibration methods using smooth pursuit eye movements.

2. Related Work

2.1 Gaze for Interaction

Gaze became a compelling modality for human-computer interaction (HCI): our eyes are involved in most of our daily interactions. They provide very valuable information as they are directly connected to the brain [40]. Further, eye tracking technology is becoming available at a low cost making this technology a potential everyday commodity.

State of the art applications expand gaze interaction from a psychology research tool used in the evaluation of user experience (sensor), to novel compelling control modalities (input). The former tries to understand human perception, cognition, and attention to evaluate how the experience can be improved [8], and it is used for marketing research. The latter is dominated by techniques that use gaze as a pointer in desktop interfaces, substituting or complementing mouse interaction [15, 23].

2.1.1 Gaze as Pointer

The established gaze input techniques and design guidelines are based on the observation that gaze is a natural (spontaneous) pointer that can be used for selection. Gaze is related to the point of regard, as we look at targets before selecting them on the screen [31, 36, 37]. Gaze pointing is introduced in most gaze-based applications as we can select items by just looking at them, quicker than with the mouse [15, 23] by just focusing on them during a dwell time, performing a fixation. Fixations consist of the ability of the eyes to focus on a static element and hold its image reducing ocular drifts; for example, the time our gaze spends stopped on a word when reading (see *Figure 1*).

Further research that emerged from that work was designed for the desktop interfaces of the 90s, characterized by windows and icons and menus that can be selected just by gaze pointing [75]. As a result, research in HCI has considered gaze only in conjunction with interfaces that do not involve movement. However, with the appearance of more dynamic content, new techniques were introduced.

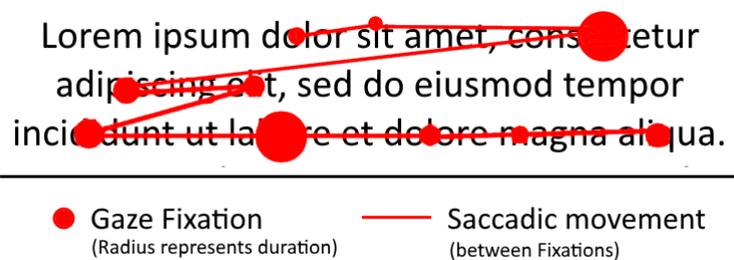


Figure 1: Representation of gaze fixations and saccadic movements when reading.

Nevertheless, gaze interaction as a pointer can be affected by the Midas effect [25] or the natural jitter of the eyes, causing the triggering of unwanted outcomes. Users accidentally look at an item without the intention of selecting it as their gaze is often attracted to nearby items [33, 36, 60]. *Zhai et al.* [75] proposed using gaze not as the pointer that substitutes the traditional input control but as a helping tool. The mouse remains as the main input method but gaze provides fine manipulation by moving the mouse cursor toward the gaze area (interest).

Accordingly, the use of gaze pointing as main interaction technique is not always recommended when accuracy is compromised. With the appearance of interfaces containing animations and motion, a new challenge appeared for user input using gaze. Presented motion makes it hard to select with the mouse, and even knowing that gaze is faster than the former selection method (the mouse) [42], and how to select with gaze is not trivial.

2.1.2 Gaze Gestures

Gaze gestures appeared as a response to the problem. State of the art research takes saccadic eye movement in order to encode gaze gestures for interaction [16], but there is a limited understanding of how gaze reacts to motion. Saccades are quick eye movements between fixations (see *Figure 1*) that can be performed by changing the object of focus or intentionally as a gesture.

Eye gestures are performed when the eyes move their focus towards another point, encoding a directional movement. Examples using gaze gestures avoid calibration and intend to provide rich interaction in the wild by translating eye movement to interface commands [16]. *Zhang et al.* [76] show how eye movements can be decoded to detect sideways eye movement when looking at the interface, causing a sliding response of the shown items.

This approach has also been used as a tool for accessibility for people with motor disabilities [67]. Projects like *The EyeWriter Initiative* [30] developed a framework to be able to draw graffiti characters with the movement of your eyes, allowing, for instance, an artist with ALS (Amyotrophic Lateral Sclerosis) to be able to create art again. However, performing intended saccadic movements is unnatural, leading to interaction techniques that are difficult to master and create eyes tiredness.

Although gaze gestures provide more flexibility and accessibility when designing interaction as calibration is not needed, less grained output information is provided. This approach for gaze interaction is only able to detect attention to the display [55] or differentiating whether users are looking to the center or left/right side of the screen [6], limiting the potential and complexity of created interfaces.

2.1.3 Implicit gaze

Implicit gaze sensing (or information about gaze position) has been used in psychology and behavioral science research to understand human behavior in different contexts. In this context, the accuracy of gaze input information is very important in order to

discriminate where users are looking. Previous research shows results on where the eyes look while reading [7], browsing web searches [29] or when they are exposed to different stimuli like advertisement [47].

Implicit gaze input has also been used to investigate on how to create attentive applications and interfaces that change their dynamics and improve according to user's gaze information. One of the most famous and iconic examples is work by *Zhai et al.* on *MAGIC* gaze pointing [75], that leverages the gaze point information to effectively control the speed when moving the mouse, allowing a quicker performance. Other examples include games that adapt their engine and difficulty according to gaze information [44], or reading assistants [53].

2.1.4 Multimodal interaction

Multimodal interaction has been another approach to solve the lack of precision in systems using gaze pointing as the main interaction. This new solution joined implicit gaze input, with other input methods. From the traditional mouse [75], mid-air gestures [63], voice [46], touch [48], joysticks and gamepads [10], different alternatives have been used to avoid unwanted results or a wrong experience during interaction with gaze.

Most examples are related to the *Midas problem* and use gaze as an interaction enhancer rather than as a control tool, providing information about the point of interest. What users want is selected by gaze and later confirmed by the second mode of interaction. As a result, the gaming mass market benefited from gaze interaction as a new compelling interaction method.

2.2 Gaze interaction for Gameplay

Following work on multimodal interaction, research on games introduced gaze to replace original controls [31, 35], or complimenting them [32, 54]. *Isokoski et al.* [31] reviews the first trends when using gaze interaction for gaming. In their work it is highlighted how gaze interaction created attention-aware games and richer input, widening accessibility for people with decreased mobility.

In further research, gaze pointing is introduced as a tool for immersion in games. *Nacke et al.* [45] proposed gaze as a controller of the camera point of view in 3D environments. Others introduced it to understand the context [44]. Both provided a greater sense of realism, as gaze triggered events that resemble the usual behavior of the eyes. *Hillaire et al.* [27] used the eyes' information to adapt the game graphics' rendering. In our vision, only the focus of our foveal vision is clear, whereas our peripheral vision remains blurred. By rendering only the focus of our gaze point they provided a more realistic experience while improving the game engine's performance.

Further, gaze has also been proposed as a tool for social interaction with avatars. Both works by *Smith et al.* [54] and *Vidal et al.* [69] show how gaze interaction can provide richer interaction with avatars. In their applications, they used gaze point information to interact with them as we would do with another human being in a social context. *Vidal*

et al. work show how information of where our eyes are looking while interacting with an avatar can trigger different responses from the virtual world as our gaze can show distraction, disinterest, challenge or shyness among others.

Alternatively, gaze interaction has been introduced as a tool for storyline adaptation based on attention direction [9], or re-engaging the lack of it [12]. By using gaze information, game engines can adapt their content to improve their experience. *Munoz et al.* [44] used gaze information to predict players behavior when navigating in the game and create more challenging game dynamics.

On the other hand, gaze interaction also introduced new game dynamics. The work in the game *Shynosaurus* by *Vidal* [68] introduced gaze interaction to create an attention dilemma. The aim of the game is to rescue "cuties" by dragging them with the mouse before the "Shynosaurus" get them. However, if players look at the villains they would "get shy" and stop moving. This way they introduced a difficulty for players that need to manage where they are looking at, creating a challenging dynamic.

2.2.1 Gaze in Games

Research using gaze in gameplay became more tangible thanks to the decrease of eye trackers cost. Mass market game franchises like *Tomb Rider* by *Crystal Dynamics*, *Tom Clancy's* or *Assassin's Creed* by *Ubisoft* introduced gaze interaction in their gameplay to manage graphics rendering, control camera field of view, affect main characters movement by giving direction, or provide players with information of where they are looking (see *Figure 2*).

Gaze interaction has the potential to become a great asset for games. Work by *Velloso et al.* [61] refer to the term *EyePlay* [59] to talk about the *playful experiences that take input from the eyes*. Their work reviews eye movements literature to develop a list of eye-enabled game dynamics. The analysis takes into account eye fixations (pointing with gaze), saccades (gaze gestures), smooth pursuits (gaze attention to movement), vergence (gaze focus on nearby and further objects according to depth), compensatory eye movements (while moving the head when fixations occur) and Optokinetic Nystagmus



Figure 2: (Left) Using gaze to aim targets in *Tomb Raider*. (Center) Gaze is used to render the scene with more or less light in *Assassin's Creed*. (Right) Gaze is used to give information about the environment, such as enemies in *Tom Clancy's*.

(combination of saccades and pursuits) in order to list a Game Mechanics Taxonomy.

Gaze in games has been used for multiple purposes. Gaze point has been used as a position indicator to be used for game navigation. Examples range from using a 1:1 mapping of gaze and player element to move the paddle in the popular games *BreakOut* [14] or *GuitarHero* [67]; looking to a specific point to make the player character move towards the gaze point [10, 17, 54], and looking at different parts of the screen that are used as Virtual WASD buttons [33, 46, 65, 66] or gradients for moving directions [58].

Related to navigation, gaze has also been used as a pointer to direct camera navigation (viewport), controlling where it points [10], moving it sideways [31], or control its speed [54]. Further, given that the eyes show the point of interest, it has also been used in shooting games to either aim the direction of the weapon [62] or trigger it by using gestures [31], blinks [73], or pursuits [70].

Moreover, the point of regard of gaze has also been used in games to activate game objects when they are stared at [17, 34], manipulate them (by picking and dropping) [4, 5, 23, 56, 74] or interacting with in-game widgets and menus [3].

Gaze is also a tool for communication. The way we look can mean different things depending on how and where we stare. From distraction, submission, attention to challenge, games have used gaze interaction with avatars to detect users behavior [69, 12]. Other uses of social gaze have explored an attention dilemma [68] or interaction between two players' gaze [2]. Similarly, gaze can also interact with the game environment, to make the game engine adapt the level of difficulty according to gaze behavior [44], predict players behavior [27], triggering different narrative stories depending on what players look at most [64] or make something appear in the interface when players look at certain areas [22, 9, 50, 41].

Finally, the last game dynamic based on gaze interaction used players' attention to adapt the visual effects. Examples include adaptive depth-of-field blur to improve the perception of 3D game objects and compensating camera motion [27]; the smart management of graphic rendering [50]; or the graphic control for immersion [2].

Nevertheless, in order to use gaze as an accurate input for a successful interaction in games, it needs to be calibrated. Calibration is usually presented as a separate application in which users are asked to look at specific points of the screen in order to accurately estimate where they are looking afterward. The task is isolated from the game context and has potential to break the game experience when the game needs to be paused in order to perform it.

2.3 Gaze Calibration

Gaze calibration consists of collecting samples of gaze at known points at the output (screen) so as to accurately perform a gaze-to-display mapping. Most commonly used calibration methods consist of performing gaze fixations on (between 7 and 9) static points placed across the screen at known positions (see *Figure 3*). Users are asked to look at displayed target points during a dwell time and paired target and gaze points on

the screen are saved for later processing. Correspondences between both are later used to compute the estimation of gaze coordinates.

Further research has been done to work on the reduction of calibration points to one or two points [24, 71]. Results compromise accuracy and do not offer as precise an estimation as a system with more points, requiring additional ones to calibrate. Commercial eye trackers, like *Tobii's EyeX* uses a standardized 7 point fixation calibration system (see *Figure 4*) and has been evaluated to be considered a successful tool for eye tracking movements research [21]. However, calibration tasks have been reported to be unnatural, sometimes tedious, boring and tiring for the eyes [72]. They can also break the experience and are subject to deviations caused by common body movements or head rotations.

2.3.1 Calibration Games

There has been a motivation to reduce the tiresomeness of calibration tasks by turning them into calibration games [20, 51]. The approach is to gamify the process in order to make it more enjoyable as a game and effortless without affecting the acquired data. We refer to *gamification* as "the use of game design elements in non-game context" [13].

Flatla et al. [20] developed three games used for different types of trackers in order to create a design framework to help in the creation of calibration games. Their work starts with the premise that calibration is a necessary process so as to provide a good interaction experience and it is considered tiring and uninteresting. Their work analyzes the effect of game dynamics on calibration quality and suggests guidelines to guarantee it. They reviewed calibration types and isolated core tasks for each of them identifying the possible associated game dynamic.

By adding gameplay mechanics (*gamification*) they introduce the use of games for

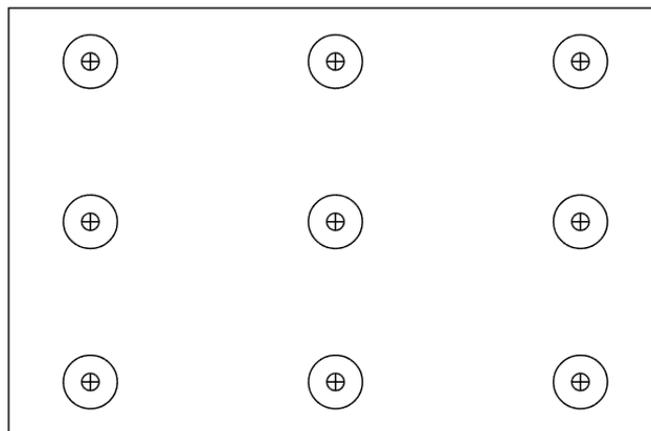


Figure 3: Schema for standard calibration task on the screen. Different targets appear (one at a time) spread around the screen in known positions. Users need to look at each target while they appear.

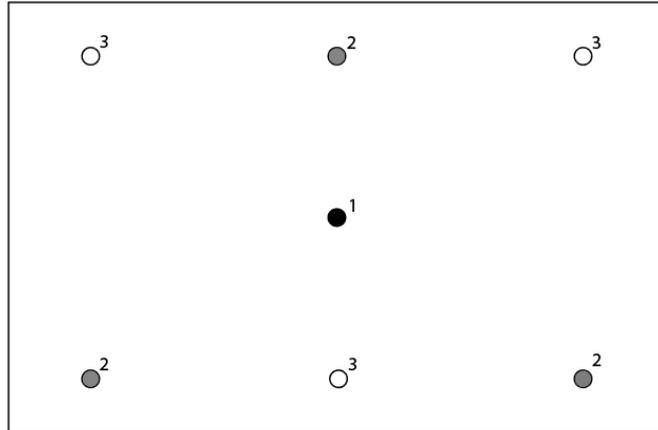


Figure 4: Schema for Tobii Calibration task on the screen. Different targets appear (in three turns, numbers shown) spread around the screen in known positions. Users need to look at each target while they appear until they 'pop' the dots.

calibration. The calibration task is transformed into a game that is reported as much more enjoyable than standard versions and able to produce accurate calibration data that is not compromised by game mechanics. However, calibration games are still using a defined isolated and external calibration task, independent from further interaction and can break the user experience.

Furthermore, *Dorr et al.* [15] integrated a gaze calibration task within a game to facilitate the turn-taking of players. In spite of the integration, the calibration task as such remained explicit and separate from the gameplay breaking the game experience. However, in a gaming context, it does not make sense to either use a calibration application before the game, or to have two separate applications, as these would interrupt the gameplay experience and immersion.

In order to tackle those limitations, research has been done fostering the study of the relationship between eye movements and motion. The expected behavior of the eyes helped the shaping of interfaces using gaze interaction and more flexible calibration tasks leveraged by the same principle.

2.4 Gaze and Motion

The relationship between gaze and motion broadens opportunities in gaze interaction research. Little research on using spontaneous gaze behavior has been done and only systems based on fixations and saccades eye movements are designed. The eyes can naturally be attentive to moving cues and cannot help but follow them without being aware with their gaze [38, 49] performing smooth pursuit movements. This neurological approach gives new systems the ability to detect when users follow objects in motion and widen the flexibility when designing novel interfaces using gaze interaction.

2.4.1 Smooth Pursuit eye movements

Smooth pursuit movements happen when the eyes try to fixate on a target that is in motion. When motion is presented, the eyes are capable to move smoothly and steadily when following the object of reference without performing saccadic jumps. This feature allowed new applications to use eye interaction in a new and compelling design context.

To differentiate smooth pursuit eye movements from fixations or saccades, the speed of the moving stimulus play an important role in its detection. *Holmqvist et al.* [28] suggest in their work guidelines to detect smooth pursuits. However, speed limitations can depend on the design of the stimulus. Their speed should be set as less than $1^\circ/s$ and a maximum of $30^\circ/s$ in which the eyes start performing catch-up saccades.

Eye interaction based on smooth pursuit eye movements allowed the creation of frameworks to select moving targets in dynamic interfaces. *Vidal et al.* [70] used such eye movements to detect which moving element in the interface was being selected by users' attention. Their approach is based on the correlation between the eyes and the target movement using *Pearson product-moment correlation*. If gaze movement matches the wanted item motion it is selected, providing more flexibility when designing new interfaces using gaze interaction.

In their research they designed different applications that used the continuity of pursuits to trigger different events. Examples vary from games, information navigation and other public displays applications, offering a robust selection and a spontaneous interaction. They were able to use gaze interaction independently of eye trackers calibration, demonstrating that spontaneous eye interaction is possible. Their contribution also highlighted smooth pursuit gaze interaction to be user independent. Different users could interact after another without the need to stop the application to perform the calibration task. This provided more flexibility as the system is not based on where exactly are you looking but how your eyes move.

Work by *Vidal et al.* [70] boosted the use of smooth pursuit eye movements in different interfaces as a new method for interaction. Examples like *Vspakov et al.* [57] or *Esteves et al.* [18] leveraged this attentive behavior of the eyes when following motion to provide hands-free control of widgets. *Esteves et al.* used a small device such as the smart watch proposing smooth pursuit interaction for selection. They present the inclusion of a little circular target object performing an orbital movement around each widget (or icon) as the target motion for pursuit interaction. By just looking at the small moving target, users are able to accurately select elements on the small smart watch interface without creating any hand occlusion as this method provides hands-free interaction.

Moreover, other uses of smooth pursuit eye movement have been presented to improve eye trackers calibration tasks design making them smarter and less tedious [49].

2.4.2 Smooth Pursuit calibration

Work by *Pfeuffer et al.* [49] changed the eye trackers' calibration method inspired by previous work on smooth pursuit behavioral gestures [70]. Their work developed from

the premise that humans cannot help but follow a moving target when is presented to them [38, 39]. In their approach, also using the *Pearson product-moment correlation*, they capture matching gaze and target positions considering them as a continuous sampling of calibration points. When a smooth pursuit movement is detected and matches the movement on the display, their method assumes that the object has been followed (see *Figure 5*). Furthermore, both the object in motion and uncalibrated gaze positions are paired and used to compute a new calibration profile by using a projective transformation (homography) using the sampled coordinates.

This approach showed that calibration with moving targets is feasible and can obtain good accuracy results, allowing less tedious methods and more flexibility for task design. Their work contribution presented different examples demonstrating that the integration of the calibration task is possible within the context of the interface. They also showed how the integration of smooth pursuit calibration inside of the interface dynamics remains invisible to users, providing compelling possibilities for novel interfaces design such as games.

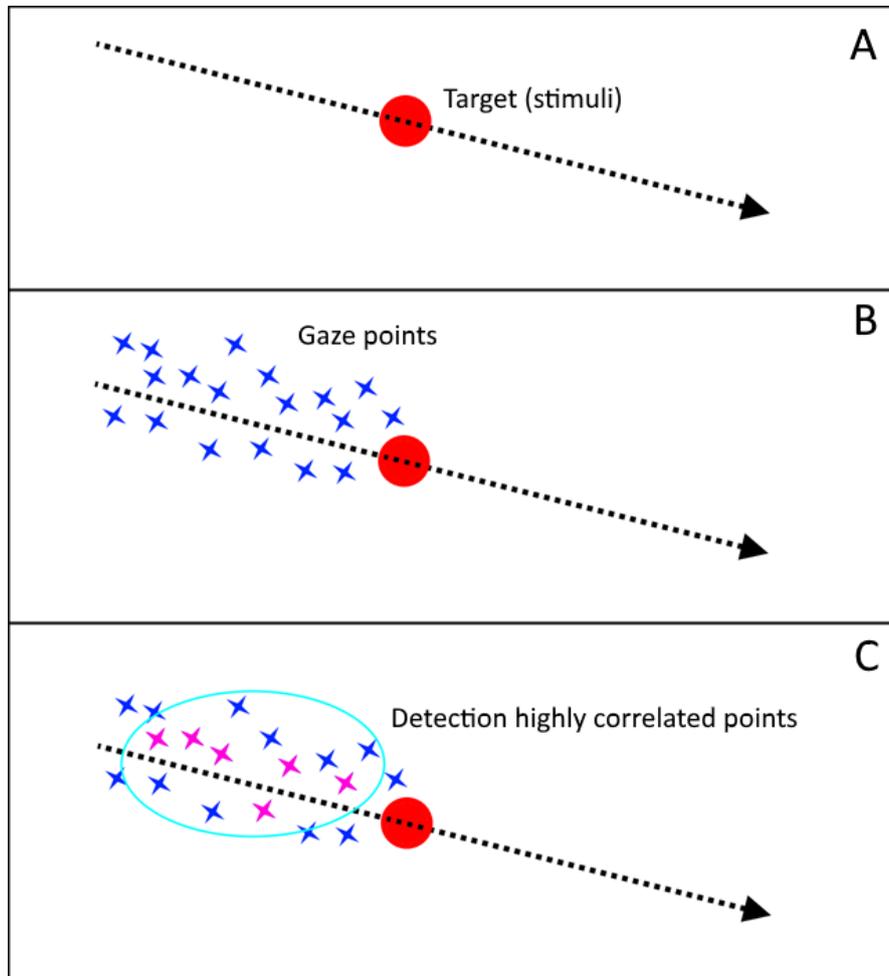


Figure 5: Smooth Pursuit detection for calibration. (A) The stimuli acts as the presented moving target. (B) Gaze follows the moving target and the points are stored. (C) Pearson's correlation is used to detect highly correlated points, and they are captured for later use during calibration.

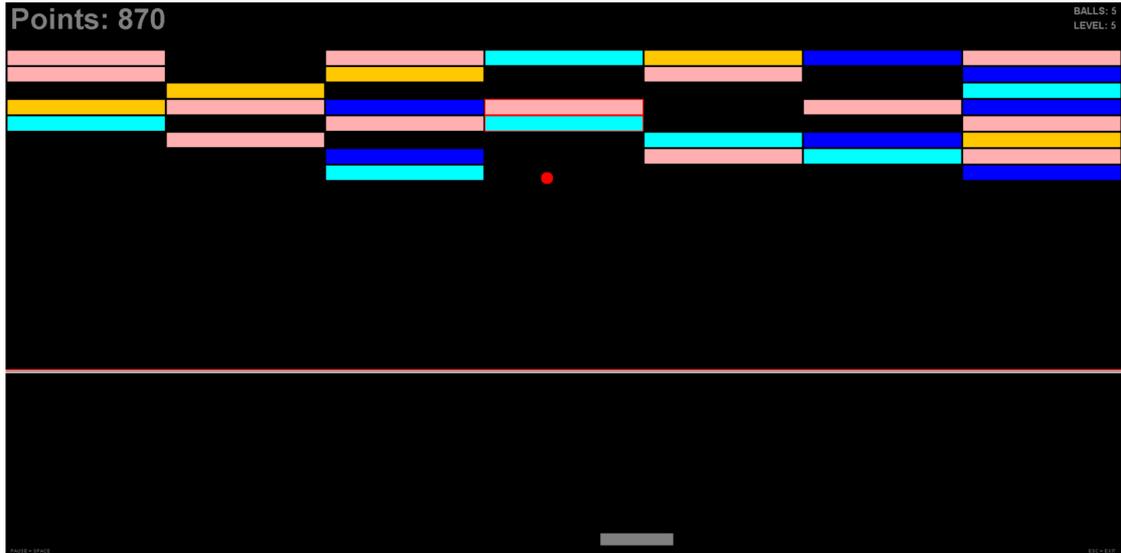


Figure 6: *GazeBall* is a gaze-enabled extension of *BreakOut*. The player’s expected gaze-following of the ball is used to sample points for continuous re-calibration of their gaze. Bricks provide power-ups that enable users to influence the ball’s direction with their gaze.

3. Smooth Pursuit re-calibration

Research using smooth pursuit eye movements leverages the spontaneous behavior of the eyes as interaction input allowing more flexible and accessible interfaces. However, the nature of such continuous eye movement is not fully understood. There is not enough information about precision or what is the concrete behavior of the eyes when following motion. Understanding the background of the movement will help in developing the next generation of interfaces using smooth pursuit eye movement as input.

Therefore, we proposed with *GazeBall* that the integration of gaze calibration inside games is possible by using continuous re-calibration based on smooth pursuit eye movements [49]. Our approach allows independence from the game design and enhances its dynamics by using unconscious gaze behavior for unaware re-calibration and playful interaction.

We followed work by *Pfeuffer et al.* [49] in which a smooth pursuit calibration task is embedded in the application context but separated from the end application containing gaze interaction. We agreed that smooth pursuit movements provided more flexibility when designing the calibration task, demonstrating being a good asset that could be integrated in the interface dynamics.

Accordingly, we hypothesized that smooth pursuit calibration can be performed implicitly and invisibly to the users; and it can improve and correct gaze estimation accuracy from possible deviations (like movement and change of position). Moreover, we hypothesize that such a task doesn’t need to be separated from the end task, and

that it can coexist with gaze interaction in order to solve the gap between experience and calibration.

3.1 Game Design

We developed *GazeBall*, a 2D retro-inspired videogame based on the popular Arcade game *Breakout* by *Atari, Inc.*. We used the original game dynamics implementing new ones based on gaze interaction, in order to influence the ball's direction and mirror it towards the position players look at. There is no need to initially calibrate the eye-tracker before playing the game as unaware continuous re-calibration, based on smooth pursuit movement, is implicitly integrated in ongoing game mechanics.

The *GazeBall* playing field consists of an automatically moving ball, and a paddle controlled by the player with either the keyboard or a gamepad, that could only be moved on the horizontal axis and is situated on the lowest part of the screen. Moreover there is a group of static rectangular shapes (the bricks) situated in different rows on the top part of the screen (see Figure 6), resembling the game target.

The aim of the game is to control and move the paddle so as to make the moving ball bounce on it in order to hit and break all the bricks. Depending on the point of the surface the ball touches along the paddle width, there is an influence on the angle of reflection used to bounce away (the closer to the edges the bigger the bouncing angle). Moreover, the ball can also bounce against the borders of the playing field and the bricks. When ball meets the lowest boundary of the game interface, a *life* is lost and the ball is situated in the center of the playing field.

The ball speed increases each level up to a maximum value and the amount of bricks is randomly established according to the same complexity increase criteria.

The bricks have two main features, strength (or resistance) and power, both set randomly. The former is represented by their color and is the amount of times that the brick has to be hit (up to 4) in order to disappear and add points to the player's score. The latter is whether they contain a power-up or not.

Power-ups are elements containing special game features and can appear from a brick when it is destroyed, falling down from it. If a power-up is collected by the paddle, gaze interaction event is triggered (see Figure 7), allowing users to influence the ball's direction towards their gaze point.

3.1.1 Gaze Interaction

We designed a power-up feature to be able to influence on the ball direction when it is going up, with player's gaze position. We based this interaction according to how players would play the game. The common strategic behavior of the eye in order to succeed in the game is to first look at the ball and later check the remaining bricks, as players wonder if they directed the ball towards any of them [37].

Interaction with power-ups during game play is illustrated in Figure 7, in which we can see how after destroying a brick, a power-up is released and when collected with the

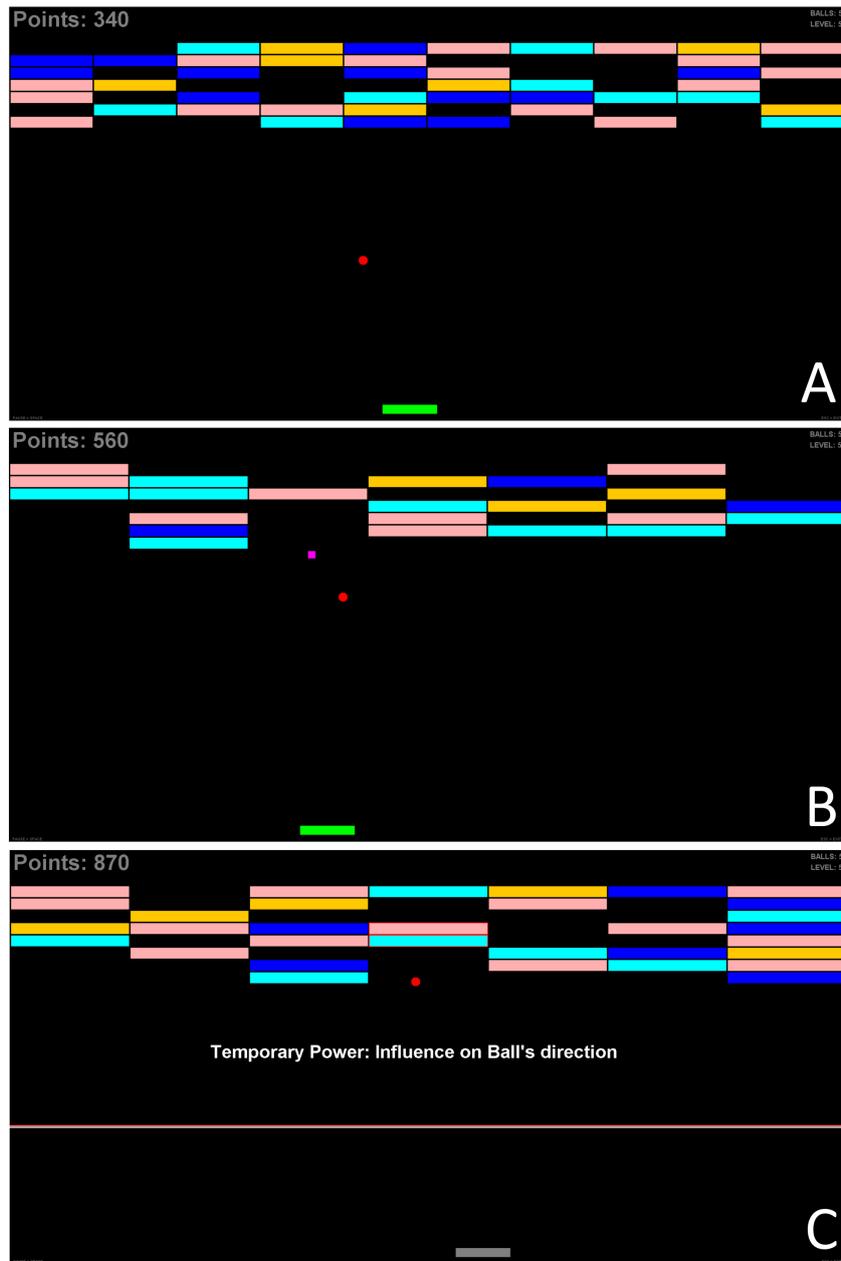


Figure 7: Game Interaction in *GazeBall*. (A) Players have to move the paddle to make the ball bounce so as to break the bricks. (B) Some bricks contain special power-ups that are released and fall down when the brick disappears after being hit. (C) If the paddle collects a power-up, the ability to influence ball's direction when it's going up with player's gaze, is triggered during 20 seconds (time associated to be able to make the ball bounce between 2 and 4 times). It also gives visual feedback by changing the paddle color into gray, the appearance of horizontal lines representing the scope of the gaze interaction (top two thirds of the screen), and showing during 2 seconds a text defining the power-up attribute ("*Temporary power: Influence on Ball's direction*").

paddle, it triggers gaze interaction to influence on ball's behavior.

Each power-up lasts for 20 seconds and gives visual feedback consisting of the paddle changing color, horizontal lines that represent the scope of the gaze interaction in the top two thirds of the screen, and a text defining the power-up attribute ("*Temporary power: Influence on Ball's direction*"). During all the game, the bricks are highlighted (see Figure 15) when the gaze point matches their position in order to help players understand gaze interaction.

When a power-up is triggered, players are able to influence the ball's direction angle towards the position of their gaze by just pointing. Influence on ball's behavior does not allow players to change its trajectory and move it to their gaze position, but mirroring (and flip) its motion direction on the horizontal axis according to it. If the ball was going to the left and gaze points on its right, the ball will change its direction and go to the right with the same angle (see *Figure 8*).

Therefore, the ball would move towards the players point of regard in the playing field. It would demand their full attention to look at the targeted position so as to keep the ball's trajectory until it hits either a brick or a boundary. If players do not focus on its target, the ball's direction would be constantly changing based on the gaze point, until it finds a boundary or brick and bounce back down. By not making the transition smooth and adding a "jittery" effect to the movement of the ball, we wanted to make obvious that it was being controlled by gaze and let players discover by themselves this novel interaction for the game without previous knowledge.

Gaze interaction adds an influence on how players act by not only demanding them to steer the ball, but also keeping their gaze focus on the bricks, changing their behavior completely. Moreover, as implicit gaze re-calibration is introduced, the more they play, the more they spontaneously follow the ball's movement, the more accurate is gaze estimation, and the more effective gaze interaction is.

3.2 Behavioral Continuous Re-calibration

When trying to succeed in playing *Breakout*, players cannot help but look constantly at the ball [15]. The eyes indicate players' point of regard and interest, they tend to look at the moving ball, as the moving target, in order to track it and predict which will be its position [37, 70] to make it bounce on the paddle.

By taking the unconscious behavior of eyes, we suggest an integration of continuous re-calibration method without compromising the game experience. We expect players would subconsciously be following the moving ball with their gaze, so this can be understood as a gaze-ball movement correlation and exploited for re-calibration. Therefore, we proposed a continuous re-calibration by using *Smooth Pursuit Calibration* [49], to detect when the user is attentive at the application and following the moving target, in order to re-estimate the gaze position without their awareness.

The calibration continuum offers the system the possibility of collecting highly correlated points between gaze and target movements during all the game play. It is

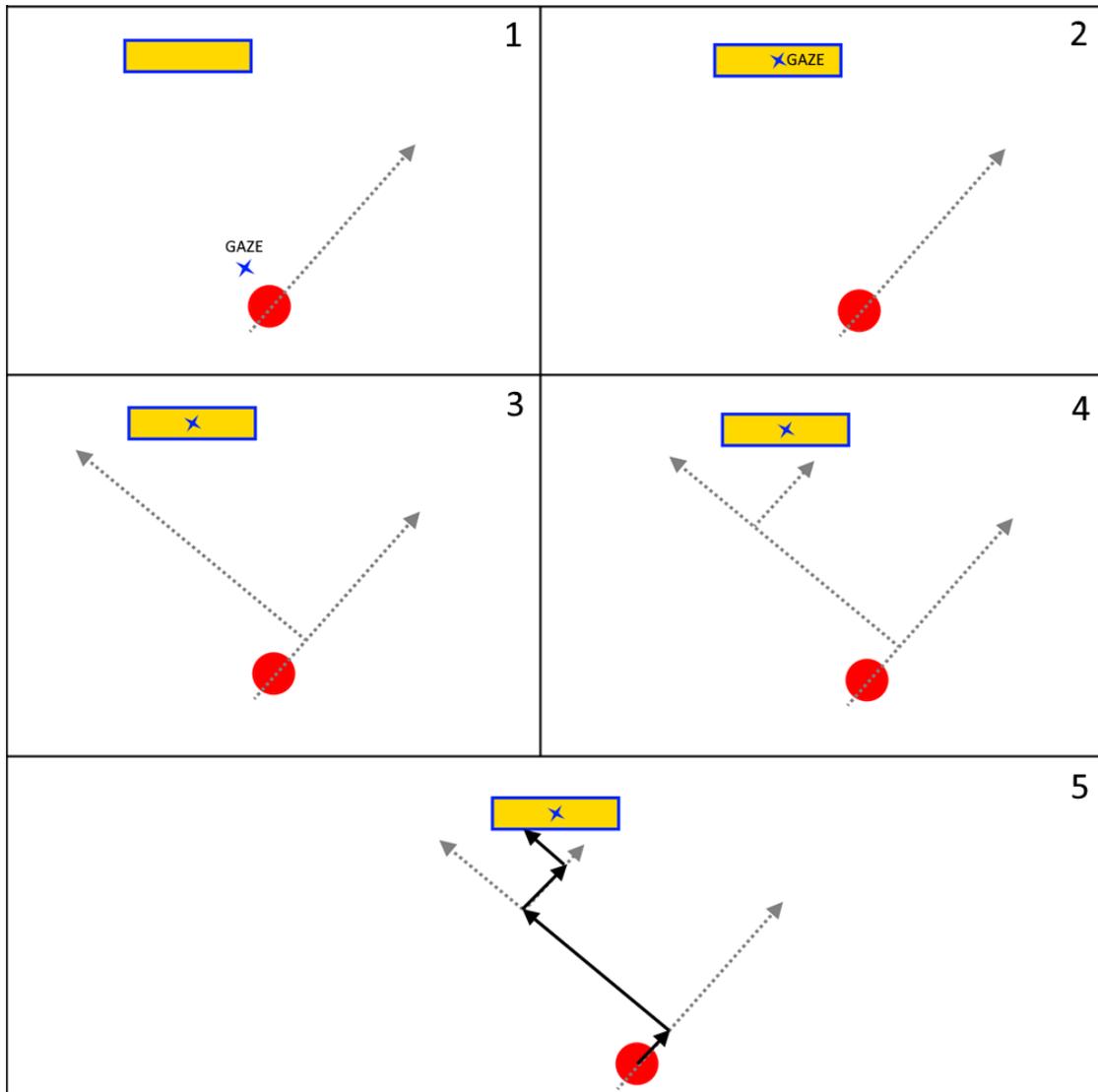


Figure 8: Gaze Interaction with power-ups. Once players made the ball bounce aiming it towards the top of the screen (1), they change their behavior and look up, aiming to the remaining bricks (2). The ball's potential direction changes towards the position of the gaze point and it is mirrored (3). This creates multiple changes of the ball's direction every time it overcomes and it is not going towards the position of players' gaze (4). Finally, the ball follows a "jittery" path that illustrates and emphasize the existence of gaze interaction to novel users.

able to compute the corresponding gaze estimation so as to update it automatically, in every frame, from possible deviations while playing. With this approach we can correct involuntary body movements involved in game play (such as body movement showing frustration, anger or celebration), without the need to stop the game and perform an isolated calibration task.

3.2.1 Implementation of the calibration

Pursuit calibration takes as input unfiltered gaze data coordinates from the eye tracker and the position of assigned game moving elements for calibration. Both are matched depending if they are correlated, and stored to compute a projective transformation (homography) that estimates the corresponding calibrated gaze coordinates.

By using *Pearson's product-moment correlation* between the eye movement coordinates and the moving object ones (the ball) in the playing field, the system can discriminate eye movements that match the moving ball (see *Figure 5*). Those are identified by windowing data samples while the game is being played. Moreover, a threshold in the correlation result determines whether players are actually following the target trajectory with their eyes. Successful coordinates points (both gaze and ball) are stored in a dynamic data set array for later use for re-calibration.

We established an extra threshold between correlation and storage in order to avoid false positives detected when performing similar gaze motion but not being looking at the ball. For those highly correlated gaze points we set a maximum distance between them of less than 200 pixels (5° of visual angle) to be considered in the estimation as it doubled the distance of a considered bad accuracy (2.5° ; being 1° good score). Moreover, the size of the moving window, threshold for correlation and minimum amount of stored data points to perform calibration were established according to *Pfeuffer et al.* [49] work.

Later, the stored data samples selected for the calibration are processed to estimate the new re-calibrated gaze point. Homographies represent the perspective projection and mapping of gaze in the output plane (screen). We obtained this geometric transformation by using computer vision algorithms to compute the homography. We used *RANSAC* (Random Sample Consensus) [19] for outliers' removal with a 0.1 projective error threshold.

Once the resultant homography is computed, it can be applied as a geometric transformation to every raw gaze point from the eye tracker to estimate the resulting calibrated gaze coordinate in the screen.

We perform the data acquisition continuously to compute new homographies every time we have a new set of points to use for re-calibration. In our game we used a correlated data coordinates set of 25 samples, defined by observation as the minimum amount of points to compute an homography (see *Figure 9*). It is used as a moving window that discard and takes new points every time that a new one is detected. To assure the quality of the resultant homography, we set a condition to check its transformation value not to be null or infinite. By using this condition we therefore assure that the amount and diversity of collected data was adequate to compute an accurate gaze estimation, dismissing the discarded homography and using the previous one.

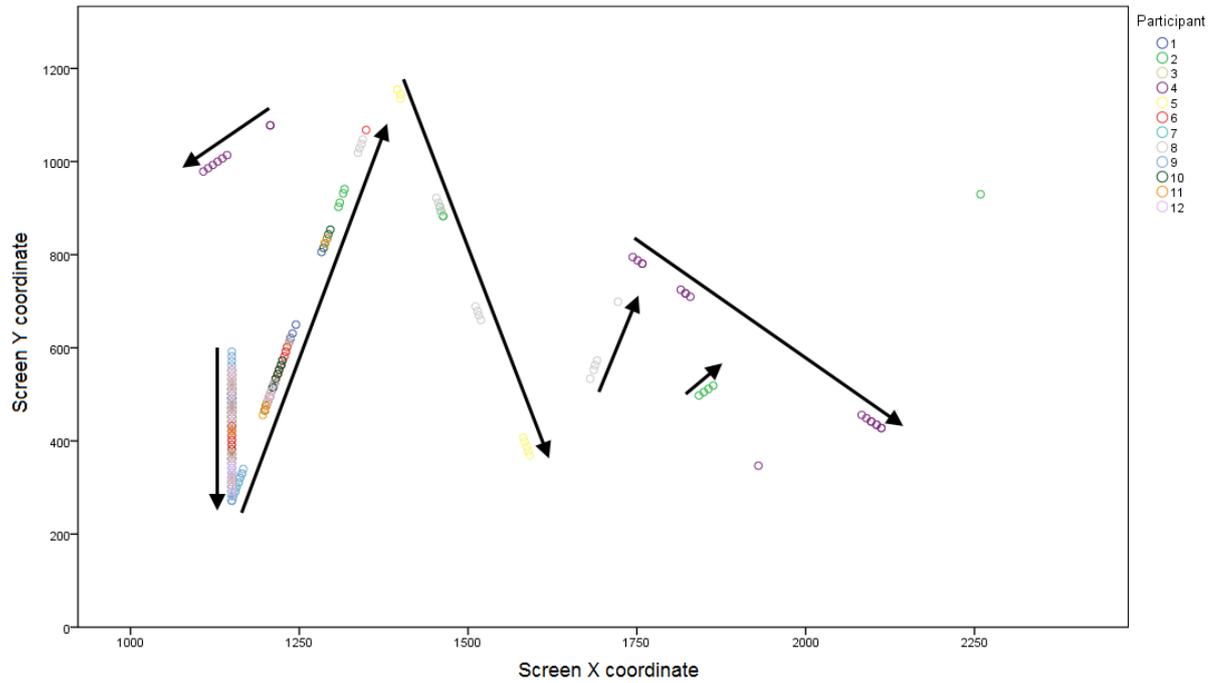


Figure 9: Example of the first 25 ball coordinates on the screen that were correlated with gaze and used to compute the first homography for re-calibration. Figure represents a portion of the screen in the x axis as balls first behavior was to bounce to the right. It shows how participants spontaneously follow the ball during its first behavior (falling down from the center of the playing field and after bouncing on the paddle or the bricks)

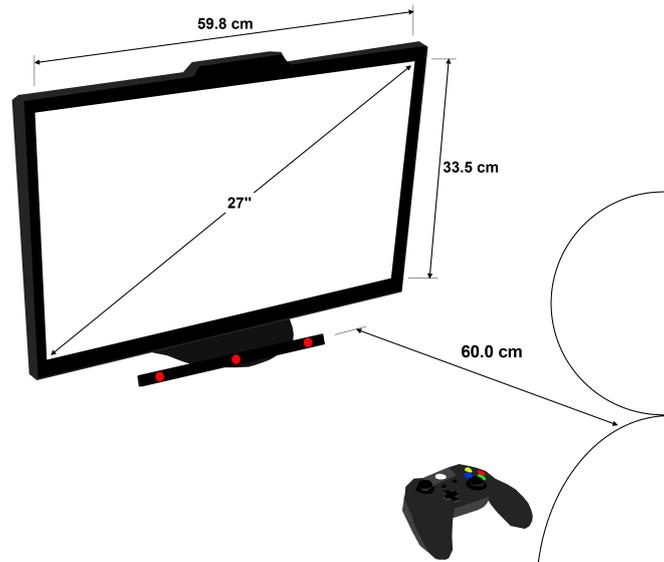


Figure 10: User Study Set-up schema, consisting of the eye tracker under the screen at 60 cm from participants and a remote controller.

GazeBall tries to assure a good gaze estimation and uses continuous re-calibrated gaze data points to enhance game dynamics. The use of the expected behavior of the eyes while playing the game makes gaze interaction possible.

GazeBall was developed in *Eclipse* using *Java* for all game dynamics, graphics and gaze calibration using *OpenCV* open-source computer vision library. The game requires an eye-tracker in order to track the player's gaze, and both game and eye-tracker were communicated with a *C#* application.

Finally, we introduced the use of a gamepad controller to interact with the application by starting, pausing or exiting the game and move the paddle.

3.3 Evaluation

Both the game experience and the calibration accuracy were evaluated in a user study so as to validate the novelty of gaze interaction, the game experience and the quality of gaze estimation considering the integration of eye tracking re-calibration.

3.3.1 Participants

12 volunteer participants between 21 and 32 years old ($\mu = 26$, $\sigma = 3$, 6 female), all students, performed the user study. 6 participants were familiar with eye tracking research, whereas the others were recruited from other fields.



Figure 11: Participant during the User Study playing GazeBall

3.3.2 Apparatus

We used a *Tobii EyeX* remote eye tracker, collecting data at 30Hz, situated in front of each participant under the screen and a *Microsoft Xbox One* remote controller to control the application (see Figure 19 and Figure 11). *Tobii EyeX* has been evaluated as a tool for research with smooth pursuit eye movements, showing reliable results [21].

3.3.3 Procedure

In the study, we explained the different stages that participants would encounter while playing with the application, levels and accuracy test tasks, and how to interact with the controller. No instructions were given about having to follow the ball while playing or how to use gaze interaction.

The study consisted of playing *GazeBall* during 6 levels, performing a gaze accuracy test between all of them. The accuracy tests were integrated into the game and appeared automatically after completing each level. Both fixations and smooth pursuit calibrated gazes were evaluated. We asked participants to look at 16 static points, equally distributed in the screen area, that appeared in random order for two seconds, in each accuracy test.

After signing a consent form and filling a demographics questionnaire, participants were asked to play the game during the first three levels, performing accuracy tests after each of them. After the three first levels, they were asked to pause the game.

During the pause, participants were asked to move from the experiment desk into another one and fill a questionnaire about the experience on Gaze Interaction. Later, they were asked to perform a new gaze accuracy test and play with the game during three more levels. At the end, they were asked to fill a new questionnaire on gaze interaction experience and calibration awareness. After the study a brief interview on the experience was performed.

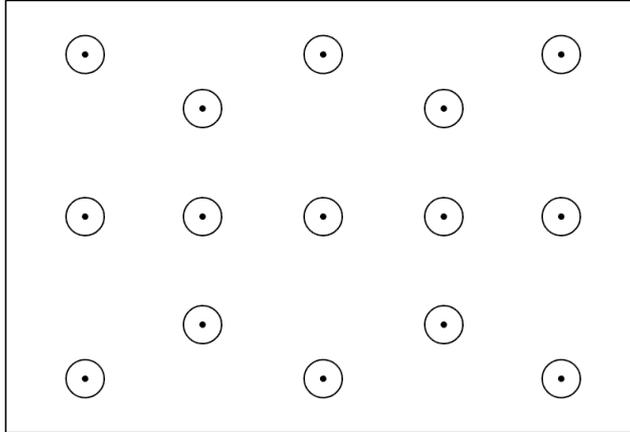


Figure 12: Accuracy test schema for GazeBall’s user study showing the 15 circular targets evenly distributed around the screen.

During the game play session, data logs were saved containing time-stamps; game events information (Detection of gaze correlations, Gaze Interaction or Free gameplay); the ball position; and the unfiltered and the re-calibrated gaze position with the accuracy test results.

The study lasted approximately 40 minutes and was subjected to each participant performance spending up to 6 minutes per level (maximum).

Results showed how the integrated gaze re-calibration was accurate and unnoticeable. Gaze interaction in the dynamics of the game was reported to be enjoyable, novel and easy to learn how to use, providing a new positive gaming experience.

3.4 Results

3.4.1 Calibration Accuracy

We asked participants during the user study to perform different accuracy tests in order to assess the quality of our integrated continuous smooth pursuit re-calibration. Accuracy is evaluated in degrees of visual angle, and computed considering the distance between the target and the gaze point, the screen resolution, the screen dimensions, and the users’ distance to the display.

Each accuracy test is composed of 15 circular targets (40 pixels radius with a marked center) evenly distributed around the screen (see *Figure 12*). Each target appeared on its own in a random order during a period of 2 seconds. The test consisted on fixating on each target while it was displayed.

We obtained the accuracy values by calculating the distance between the gaze point on the screen and the target point position on it. We dismissed the information obtained

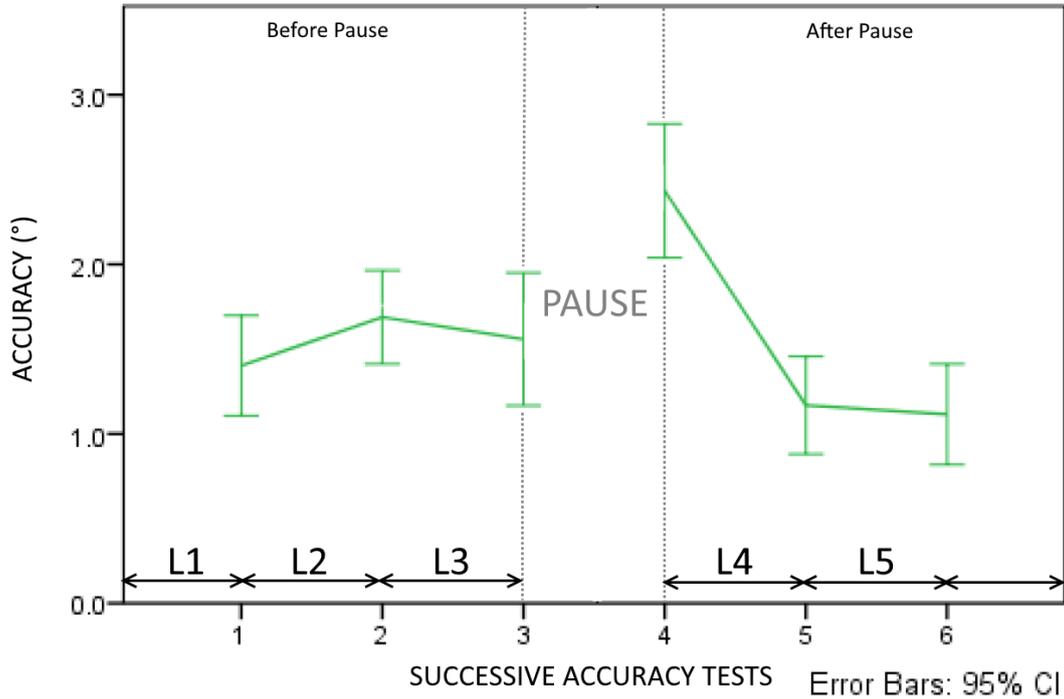


Figure 13: Mean Accuracy in degrees ($^{\circ}$) of visual angle for smooth pursuit based re-calibration method over the performed successive Accuracy Tests before and after the pause. Each test is performed right after playing each level and after the pause (when the user moved away and came back). Results showed that smooth pursuit re-calibration can correct deviations from changing positions scoring a mean ($\mu = 1.56^{\circ}$, $\sigma = 0.66^{\circ}$).

during the first 0.8 seconds and the last 0.2 seconds related to the time participants' eyes need to travel to reach the target and anticipatory movements [49]. The result for each accuracy test was computed by first taking the median distance from all the gaze points to each target position, and later the mean between all targets results.

Each participant played 6 levels, hence performed 6 different accuracy tests plus the one done after the break. A total of 7 results on accuracy were obtained but data from the last one was dismissed as more than half the number of participants failed when performing the test (mean failed targets: 4.75 ± 2.9), showing a lack of attention and lack of interest, as they knew that was the end of the study.

Results in Figure 13 show the resultant accuracy mean values after each test, including the one performed right after the pause. They show how the re-calibration method is subjected to drops on accuracy over time, especially when participants came back to play after the pause. Integrated continuous re-calibration showed improvement after the pause whereas standard one-time-calibration would not, as is not readjusted automatically. It would have the same decreases effect on its accuracy and would not be able to recover

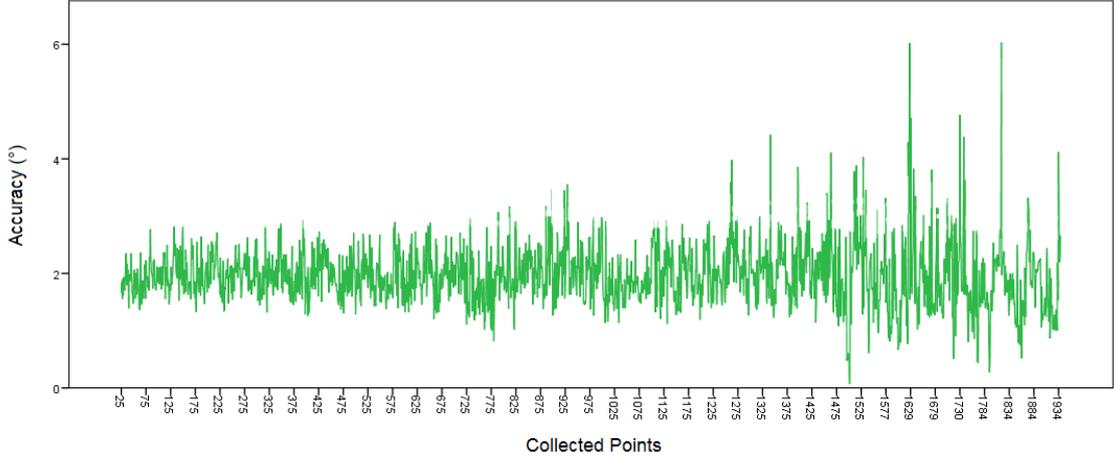


Figure 14: Results on the offline analysis of the evolution of calibration accuracy in $^{\circ}$ of visual angle according to the data points used as a time line in game playing. Smooth Pursuit re-calibration reported ($\mu = 1.98^{\circ}$, $\sigma = 1.09^{\circ}$).

unless the experience is stopped and another external calibration task is performed.

Therefore, Figure 13 shows how continuous smooth pursuit re-calibration reports auto-correction of accuracy ($\mu = 1.56^{\circ}$, $\sigma = 0.66^{\circ}$) being able to improve over time. Accuracy visual angles of 1° are considered good results, suggesting that our integrated implementation is able to perform a good approximation.

As our method is implemented to be re-calibrating continuously, in a later offline analysis based on the game data logs that we created during the study, we analyzed how accuracy evolved during the game playing session. Based on the 6 accuracy tests used for our analysis, we created a mean performance profile to compute the results. The profile was made of unfiltered gaze points per each target during the test. It was used to model each participant behavior during their session when performing accuracy tests, and was applied to our further analysis on accuracy evaluation.

We applied a sliding window of 25 samples to compute the homography that would estimate the gaze point. Applying the transformation to the created profile, it provided us with results on accuracy during all the study considering all correlated gaze and ball paired points collected (see Figure 14).

Results comparing calibration and re-calibration methods show how integrated and continuous smooth pursuit re-calibration had a high variability, but was automatically corrected and showed a good mean accuracy ($\mu = 1.98^{\circ}$, $\sigma = 1.09^{\circ}$).

3.4.2 Gameplay Experience

We based our results from the answers provided in the questionnaire participants filled out during the study, consisting of 5-point *Likert*-scale statements ($1 = not\ at\ all$, 5

= *very*). During the experiment the game without gaze interaction (control condition) wasn't considered as participants were familiar with the studied game dynamics.

Participants reported an evolution on how much they were enjoying playing using gaze in the game, before and after the pause from neutral positive experience ($\mu = 3.75$, $\sigma = 1.05$) to quite enjoyable ($\mu = 4.00$, $\sigma = 1.12$). This result is correlated with their perception on how easy to learn and use was gaze for interacting with the game, reporting and evolution from neutral positive easiness ($\mu = 3.75$, $\sigma = 1.42$) to quite easy ($\mu = 4.08$, $\sigma = 0.99$).

In terms of calibration, after the study we explained to participants how gaze and re-calibration was integrated in the game dynamics and asked them in the questionnaire if they felt being re-calibrated while they were playing. Results report that participants, without gaze interaction background, did not feel anything was happening during the game play (6 participants).

In terms of the game experience, we designed a game using gaze interaction that was reported to be enjoyable the more you played as the re-calibration was continuously being corrected at the same time without being noticed. Participants reported they felt the game was very innovative in terms of using gaze to interact as a special power and not to control the game. Furthermore, they reported considering the game more enjoyable and dynamic than original version. They also reported that gaze interaction was easy to learn how to be used without previous experience or explanation. However, most of them reported not having noticed the brick highlighting (see Figure 15) as they were immersed in the game experience.

3.5 Discussion

Our approach introduced the integration of continuous Smooth Pursuit re-calibration in ongoing game dynamics rather than designing a separate calibration task. It enhanced the game experience by considering players' common behavior when playing, to induce them to unconsciously re-calibrate the system.

Results showed how during the game session integrated re-calibration reported high variability with good mean accuracy. We believe that as players are constantly correlating their gaze with ball movement [15], we are acquiring new points that quickly correct such slight deviations in the accuracy estimation. Nevertheless, we can see in Figure 14 how variability increased its amplitude by the end of the last levels played. The more levels are played, the more challenging the game is, having more power-ups that triggered more times gaze interaction. Results on questionnaires show that the experience of using gaze in the game improved the more they played, meaning that they might have used more power-ups and were more exposed to gaze interaction. We can conclude that the last quarter of Figure 14 represents latest levels in which gaze interaction increased and players changed their behavior while playing. Samples of lower quality were obtained as players were no longer following the ball and focused more in looking at the bricks waiting for power-ups, and more time interacting with gaze as it is reported in our game logs.

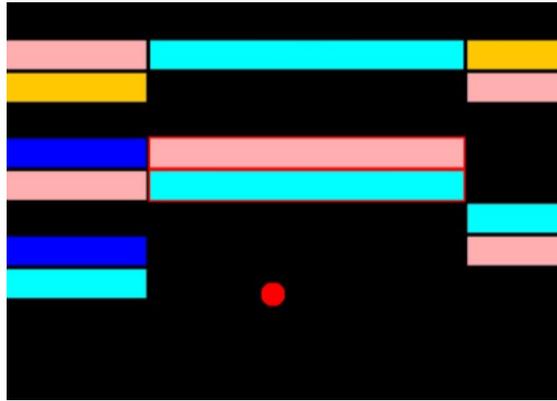


Figure 15: Zoomed capture of gaze interaction when looking at a brick, as it is highlighted in red.

On the other hand, results on calibration evolution were offline calculations as if we would have paused the game at that point to perform the accuracy test, and they do not represent the evolution of the study. Having such variability did not affect the quality of our re-calibration. (We consider that continuous smooth pursuit re-calibration mean accuracy approximates to good results). Moreover, participants reported in their questionnaires that they were not aware of being re-calibrated and they were enjoying the resulting gaze interaction.

Quality of calibration data is a priority in order to have a good game experience in a gaze-based interaction game. Results in Figure 13 suggest that introducing continuous re-calibration is beneficial, being continuously correcting the estimation of data while playing, removing the need to recalibrate using the 9 point gaze fixation calibration. There is no need to use an external application, pausing the experience, after a period of time.

The introduction of gaze influence in the game was reported to be felt as something that made the game easier. However, we did observations during each session that show how the more users played with gaze interaction, the more they focused on influencing the ball direction. Participants, while influencing the ball, "forgot" to move the paddle so as to make it bounce again, causing them losing the ball and a game life (reported in all game logs). Moreover, as participants were enjoying the use of gaze in the game, during repetitive times, they preferred to collect a power-up rather than making the ball bounce, losing a life and the power as the game was restarted.

Therefore, by introducing gaze as a power-up to be able to influence the ball behavior, we introduced the dilemma on whether picking a power-up and controlling the ball direction or following the ball with gaze to estimate where to place the paddle. Besides, we were able to turn the expected behavior of aiming at bricks into a novel gaze-based interaction mechanic to influence on the ball's direction. It was presented not to make the game easier or introduce a game advantage, but to enhance the experience by adding

new gaze-based mechanics that were able to influence on players' behavior. The more players learn how to use gaze in the game, the more they changed their behavior while playing. They turned from being constantly looking at the ball and quickly looking at bricks into keeping their gaze at targeted bricks so as to benefit from the new challenging dynamic.

3.6 Future Work

We believe that further implementations of our proposed continuous unaware re-calibration based on smooth pursuit eye movement need to be done by testing them in different game mechanics so as to validate the method.

Moreover, we would like to develop new game features in *GazeBall* based on the proposed gaze interaction. As it is only possible when having previously followed the moving target for later gaze re-calibration, we would like to introduce the concept of rewarding or not this calibration "reinforcement". We would like to aim the influence on players behavior to being induced to re-calibrate by adding two main features:

1. Reward players who follow (and correlate with gaze) power-up movements to make them bigger and be able to use gaze interaction during a longer period of time.
2. Sanction for players that during game play have not been following the ball in order to calibrate. It would make a brick immune to the ball hit when players are looking at them rather than following the ball, hence the brick would not be destroyed.

3.7 Conclusion

We successfully implemented in *GazeBall* new game dynamics by introducing gaze interaction in a special power-up, being able to influence the ball's direction. The use of enjoyable gaze interaction is achieved by the integration of an unaware continuous gaze re-calibration based on smooth pursuit eye movements when following a moving target during ongoing game dynamics. Gaze integrated re-calibration does not compromise the game experience and it reported good accuracy results. Moreover, the continuous re-calibration helped preventing an unsuccessful game experience when tracking and estimating gaze while playing. It also introduced an automatic correction without the need to stop the game and re-calibrate gaze in an external application, letting players enjoy the experience.

GazeBall was not created as a gamified gaze calibration task. It integrates re-calibration in ongoing game dynamics without affecting interaction but enhancing game enjoyment. It introduced both novel interaction and naturalness, as it uses gaze to interact with the game features based on how players would commonly behave and use their gaze while playing. Interaction mechanics were able to lead to an easy to learn gaze-based interaction, capable of influencing players' own behavior.

GazeBall differentiated between playing and calibration. It was not the creation of a calibration process, but the integration of such re-calibrating method in already existing

game dynamics. Re-calibration does not prevent the game to work but enhances it being both developed independently. Moreover, as re-calibration was only performed when users followed a trajectory, the eye and target paired movement correlation created a balance between play and interaction; the more you play, the more you calibrate; the more you interact, and the more you enjoy.

However, results from the study using GazeBall do not show evidence of how accurately the eyes are able to follow movement. We assumed that the correlated gaze was following the target's trajectory and used this data for re-calibration. We cannot be certain that the results were just a coincidental synchrony, the eyes could be performing a similar movement without being directly looking at the target motion, creating an deviated result.

We did not have any information on how easy was to follow the movements or which were the characteristics of it to get good results for re-calibration. Moreover, getting an insight on the precision of the matched movements would help to further develop and improve smooth pursuit as calibration method. Further research to understand the nature of continuous eye movements could lead to better integrated calibration algorithms or even integrated gaze estimation accuracy tests that use smooth pursuit movements.

The understanding of the spontaneous behavior of smooth pursuit eye movements with movement on-screen was targeted during a second user study.

4. A Further Insight on Smooth Pursuit eye movements

Our contribution with *GazeBall* demonstrated that the integration of the eye tracker calibration task is feasible by leveraging the spontaneous behavior of the eyes. However, how smooth pursuit eye movements are performed while interacting with dynamic interfaces is not fully understood. Gaze interaction using smooth pursuit movements has been designed regardless of calibration, thence no knowledge about precision or what is the concrete behavior when following motion has been considered.

4.1 Motivation

Previous work on calibration assumed that the matched movement belonged together without acknowledging how accurate the sampled gaze point on the screen is. The aim of the study was to investigate how precise smooth pursuit eye movements are when predicting where gaze is looking at.

Given that pursuits calibration [49] was design for effective calibration considering uncalibrated gaze points, we wondered if we can be sure of where the user is looking when following movement and how can we use their behavior to create a smarter calibration. This study became an exploration of smooth pursuit eye movements to comprehend how current algorithms for detection of the movement and calibration could be improved. We wanted to understand what the eyes behavior is, and if smooth pursuit is a good position estimator. Additionally, we wanted to find if there is a systematic offset between the moving target and gaze position when following different target directions and speeds.

Moreover, we will investigate which are the movement features that best favor smooth pursuit detection and provide useful data to improve the smooth pursuit re-calibration method. The detection of smooth pursuit eye movements will help us determine which movement features can guarantee an accurate prediction of where gaze is looking.

We hypothesized that the speed and direction considered for the design of the stimuli can modulate gaze behavior and matter when detecting and using smooth pursuit movements.

4.2 Evaluation

In order to understand how gaze can adapt and it evolves when following an object in motion performing smooth pursuit movement we asked participants to follow a moving target with their gaze. We designed the movement as a closed path, creating a shape formed loop (square, circle, diamond and random shape based on four randomized Bezier curves) in order to contain the maximum number of possible directions (see *Figure 16*). Every shaped path had a similar size (600 pixels of travel per quarter of the shape or diameter) and was situated in the center of the screen. The horizontal and vertical directions were encapsulated in a square-shaped motion loop. The different diagonal directions were contained in the diamond shape, and circular movement was included in a circle shape (using both clockwise and anti-clockwise movements separately). The

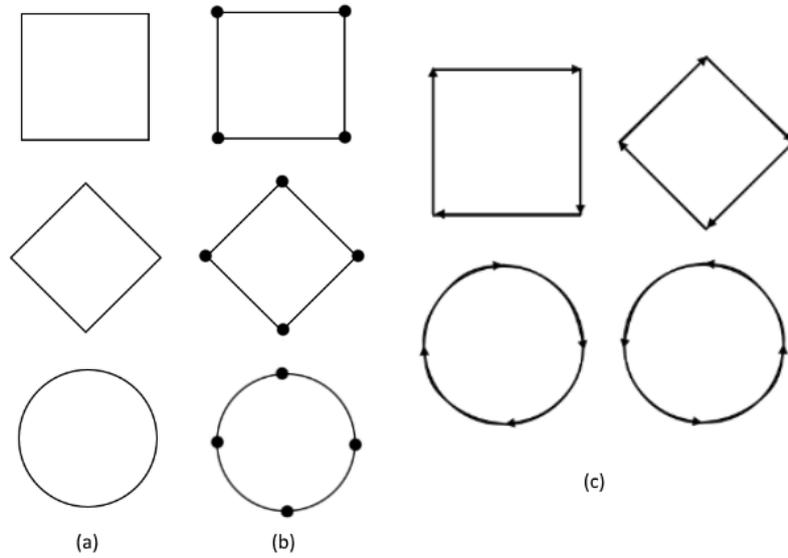


Figure 16: Target motion Directions design. (a) Abstract representation of groups of direction during motion paths as the shaped loops. Square, Diamond, Circle and Random shape (not shown, combination of all based on four random Bezier curves). (b) Different possible target starting points in each shaped motion loop. (c) Representation of different directions inside each shape loop during target motion. Horizontal (Left to right and Right to left), Vertical (Top to bottom and Bottom to top), Circular (Fragmented in quarters, clockwise and anti-clockwise direction).

random shape was a combination of the others. The target was presented as a filled circle (10 pixel radius) and could start its movement from a randomized starting position inside each group of directions closed loop (four possible points in each quarter of each shaped path).

Furthermore, in the study we tested the influence of speed when performing smooth pursuit movements. We used 6 different constant speeds, measured in $^{\circ}$ (degrees) of visual angle per second, ($1.5^{\circ}/s$, $6^{\circ}/s$, $12^{\circ}/s$, $18^{\circ}/s$, $24^{\circ}/s$ and $30^{\circ}/s$) so we could cover all range of target speed between performing eye fixations and saccadic movements [28]. Both target and gaze position were recorded.

4.2.1 Participants

12 volunteer participants, between 20 and 40 years old ($\mu = 27$, $\sigma = 5$, 6 female), all students, performed the user study. Only 4 participants were wearing glasses.

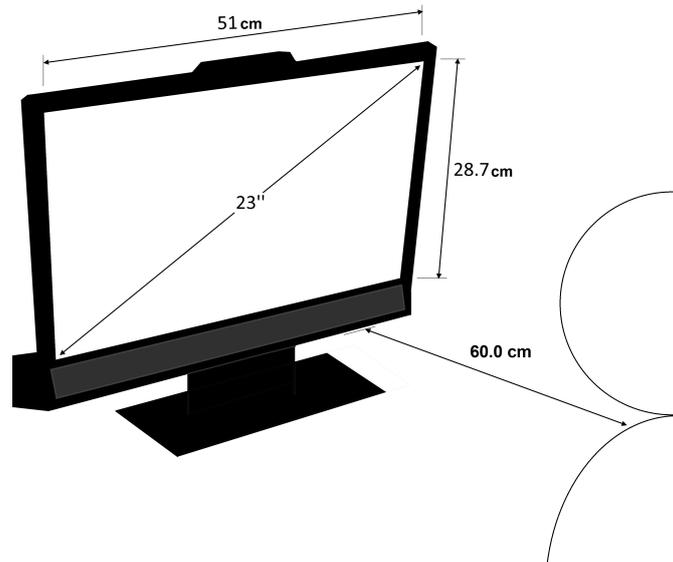


Figure 17: User study set-up, using Tobii Pro TX300 screen-based eye tracker integrated under a 23” monitor at 60cm away from participants.

4.2.2 Apparatus

We used a Tobii Pro TX300 screen-based eye tracker integrated under a 23” monitor, capturing gaze data at 300Hz, in front of each participant at 60cm (see Figure 17).

4.2.3 Procedure

During the study, we asked participants to perform the task consisting of following a moving circle displayed on the screen with their gaze. There were 30 randomized trials in each round (corresponding to the 6 different speeds used in the 5 distinct closed loop paths). Between each loop there was a 3 seconds interval used as a break to rest the eyes. Two rounds were carried out for each participant to complete the study. Before each round, the eye tracker was calibrated to the participant’s features and a gaze accuracy test was performed to avoid a drop in the quality of estimated gaze data between rounds. After each round, another gaze estimation accuracy test was performed, in order to assure that the accuracy was maintained during all the trials composing the round.

Prior to the experiment participants received information about the study and signed the consent form. Then, they completed a demographics questionnaire.

During the gaze tracking calibration process we used Tobii Engine calibration system based on fixations, using 24 points spread around the screen to try to obtain the higher grained accuracy possible.

The study took proximately 30 minutes and both gaze position on the screen and target position information were recorded for every participant trial. Further, gaze

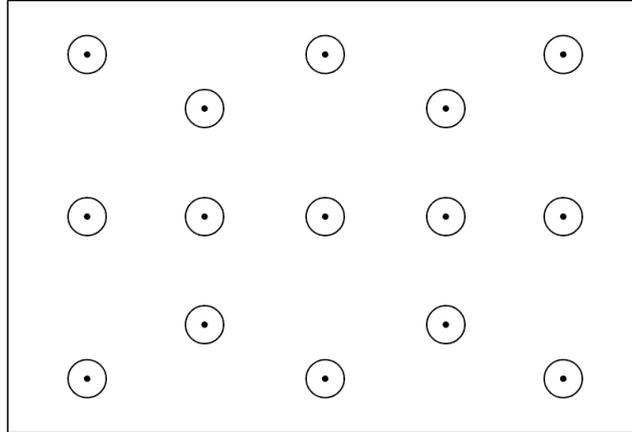


Figure 18: Accuracy test schema for the second study showing the 15 circular targets evenly distributed around the screen.

estimation accuracy information and scores were stored after each accuracy test.

4.3 Results

4.3.1 Calibration Accuracy

During the study we aimed to maintain the estimated gaze accuracy level between rounds. By assuring that there was no drop in the accuracy level could help to determine whether the recorded gaze data was compromised or not. To date smooth pursuit eye movement has been used in HCI regardless of calibration or gaze accuracy level. However, with an accurate gaze estimation we can understand how precise are smooth pursuit eye movements when detected.

Each accuracy test is composed by 15 circular targets (40 pixels radius with a marked center) evenly distributed around the screen (see *Figure 18*). Each target appeared on its own in a random order during a period of 2 seconds. The test consisted on fixating on each target while it was displayed.

Participants were asked to perform two different accuracy tests, one before and one after each study round. Accuracy is evaluated in degrees of visual angle, and computed considering the distance between the target and the gaze point, the screen resolution, the screen dimensions, and the distance to the display.

We obtained the accuracy values by computing the distance between the gaze point in the screen and target point position on it. We dismissed information during the first 0.8 seconds and the last 0.2 seconds related to the time participants' eyes need to travel to reach the target and anticipatory movements [49]. Results for each accuracy test are computed by first taking the median distance from all the gaze points to each target

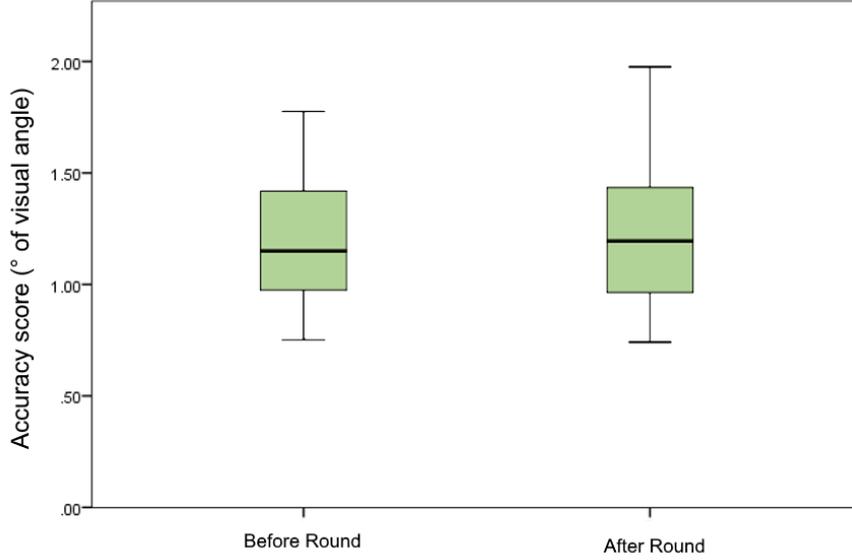


Figure 19: Box plot showing accuracy test score results before and after study rounds. Previous to the tests mean accuracy score was $\mu = 1.21^\circ$, $\sigma = 0.34^\circ$, after the study the score was $\mu = 1.23^\circ$, $\sigma = 0.33^\circ$.

Table 1: Accuracy results for different areas of the screen

	<i>Before</i>	<i>After</i>	Mean
<i>Center screen</i>	$\mu = 0.95^\circ$, $\sigma = 0.32^\circ$	$\mu = 1.04^\circ$, $\sigma = 0.43^\circ$	$\mu = 0.99^\circ$, $\sigma = 0.38^\circ$
<i>Left-Right Margins</i>	$\mu = 1.31^\circ$, $\sigma = 0.41^\circ$	$\mu = 1.29^\circ$, $\sigma = 0.34^\circ$	$\mu = 1.30^\circ$, $\sigma = 0.37^\circ$
<i>All screen</i>	$\mu = 1.21^\circ$, $\sigma = 0.34^\circ$	$\mu = 1.23^\circ$, $\sigma = 0.33^\circ$	$\mu = 1.22^\circ$, $\sigma = 0.33^\circ$

(point cloud) position, and later the mean between all targets results, to get a score around all the screen. Accuracy scores around 1° visual angle are considered good results.

Results in Figure 19 show how gaze estimation’s evaluation evolve from before to after the study. We can observe that there is no much difference between the accuracy scored at the beginning of the study ($\mu = 1.21^\circ$, $\sigma = 0.34^\circ$) with the value obtained at the end ($\mu = 1.23^\circ$, $\sigma = 0.33^\circ$), resulting in a mean accuracy of $\mu = 1.22^\circ$, $\sigma = 0.33^\circ$ during the experiment based on the whole screen.

Moreover, as in our study design we decided to display all the shaped loops in the center of the screen, in a further analysis we checked which was the overall accuracy in the area of action, by excluding the accuracy points that were situated outside the center of the screen (Left and Right margins). Results in Table 1 show how in the center of screen we scored a mean accuracy lower than 1° of visual angle ($\mu = 0.99^\circ$, $\sigma = 0.38^\circ$), demonstrating that the data acquired during the study was not affected by inaccuracies from the tracker and results in further analysis are valid.

Table 2: Pearson’s Correlation mean coefficient showing the effect between Direction and Speed

	1.5	6	12	18	24	30 (°/s)
<i>Horizontal</i>	0.93±0.13	0.90±0.17	0.85± 0.21	0.86±0.16	0.81±0.16	0.75±0.17
<i>Vertical</i>	0.94±0.12	0.91±0.14	0.83±0.19	0.69±0.28	0.58±0.31	0.51±0.33
<i>Diagonal</i>	0.89±0.19	0.80±0.27	0.76±0.28	0.72±0.27	0.66±0.26	0.64±0.24
<i>Circular</i>	0.93±0.15	0.89±0.19	0.82±0.24	0.77±0.25	0.72±0.25	0.66±0.25
<i>Random</i>	0.9±0.11	0.93±0.13	0.87±0.17	0.83±0.19	0.75±0.22	0.71±0.23

4.3.2 Smooth Pursuit Performance

We acquired gaze position data during the whole study and filtered out the points that were outside the screen (Resolution: 1920x1080). Smooth pursuit, against saccadic eye movement, is characterized by the continuity of the motion that happens when gaze follows a moving target. Based on previous research on smooth pursuit eye movement interaction [49, 70] we could determine when a series of eye points are performing smooth pursuit, by using Pearson’s correlation. Those points which correlation coefficient was higher than 0.7 were considered highly correlated and tagged as smooth pursuit eye movement points [70].

We used Pearson’s correlation using a moving window of 30 samples on our filtered eye data. *Table 2* and *Figure 20* show the effect that direction and speed have in the resulted mean correlation coefficient between target and gaze movement. In the first range of speeds (slow speeds: 1.5, 6 and 12 °/s) correlation coefficient scored higher mean values than for faster speeds (18, 24 and 30 °/s), showing that smooth pursuit eye movements are harder to perform in high speed conditions, and suggesting that pursuits are good indicators of gaze position for the lower range of speeds.

A two-way repeated measures ANOVA was run to determine the effect of different directions over speed on the correlation coefficient values. The main effect of direction showed a statistically significant difference in gaze correlation $F(2.199, 24.187) = 132.734$, $p < .001$. Moreover, speed also showed a statistically significant difference in gaze correlation $F(5, 55) = 336.633$, $p < .0005$. Results showed that there was a statistically significant interaction between directions and speed on gaze points correlation, $F(20, 220) = 45.633$, $p < .0005$.

Therefore, simple main effects were run. Correlation coefficient results were not statistically significantly different for Horizontal movement (0.91 ± 0.15) compared to Random one (0.92 ± 0.15), with $p = .098$. Similarly, there was no difference between Vertical (0.89 ± 0.19) and Diagonal (0.84 ± 0.23) movements, with $p = 1$. Circular movement was significantly different to the rest of directions, with $p < .001$. Vertical, Circular and Random movements showed to be dependent on speed, whereas Horizontal and Diagonal showed less difference between consequent speeds.

Further data analysis was performed only on the detected smooth pursuit eye movements. Results in *Figure 21* show how for all directions around 50% of the samples were discarded (correlation coefficient less than 0.7). This suggest that smooth pursuit

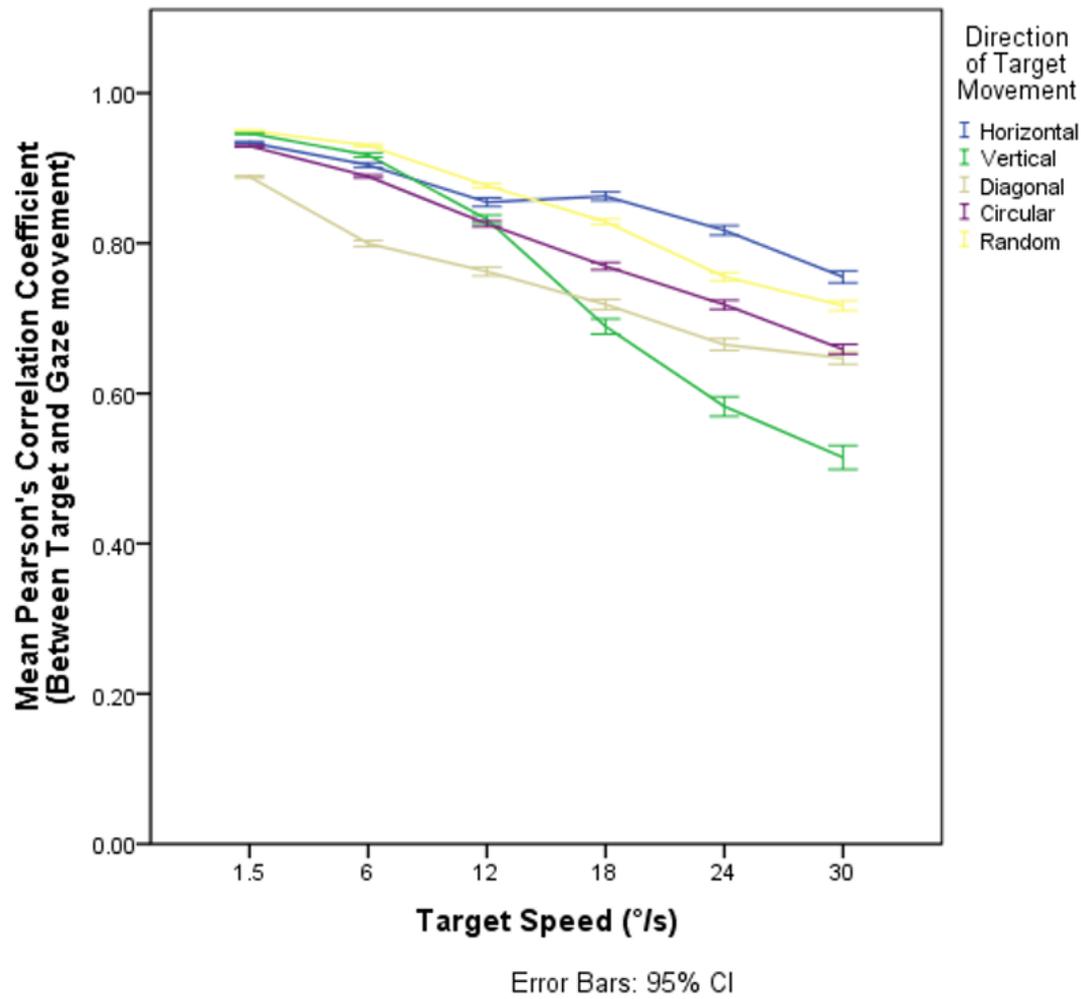


Figure 20: Pearson's correlation coefficient evolution depending on speed per each different type of movement for filtered values of eye movements.

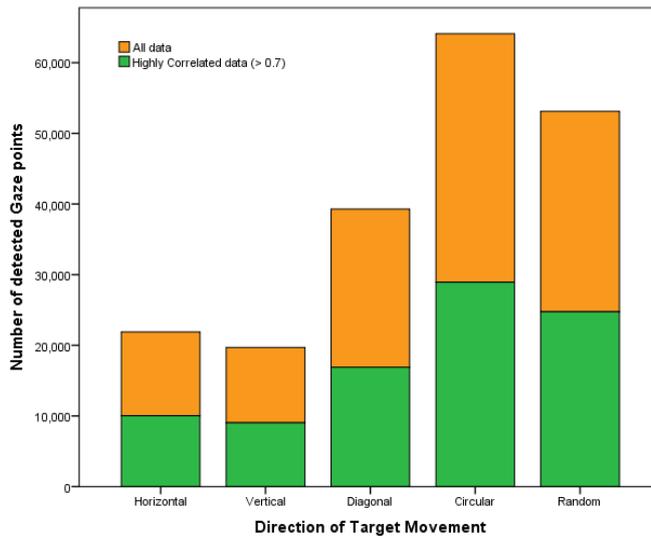


Figure 21: Amount of filtered gaze data point detected versus the amount of filtered points that were highly correlated ($p > 0.7$), corresponding to smooth pursuit eye movements.

detection is a good filter for accurate gaze points.

A new two-way repeated measures ANOVA was run to determine the effect of different directions over speed on the distance between target and gaze points. The main effect of direction showed no statistically significant difference in gaze and target distance. Moreover, there was no statistically significant difference between speeds. However, the difference between slower and faster speeds, resulted significantly different results, with $p = .027$. Finally, there was no statistically significant interaction between directions and speed on gaze and target points distance.

As we did not find any statistically significant effect, in a further analysis we decided to divide our generalized motion direction into different segments of motion and directions inside each movement. Horizontal motion was divided in two directions (Left to right and right to left), Vertical also in two (top to bottom and bottom to top), all being the four different segments of the square path movement displayed to users. Accordingly, Diagonal and both circular motions (clockwise and anti-clockwise) were divided into four different segments of direction. Random direction was discarded from further analysis as it was very difficult to assess the different motion segments, as they were changing between trials.

Further analysis will help to understand what the behavior of gaze is when performing smooth pursuit eye movements. Our aim is to know which is the position of gaze, and its distance to the target point according to the presented movement features when detecting pursuits. Although smooth pursuits are suggested to represent a good estimator of the gaze position, we want to understand its behavior when following motion.

4.3.3 Smooth Pursuit Motion Analysis

We analyzed per each possible fragment of motion which was the mean distance between the estimated gaze and the target point. We used aggregation via geometrical center by using each target (stimuli) position as the reference. We normalized all paired coordinates (target(x,y) and gaze(x,y)) and translated them to coordinate (0,0) according to their reference and have them superposed for better understanding of the behavior of gaze during the segment (see *Figure 22*). The results represented the computed distance (both in x and y axis) in $^{\circ}$ of visual angle. The results were represented in a scatter plot, showing gaze points cloud, its mean point and standard deviation.

Results in *Table 3* and *Table 4* show the mean distance ($^{\circ}$ of visual angle) between target and gaze points during each segment of motion direction. We can observe how the distance (offset) between target and gaze positions increases as the speed does. Accordingly, slower speeds scored a better accuracy to the target point when following its movement. Overall, results suggest how smooth pursuit is an accurate predictor of where the gaze is looking, scoring a mean distance of around 1° for most of the tested speeds. For the fastest speed and vertical movement, results suggest how gaze cannot follow the presented movement accurately.

We computed the relative position of the mean gaze point adopted according to target position during every segment of the motion. *Table 5* and *Figure 24* show how segments of direction involving descending motion reported gaze trajectory being ahead from target movement. Horizontal (from left to the right) movement also showed such anticipatory behavior of the eyes when following target movement for all speeds.

Figure 23 shows how the mean gaze position is advanced ($0.85 \pm 1.21^{\circ}$) regarding target points during the performance of anti-clockwise circular movement (quarter of the circle going down to the left) at $18^{\circ}/s$.

On the other hand, movements involving ascending motion or horizontal motion to the left reported gaze trajectory to be delayed from target movement for slower speeds ($1.5^{\circ}/s$, $6^{\circ}/s$ and $12^{\circ}/s$), whereas for the faster range behavior changed to be onward target trajectory. *Figure 26* represents the scatter plots for Horizontal direction segment going from right to the left at both speeds $6^{\circ}/s$ and $24^{\circ}/s$. We can see how for the lower speed, gaze mean point (with distance $0.42 \pm 1.51^{\circ}$) show a delayed behavior on its movement when following target's trajectory. Faster speeds, show how the gaze point (with distance $0.83 \pm 2.36^{\circ}$) changed its behavior and moves being ahead from the target trajectory.

Although we would have expected gaze to be delayed regarding target's trajectory as gaze is following such movement when performing smooth pursuits, we found a very different behavior (see *Figure 26*) that suggests that there is a prediction of the target movements and a tendency to overcome its position and be advanced to the motion. However, we found that for certain segments of direction (descending or to the left) gaze position remains delayed, but only at slow speeds.

In order to better understand the behavior of gaze based on our results, we took pieces of the motion segments and represented both target and gaze positions (without

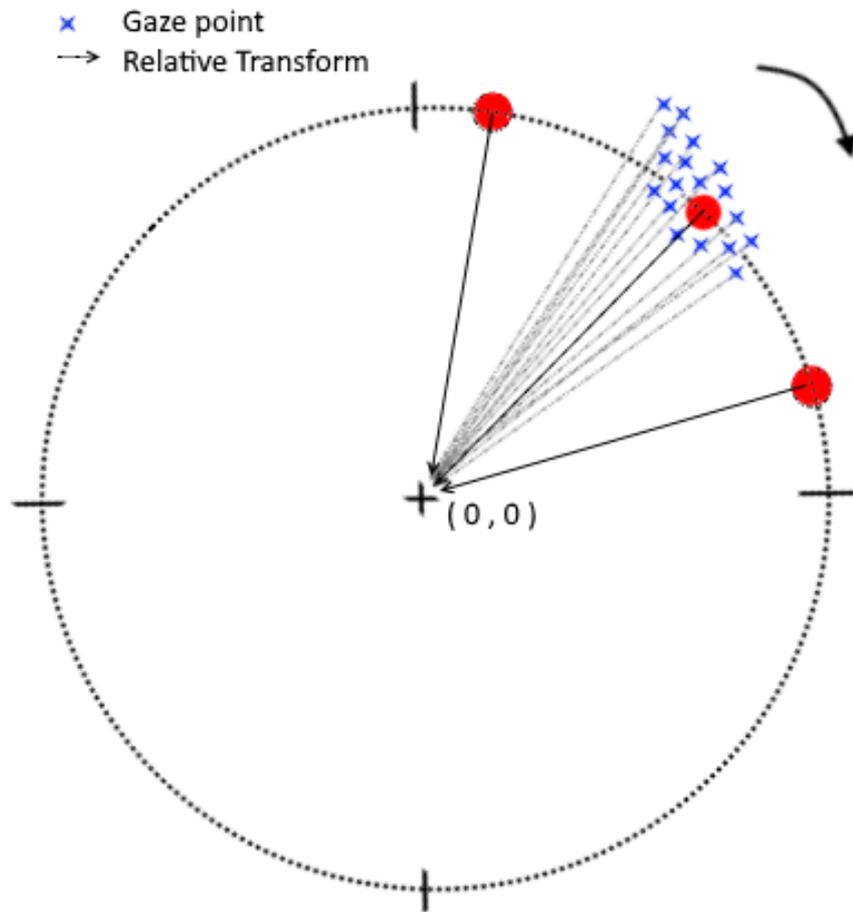


Figure 22: Methodology for aggregation of paired gaze and target points via geometrical center. The direction shape is divided by segments (example circle going clockwise). Each stimuli point is considered referenced and it is normalized and translated to the coordinate $(0,0)$ together with the corresponding gaze points. Resultant figure shows a superposition of all the points showing the mean target-gaze distance and standard deviation (see next page).

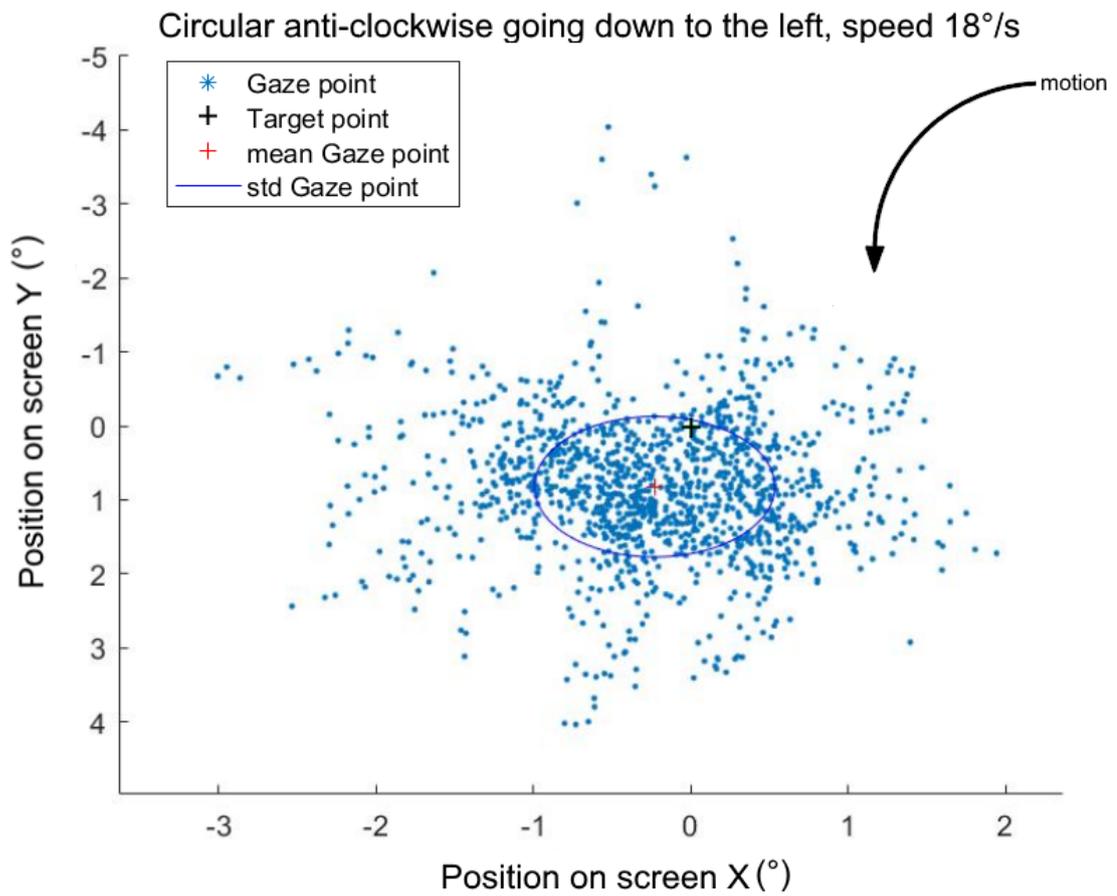


Figure 23: Anti-clockwise circular movement scatter plot, belonging to the direction segment corresponding to the quarter of the circle going down and to the left at speed $18^\circ/s$. Gaze mean point against target points position (with distance $0.85 \pm 1.21^\circ$) show gaze motion behavior to be ahead of target trajectory.

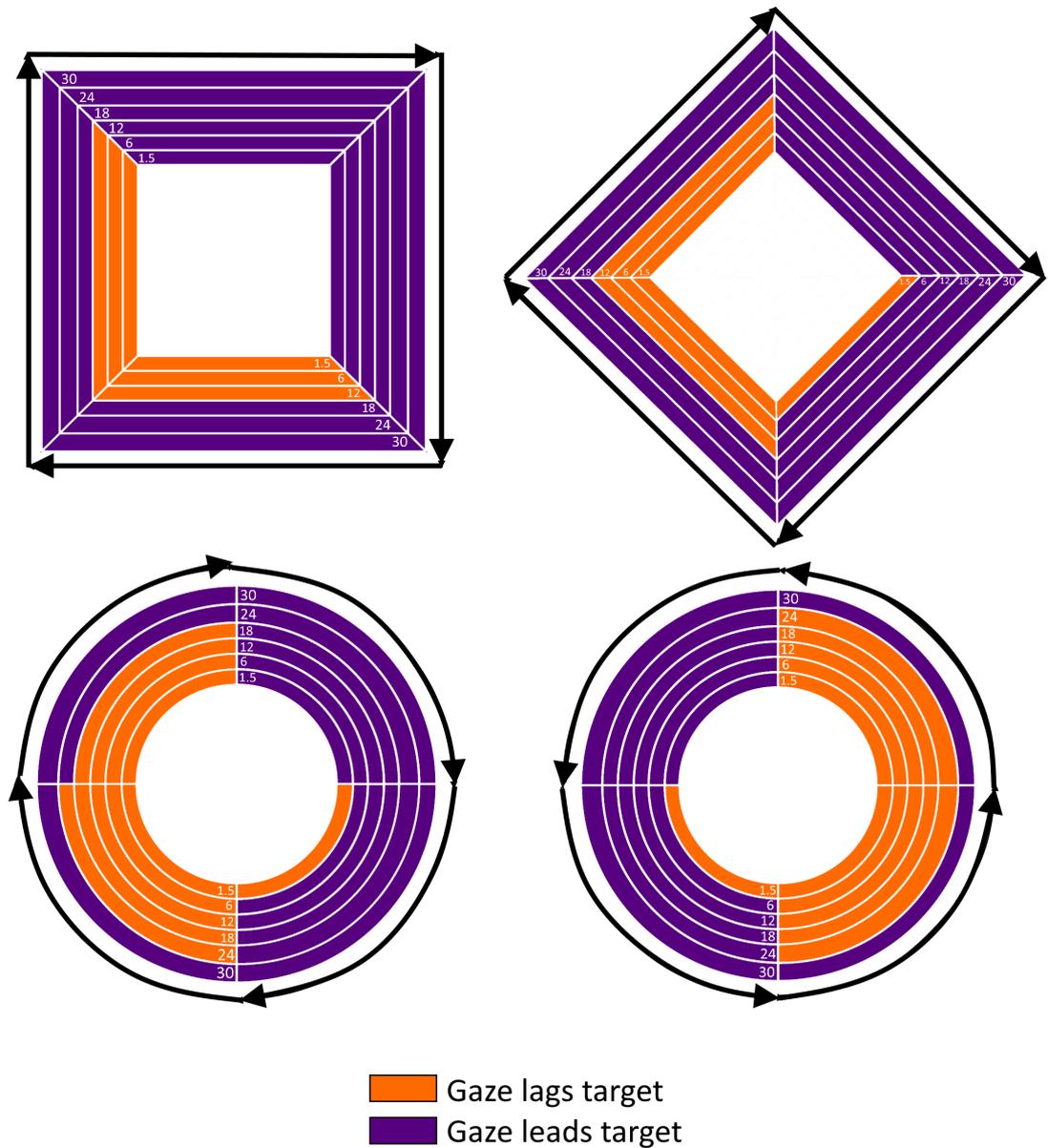


Figure 24: Infographic showing motion behavior analysis results. Gaze position versus Target position influenced by different motions and speeds. Outer represents the fastest speed and decreases towards the center of the shape.

being normalized) on the screen by linking pairs of both element positions with lines so as to give a representation of the offset (or distance) between positions and analyze the gaze performance.

We displayed the first recorded points (between 500 and 1000 samples) of motion for every direction and speed for all participants. *Figure 27* represents the smooth pursuit samples captured for all participants for the first part of circular clockwise movement when ascending to the right at $18^\circ/\text{s}$. We can observe how most of the vectors that most of gaze samples seem to be delayed. Moreover, *Figure 28* representing the samples captured for horizontal motion going from right to the left at $6^\circ/\text{s}$, shows a similar behavior.

On the other hand, *Figure 29* shows how for the first samples during circular clockwise movement when target moves down to the left at $12^\circ/\text{s}$, gaze starts presenting a delayed trajectory and then changes its behavior to tend to be ahead of target's motion.

4.4 Discussion

Research in Human-Computer interaction using Smooth Pursuit eye movements consider that when motion is present, gaze cannot help but follow that movement. While this statement could imply a delayed trajectory when following the target's motion, our approach to understand the characteristics of eye movement showed more complex behavior.

Results show how our ability to follow movement is dependent on the speed of the

Table 3: Mean Gaze position versus Target position (Offset or distance in $^\circ$ of visual angle) influenced by different motion directions and slow speeds ($^\circ/\text{s}$)

	1.5	6	12
<i>Horizontal (left-right)</i>	0.51±1.95	0.50±2.15	0.88±1.58
<i>Horizontal (right-left)</i>	0.49±1.64	0.42±1.51	0.22±1.81
<i>Vertical (top-bottom)</i>	0.12±1.98	0.34±1.67	0.50±1.81
<i>Vertical (bottom-top)</i>	0.53±1.93	0.57±1.69	0.50±1.80
<i>Diagonal (up-left)</i>	0.66±2.07	0.64±2.24	1.19±3.21
<i>Diagonal (up-right)</i>	0.56±1.76	0.48±1.63	0.58±1.66
<i>Diagonal (down-left)</i>	0.25±1.68	0.28±1.56	0.58±1.79
<i>Diagonal (down-right)</i>	0.16±1.67	0.41±1.93	0.18±3.38
<i>Circular clk (down-left)</i>	0.17±1.71	0.23±1.35	0.33±1.56
<i>Circular clk (down-right)</i>	0.29±1.64	0.47±1.29	0.75±1.46
<i>Circular clk (up-left)</i>	0.37±1.92	0.22±1.41	0.43±2.36
<i>Circular clk (up-right)</i>	0.49±1.82	0.65±1.58	0.75±1.47
<i>Circular no-clk (up-left)</i>	0.24±1.64	0.29±1.70	0.29±1.59
<i>Circular no-clk (up-right)</i>	0.15±2.37	0.28±1.88	0.29±2.01
<i>Circular no-clk (down-left)</i>	0.23±1.67	0.46±1.41	0.50±1.23
<i>Circular no-clk (down-right)</i>	0.26±1.62	0.25±1.34	0.42±1.74

Table 4: Mean Gaze position versus Target position (Offset or distance in ° of visual angle) influenced by different motion directions and fast speeds (°/s)

	18	24	30
<i>Horizontal (left-right)</i>	0.87±2.25	0.91±2.22	1.17±1.76
<i>Horizontal (right-left)</i>	0.57±1.53	0.83±2.36	1.39±2.11
<i>Vertical (top-bottom)</i>	1.02±1.90	1.38±2.44	2.03±2.80
<i>Vertical (bottom-top)</i>	0.39±2.03	3.78±4.48	2.55±2.86
<i>Diagonal (up-left)</i>	0.56±2.10	0.53±1.93	1.22±2.82
<i>Diagonal (up-right)</i>	0.60±1.66	0.63±1.80	2.21±3.10
<i>Diagonal (down-left)</i>	1.12±2.07	1.63±2.20	2.47±2.59
<i>Diagonal (down-right)</i>	0.74±1.79	0.92±1.79	1.01±2.91
<i>Circular clk (down-left)</i>	0.71±1.65	1.15±1.49	1.58±1.70
<i>Circular clk (down-right)</i>	0.91±1.44	1.15±1.43	1.44±2.03
<i>Circular clk (up-left)</i>	0.31±1.89	0.57±1.63	0.97±1.86
<i>Circular clk (up-right)</i>	0.70±1.35	0.71±1.37	0.99±1.92
<i>Circular no-clk (up-left)</i>	0.29±1.82	0.49±1.59	0.76±1.81
<i>Circular no-clk (up-right)</i>	0.39±2.86	0.33±2.72	0.65±2.32
<i>Circular no-clk (down-left)</i>	0.85±1.21	1.03±1.64	1.36±2.01
<i>Circular no-clk (down-right)</i>	1.30±3.14	1.61±2.51	2.20±2.01

Table 5: Gaze position versus Target position influenced by different motion directions and speeds (°/s) (fwd: forward, d: delayed)

	1.5	6	12	18	24	30
<i>Horizontal (left to right)</i>	fwd	fwd	fwd	fwd	fwd	fwd
<i>Horizontal (right to left)</i>	d	d	d	fwd	fwd	fwd
<i>Vertical (top to bottom)</i>	fwd	fwd	fwd	fwd	fwd	fwd
<i>Vertical (bottom to top)</i>	d	d	d	fwd	fwd	fwd
<i>Diagonal (up to left)</i>	d	d	d	fwd	fwd	fwd
<i>Diagonal (up to right)</i>	d	d	d	fwd	fwd	fwd
<i>Diagonal (down to left)</i>	d	fwd	fwd	fwd	fwd	fwd
<i>Diagonal (down to right)</i>	fwd	fwd	fwd	fwd	fwd	fwd
<i>Circular clock (down left)</i>	d	fwd	fwd	fwd	fwd	fwd
<i>Circular clock (down right)</i>	fwd	fwd	fwd	fwd	fwd	fwd
<i>Circular clock (up left)</i>	d	d	d	d	d	fwd
<i>Circular clock (up right)</i>	d	d	d	d	fwd	fwd
<i>Circular anti-clock (up left)</i>	d	d	d	d	d	fwd
<i>Circular anti-clock (up right)</i>	d	d	d	d	d	fwd
<i>Circular anti-clock (down left)</i>	fwd	fwd	fwd	fwd	fwd	fwd
<i>Circular anti-clock (down right)</i>	d	fwd	fwd	fwd	fwd	fwd

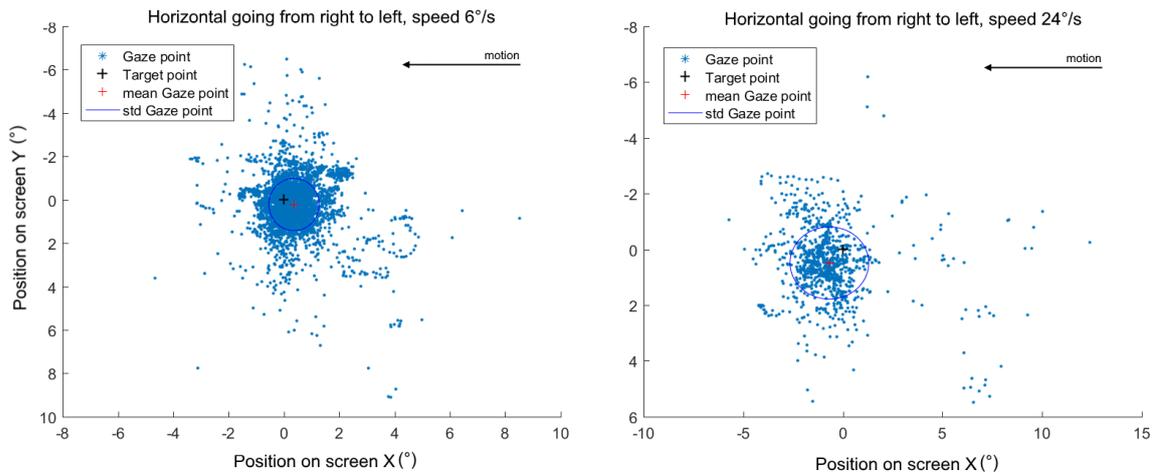


Figure 25: (Left) Horizontal direction segment going from right to left at $6^\circ/\text{s}$ scatter plot. (Right) Horizontal direction segment going from right to left at $24^\circ/\text{s}$ scatter plot. Both show how for slow speed gaze is delayed regarding target movement, whereas for fast speeds the eyes are reported to move ahead of the target.

target, affecting the detection of smooth pursuit eye movements. *Figure 20* suggest that in order to be able to detect the performed movement, targets' speed should be lower than $24^\circ/\text{s}$. However, all participants reported after the study feeling eye tiredness and discomfort when following targets at the lowest speed ($1.5^\circ/\text{s}$). Therefore we believed the minimum speed to be considered should be higher than $1.5^\circ/\text{s}$.

In terms of the targets' motion direction, all horizontal, vertical, circular and diagonal movements showed good results in detection of smooth pursuit movement when slow speeds are used. Nevertheless, results also showed that avoiding vertical movements is desired when faster speeds are used. Vertical (both ascending and descending) eye movements are not common in our everyday activities, therefore our eye muscles can feel more difficult following a target that moves using that scope [11].

Moreover, all directions reported no statistically significant difference in results concerning the distance between gaze and target position during movement. However, against the premise that locates gaze behind target motion as the eyes follow its movement, we reported in *Table 3*, *Table 4* and both *Table 5* and *Figure 24* a different gaze motion behavior depending on direction and speed. We found in slower speeds a delayed gaze behavior performed for horizontal right to left movement and directions involving ascending motion. On the other hand, when speed increased, its position turns to be ahead, as reported in *Figure 25* and *Figure 24*.

Additionally, *Figure 24* suggests that the eyes don't have the same ability to follow movement going up or right to left, and going down or left to right. Our study results indicate that gaze behavior changes depending on such directions and according to speed increase. We believe this is caused by the lack of familiarity in our everyday eye

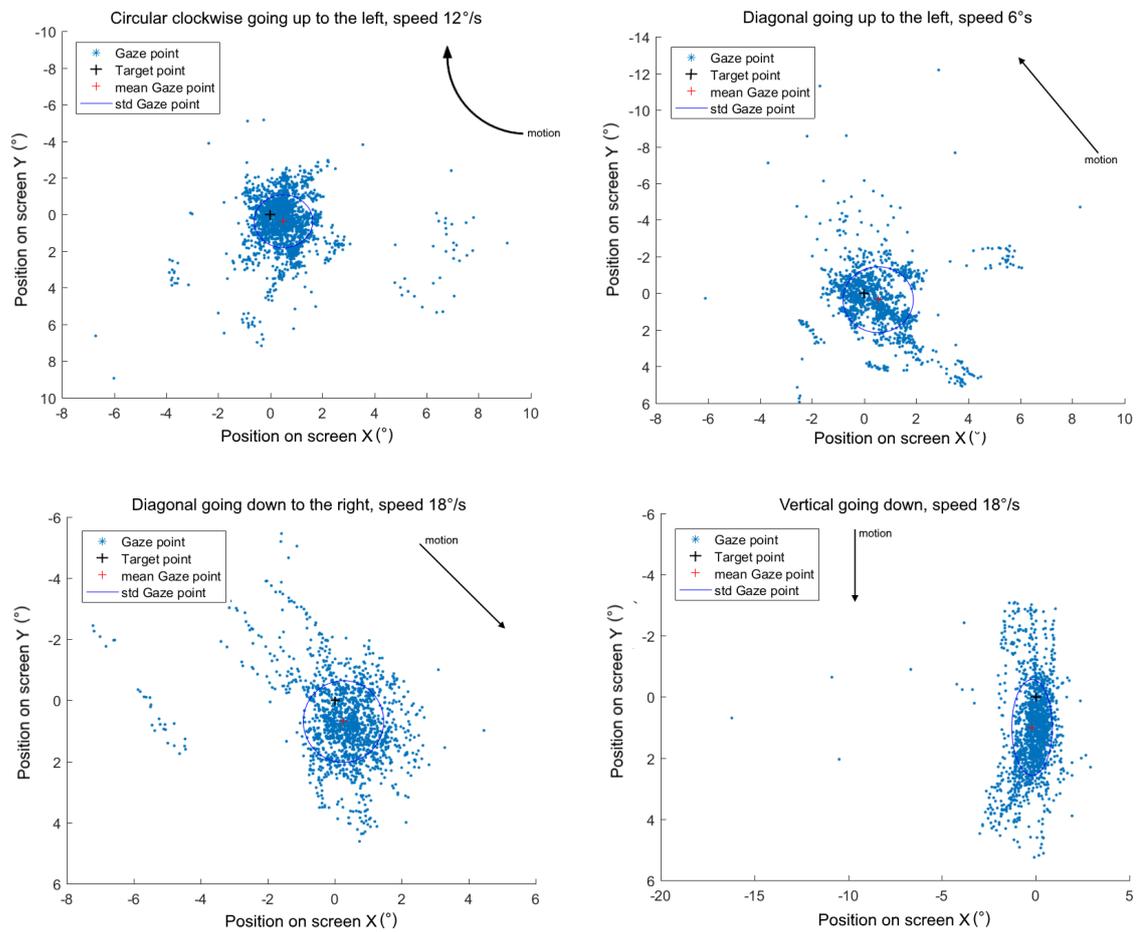


Figure 26: Gaze position versus target trajectory scatter plots. (Top-left) Clockwise circular movement belonging to the quarter of circle going up to the left at $12^\circ/s$, shows how gaze is delayed. (Top-right) Diagonal movement going up to the left at $6^\circ/s$, show how gaze is delayed. (Bottom-left) Diagonal movement going down to the right at $18^\circ/s$, shows how gaze is ahead. (Bottom-right) Vertical movement going down at $18^\circ/s$, shows how gaze is ahead.

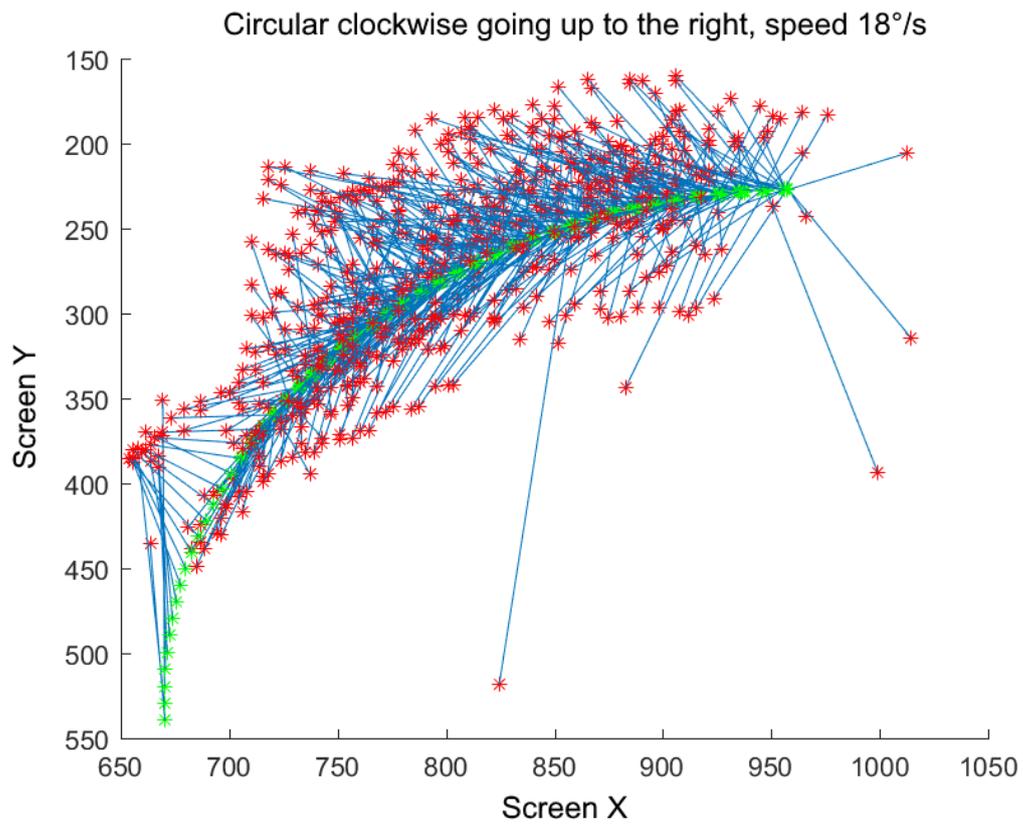


Figure 27: Representation of paired points (Gaze and target position) during the first smooth pursuit samples of circular clockwise movement going up to the right at 18°/s for all participants.

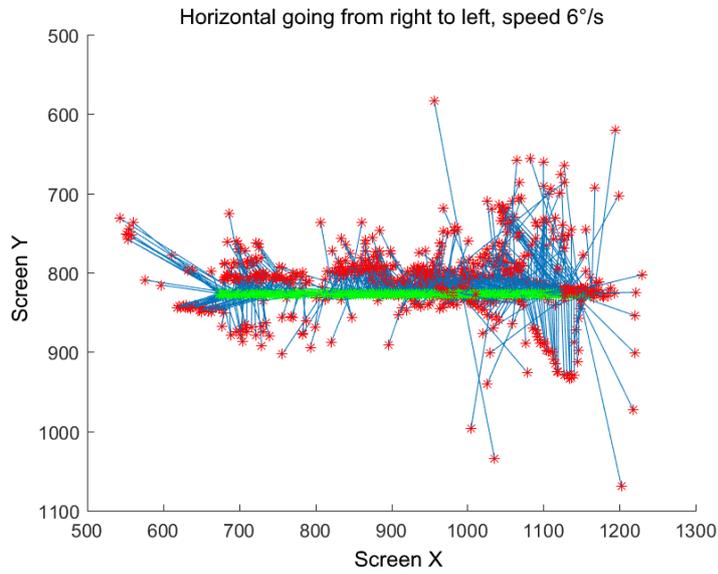


Figure 28: Representation of paired points (Gaze and target position) during the first smooth pursuit samples of horizontal movement going right to the left at 6°/s for all participants.

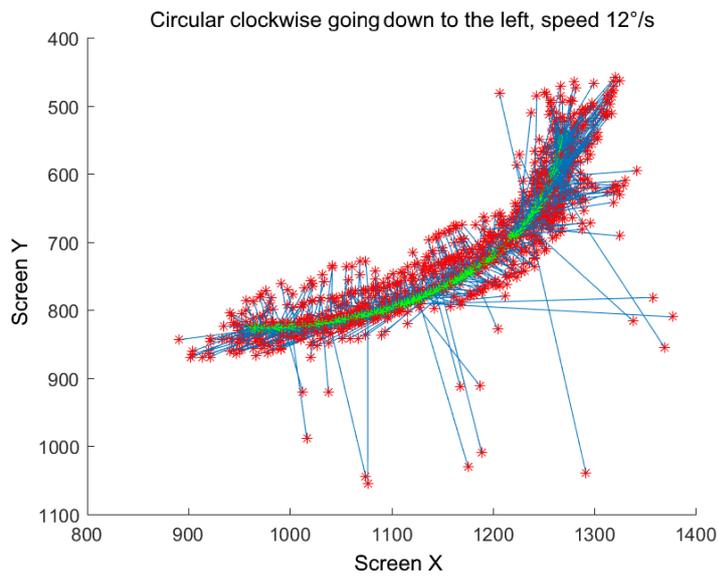


Figure 29: Representation of paired points (Gaze and target position) during the first smooth pursuit samples of circular clockwise movement going down to the left at 12°/s for all participants.

movements. For instance, when we read (European languages), there is an eye movement continuity involving horizontal left to right and descending directions, whereas vertical motion is not as present.

Results indicate the presence of two possible factors that can act as uncontrolled variables in our study. When assessing vertical movements we find both physiological constraints and technological limitations. While it is shown in human anatomy that the muscles of the eyes have a higher scope on the horizontal axis rather than the vertical one, state of the art eye trackers are known to have less reliable data on vertical movement due to the inability of the tracker cameras to locate the eye in such movement. Moreover, results suggesting a difference in left to right and right to left movements might indicate a possible cultural (experience) difference. This variable was uncontrolled during the study and we do not know if users from a cultural background where predominant eye movements (for example when reading) are opposite to that of European culture.

Accordingly, as shown in *Figure 26*, movements in horizontal (left to right) and descending directions reported gaze position to be ahead of target trajectory. Further analysis results, reported in *Figure 29*, suggest that gaze starts following the movement being delayed, but there could be a prediction of the movement. Therefore, our results indicate that the behavior of the eyes is to be ahead of the target movement tending to try and predict the forthcoming target motion [28].

Furthermore, in terms of gaze accuracy our results suggest that smooth pursuit eye movements are a potential good predictor of where gaze is pointing. However, not only does the demonstrated anticipatory behavior has to be taken into consideration, so do the features of the presented movement. We determined that for very fast speeds (higher than $24^\circ/\text{s}$) and directions that our eyes muscles are not used to perform in everyday activities (vertical), results (Table 3 and Table 4) show that gaze accuracy is lost as users cannot follow the motion properly.

4.5 Conclusion

Our study helped to gain a better understanding of smooth pursuit eye movements and how the eyes react and behave when on-screen motion is presented.

Providing users with different motion directions, based on previous studies we assumed that the eyes cannot help but follow the target that is moving, performing smooth pursuits. However, our user study showed results suggesting that there is an anticipation of the movements pointing towards a predictive behavior of the eyes when motion is presented. We determined which movement features are best to avoid when designing in terms of speed and direction, as no unnatural or unusual eye movements can guarantee a good gaze estimation when performing smooth pursuits. Our results suggest that vertical directions and speeds higher than $24^\circ/\text{s}$ should be avoided when designing targets for smooth pursuit interaction. Moreover, when designing for smooth pursuit detection, the anticipatory behavior needs to be considered.

Understanding what the behavior of the eyes is when content in motion is presented will help when designing new interfaces that leverage smooth pursuit eye movements for

hands-free interaction. We discovered that there is a prediction in the eyes movement which might help in the development of new smooth pursuit algorithms detectors and further applications using this continuous movement of the eyes.

5. Discussion

Smooth Pursuit eye movements have become a very promising input for interactive applications providing hands-free interaction. New dynamic interfaces made the integration of eye movements into applications using gaze interaction possible. Smooth pursuit allows users to interact by following movement with their gaze, without being compromised by eye trackers accuracy. It also benefits the user experience by avoiding triggering unwanted outcomes thanks to its robustness, and this is not affected by the calibration process. However, very little is known about the behavior of smooth pursuit eye movements when used as input.

Applications using smooth pursuit interaction are based on the spontaneous behavior that allows our eyes to show attention to a presented object in motion. By performing the same motion pattern as the moving target, our eyes are capable to move continuously and smoothly while following the movement. This feature of our eyes broaden opportunities for the design of new interfaces using gaze interaction, that provide more flexible and a better user experience.

Previous research demonstrated that smooth pursuit detection is robust and able to be embedded in different interfaces' contexts. In our first study, we demonstrated that the integration of calibration processes using smooth pursuit movements is feasible and possible when content in motion is presented on the interface. Rather than having a separate task for calibration we proposed leveraging the spontaneous behavior of the eyes when performing pursuits to be used in interfaces containing moving content. We illustrated this principle by using a videogame in which motion is always present.

We proposed to embed smooth pursuit re-calibration and be run in the interface's background without being perceptible to users. We used the direction and motion matching of the game's moving objects and eyes movements to select samples for calibration. Given that the dynamics of the used interface are in a constant movement we constantly assess the eyes movement and use its data to calibrate the gaze estimation process. Such repeatedly performed re-calibration helped to improve the accuracy of the estimated gaze that could be affected by deterioration over time.

Results from the user study show how calibration remained invisible to users while reporting good accuracy results that improved over time. Participants in our study were not given any instruction and they were only asked to play the game in order not to affect their behavior. During the course of the study, participants were asked to move from their position and come back again. The eye estimation accuracy was affected but smooth pursuit re-calibration was able to correct the created deviations.

Our first study results provided us with a better understanding of smooth pursuit detection and integration. Smooth pursuit re-calibration showed potential to be integrated into different applications where items that move are present, allowing gaze estimation to happen within and among different interfaces. Overall, we demonstrated that the coexistence of calibration and interaction is possible without adding separating them in different applications.

Nevertheless, the detection of smooth pursuits is based on a speed and direction matching process without considering any information on how accurate our eyes are pointing at the movement. There is no knowledge of whether gaze is close to the target when pursuing its motion. Based on the related background literature, during our first study, we assumed that the matched motion was accurate. However, we did not have enough evidence to assure the quality of the detected data used for later re-calibration. Smooth pursuit movements for interaction have been used regardless of calibration, hence the use of matched points in motion for later calibration made us wonder whether pursuits calibration procedure was reliable enough.

In our approach, eye motion matched target movement but no information about gaze precision when following the different directions was gained as we could not be sure if users were specifically looking at the target during its movement. Our results show promising and compelling characteristics that demonstrate that smooth pursuit calibration can shape the future of calibration tasks. Notwithstanding, we believe our approach was too naive and simple. The embedded re-calibration method was not smart enough, it was running all the time without assuring the quality of the collected data.

During the gameplay, we created an attention dilemma by offering gaze interaction to players. There is not enough evidence to report that such behavioral change affected calibration, but some errors might have been stored. Thence, a need to understand if pursuits are reliable for calibration arose. Obtaining such knowledge would help in the creation of a smarter calibration system using smooth pursuit movements. If we were able to assess the accuracy of the correlated motion between eye and target movements, we could be able to evaluate the quality of the estimated gaze and decide whether re-calibration is needed or not.

Nonetheless, in order to design and improve state of the art gaze calibration using smooth pursuits we need further information about the behavior of gaze when different types of motion are presented.

Against the approach used during our first experiment, in our second study, we took this exploratory step backward to find which are the characteristics of pursuits. We explored the spontaneous behavior of the eyes when following a target moving in different directions. We asked participants to look at and follow the target while moving, without assuming that the eyes are expected to pursue movement when is presented.

During the study, eye tracking calibration process based on fixations was repeatedly performed between rounds. We decided to maintain the calibration accuracy level in order not to compromise the gaze data during the experiment for later analysis. Our second target was to try not to affect the accurate estimation of gaze behavior with the tracker inaccuracy. Highly accurate gaze data was recorded so as to analyze smooth pursuit eye movements' performance.

Results reported a tendency for the eyes' movement to be ahead of the target motion, showing a predictive behavior of the displayed movement. We determined the characteristics of the target motion that benefit the detection of smooth pursuit eye movement. We also got more information on how the eyes follow movement depending

on different speeds and directions.

In summary, our second study helped in gaining a better understanding of the spontaneous behavior the eyes perform when following different target movement directions. We discovered how different target movement patterns affect the ability to accurately follow them. Furthermore, some directions reported being better than others for the collection of highly correlated gaze and target position samples.

Results from our second study helped gain a deeper insight on smooth pursuits behavior that can modulate the performance of smooth pursuit calibration. By providing a set of guidelines we intend to help future designers and developers to create effective detection of smooth pursuit eye movements, not only for calibration but also for interaction.

5.1 Guidelines for Smooth Pursuit Design

The presented research can be generalized in different indications. When designing for smooth pursuit detection and interaction speed and direction of the target stimulus matter.

For better detection of smooth pursuit eye movements, it is desirable to avoid vertical directions as eye muscles have less scope on the y-axis and eye trackers might fail when trying to identify users' eye features. Horizontal, circular and random movements are highly recommended. However, the latter might not allow a fully controlled recognition of pursuits. Moreover, circular motion might provide the system with a higher rate of detection.

Furthermore, we highly recommend being sure of your target audience, as culture or ethnicity might have an effect on your detection parameters. Our results suggest there is a difference in smooth pursuit eye movements' behavior from left to right and right to left. We identified this effect as an influence of day to day eye movements in reading habits (European cultures), advising that it might have an impact on users' gaze conduct. Such awareness can help when designing pursuits detection by optimizing the method to whether expect an anticipatory conduct or not.

On the other hand, the target's (stimulus) speed is another important factor in the design. Whereas slow speeds are easy to detect, they are not user-friendly, reporting discomfort and tiredness. Alternatively, very fast speeds are not desirable either, they are reported hard to follow and will be hard to detect as pursuits as the eyes start performing saccadic movements (quick jumps to catch up with the movement) while also anticipating them. Finally, our results suggest designing with speeds no slower than $6^\circ/\text{s}$ and no higher than $24^\circ/\text{s}$.

5.2 Direction, Generalization and Limitations

The understanding of the eyes behavior can benefit current gaze input systems and help develop new applications not only in a desktop context but also in public displays or Virtual Reality, creating new questions and opportunities in HCI research using gaze.

In the case of videogames, popular game franchises have been integrating gaze interaction in their engines. We believe that the introduction of smooth pursuit eye movements will benefit the gaming industry. First, embedded smooth pursuit re-calibration is easy to introduce within game contexts as motion is always presented. Second, the integration of the calibration task maintains the user experience without breaking it for re-calibration. Third, smooth pursuit movements offer different possibilities to create new interaction paradigms.

Smooth pursuits can be used for both helping the player and the game engine. In the case of *GazeBall* we need the player to look at the ball in order to calibrate the system. However, if other moving elements were introduced, we could use pursuits movement correlation to speed up the ball and make the game more challenging. On the other hand, pursuit movement could help identify players behavior to help them during gameplay.

The proposed system can be easily implemented in any gaming interface, the only requirement is to have an eye tracker to provide the gaze signal. We can identify movement happening in several game scenes, there is always a character to follow, a target that needs to be shot or an element that is moving across the screen. The implementation of the smooth pursuit calibration will be aligned with the level designer responsibility. It is during the designing process that the guidelines on smooth pursuit have to be followed in order to include elements that would be potentially able to get players attention and trigger the calibration. Moreover, this method can be embedded in other applications and interfaces outside gaming industry like simulations or video streams only if those contain content that moves. Without a moving stimulus, the designed calibration becomes ineffective.

On the other hand, designers need to consider for gaze interaction that only areas in which something have been followed (in pursuit calibration) have an accurate estimation of gaze. The proposed calibration approach is limited to the "visiting" of enough areas spread across the screen to be able to interpolate the gaze position. However, we can imagine a system that has already featured a standardized calibrated gaze profile that is corrected and adapted to the current user by smooth pursuit calibration.

In terms of hardware, pursuit calibration can rely on any eye tracker as it transforms the device signals obtained from the eyes position to coordinates that correspond to different points on the screen. The system is able to discriminate any format for coordinates and will be able to transform them into screen points.

Our implemented system was dependent on the eye tracker calibration profile, as there is a limited access to Tobii's trackers' raw data. Therefore, the implemented smooth pursuit calibration was performed on top, creating a double correction from the tracker raw data. The participants of our study performed the tasks relying on an already existent and independent calibration profile and an additional correction obtained with our system was added. Whereas this method might look like a limitation, the system proved that can be robust with any input gaze data.

All in all, we believe that smooth pursuits offer new opportunities for a hands-free and independent tool for interaction.

5.3 Future work

Both studies provided a better understanding of the eyes movement performance. Leveraging this knowledge of the eyes' behavior when performing smooth pursuit movements would help improving current systems using such gaze interaction paradigm. In the case of our first study, knowing the best strategy to collect highly correlated paired points would improve the quality of the estimated gaze data.

Furthermore, when designing new interfaces containing motion, we would be able to establish different rules to perform re-calibration based on the displayed movement and accuracy of gaze. Moreover, when validating the accuracy of the gaze point while performing smooth pursuit movements, we are putting forward the possibility to use smooth pursuit movements for accuracy control.

We believe that in future work smooth pursuit eye movements will show flexibility for the design of a very compelling embedded automatic re-calibration method. This new method has the potential to be integrated into applications that contain motion. Moreover, this novel approach would be able to assess which is the quality of the accuracy level of the estimated gaze point in order to activate a new re-calibration process.

The creation of such method would validate smooth pursuit eye movement as a robust and efficient approach for gaze calibration, providing a compelling future of gaze as a powerful tool for interaction.

With this work findings we propose the creation of the *Smooth-i* calibration framework, a smart smooth pursuit calibration method able to detect when calibration is necessary and capable of updating users profile in an effective way.

We devise *Smooth-i* as a smart calibration method, based on smooth pursuit calibration [49], able to detect when the predicted gaze is uncalibrated and able to efficiently perform re-calibration by updating the user profile and maintaining only the necessary points for calibration.

With *Smooth-i* we intend to apply the knowledge gained through this project and leverage that it is able to detect where users look while performing smooth pursuits. This detection would allow us to perform an accuracy check and decide whether re-calibration needs to be performed or not. Moreover, *Smooth-i* will ensure that as soon as new screen areas are discovered, those points are saved and don't affect previous calibration areas. If they do, *Smooth-i* will effectively decide which points are suitable for the gaze estimation.

We believe our devised calibration method makes a contribution that can be very beneficial for systems based on Artificial Intelligence (AI). Although future calibration methods can use AI to learn how to users move and help to correct such gaze accuracy estimation deviations and improve calibration profiles, we propose a coexistence with smooth pursuit re-calibration that offers information about the quality of points collected that would enhance such smart approach. Nevertheless, machine learning can be very useful to manage calibration profiles with smooth pursuit re-calibration. As the system only have information and can accurately estimate points that have already been visited, AI methods might be able to select and switch between different profiles, being able to

adapt the proposed algorithm in a much smarter way, whereas the proposed *Smooth-i* would depend on the designer configuration.

All in all, we think smooth pursuit calibration offers compelling features to the evolution of gaze calibration methods, offering more flexible and smarter interfaces design.

6. Conclusion

Previous research on eye tracking calibration methods for gaze interaction separates the calibration task from the actual experience. Our proposal is to use smooth pursuit movements that proved to be able to be embedded within the interface dynamics.

Using a videogame as our testing workspace, we demonstrated that smooth pursuit re-calibration remains robust and invisible to users. Our approach was able to correct accuracy deviations over time reporting good and accurate estimation of gaze positions while introducing new gaze interaction paradigms.

Moreover, further research allowed us to understand an anticipatory behavior happening when performing smooth pursuit eye movements. We were able to identify which displayed target directions and speeds are best to obtain highly correlated and accurate smooth pursuit gaze points.

Furthermore, previous research assumed that gaze was accurately pointing at the target moving object position when performing pursuits used for the correlation between gaze and target motion. In our approach, we demonstrated how accurate smooth pursuit movements are when following, and we contributed possible design guidelines.

We believe our work contributed to providing a better understanding of the use of smooth pursuit movements for calibration in dynamic interfaces such as videogame contexts. Our results aim to design a potentially compelling smart tool for automatic re-calibration that would shape the technology of the future being able to utilize gaze interaction in an efficient and effortless way.

References

- [1] Inc. Atari. 1976. *Breakout*. Game [Arcade]. (April 1976).
- [2] Paulo Bala, Lucilia Noóbrega, Guilherme Neves, Lai's Lopes, Joana Morna, João Camacho, and Cristina Freitas. 2015. Keyewai: Looking at Cooperation in a Holographic Projection Screen. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*. ACM, 61–64.
- [3] Richard Bates, Stephen Vickers, and Howell O Istance. 2010. Gaze interaction with virtual on-line communities: levelling the playing field for disabled users. *Universal Access in the Information Society* 9, 3 (2010), 261–272.
- [4] Roman Bednarik, Tersia Gowases, and Markku Tukiainen. 2009. Gaze interaction enhances problem solving: Effects of dwell-time based, gaze-augmented, and mouse interaction on problem-solving strategies and user experience. *Journal of Eye Movement Research* 3, 1 (2009).
- [5] Roman Bednarik, Tersia Gowases, and Markku Tukiainen. 2013. Gaze-augmented interaction improves problem-solving: new evidence from verbal protocols. *Behaviour & Information Technology* 32, 8 (2013), 836–844.
- [6] Abdelkader Nasreddine Belkacem, Supat Saetia, Kalanyu Zintus-art, Duk Shin, Hiroyuki Kambara, Natsue Yoshimura, Nasreddine Berrached, and Yasuharu Koike. 2015. Real-time Control of a Video Game using Eye Movements and Two Temporal EEG Sensors. *Computational intelligence and neuroscience* 2015 (2015), 1–10.
- [7] David Beymer and Daniel M Russell. 2005. WebGazeAnalyzer: a system for capturing and analyzing web reading behavior using eye gaze. In *CHI'05 extended abstracts on Human factors in computing systems*. ACM, 1913–1916.
- [8] Aga Bojko. 2013. *Eye tracking the user experience*. Rosenfeld Media.
- [9] Richard A Bolt. 1990. A gaze-responsive self-disclosing display. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 3–10.
- [10] Emiliano Castellina and Fulvio Corno. 2008. Multimodal gaze interaction in 3D virtual environments. *COGAIN* 8 (2008), 33–37.
- [11] Han Collewyn and Ernst P Tamminga. 1984. Human smooth and saccadic eye movements during voluntary pursuit of different target motions on different backgrounds. *The Journal of Physiology* 351 (1984), 217.
- [12] Matthieu Perreira Da Silva, Vincent Courboulay, and Armelle Prigent. 2007. Gameplay experience based on a gaze tracking system. In “*Gaze-based Creativity, Interacting with Games and On-line Communities*” *INPROCEEDINGS in proceedings of COGAIN 2007 (Communication by Gaze Interaction IST FP6 European Project)*. 25–28.

- [13] Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart Nacke. 2011. From game design elements to gamefulness: defining gamification. In *Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments*. ACM, 9–15.
- [14] Michael Dorr, Martin Böhme, Thomas Martinetz, and Erhardt Barth. 2007. Gaze beats mouse: a case study. *Proceedings of COGAIN (2007)*, 16–19.
- [15] Michael Dorr, Laura Pomarjansch, and Erhardt Barth. 2009. Gaze beats mouse: A case study on a gaze-controlled breakout. *PsychNology Journal* 7, 2 (2009), 197–211.
- [16] Heiko Drewes and Albrecht Schmidt. 2007. Interacting with the computer using gaze gestures. In *Human-Computer Interaction—INTERACT 2007*. Springer, 475–488.
- [17] Inger M Ekman, Antti W Poikola, and Meeri K Mäkäräinen. 2008. Invisible eni: using gaze and pupil size to control a game. In *CHI'08 extended abstracts on Human factors in computing systems*. ACM, 3135–3140.
- [18] Augusto Esteves, Eduardo Velloso, Andreas Bulling, and Hans Gellersen. 2015. Orbits: Gaze Interaction for Smart Watches using Smooth Pursuit Eye Movements. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. ACM, 457–466.
- [19] Martin A Fischler and Robert C Bolles. 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* 24, 6 (1981), 381–395.
- [20] David R Flatla, Carl Gutwin, Lennart E Nacke, Scott Bateman, and Regan L Mandryk. 2011. Calibration games: making calibration tasks enjoyable by adding motivating game elements. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*. ACM, 403–412.
- [21] Agostino Gibaldi, Mauricio Vanegas, Peter J Bex, and Guido Maiello. 2016. Evaluation of the Tobii EyeX Eye tracking controller and Matlab toolkit for research. *Behavior research methods* (2016), 1–24.
- [22] James Gips and Peter Olivieri. 1996. EagleEyes: An eye control system for persons with disabilities. In *The Eleventh International Conference on Technology and Persons with Disabilities*. 1–15.
- [23] Teresia Gowases, Roman Bednarik, and Markku Tukiainen. 2008. Gaze vs. mouse in games: The effects on user experience. (2008).
- [24] Elias Daniel Guestrin and Moshe Eizenman. 2008. Remote point-of-gaze estimation requiring a single-point calibration for applications with infants. In *Proceedings of the 2008 symposium on Eye tracking research & applications*. ACM, 267–274.

- [25] Dan Hartescu and Andreas Oikonomou. 2011. Gaze tracking as a game input interface. In *Computer Games (CGAMES), 2011 16th International Conference on*. IEEE, 126–133.
- [26] Khalad Hasan, Tovi Grossman, and Pourang Irani. 2011. Comet and target ghost: techniques for selecting moving targets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 839–848.
- [27] Sébastien Hillaire, Anatole Lécuyer, Rémi Cozot, and Géry Casiez. 2008. Using an eye-tracking system to improve camera motions and depth-of-field blur effects in virtual environments. In *Virtual Reality Conference, 2008. VR'08. IEEE*. IEEE, 47–50.
- [28] Kenneth Holmqvist, Marcus Nyström, Richard Andersson, Richard Dewhurst, Halszka Jarodzka, and Joost Van de Weijer. 2011. *Eye tracking: A comprehensive guide to methods and measures*. OUP Oxford.
- [29] Jeff Huang, Ryen White, and Georg Buscher. 2012. User see, user point: gaze and cursor alignment in web search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1341–1350.
- [30] EyeWriter Initiative and others. 2009. The EyeWriter. (2009).
- [31] Poika Isokoski, Markus Joos, Oleg Spakov, and Benoît Martin. 2009. Gaze controlled games. *Universal Access in the Information Society* 8, 4 (2009), 323–337.
- [32] Poika Isokoski and Benot Martin. 2006. Eye tracker input in first person shooter games. In *Proceedings of the 2nd Conference on Communication by Gaze Interaction: Communication by Gaze Interaction-COGAIN 2006: Gazing into the Future*. 78–81.
- [33] Howell Istance, Richard Bates, Aulikki Hyrskykari, and Stephen Vickers. 2008. Snap clutch, a moded approach to solving the Midas touch problem. In *Proceedings of the 2008 symposium on Eye tracking research & applications*. ACM, 221–228.
- [34] Howell Istance, Aulikki Hyrskykari, Lauri Immonen, Santtu Mansikkamaa, and Stephen Vickers. 2010. Designing gaze gestures for gaming: an investigation of performance. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. ACM, 323–330.
- [35] Howell Istance, Aulikki Hyrskykari, Stephen Vickers, and Thiago Chaves. 2009. For your eyes only: Controlling 3d online games by eye-gaze. In *Human-Computer Interaction-INTERACT 2009*. Springer, 314–327.
- [36] Robert JK Jacob. 1991. The use of eye movements in human-computer interaction techniques: what you look at is what you get. *ACM Transactions on Information Systems (TOIS)* 9, 2 (1991), 152–169.

- [37] Erika Jönsson. 2005. If looks could kill—an evaluation of eye tracking in computer games. *Unpublished Master’s Thesis, Royal Institute of Technology (KTH), Stockholm, Sweden* (2005).
- [38] Incheol Kang and Joseph G Malpeli. 2003. Behavioral calibration of eye movement recording systems using moving targets. *Journal of neuroscience methods* 124, 2 (2003), 213–218.
- [39] Yuki Kondou and Yoshinobu Ebisawa. 2008. Easy eye-gaze calibration using a moving visual target in the head-free remote eye-gaze detection system. In *Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2008. VECIMS 2008. IEEE Conference on*. IEEE, 145–150.
- [40] R John Leigh and David S Zee. 2015. *The neurology of eye movements*. Oxford University Press, USA.
- [41] Ubisoft Massive. 2016. *Tom Clancy’s The Division*. Game [Microsoft Windows]. (8 March 2016).
- [42] Julio C Mateo, Javier San Agustin, and John Paulin Hansen. 2008. Gaze beats mouse: hands-free selection by combining gaze and emg. In *CHI’08 extended abstracts on Human factors in computing systems*. ACM, 3039–3044.
- [43] Carlos H Morimoto and Marcio RM Mimica. 2005. Eye gaze tracking techniques for interactive applications. *Computer Vision and Image Understanding* 98, 1 (2005), 4–24.
- [44] Jorge Munoz, Georgios N Yannakakis, Fiona Mulvey, Dan Witzner Hansen, German Gutierrez, and Araceli Sanchis. 2011. Towards gaze-controlled platform games. In *Computational Intelligence and Games (CIG), 2011 IEEE Conference on*. IEEE, 47–54.
- [45] Lennart E Nacke, Sophie Stellmach, Dennis Sasse, and Craig A Lindley. 2010. Gameplay experience in a gaze interaction game. *arXiv preprint arXiv:1004.0259* (2010).
- [46] J O’Donovan, J Ward, S Hodgins, and V Sundstedt. 2009. Rabbit run: Gaze and voice based game interaction. In *Eurographics Ireland Workshop, December*.
- [47] Johanna Palcu, Jennifer Sudkamp, and Arnd Florack. 2017. Judgments at Gaze Value: Gaze Cuing in Banner Advertisements, Its Effect on Attention Allocation and Product Judgments. *Frontiers in Psychology* 8 (2017).
- [48] Ken Pfeuffer, Jason Alexander, Ming Ki Chong, and Hans Gellersen. 2014. Gaze-touch: combining gaze with multi-touch for interaction on the same surface. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. ACM, 509–518.

- [49] Ken Pfeuffer, Melodie Vidal, Jayson Turner, Andreas Bulling, and Hans Gellersen. 2013. Pursuit calibration: Making gaze calibration less tedious and more flexible. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM, 261–270.
- [50] Ubisoft Quebec. 2015. *Assassin’s Creed Syndicate*. Game [Microsoft Windows]. (19 November 2015).
- [51] Patrick Renner, Nico Lüdi, Jens Wittrowski, and Thies Pfeiffer. 2011. Towards Continuous Gaze-Based Interaction in 3D Environments—Unobtrusive Calibration and Accuracy Monitoring. In *GI VRAR workshop*, Shaker Verlag.
- [52] Susan K Schnipke and Marc W Todd. 2000. Trials and tribulations of using an eye-tracking system. In *CHI’00 Extended Abstracts on Human Factors in Computing Systems*. ACM, 273–274.
- [53] John L Sibert, Mehmet Gokturk, and Robert A Lavine. 2000. The reading assistant: eye gaze triggered auditory prompting for reading remediation. In *Proceedings of the 13th annual ACM symposium on User interface software and technology*. ACM, 101–107.
- [54] J David Smith and TC Graham. 2006. Use of eye movements for video game control. In *Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology*. ACM, 20.
- [55] John D Smith, Roel Vertegaal, and Changuk Sohn. 2005. ViewPointer: lightweight calibration-free eye tracking for ubiquitous handsfree deaxis. In *Proceedings of the 18th annual ACM symposium on User interface software and technology*. ACM, 53–61.
- [56] Oleg Špakov. 2005. EyeChess: the tutoring game with visual attentive interface. *Alternative Access: Feelings & Games* 5 (2005).
- [57] Oleg Špakov, Poika Isokoski, Jari Kangas, Deepak Akkil, and Päivi Majaranta. 2016. PursuitAdjuster: an exploration into the design space of smooth pursuit-based widgets. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*. ACM, 287–290.
- [58] Sophie Stellmach and Raimund Dachselt. 2012. Designing gaze-based user interfaces for steering in virtual environments. In *Proceedings of the Symposium on Eye Tracking Research and Applications*. ACM, 131–138.
- [59] Jayson Turner, Eduardo Velloso, Hans Gellersen, and Veronica Sundstedt. 2014. EyePlay: applications for gaze in games. In *Proceedings of the first ACM SIGCHI annual symposium on Computer-human interaction in play*. ACM, 465–468.

- [60] Boris Velichkovsky, Andreas Sprenger, and Pieter Unema. 1997. Towards gaze-mediated interaction: Collecting solutions of the “Midas touch problem”. In *Human-Computer Interaction INTERACT’97*. Springer, 509–516.
- [61] Eduardo Velloso and Marcus Carter. 2016. The Emergence of EyePlay: A Survey of Eye Interaction in Games. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play*. ACM, 171–185.
- [62] Eduardo Velloso, Amy Fleming, Jason Alexander, and Hans Gellersen. 2015a. Gaze-Supported Gaming: MAGIC Techniques for First Person Shooters. In *Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play*. ACM, 343–347.
- [63] Eduardo Velloso, Jayson Turner, Jason Alexander, Andreas Bulling, and Hans Gellersen. 2015b. An empirical investigation of gaze selection in mid-air gestural 3D manipulation. In *Human-Computer Interaction–INTERACT 2015*. Springer, 315–330.
- [64] Tore Vesterby, Jonas C Voss, John Paulin Hansen, Arne John Glenstrup, Dan Witzner Hansen, and Mark Rudolph. 2005. Gaze-guided viewing of interactive movies. *Digital Creativity* 16, 04 (2005), 193–204.
- [65] Stephen Vickers, Howell Istance, and Michael J Heron. 2013a. Accessible gaming for people with physical and cognitive disabilities: a framework for dynamic adaptation. In *CHI’13 Extended Abstracts on Human Factors in Computing Systems*. ACM, 19–24.
- [66] Stephen Vickers, Howell Istance, and Aulikki Hyrskykari. 2013b. Performing locomotion tasks in immersive computer games with an adapted eye-tracking interface. *ACM Transactions on Accessible Computing (TACCESS)* 5, 1 (2013), 2.
- [67] Stephen Vickers, Howell Istance, and Matthew Smalley. 2010. EyeGuitar: making rhythm based music video games accessible using only eye movements. In *Proceedings of the 7th International Conference on Advances in Computer Entertainment Technology*. ACM, 36–39.
- [68] Melodie Vidal. 2014. Shynosaurs: A game of attention dilemma. In *Proceedings of the first ACM SIGCHI annual symposium on Computer-human interaction in play*. ACM, 391–394.
- [69] Melodie Vidal, Remi Bismuth, Andreas Bulling, and Hans Gellersen. 2015. The Royal Corgi: Exploring Social Gaze Interaction for Immersive Gameplay. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 115–124.
- [70] 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 the 2013 ACM international joint conference on Pervasive and ubiquitous computing*. ACM, 439–448.

- [71] Arantxa Villanueva and Rafael Cabeza. 2008. A novel gaze estimation system with one calibration point. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* 38, 4 (2008), 1123–1138.
- [72] Arantxa Villanueva, Rafael Cabeza, and Sonia Porta. 2004. Eye tracking system model with easy calibration. In *Proceedings of the 2004 symposium on Eye tracking research & applications*. ACM, 55–55.
- [73] Tom Wilcox, Mike Evans, Chris Pearce, Nick Pollard, and Veronica Sundstedt. 2008. Gaze and voice based game interaction: the revenge of the killer penguins. *SIGGRAPH Posters* 81 (2008).
- [74] Michiya Yamamoto, Munehiro Komeda, Takashi Nagamatsu, and Tomio Watanabe. 2011. Hyakunin-eyesshu: A tabletop hyakunin-issu game with computer opponent by the action prediction based on gaze detection. In *Proceedings of the 1st Conference on Novel Gaze-Controlled Applications*. ACM, 5.
- [75] Shumin Zhai, Carlos Morimoto, and Steven Ihde. 1999. Manual and gaze input cascaded (MAGIC) pointing. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. ACM, 246–253.
- [76] Yanxia Zhang, Andreas Bulling, and Hans Gellersen. 2013. Sideways: A gaze interface for spontaneous interaction with situated displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 851–860.