Article

A Study on the Use of Eye Tracking to Adapt Gameplay and Procedural Content Generation in First-Person Shooter Games

João Antunes and Pedro Santana * iD

ISCTE-Instituto Universitário de Lisboa (ISCTE-IUL) and Instituto de Telecomunicações (IT), 1649-026 Lisboa, Portugal; jmmea@iscte-iul.pt

* Correspondence: pedro.santana@iscte-iul.pt; Tel.: +351-217-650-558

Abstract: This paper studies the use of eye tracking in a First-Person Shooter (FPS) game as a mechanism to: (1) control the attention of the player's avatar according to the attention deployed by the player; and (2) guide the gameplay and game's procedural content generation, accordingly. This results in a more natural use of eye tracking in comparison to a use in which the eye tracker directly substitutes control input devices, such as gamepads. The study was conducted on a custom endless runner FPS, Zombie Runner, using an affordable eye tracker. Evaluation sessions showed that the proposed use of eye tracking provides a more challenging and immersive experience to the player, when compared to its absence. However, a strong correlation between eye tracker calibration problems and player's overall experience was found. This means that eye tracking technology still needs to evolve but also means that once technology gets mature enough players are expected to benefit greatly from the inclusion of eye tracking in their gaming experience.

Keywords: computer games; eye tracking; gaze-oriented gameplay

1. Introduction

With eye tracking dating back from the XVIII century [1], many have been the applications for this technology, such as medicine [2]), robotics [3,4]), advertising [5]) and, more recently, computer games [6]. In the past decade, research on computer games tried to compare traditional input (e.g., mouse, keyboard, and gamepad) with eye tracking input, in terms of action accuracy and responsiveness[6–10]. These comparisons were often made after asking players to compete against each other or asking users to complete a given task, using the different input methods. Overall, these comparisons provided mixed results, with some studies claiming that the use of eye tracking contributed to better task completion [10], while others claiming that traditional input devices provided better overall results [7]. Hence, these previous studies have provided contradictory results regarding the effectiveness of the use of eye tracking in computer games as a direct control input, which may be an indication that eye tracking is not best suited to direct input control.

Bearing the limitations of eye tracking as a simple direct control input in mind, in this paper, we propose to use it to control the attention of the player's avatar and the game's procedural content generation. This use of eye tracking is focused in mapping the player's and avatar's attention processes, which we believe to be much more natural and useful than allowing the player to directly control a pointer with the eyes, which has no actual mapping to real life. That is, instead of replacing the traditional input for an eye tracker, we are more interested in studying meaningful ways about how eye tracking can improve gameplay, which includes co-existence with traditional inputs. This also means that, instead of analysing objective performance-based data, as in previous studies, we are more

interested in a subjective analysis of how eye tracking improves enjoyability and how well the player adapts to the technology.

To demonstrate the value of the herein proposed alternative uses of eye tracking in computer games, we developed and tested our own endless runner First-Person Shooter (FPS), Zombie Runner (see Figure 1). In this game, shot accuracy, automatic obstacle avoidance, and procedural obstacle spawn probability are controlled as a function of the avatar's attention model, which, in turn, operates according to the player's gaze estimated with an affordable eye tracker. The goal is to better represent the player's actions in the game, thus contributing to a more immersive experience. Based on a set of testing sessions, we conclude that the use of eye tracking provides a more challenging and immersive experience to the player. Participants reported better levels of satisfaction while playing the game with the gaze tracking turned on. However, a strong correlation between eye tracker calibration problems and player's overall experience was found. This means that eye tracking technology still needs to evolve but also means that once technology gets mature enough players are expected to benefit greatly from the inclusion of eye tracking in their gaming experience.

Figure 1. A zombie being killed in Zombie Runner.

This article is an extended and improved version of a poster paper [11] and it is organised as follows. Section 2 presents an overview of previous and related work. Then, in Section 3, Zombie Runner is described and its implementation detailed. Section 4 describes the experimental setup and analysis the obtained results. Finally, Section 5 presents some conclusions and provides some future work directions.

2. Related Work

Eye tracking is the process of estimating one's gaze direction, identifying the object in which the the subject is focused [12,13]. Eye tracking dates from the XVIII century, in which persistent images to describe the human eye movements were used [14]. In the XX century, the first eye movement measures through a non-intrusive method using photographs and light reflections were made [15]. In the 1980s, with the evolution of computing capacity, it became possible to perform real-time eye tracking with access to video, what opened the possibility of human–machine interaction [16]. With increasingly more accessible prices [17,18], the use of eye trackers increased in several areas, such as marketing [5], psychology [2] and, more recently, computer games [6].

Eye tracking has been explored in computer games as an alternative to traditional input methods, such as mouse or keyboard [6,9]. By testing gaze input versus mouse input in three different computer games, Smith and Graham [6] concluded that the use of eye tracking can provide a more immersive

experience to the player. Isokoski and Martin [8] conducted a preliminary study on the use of eye tracker in First Person Shooters. Each participant in the test was asked to play the same game using three different input method schemes: (1) mouse, keyboard, and eye tracker; (2) only mouse and keyboard; or (3) a console gamepad. The conclusions were not exactly encouraging, suggesting that the performance with the eye tracker was quite inferior to the other two. However, Isokoski and Martin attributed these results to the players' greater knowledge and contact with the traditional input methods, suggesting that this scenario could change with more training.

Other studies achieved similar conclusions. Leyba and Malcolm [7] created a simple test in which the player was asked to eliminate twenty-five balls that moved around the screen at different velocities. The player would move the pointer using the mouse or the eye tracker and would eliminate the balls by clicking on them with the mouse. Two conditions were tested: with and without time limit to complete the task. The results showed that, without time limit, precision and time to complete the task was worse while using the eye tracker than when using a mouse. The same results were obtained for the no time limit condition, in which performance was based on percentage of balls eliminated by the player. Michael Dorr et al. [10] achieved totally opposite results. After creating and adapting a clone of the classic game Breakout, twenty players were asked to participate in a tournament. Players were separated in pairs. The two players of each pair played against each other, one using the mouse and the other using an eye tracker. The control inputs were swapped between rounds. The results showed that the players who used the eye tracker achieved higher scores and won more rounds. The players also stated that using the eye tracker was highly enjoyable. These discrepant results between studies suggest that the type of game and development method of the same game are key elements to achieve a satisfying final result.

Bearing the limitations of using eye tracking as a simple direct control input in mind, we propose to use it to control the attention of the player's avatar and the game's procedural content generation. This use of eye tracking is focused in mapping the mental state of the player and her/his avatar, which we believe to be much more natural and useful than controlling a pointer with the eyes, which has no mapping to real life. Another alternative and interesting use of eye tracking, not tackled in this paper, is to know when to actively redirect the player's attention [19].

Procedural Content Generation (PCG) concerns all creation of game content (e.g., sounds, levels, objects, characters, and textures) with algorithms, with limited or indirect stimuli from the user [20]. PCG is a way of coping with the daunting task of manually creating and populating massive open worlds. Besides this more traditional use of PCG, some researchers proposed that, through the analysis of the interaction between the player and the game, PCG could be used to create a playing experience that adapts itself to the player [21–23], improving the game's replay value. This new approach is known as Experience-Driven Procedural Content Generation (EDPCG) [24].

Within the EDPCG framework, it is possible to generate levels that are adapted to the strengths and limitations of the player, in an attempt to maximize the fun factor. Many studies propose that the fun and challenge factors are directly linked [25–31], which means that, to tune the fun factor, one often has to tune the challenge factor. In this line, in this paper, we propose the integration of PGC with gaze tracking so as to adapt the challenge the player as to face according to his/her form of playing and, as a result, to improve the fun factor.

3. Zombie Runner

The game herein presented, Zombie Runner, was purposely designed to integrate gaze tracking at its core mechanic as a way of estimating the player's attention and use those estimates to control the avatar's attention, that is, as a way of implementing a mapping between the player's and the avatar's attention processes. This is in contrast to the typical use of eye tracking to control avatar's motor actions (e.g., to aim the weapon). In Zombie Runner, motor actions are controlled via a traditional input, a gamepad. This way, gaze and hand orthogonally and, thus, more naturally, control their virtual counterparts. To better analyse the advantages and disadvantages of the use of eye tracking in games,

focus should be given to game genres where the player's visual attention has to be shared between different elements in the scene under tight temporal restrictions, where the eye tracking use is more central and challenging. For this reason, Zombie Runner has been designed and implemented as the common genre (of high impact) FPS. The character's running action is fully automated to reduce the number of actions the player had to memorize and control and, hence, reduce variability in the game evaluation phase.

3.1. System Configuration

The game, developed in Unreal Engine, has been with the following hardware configuration in mind (see Figure 2): a low-cost Gazepoint GP3 eye tracker; a Microsoft Xbox gamepad; and a 32 inch computer screen. The player sits in front of the computer screen, with the gamepad in hand. The eye tracker sits below the computer screen. Behind it, a laptop running the game is available for the research team during the evaluation sessions.
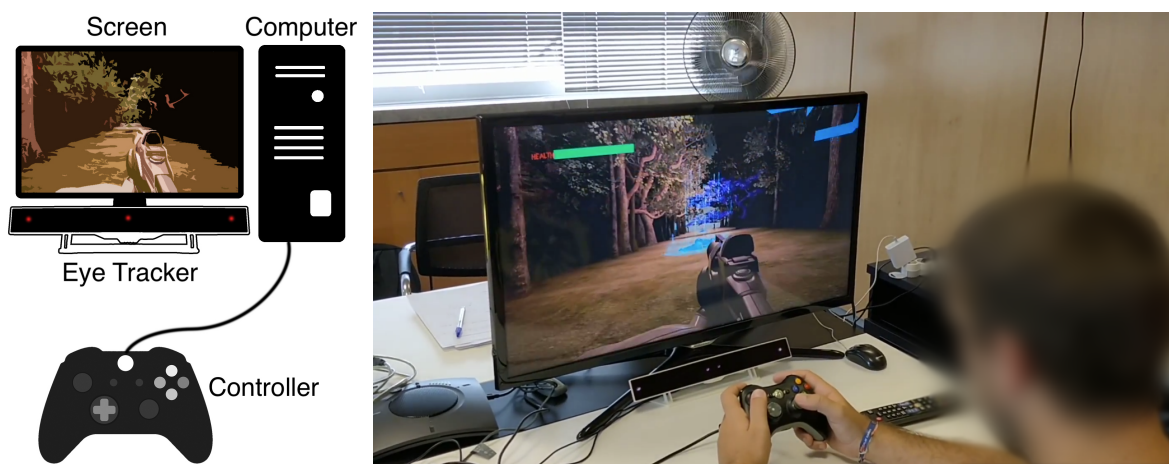

Figure 2. The hardware configuration and a test subject during an evaluation session.

In Zombie Runner, eye tracking is ensured by the Gazepoint's control software [32], which takes control of the mouse cursor to direct it towards the gaze point in the screen. Thus, Zombie Runner only needs to be sensitive to the mouse's cursor to determine the player's gaze in screen coordinates. The eye tracker's vendor claims a visual angle accuracy between 0.5° and 1.0° at 60 Hz, which we consider to be sufficient to assess which in-game elements are being attended by the player. A recent study confirms the vendor's claimed accuracy [33], provided that the user does not wear glasses, the testing environment offers proper lighting conditions, and a correct calibration procedure is carried out.

To attain a correct calibration, Zombie Runner uses a nine-point calibration procedure shipped with the Gazepoint's control software. Before playing the game, the user must confirm that the calibration is correct by running a calibration validation procedure. This procedure consists of gazing at the centre of several circles displayed in the screen (see Figure 3). This calibration is assumed to be successful (good enough) if the gaze never lands outside the circles being attended by the user.

Figure 3. The Gazepoint Control software's screen used to test the calibration results. The user is asked to look at the centre of each circle. The calibration is assumed to be successful if the gaze, represented in green, never lands outside the circle being attended by the user.

3.2. Game Rules and Mechanics

Zombie Runner follows the following set of rules. The main objective of the game is to ensure that the avatar survives for as long as possible while running along a corridor with a non controllable constant forward movement. The player can achieve this by killing enemies and by *noticing* elements in the scene (enemies and obstacles). The player can kill enemies by aiming and sooting the avatar's gun via the gamepad. The player can *notice* the several elements in the scene by actively looking at them on the screen for a sufficient amount of time. The gaze of the player is estimated with the eye tracker. An element noticed by the player is also noticed by the avatar.

If the avatar approaches a previously *noticed* obstacle, it will automatically avoid that obstacle (i.e., jump over a rock or dodge a hanging tree branch). If this obstacle were not noticed early enough, it will not be seen by the avatar, which will result in a collision and subsequent avatar's health decrement. A shot enemy that was noticed dies instantly, whereas it will only be hurt if unnoticed. This intends to simulate the lack of shot accuracy resulting from an insufficient focus on the enemy. If a noticed enemy approaches the avatar, it will attack, causing the avatar to lose health. If the enemy was not noticed early enough, its attack will cause instant avatar's death. The idea is that the avatar is able to dodge the attack provided the attacker had been noticed. The avatar dies when its health reaches zero, leading to the game being over. Figure 4 depicts the state diagrams associated to these game rules.
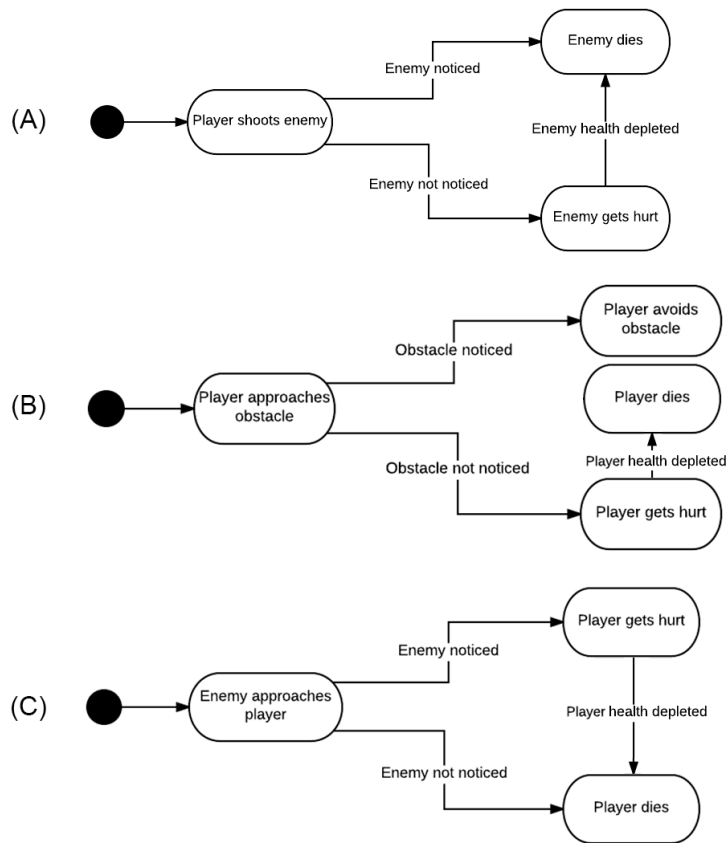
Figure 4. State diagrams for: when the avatar shoots an enemy (**A**); when the avatar approaches an obstacle (**B**); and when an enemy approaches the avatar (**C**). The arrows indicate state transitions and the text in them are the conditions necessary to trigger another state of the interaction.

3.3. Game Implementation

Zombie Runner was developed on Unreal Engine. The programming of Zombie Runner was split between C++ and Blueprints, the Unreal's visual scripting language. C++ was used for the core algorithms while Blueprints was used to program actor behaviour, such as enemies and obstacles. The game is based on the Unreal's FPS template, which already implements the expected behaviour for a generic game of the genre, including shooting, walking, and aiming mechanisms.

The corridor in which the game takes place is made of an unidimensional array of 3D tiles (see Figure 5), which are procedurally generated and removed from the scene once the avatar passes by the them. Tiles are comprised of one plane for the floor, two planes for the side walls, an array of marker points distributed randomly on the sides of the floor (not in the region in which the avatar will be crossing), and an array of tree models. Every time a tile is spawned, the array of marker points is traversed. For each marker, there is a 66% probability of spawning a tree (non-obstacle) in its position, plus a small random offset. This tree is spawned with a random rotation. The tree's height is also randomly set between reasonable values. The result are sufficiently different tiles that, when placed in succession, give the feeling of a dense and varied forest on each side of the player's sight. Figure 5 shows three different results of this algorithm applied to the tile generation.
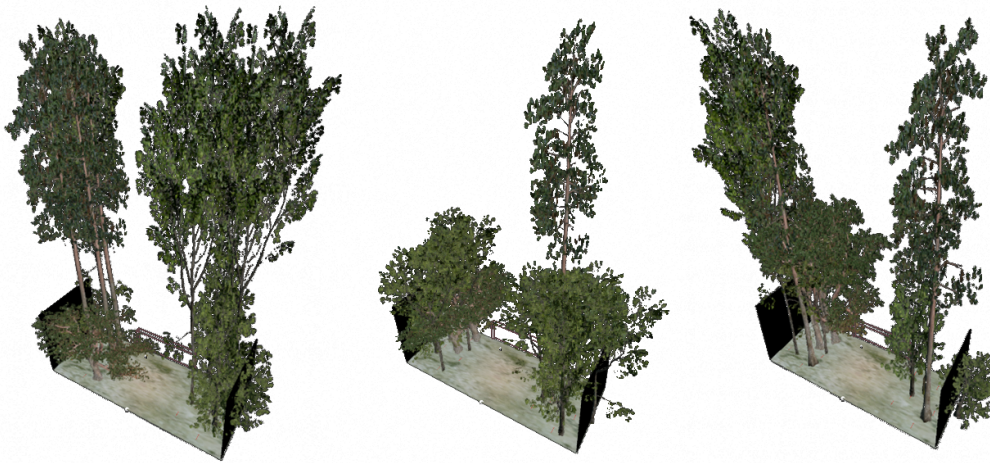
Figure 5. Tiles populated with procedurally placed obstacles. Note that the tree branches in the central region of the tiles are implemented as small rotated trees.

Fifteen tiles are initially generated. After the eighth tile is spawned, obstacles or enemies can start being spawned with them. This value was achieved by trial and error and produces an initial tile generation that gives enough time for the player to settle in and prepare for the incoming obstacles and enemies. After this initial generation, all subsequent tile generation is recursively and procedurally handled by the tiles themselves. An obstacle (i.e., a rock or an hanging tree branch) or an enemy has a 33% probability of being procedurally spawn in each new tile. If it does occur, there is a 55% probability for either spawning an obstacle or an enemy. A spawned enemy has a 20% probability of being spawned as a *runner*, which has double the speed of a *walker*, the default behaviour of an enemy. These percentages have been tuned according to a set of informal tests to ensure that the game was playable without training.

An obstacle or an enemy can be spawned in three different regions in the tile (see Figure 6): central region, left region, or right region. If the previous spawned obstacle or enemy was spawned in either the left or the right regions, the new one will be spawned in the central region. This adds variety to the game and avoids objects of the same type to be spawned close to each other. If the previous spawned obstacle or enemy was spawned in the central region, the new one will be spawned in either left or right regions, depending on the player's gaze. Concretely, the obstacle or enemy is spawn in the region of the tile least gazed by the player during the time spent running over the two previous tiles. This forces the attention of the player to be often alternating between regions, thus increasing the challenge. When the obstacle is spawned in the central region, its asset is a rock, being a tree branch when spawned on one of the side regions. Tree branches are implemented as laying small trees (see Figure 5).
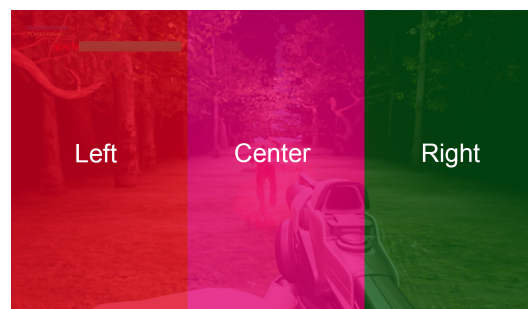

Figure 6. Tile/screen regions used to determine where to spawn an obstacle or enemy.

Zombie Runner relies on ray casting to determine which *object* (term hereafter used to generally refer to obstacles and enemies) present in the virtual scene is being attended by the player, given the gaze position in the screen. Assuming that the avatar's camera (the one rendering the images presented to the player) is located at world coordinates **o** and the player is gazing towards the screen's local coordinates **s**, the system determines which object is being attended by the player by casting into the scene a parametric ray $\mathbf{r}(t) = \mathbf{o} + t(\Phi(\mathbf{s}) - \mathbf{o})$, where the function $\Phi(\cdot)$ transforms a point from screen coordinates to world coordinates, given the virtual camera's field of view and pose. The closest intersected object is taken as the one being attended by the player.

Due to the rapid nature of FPS games (which induces frequent gaze shifts) and the limited accuracy of low-cost eye tracking technology, demanding for gaze fixations to occur with high certainty (i.e., to lock in a given object for a significant amount of time) before accepting that a given object was attended by the player could result in many false negatives. Here, a false negative means penalizing the player because a given object was not attended when, in practise, the player feels that the object was attended (even if only momentarily). These events are more harmful to the player's engagement level than the other way around, that is, to erroneously assume that the player has attended to a not actually attended object. Hence, the approach followed in Zombie Runner is to label a given object as noticed if the gaze of the player intersects that same object over an *accumulated* (i.e., time is not reset when gaze leaves the object) period of 0.5 s. This period was obtained from a set of informal tests. The approach of considering accumulated time ensures that an object is marked as attended/noticed even if the player frequently gazes across several objects or the eye tracker estimates jitter around the player's gaze, meaning it frequently lands off the attended object. Thus, this approach implements a filtering process that trades-off false positives and false negatives in a way that suits the game's needs.

To speed up computation, ray–object intersections (for determining which object is being attended by the player) are tested using pre-computed accessory intersection bounding boxes, rather than using the objects' triangular meshes directly. That is, instead of testing ray intersections against the several triangles present in an object's mesh, intersections are tested against a bounding box properly placed in front of the object. Additional bounding boxes are also associated to all objects present in the scene to speed-up avatar–object intersection tests. Again, instead of testing intersections between the avatar's and the objects' meshes, intersections are tested between the avatar and two bounding boxes, one placed in front of the object and another placed behind the object. The former is used to detect the moment the avatar is close enough to the object to trigger a proper interaction (e.g., enemy attack and obstacle dodging) and the latter to detect the moment the avatar passes beyond the object, allowing the object to be removed from the scene. Figures 7 and 8 depict the intersection bounding boxes employed in obstacles and enemies, respectively. The sizes of the intersection bounding boxes had to be enlarged so as to accommodate the inaccuracy of the eye tracker. These adjustments were carried out during a set of informal tests.
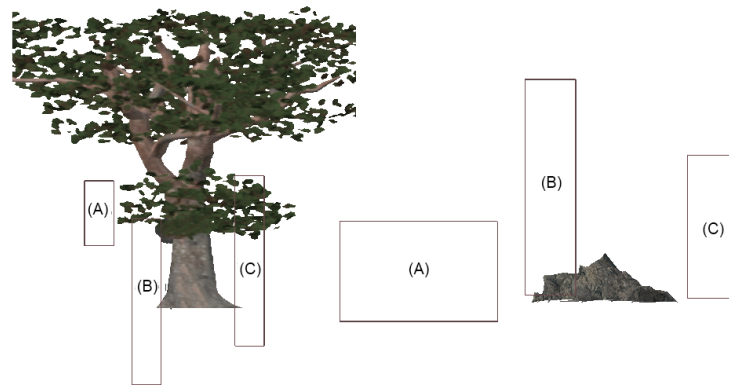
Figure 7. A lateral view of the meshes composing the two types of obstacles and their associated three intersection bounding boxes, assuming that the avatar approaches the obstacles from the left. Boxes labelled as (**A**) are used for detecting ray–obstacle intersections (to determine which object is being attended by the player). In the case of the tree, box (**A**) surrounds the tree's horizontal hanging branch, that is, the actual obstacle to the avatar (see Figure 5). Boxes labelled as (**B**) are used to detect the moment the avatar is about to collide against the obstacle. Boxes labelled as (**C**) allow the system to detect the moment the avatar passes beyond the obstacle.
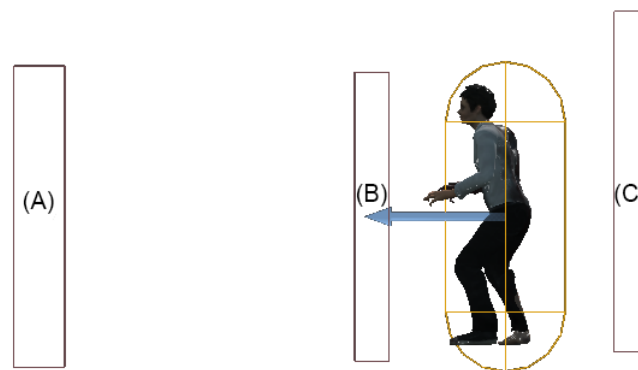
Figure 8. A lateral view of the mesh composing the enemy and its associated three intersection bounding boxes, assuming that the avatar approaches the enemy from the left. Box labelled as (**A**) is used to detect the moment the avatar is close enough to the enemy to trigger an enemy attack. Box labelled as (**B**) is used for detecting ray–enemy intersections (to determine which object is being attended by the player) as well as bullet-enemy intersections. Box labelled as (**C**) allow the system to detect the moment the avatar passes beyond the enemy.

All assets (e.g., obstacles) were freely obtained from the Unreal's dedicated store. The rigged 3D model and basic animations for the enemies were obtained from Adobe Mixamo. An Unreal Animation Blueprint was built as a state machine which defining the different animation states and transitions between them (see Figure 9). To provide the player with compelling visual feedback, the appearance of the objects changes when they are first gazed by the player and also later on when they become actually labelled as *noticed*. The first change consists in rendering the object in wireframe with shades of purple. The second change, more dramatic, consists in turning the object in light blue and in adding a particle effect representing a blue shock wave. Figure 10 shows the evolution of the materials and effects used on a rock obstacle through the process of being noticed.
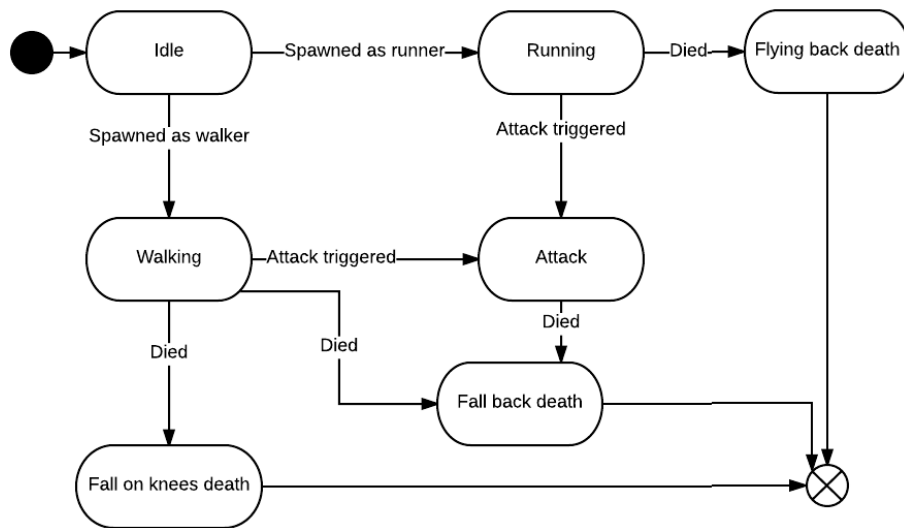
Figure 9. The enemy's animation blueprint. The death animation that is triggered from the Walking state is randomly selected to bring variety to the game. The arrows indicate state transitions and the text in them are the conditions necessary to trigger another animation state.
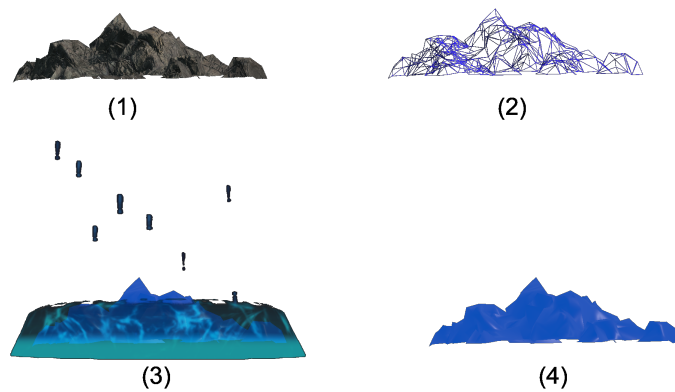

Figure 10. The material evolution of a rock obstacle being noticed.

4. Evaluation and Discussion

As mentioned, several informal tests were run to guide the development and tune the game's parameters. Then, the whole game was played by a set of ten people in formal game evaluation sessions to systematically assess what eye tracking technology brings to the game in terms of overall enjoyability.

4.1. Evaluation Method

Test sessions were carried out privately in a room without the presence of anyone but the participant and the research team. An email-based call for participants that considered themselves gamers was issued to an universe of 170 people. The first ten that responded to the email and complied with the requirements were selected for the test sessions. The participants had no previous knowledge about the game and experience. The ages of the ten participants spanned from 25 to 49 years old (see Figure 11), with different occupations such as software developer, quality assurance tester, and student. All participants were male, with no female subjects volunteering for the experience.
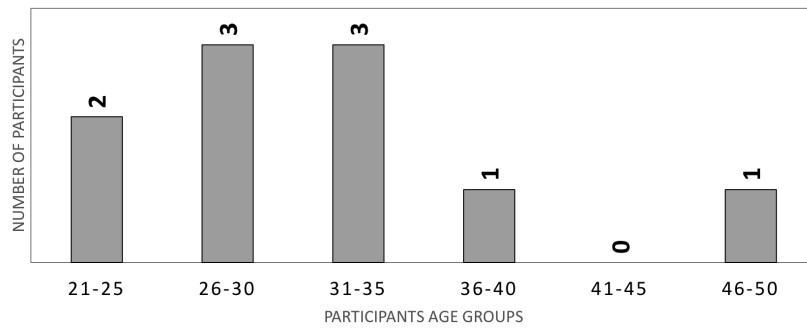
Figure 11. Distribution of participants according to their age group.

In addition to bottom-line data stored during the test sessions, we also enquired the participants with Game Experience Questionnaires (GEQ) [34], which has been widely applied in previous studies related to control inputs in computer games [35,36]. By using GEQ, we intended to evaluate if the use of eye tracker is enjoyable and comfortable for the player. We also intended to pinpoint possible advantages and disadvantages of the technology in the way it impacts the overall experience of the player and its relationship with the game, comparing the player experience with and without eye tracking.

4.1.1. Test Sessions

Figure 12 presents the flowchart followed in each test session (per participant). Each test session started with a brief questionnaire the participant had to fill in, regarding personal details, such as age and occupation, if the player had some degree of visual impairment, as well as classifying their experience as video game players, with the use of eye trackers, and with the use of gamepads in FPS games. As Figure 13 shows, the average participant had a considerable amount of gaming experience, little to no previous exposure to eye tracking technology, and was moderately experienced using gamepads in FPS games.
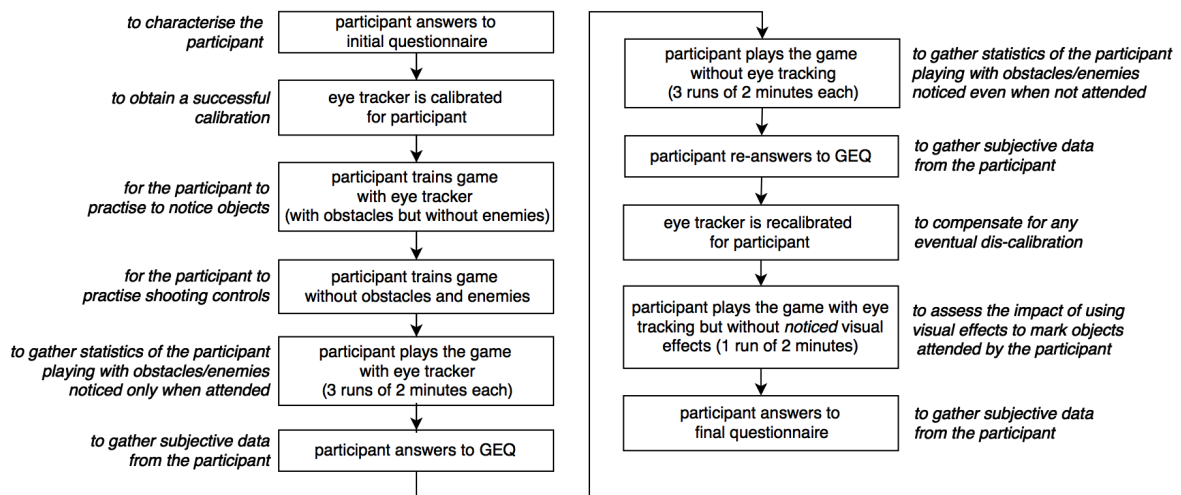

Figure 12. Flowchart applied in each test session (per participant). Each box represents a step in the test session and the accompanying off-box italicised text summarises the purpose of applying each step.
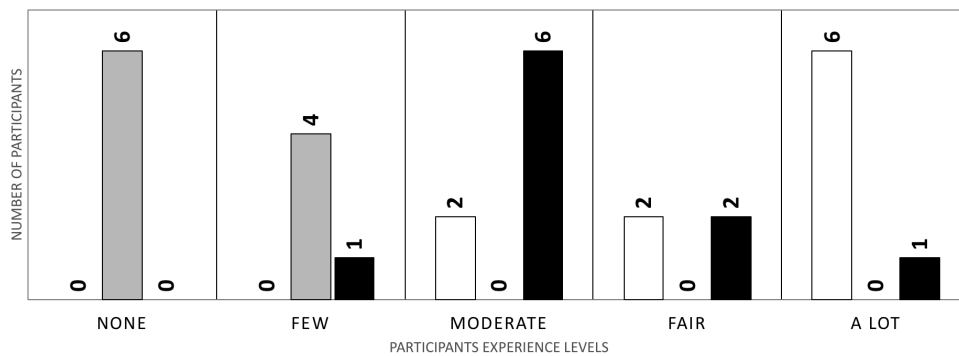
Figure 13. Distribution of participants according to their experience with video games (white bars), eye tracking technology (gray bars), and use of gamepads in FPS games (black bars).

To reduce any pleasing biases, the participants might have, we conveyed the message that we are agnostic regarding the use of eye tracking in games by means of a brief paragraph on the top of the questionnaire with "the advent of eye tracking technology some developers are including this technology in games" and "this study aims at providing scientific validation of the advantages and disadvantages of such inclusion in terms of gameplay and comfort to the user". Then, a brief explanation of the game's rules and objectives was given, followed by a briefing on the eye tracker calibration process. This process was run as many times as deemed necessary until the calibration was successful (as described in Section 3.1). When this successful calibration was achieved, the participants were asked to state their feelings towards the process and if they would imagine themselves doing it at home before playing a game.

To get the participant acquainted with the game, two separate runs were done, with each one only having an input form enabled (gamepad versus eye tracker). These runs had no time limit and did not count for the evaluation. On the first run, only the eye tracker was enabled and no enemies were spawned. The result was a play session with only obstacles being spawned. This allowed the participants to freely use their eyes to notice the obstacles in front of them and get used to this mechanic, without having to worry about the gamepad. The participants were also asked to state any false positives or obstacles that they had noticed but were not tagged as such by the game. On the second run, no obstacles or enemies were spawned. This allowed the participants to get acquainted with aiming and shooting with the gamepad, adapt to its sensibility and button scheme. The main objective was for the participant to feel comfortable with the different inputs and the test session would only advance when the participants confirmed they felt acquainted with the controls.

The participant was then asked to play the game in its original form, as described in Section 3, for three sessions of 2 min each. For each session, the ratio between enemies killed and spawned, the ratio between overall (enemies and obstacles) noticed count and overall spawned count, as well as the number of deaths experienced by the player, were registered. After these three sessions, the player was asked to fill the core and post-game modules of a Game Experience Questionnaire (GEQ) [34]. This questionnaire is filled by answering a set of questions whose answers are provided on a Likert-type scale scored from 0 to 4. Afterwards, the participant was asked to play another set of three sessions of 2 min, but this time with eye tracking disabled, which means that all obstacles and enemies were automatically labelled as noticed, with the participant only being required to shoot the enemies. With these three sessions over, another GEQ was filled in by the player. Then, the calibration process was performed again and the player was asked to play one 2 min session of the game but this time with the visual effects, that occur when the obstacle or enemy is noticed, disabled.

Finally, an end questionnaire was then handed to the participant. This questionnaire was more subjective and consisted of three questions: (1) "What do you think of the visual effects used on the first sessions, comparing with the last session you played?" (2) "How was the overall experience of playing the game?" (3) "Would you consider an eye tracker as part of your gaming setup and why?"

4.2. Experimental Results

All test sessions were concluded with success, in the sense that all test subjects were able to perform the tasks required from them.

4.2.1. Eye Tracker Calibration Process

The calibration process revealed itself as the most challenging step of the testing session. A calibration was deemed good enough if the accuracy in the areas where the player mostly interacts with in the game, i.e., the whole screen except the top corners (see Figure 14), was considered as successful according to the criterion defined in Section 3.1. Two participants had to remove their glasses so that their gaze could be properly detected. Figure 15 shows the distribution of participants according to the number of calibration tries each participant required to obtain a good enough calibration. As the table shows, a single calibration process was enough for half the participants.
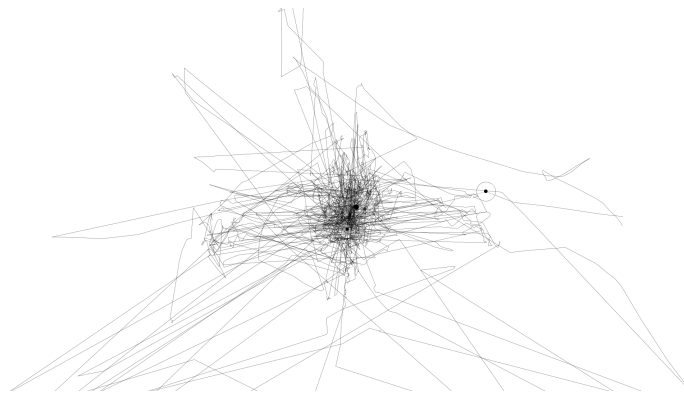

Figure 14. Gaze movement of a participant during a typical play-through of Zombie Runner. It can be observed that the player spends most of the time gazing at the central region of the screen.
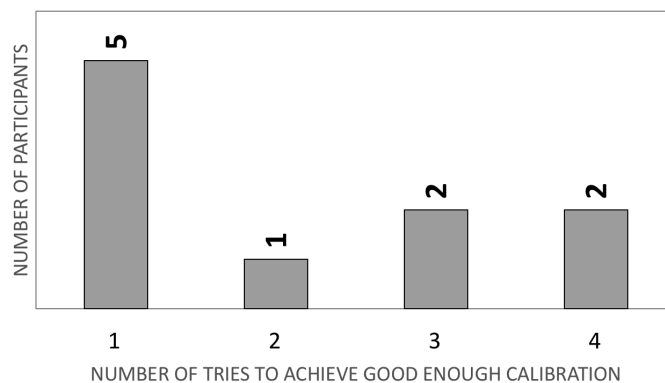
Figure 15. Distribution of participants according to the number of eye tracking calibration tries required to achieve a good enough calibration.

After achieving a calibration deemed successful, the participants were asked about their feelings towards it and if they would see themselves repeating this process at home before playing a game. From all participants, eight said that they would see themselves doing it, stating that a calibration process also had to be performed with other controllers. Three of these participants reported that the process was easy and fast, while others stated that it should be easier, but it is still acceptable. The other two participants did not see themselves calibrating an eye tracker each time they wanted to play, expressing that they wished the process was easier.

4.2.2. Play Sessions

The three 2 min play sessions produced results that suggest an approximation to the ideal game flow, with the participant being able to progressively learn how to use play more competently. This can be observed in Table 1, which shows the evolution along the three play sessions of: (1) the ratio between the number of killed enemies (by the player) and the total number of spawn enemies; (2) the ratio between the number of noticed (by the player) elements (obstacles and enemies) and the total number of spawn elements (obstacles and enemies); and (3) the number of avatar (player) deaths. The two ratios evolved positively, with the player improving in the tasks of killing enemies and noticing them, along with obstacles. The number of deaths went down abruptly from the first to the second session and then went up slightly on the third one, but to a value close to the lowest one obtained in the second session. We suspect this slight increase in deaths can be a consequence of the better results achieved by the participants in terms of killing enemies and noticing objects. To excel in these tasks, participants had to better coordinate the two input forms, which led to a greater risk of being killed.

Table 1. Results per play session (ratios represented as percentages), provided as STD $\pm$ STE, where STD stands for standard deviation and STE for standard error of the mean.

Play Session	Ratio [%] of Killed Enemies	Ratio [%] of Noticed Elements	Number of Deaths
1st	65.6 ± 4.4	49.7 ± 6.3	2.3 ± 0.2
2nd	69.5 ± 7.2	49.8 ± 6.1	1.3 ± 0.4
3rd	79.0 ± 5.0	52.7 ± 5.4	1.5 ± 0.2

The GEQ was filled out by the participants after these three play sessions. As mentioned, after these sessions, the participants were asked to play an additional set of three play sessions, this time without using eye tracker, i.e., with all obstacles and enemies automatically labelled as noticed. Table 2 compares, for each of the components of GEQ core module (competence, sensory and imaginary Immersion, flow, tension/annoyance, negative affect, and positive affect), the sessions with and without eye tracking. In the Competence component, the participants felt more competent and able to complete tasks without the eye tracker than with it, which can be explained by the higher degree of difficulty of the game when eye tracking is on. In the Sensory and imaginative immersion component, the participants reported higher scores with eye tracking, which means the feeling of immersion was felt at a greater level with the full game experience. The Flow component was also higher with eye tracking on, which reveals the participants felt a stronger feeling of game flow, a better balance within the game's Challenge, another component better graded with eye tracking on. participants reported higher levels on the component of Tension/annoyance with the eye tracker in use, which reveals the full game experience can lead to greater levels of frustration. This may be the result of the game being more challenging with eye tracking or of the issues related to the accuracy of the eye tracker (more on this below). Both positive and negative affects are slightly higher when eye tracking was on, suggesting that players were more emotionally invested in that condition. In both conditions, the positive affect is considerably higher than the negative one, suggesting that the game provides an overall positive experience.

Although GEQ scores are used in this study to compare two designs, we can also analyse the absolute meaning of the obtained scores. In this regard, although the GEQ scores are low when compared to the maximum of the scale, these are not substantially different from the ones obtained for full-fledged commercial FPS games, reported in a previous study [35]. The lower scoring can be explained by the awareness players have of the current state of the art in commercial games.

Table 2. Average values for the different components on the GEQ core module for the play sessions with and without Eye Tracking (ET). Results provided as STD ± STE, where STD stands for standard deviation and STE for standard error of the mean.

	GEQ Average Score	
Component	**With ET**	**Without ET**
Competence	2.22 ± 0.27	2.8 ± 0.25
Immersion	1.33 ± 0.3	0.93 ± 0.3
Flow	2.08 ± 0.24	1.68 ± 0.26
Tension	1.06 ± 0.28	0.3 ± 0.14
Challenge	1.56 ± 0.2	0.58 ± 0.19
Negative affect	0.68 ± 0.2	0.45 ± 0.21
Positive affect	2.48 ± 0.26	2.26 ± 0.22

The results were treated in the same way for the GEQ post-game module, whose results are summarised in Table 3. The differences in levels of positive and negative experience are too small (≈0.05) to allow any analysis from them. The more significant difference in Tiredness and Returning to reality components provide additional support to the idea that eye tracking renders the experience more immersive and engaging. The similar values obtained with and without the eye tracker suggest that the participant's perception of the overall experience was more shaped by the game itself than by the use or lack of the eye tracker. However, these results become more meaningful and easier to understand when compared with the answers the test subjects gave to the last set of informal questions (see below). The main goal of these questions was to extract from the participants more subjective perceptions they had from the game, which could help us understand the GEQ scores.

Table 3. Average values for the different components on the GEQ post-game module for the play sessions with and without Eye Tracking (ET). Results provided as STD ± STE, where STD stands for standard deviation and STE for standard error of the mean.

	GEQ Average Score	
Component	**With ET**	**Without ET**
Positive experience	1.27 ± 0.31	1.33 ± 0.28
Negative experience	0.13 ± 0.06	0.15 ± 0.1
Tiredness	0.5 ± 0.35	0.23 ± 0.12
Returning to reality	0.7 ± 0.26	0.5 ± 0.18

4.2.3. Informal Questions

The first question asked participants for their opinion about the used visual effects signalling which objects were labelled as noticed, that is, as an attention feedback mechanism. Nine of the ten participants stated *the importance of the effects as a means to give feedback to the player*, with some pointing out that *without effects the player may be forced to look more than needed as the player never knows if the obstacle or enemy was actually tagged as noticed*. The use of effects was also pointed out as *more rewarding to the players actions*. From all the participants, seven reported that *without the visual effects, the game is more immersive and the experience more realistic, making the way the player looks at things more natural*. This may suggest that being more immersive does not mean that a game is necessarily more rewarding. In order for it to be both rewarding and immersive, a different, more subtle attention feedback mechanism should be implemented as to avoid breaking the suspension of disbelief. The development of such a feedback mechanism (more subtle than the tested visual effects) still demands for additional research.

It is also worth studying the value of such an attention feedback mechanism when using highly accurate eye tracking technology. On the one hand, there is the possibility that users stop feeling the need for attention feedback as soon as they feel that the system is accurately tracking their attention. On the other hand, highly accurate eye trackers are expensive and, even those, are unable to fully predict attention deployment, as humans also exploit vision periphery to scan the environment. Some participants, to whom the eye tracker calibration was not fully successful, stated that *the visual effects used as attention feedback mechanism may induce frustration on them as they could see the discrepancy between where the eye tracker thought they were looking at and where they were actually looking at*. This issue calls for more robust eye tracking and calibration techniques if we wish to provide universal access to this technology.

To the second question about the overall experience of playing the game, three out of the ten participants complained about *having to be as static as possible to avoid the eye tracking de-calibration*. Although only two of the participants reported *problems with the game registering when they look at enemies and rock obstacles*, some participants complained about *hardware problems and the frustration of going to the process of calibration and then the technology still not working right*. A participant that had problems with getting the eye tracker to work while wearing glasses *wished that the technology was more prepared for people with glasses*. These issues highlight the biggest practical caveats on the application of eye tracking to video games (and to other related domains): the time-consuming calibration process and brittleness of the system when used in non-ideal scenarios. We expect that further research in eye tracking technology will produce more robust solutions, fostering their use in the wild. Regarding the game, a participant stated that *it is well designed*, also expressing *appreciation for the feedback given when the player loses life*. Another participant said that *the mechanic of noticing things is fun, but that the game lacks progression, having nothing new after the first minute*. This might suggest that a more complex game could have been a better fit for this test session, as it would avoid frustration resulting from lack of novelty. In fact, although not so relevant in the context of this study, in which the game sessions were setup to last only a couple of minutes, maintaining player engagement in long testing sessions may demand for more compelling game mechanics. One participant felt that *the time to set an obstacle or enemy as noticed is too long*, which leads to the consider that in future studies these timings should be learned for each player, taking into account the player-dependent uncertainty of the eye tracker. The experience was *classified as immersive* by three participants, with one of them stating enthusiastically that *eye tracking is amazing*.

With the goal of emphasising potential points of improvement, Table 4 provides a compilation of the complaints described in the two previous paragraphs, alongside the number of participants supporting these complaints. As the table highlights, most of the complaints are related to the limitations of current eye tracking affordable technology. The most frequent complaint concerns the lower level of immersion and realism induced by the presence of visual effects used to highlight noticed elements. However, as aforementioned, these visual effects had a positive impact on the overall experience, rendering it more rewarding. Hence, future studies are required to better analyse this (at least apparent) trade-off between rewarding experience and game immersion/realism in the context of gaze-directed gameplay.

Table 4. Number of participants agreeing with a given complaint.

Complaint	Nr. of Participants
Impact of noticed-related visual effects to feeling of immersion/realism	7
Need for being as static as possible for proper eye tracking operation	3
Failures in mis-registering enemies/obstacles as noticed	2
Time required for obstacles/enemies to be considered as noticed	1
Problems in tracking the eyes of people wearing glasses	1
Game lacks progression	1

To the question about whether the participants would consider the inclusion of an eye tracking camera in their gaming setup, the responses were mixed (see Table 5). From the entire group, two of the participants stated *they would not do it*, with reasons such as *it being another piece of hardware that has little application and would be quickly abandoned after the novelty effect wore off*. Fortunately, we expect eye tracking technology to become embedded in computing devices, reducing the limitations raised by these participants. Three other participants stated that *they would not in the current state but that they could try it in the future*, stating that *depending on the game it could facilitate precision tasks such as aiming in FPS games or passing the ball to another player in a football game*. This shows that participants see the value of eye tracking as a means to map player's and avatar's attention processes. The reasons these participants presented as to holding off in the adoption of the technology regarded its poor performance while wearing glasses, and the fact that it was not stable or precise, which allied with the calibration process ruined the experience. These issues were already raised by participants in the responses to the two previous questions (please refer to the three previous paragraphs). The other five participants said *they would adopt the technology*, stating that *it was a new form of interaction, that opened new possibilities and created more immersive experiences*.

Table 5. Distribution of responses to the question "Would you consider the inclusion of an eye tracking camera in your gaming setup?".

Response	Nr. of Participants
No	2
Maybe, when eye tracking becomes more reliable	3
Yes	5

These answers showed, along with the previous questionnaires, that the use of eye tracking in games has both pros and cons. On the positive side, the use of gaze-oriented gameplay provided a more immersive and richer experience, providing a better game flow. On the negative side, the technology's limitations raised feelings in the participants that were not desired. The calibration process and its results, along with some disbelief that eye tracking could have an important role in a computer game, are the main reasons for the participants being adamant about adopting the technology.

5. Conclusions and Future Work

This paper presented Zombie Runner, an endless runner FPS game, whose core mechanics and procedurally generated content are modulated by the player's gaze estimated with an affordable eye tracker. A set of testing sessions were carried out to assess the impact of eye-tracking in the player's satisfaction when playing Zombie Runner. The results obtained from testing sessions show that the use of eye tracking provides a more challenging and immersive experience to the player. These results complement previous studies, which were mostly focused on performance-based metrics when using eye tracking as a direct control input. Conversely, Zombie Runner exploits eye tracking as a

mechanism to match the avatar's attention model with the one of the player. Moreover, Zombie Runner also exploits eye tracking to guide the procedural content generation of the game's environment, contributing in an original way to the emerging field of Experience-Driven Procedural Content Generation (EDPCG) [24].

During the evaluation, a strong correlation between problems that surfaced with the eye tracker calibration and participants' overall experience was observed. Participants for whom the hardware worked without major flaws reported better levels of satisfaction when contrasting with participants for whom the calibration process was not perfect or took a longer time. Among the participants' complaints about the eye tracking technology, many were related the need to have the head mostly static during the play session, the calibration process requiring repetitions to meet the required accuracy, and overall eye tracker's lack of precision. These complaints show that the affordable eye tracking technology still has to grow and develop until it is in a state that can be accepted by the overall gaming community. However, positive player's experience when calibration was easily attained is a sign that the method will be a valuable asset for the game designer as soon as the eye tracking technology matures.

As future work, we intend to validate the use of eye tracking in other types of video games and allow for free avatar's movement. We also intend to extend this testing framework, including the philosophy behind the way the player's attention is integrated in the core gameplay, to games with other types of camera perspective, such as third-person games. Finally, we also intend to perform a more intensive set of tests, enlarging the participant population, to obtain more robust statistics, in particular to allow an in-depth correlation analysis between gaze patterns, game progression data, player profiles, and GEQ scores.

Author Contributions: J.A. participated in the design of the study, developed the software, participated in the analysis of the results, and participated in the preparation of the manuscript. P.S. participated in the design of the study, participated in the design of the developed software, participated in the analysis of the results, participated in the preparation of the manuscript, and coordinated all activities.

References

1. Javal, L. Essai sur la physiologie de la lecture. *Ann. d'Ocul.* **1879**, *82*, 242–253.
2. Holzman, P.S.; Proctor, L.R.; Hughes, D.W. Eye-Tracking Patterns in Schizophrenia. *Science* **1973**, *181*, 179–181.
3. Mcmullen, D.P.; Hotson, G.; Katyal, K.D.; Wester, B.A.; Fifer, M.S.; Mcgee, T.G.; Harris, A.; Johannes, M.S.; Vogelstein, R.J.; Ravitz, A.D.; et al. Demonstration of a Semi-Autonomous Hybrid Brain-Machine Interface Using Human Intracranial EEG, Eye Tracking, and Computer Vision to Control a Robotic Upper Limb Prosthetic. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2014**, *22*, 784–796.
4. Gomes, J.; Marques, F.; Lourenço, A.; Mendonça, R.; Santana, P.; Barata, J. Gaze-Directed Telemetry in High Latency Wireless Communications: The Case of Robot Teleoperation. In Proceedings of the 42nd Annual Conference of the IEEE Industrial Electronics Society (IECON), Florence, Italy, 24–27 October 2016; pp. 1–6.
5. Krugman, D.M.; Fox, R.J.; Fletcher, J.E.; Fischer, P.M.; Rojas, T.H. Do adolescents attend to warnings in cigarette advertising?: An eye-tracking approach. *J. Adv. Res.* **1994**, *34*, 39–39.
6. Smith, J.D.; Graham, T.C.N. Use of Eye Movements for Video Game Control. In Proceedings of the 2006 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology-ACE '06, Hollywood, CA, USA, 14–16 June 2006.
7. Leyba, J.; Malcolm, J. Eye tracking as an Aiming Device in a Computer Game. In *Course work (CPSC 412/612 Eye Tracking Methodology and Applications by A. Duchowski)*; Clemson University: Clemson, CA, USA, 2004.
8. Isokoski, P.; Martin, B. 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, Turin, Italy, 4–5 September 2006; pp. 78–81.
9. Isokoski, P.; Joos, M.; Spakov, O.; Martin, B. Gaze controlled games. *Univ. Access Inf. Soc.* **2009**, *8*, 323–337.
10. Dorr, M.; Pomarjanschi, L.; Barth, E. Gaze beats mouse: A case study. *PsychNol. J.* **2009**, *7*, 197–211.

11. Antunes, J.; Santana, P. Gaze-Oriented Gameplay in First-Person Shooter Games. In Proceedings of the 24th Portuguese Meeting on Computer Graphics and Interaction (EPCGI), Guimarães, Portugal, 12–13 October 2017; pp. 231–232.

12. Lukander, K. Measuring Gaze Point on Handheld Mobile Devices. In Proceedings of the Extended Abstracts of the 2004 Conference on Human Factors and Computing Systems-CHI '04, Vienna, Austria, 24–29 April 2004.

13. Arai, K.; Mardiyanto, R. Eye-based HCI with Full Specification of Mouse and Keyboard Using Pupil Knowledge in the Gaze Estimation. In Proceedings of the 2011 Eighth International Conference on Information Technology: New Generations, Las Vegas, NV, USA, 11–13 April 2011.

14. Wells, W.C. An essay upon single vision with two eyes together with experiments and observations on several other subjects in optics. T. Cadell, in the Strand 1792.

15. Dodge, R.; Cline, T.S. The angle velocity of eye movements. *Psychol. Rev.* **1901**, *8*, 145–157.

16. Singh, H.; Singh, J. Human Eye tracking and Related Issues: A Review. *Int. J. Sci. Res. Publ. (IJSRP)* **2012**, *2*, 1–9.

17. Shell, J.S.; Vertegaal, R.; Cheng, D.; Skaburskis, A.W.; Sohn, C.; Stewart, A.J.; Aoudeh, O.; Dickie, C. ECSGlasses and EyePliances. In Proceedings of the Eye Tracking Research and Applications Symposium on Eye Tracking Research and Applications-ETRA'2004, San Antonio, TX, USA, 22–24 March 2004.

18. Smith, J.D.; Vertegaal, R.; Sohn, C. ViewPointer. In Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology-UIST '05, Seattle, WA, USA, 23–27 October 2005.

19. Perreira da Silva, M.; Courboulay, V.; Prigent, A. Gameplay Experience based on a Gaze Tracking System. In Proceedings of the Communication by Gaze Interaction IST FP6 European Project (COGAIN), Leicester, UK, 3–4 September 2007.

20. Shaker, N.; Togelius, J.; Nelson, M. *Procedural Content Generation in Games: A Textbook and an Overview of Current Research*; Springer: Berlin/Heidelberg, Germany, 2014.

21. Browne, C.; Yannakakis, G.N.; Colton, S. Guest Editorial: Special Issue on Computational Aesthetics in Games. *IEEE Trans. Comput. Intell. AI Games* **2012**, *4*, 149–151.

22. Yannakakis, G.N. Game AI Revisited. In Proceedings of the 9th Conference on Computing Frontiers, Cagliari, Italy, 15–17 May 2012.

23. Gow, J.; Baumgarten, R.; Cairns, P.; Colton, S.; Miller, P. Unsupervised Modeling of Player Style With LDA. *IEEE Trans. Comput. Intell. AI Games* **2012**, *4*, 152–166.

24. Yannakakis, G.N.; Togelius, J. Experience-Driven Procedural Content Generation. *IEEE Trans. Affect. Comput.* **2011**, *2*, 147–161.

25. Iida, H.; Takeshita, N.; Yoshimura, J. A Metric for Entertainment of Boardgames: Its Implication for Evolution of Chess Variants. In *Entertainment Computing*; Springer: Boston, MA, USA, 2003; pp. 65–72.

26. Spronck, P.; Sprinkhuizen-Kuyper, I.; Postma, E. Difficulty Scaling of Game AI. In Proceedings of the 5th International Conference on Intelligent Games and Simulation (GAME-ON 2004), Ghent, Belgium, 25–27 November 2004; pp. 33–37.

27. Andrade, G.; Ramalho, G.; Santana, H.; Corruble, V. Automatic Computer Game Balancing. In Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems-AAMAS '05, Utrecht, The Netherlands, 25–29 July 2005.

28. Yannakakis, G.N.; Hallam, J. Towards Optimizing Entertainment In Computer Games. *Appl. Artif. Intell.* **2007**, *21*, 933–971.

29. Olesen, J.K.; Yannakakis, G.N.; Hallam, J. Real-Time Challenge Balance in an RTS Game Using rtNEAT. In Proceedings of the 2008 IEEE Symposium On Computational Intelligence and Games, Perth, Australia, 15–18 December 2008.

30. Lankveld, G.V.; Spronck, P.; Herik, H.J.V.D.; Rauterberg, M. Incongruity-Based Adaptive Game Balancing. In *Lecture Notes in Computer Science Advances in Computer Games*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 208–220.

31. Sorenson, N.; Pasquier, P. Towards a Generic Framework for Automated Video Game Level Creation. In *Applications of Evolutionary Computation Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 131–140.

32. Gazepoint Control. Available online: http://www.gazept.com (accessed on 19 October 2017).

33. Zugal, S.; Pinggera, J. Low–Cost Eye–Trackers: Useful for Information Systems Research? In *International Conference on Advanced Information Systems Engineering*; Springer: Cham, Switzerland, 2014; pp. 159–170.

34. IJsselsteijn, W.A.; de Kort, Y.A.W.; Poels, K. *The Game Experience Questionnaire*; Technical Report; Technische Universiteit Eindhoven: Eindhoven, The Netherlands, 2013.

35. Drachen, A.; Nacke, L.E.; Yannakakis, G.; Pedersen, A.L. Correlation between Heart Rate, Electrodermal Activity and Player Experience in First-Person Shooter Games. In Proceedings of the 5th ACM SIGGRAPH Symposium on Video Games-Sandbox '10, New York, NY, USA, 26–30 July 2010.

36. Gerling, K.M.; Klauser, M.; Niesenhaus, J. Measuring the Impact of Game Controllers on Player Experience in FPS Games. In Proceedings of the 15th International Academic MindTrek Conference on Envisioning Future Media Environments-MindTrek '11, Tampere, Finland, 28–30 September 2011.