Exploring the Impact of a Spatial Tracker on Object Localization in Video Games for Visually Impaired Individuals

Daniel de Barros Rodrigues
Instituto Superior Técnico de Lisboa
Lisboa, Portugal
daniel.rodrigues@tecnico.ulisboa.pt

ABSTRACT

As the use of virtual environments becomes more relevant nowadays, accessibility for individuals with visual impairments remains a significant concern. One major obstacle for these individuals is the difficulty in locating objects within virtual environments like video games.

Having this in consideration, the aim of this project is to explore the feasibility of designing a mechanism that improves object localisation for individuals with visual impairments in virtual environments using auditory and haptic stimuli as cues to assist the users. In order to validate the effectiveness of this mechanism, we developed a video game prototype using *Unity* graphic engine and game design techniques that enhance auditory and haptic perception.

Ten individuals with visual impairments participated in a three-phase summative user evaluation of the prototype. The initial phase introduced the prototype to the participants, followed by an instructive second phase where users were trained on the usage of the designed mechanism. The final phase echoed the first but now with the designed mechanism incorporated.

Upon analysis of the data garnered from these evaluations, it was observed that the developed mechanism did not yield a significant difference in the participants' ability to locate objects within the virtual environment, regardless of its presence or absence.

KEYWORDS

Visual Impairment, Sound Localisation, Virtual Environments, Video Games, Accessibility

1 INTRODUCTION

The field of Information Technologies, particularly in entertainment and virtual environments, has witnessed unprecedented growth in recent decades. Such virtual environments, including virtual reality and video games, hold transformative potential across sectors such as education and health. Despite the strides made, the accessibility of these virtual environments for individuals with visual impairments remains a challenge. This limitation hampers the potential benefits these individuals could gain from the use of virtual environments, especially in the context of video games, which predominantly rely on visual cues [1–6].

A pertinent analysis of the American census (2002) revealed a deficiency in accessible gaming options for people with motor or visual impairments [7]. While advancements have been made in the realm of virtual environment accessibility for these individuals, challenges persist. A primary issue is spatial location within virtual environments, as the use of sound has proven only partially helpful, specifically in object identification tasks.

The goal of this project is to address the difficulty individuals with visual impairments face in locating objects in virtual environments, through the implementation of a *Spatial Tracker* mechanism within a video game prototype. The mechanism's effectiveness was validated in a experimental session with visually impaired individuals.

2 RELATED WORK

Accessibility in virtual environments is essential to promote inclusive experiences, particularly for visually impaired individuals. Understanding the unique needs and challenges faced by these individuals is vital to meet the objectives of this project. The investigation of virtual environments accessibility, video games designed for visually impaired individuals, and key technologies like sound and spatial localisation, are fundamental in crafting solutions to meet these requirements.

In the next section, an analysis is performed on studies conducted on accessibility to virtual environments for visually impaired individuals, with an emphasis on video games.

2.1 Accessibility of Virtual Environments for Individuals with Visual Impairments

Virtual environments offer numerous opportunities for entertainment, education, and therapy, but accessibility remains a challenge for individuals with visual impairments. Recent developments focus on assistive technologies, such as audio cues and haptic feedback.

Research into game accessibility for this group of individuals indicates a significant gap in inclusivity due to a lack of non-visual cues and interaction mechanisms. Audio description has been identified as a potential tool for enhancing game accessibility. It involves narrating visual elements within games, thereby conveying critical information that players with visual impairments would otherwise miss [8].

An innovative approach to improve virtual environment accessibility was the creation of an intelligent virtual cane [9] using a *Nintendo Wii* controller. By translating visual information into haptic and auditory feedback, the virtual cane enables visually impaired users to navigate and interact with virtual spaces. The successful application of this device underscores the potential of combining haptic feedback and sound localisation for providing spatial information in virtual environments.

In another study a system called *NavStick* [10] was developed with the intention of allowing visually impaired users to explore virtual environments. Using a standard game controller with a custom software interface providing auditory and haptic feedback, users can look around in virtual spaces. Participants have effectively navigated and completed tasks within games using this system, demonstrating its potential.

2.2 Video Games for Individuals with Visual Impairments

This section presents video games specifically designed for individuals with visual impairments, illustrating how challenges of accessibility can be overcome. Three distinct games that employ auditory, haptic, and other non-visual cues are analysed, *AudioDoom* [11], *Blind Hero* [12], and *A Blind Legend*[1].

AudioDoom is a prototype based on the classic game *DOOM*[2], repurposed for cognitive testing. It emphasises audio-based and spatial sound experiences, designed to evaluate if kids visually impaired aged between eight and eleven years old could construct spatial cognitive structures. The game tests these abilities through interactive hyperstories and physical structure building. Results indicate that participants could mentally render navigable spatial structures using spatialized sound.

Blind Hero uses haptic stimuli as a substitute for visual cues. Building on the popular rhythm game, *Guitar Hero*[3], this game employs a glove fitted with vibrating motors on each finger to indicate when a note should be played. The study adheres to the Beyond Accessibility To Efficiency (BATE) principle, prioritising accessible technology from the design stage. Testing results suggest that haptic feedback enhances game play and user experience in sound cues based games, with players adjusting their strategies to different haptic patterns.

A *Blind Legend* is an audio game funded through a public crowdfunding campaign. Using binaural sound and a well-defined control scheme, the game immerses players in a completely audio-based experience. The player's virtual actions are guided by sound cues while others are provided by the game protagonist's daughter. The game's success and acclaim underline its innovation and pioneering role in games designed for visually impaired players.

2.3 Graphic Engine

The evolution and accessibility of graphic engines like *CryEngine*[4], *Unity*[5], and *Unreal Engine*[6] have simplified the development of 3D virtual environments and in particular video games.

Among the popular graphic engines, *CryEngine* is powerful but complex to use, with limited documentation and less support for spatial audio frameworks.

Unity, on the other hand, stands out for its simplicity, extensive documentation and easy integration with major spatial audio frameworks. *Unity*'s package technology simplifies the integration process, allowing developers to easily incorporate spatial audio solutions into their projects.

Unreal Engine is also well-documented and supports frameworks like *Resonance Audio* and *Steam Audio*, but choosing it would require a better mastery of the *C++* language. Considering the ease of integration with spatial audio frameworks, simple management of controller motor vibration and comprehensive documentation, the decision was made to proceed with *Unity* as the graphic engine for the project.

2.4 Spatial Audio

Spatial audio technology seeks to mimic the human brain's sound localisation process to create immersive auditory illusions in a virtual environment. Key mechanisms of this technology include the Interaural Time Difference (ITD), Interaural Level Difference (ILD) and Head-Related Transfer function (HRTF). ITD, involves the difference in sound arrival times at two different locations [13, 14], while ILD is about the difference in sound intensities [15]. Together, these two mechanisms make up the duplex theory of sound localisation [16].

The duplex theory, however, only addresses lateral localisation issues. To overcome the limitations of the duplex theory, such as the cone of confusion [17], HRTF have been introduced. They describe how a sound from a specific location reaches the ear and can be used to synthesise a binaural sound [18, 19]. They account for the influence of human anatomy (head and ears) on sound waves, which assists in sound localisation.

Spatial audio technology is increasingly being used in multimedia content creation due to its potential to provide immersive user experiences. It's becoming particularly prominent in the video game industry, where producers are incorporating this technology into their products[7][8][9][10].

After careful analysis, Application Program Interface (API) were identified as suitable options for the project. *Oculus Spatializer*[11] is compatible with major operating systems and game engines like *Unity* and *Unreal Engine*. It offers a user-friendly interface, comprehensive documentation and examples. However, it lacks the implementation of the occlusion effect.

Resonance Audio[12] supports various operating systems and game engines. It provides realistic spatial audio experiences with features like real-time spatialization and sound occlusion. The documentation includes step-by-step guides and sample code. However, the occlusion effect is implemented as a basic switch, which may affect the user experience.

Steam Audio[13], works with popular game engines and operating systems. It offers real-time sound propagation and dynamic acoustic effects. Although the documentation is limited, it provides features like HRTF spatialization and dynamic reverb. The drawback is the need for integration with the graphics engine's geometry system.

Software choice was informed by the previous study on the prevalent graphics engines and spatial audio APIs, with *Unity* emerging as the preferred option due to its comprehensive documentation and widespread use. Initially, the *Resonance Audio* API was chosen

[1] http://www.ablindlegend.com/en/home-2/ (as consulted in May 2023)
[2] https://slayersclub.bethesda.net/en (as consulted in May 2023)
[3] https://support.activision.com/guitar-hero-live (as consulted in May 2023)
[4] https://www.cryengine.com/ (as consulted in May 2023)
[5] https://unity.com/ (as consulted in May 2023)
[6] https://www.unrealengine.com/en-US (as consulted in May 2023)
[7] https://playvalorant.com/en-us/news/game-updates/valorant-patch-notes-2-06/ (as consulted in May 2023)
[8] https://blog.counter-strike.net/index.php/2016/12/17260/ (as consulted in May 2023)
[9] https://hellblade.com/posts/development-diary-15-binaural-audio-test (as consulted in May 2023)
[10] https://tinyurl.com/gow-3daudio (as consulted in May 2023)
[11] https://developer.oculus.com/resources/ (as consulted in May 2023)
[12] https://resonance-audio.github.io/resonance-audio/ (as consulted in May 2023)
[13] https://valvesoftware.github.io/steam-audio/ (as consulted in May 2023)

for spatial audio solution integration, but due to its simplistic implementation of occlusion and limitations in real-time sound effects recalculations, it was later replaced by the *Steam Audio*.

3 PROTOTYPING THE SPATIAL TRACKER

With an understanding of the basic requirements needed to begin the prototype development process, it was essential to prioritise the Spatial Tracker system to ensure its development would not be constrained by potential restrictive decisions. However, it was also important to consider the requirements and feasibility of integrating the mechanism into the video game prototype.

The following section provide a analysis of the different approaches taken to define the spatial tracker mechanism, leading to its final implementation.

3.1 Using Ray Casting for Collision Detection

The development of the mechanism was influenced by the knowledge obtained from the analysis conducted on the previously mentioned works. The initial concept emerged from the idea of stimulating both auditory and haptic senses, as individuals with visual impairments heavily rely on these stimuli in their daily activities [11, 12]. The system aimed to integrate spatial localisation mechanisms with sensory stimulation mechanisms. This led to the idea of using Sound Navigation and Ranging (SONAR) as a metaphor for the mechanism, which guided the design process.

The initial approach mirrored techniques seen in video games like *Alien Isolation*[14], where a tool served as a motion tracker. Similar to the SONAR, the tool emits a signal every few seconds that reflects off the nearest objects and returns to the tracker, thus showing the enemy's location on the tool's screen [20]. The intention was to develop a similar behaviour by using ray casting to cast multiple rays in a 360° degree range to detect objects in the scene, with sound and vibration intensity corresponding to the distance of the object. However, we found ray casting could cause performance issues and consequently degrade the user experience due to its computational weight.

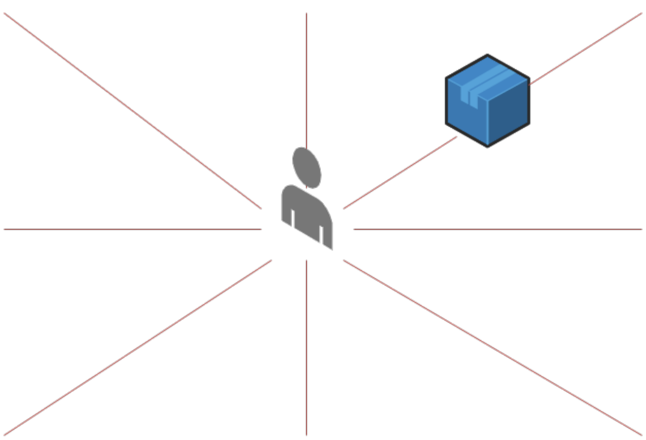

Figure 1: Ray casting technique

[14] https://www.feralinteractive.com/en/switch-games/alienisolation/ (as consulted in May 2023)

3.2 Using Sphere Colliders for Collision Detection

In the second approach, *Sphere Colliders* (refer to Figure 2) replaced ray casting for collision detection, leading to an updated design of the *Spatial Tracker* as can be observed in Algorithms 1 and 2. This adjustment simplified collision detection and offered more efficiency in handling large or complex objects.

```
begin
    // Verify if there is a intersection between
       the ObjectTracker and the Target
    if ObjectTracker.SphereCollider.bounds.Intersects(Target.BoxCollider) then
        // if there is update the stimuli signals
        UpdateStimuliSignal(Player.position, target.position)
    end
end
```

Algorithm 1: ObjectTracker - *Update* method

```
Data: Player position pp, Target Position tp
begin
    newDistance ⟵ Distance from pp to tp
    distancePercentage ⟵ newDistance / distanceOnCollision
    // If the target is getting closer, update
       the signals
    if newDistance < previousDistance then
        previousDistance ⟵ newDistance
        // Validate the according intensity
        if distancePercentage > 80% and distancePercentage ≤ 100% then
            signalSound ⟵ Very Low Intensity Signal
        else
            if distancePercentage ≥ 65% and distancePercentage < 80% then
                signalSound ⟵ Low Intensity Signal
            else
                if distancePercentage ≥ 50% and distancePercentage < 65% then
                    signalSound ⟵ Medium Intensity Signal
                else
                    if distancePercentage ≥ 35% and distancePercentage < 50% then
                        signalSound ⟵ High Intensity Signal
                    else
                        signalSound ⟵ Very High Intensity Signal
                    end
                end
            end
        end
    end
end
```

Algorithm 2: *SpatialTracker - UpdateStimuliSignal* method

A five-point scale of intensities was also introduced for auditory and haptic stimuli as shown in Figure 3. However, play-testing sessions revealed that the mechanism was dependent on the size of the *Sphere Collider*. This dependence proved problematic, with objects either spawning outside of the collider range or multiple objects causing confusion when intercepted simultaneously.

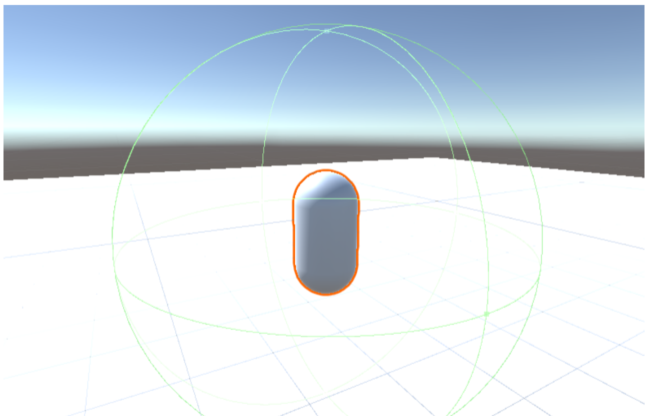

Figure 2: Sphere Collider in Unity

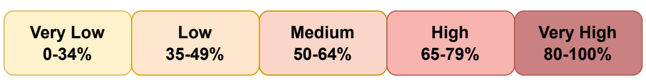

Figure 3: Five point scale of intensities for both stimuli.

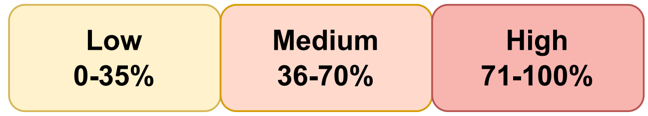

Figure 4: Updated three point scale of intensities for both stimuli.

```
Data: Player position pp, Target Position tp
begin
    newDistance ← Distance from pp to tp
    distancePercentage ← newDistance / distanceOnCollision
    // If the target is getting closer, update
       the signals
    if newDistance < previousDistance then
        previousDistance ← newDistance
        // Validate the according intensity
        if distancePercentage > 70% then
            signalSound ← Low Intensity Signal
        else
            if distancePercentage ≥ 35% and
              distancePercentage < 70% then
                signalSound ← Medium Intensity Signal
            else
                signalSound ← High Intensity Signal
            end
        end
    end
end
```

Algorithm 3: *SpatialTracker - UpdateStimuliSignal* method

3.3 Avoid Handling Collision Detection

Two main issues needed to be addressed at this stage, improve the discretization of the intensity scale for both stimuli and establish an effective methodology for object detection in the virtual environment.

To address the first problem, the intensity scale was simplified to three levels as can be observed in Figure 4. The goal was to enhance user understanding by having a more concise scale.

The object detection strategy was updated, shifting the responsibility from the player to the target object. Each target, once generated, was aware of the user's location and could calculate the distance to the player. This decentralised approach eliminated the need for complex collision management when dealing with multiple targets.

Consequently, an improved prototype of the spatial tracking mechanism was implemented, offering a more effective and user-friendly tool for object identification in a video game prototype. Refer to the algorithm 3 pseudo code for a better understanding of the mechanism.

4 GAME DESIGN AND IMPLEMENTATION

With the *Spatial Tracker* mechanism completed, the process of designing and implementing the game prototype began. This involved establishing the game world and creating a *High Concept* document. Several components of the game prototype, including its theme, type, and actions, were defined, paving the way for the development of the actual game prototype.

4.1 Game World

The game's narrative is set in a futuristic universe where Earth faces an alien invasion by a race known as the *Nauds*.

Players take on the role of a super soldier who, despite losing their sight as a result of previous invasions, fights against these invaders using advanced technologies, such as the *Spatial Tracker* mechanism. The *Spatial Tracker* serves as a tool for the player, allowing them to detect enemies and engage in combat.

4.2 High Concept Document

The *High Concept* document succinctly encapsulates the main ideas of the game, providing clarity and focus. It encompasses the game's premise, design goals, target audience, genre, selling points, and target platform, while maintaining a concise, readable format [21].

The document established the video game genre as an action game, specifically a *Rail Shooter* game, with a First Person Shooter (FPS) perspective. This genre allows players to concentrate on object location rather than navigation. Drawing inspiration from the game *Blind Hero*, a lane system was introduced, creating clearly defined interaction areas, as depicted in Figure 5.

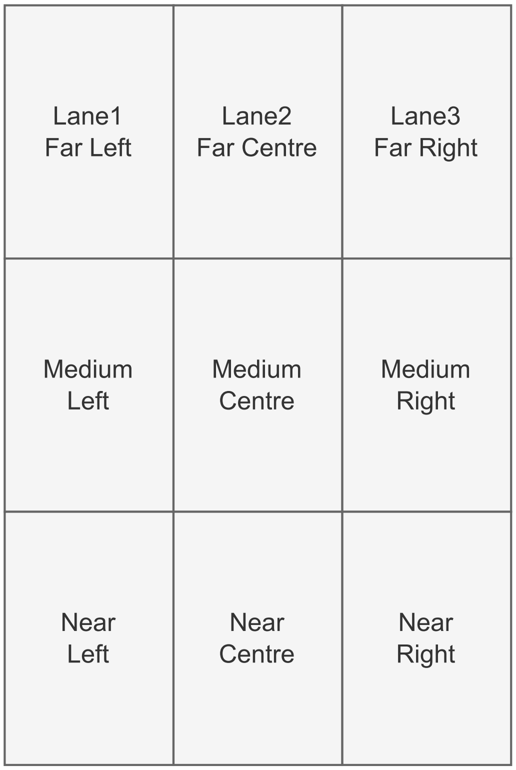

Figure 5: Lanes Layout.

This way, when enemies or health kits and ammunition are generated, they slide to the player's position, similar to what happens with notes in *Blind Hero*. While objects (enemies, health kits, etc.) slide towards the player, looped sound effects are played taking into account the object distance to the player. For instance, when an enemy is at the *Far Left* position, a low-frequency beep sound effect is played. However, as the enemy approaches the player and reaches the *Medium Left* position, the frequency of the beep sound increases.

In combination with the spatial audio framework, these sounds assist in distinguishing the object and its position in the scene.

The game prototype comprises a level with four sections, each with its own objective, refer to 4.3.6 for more details. Upon completing the challenges of a section, the player automatically advances to the next section through a gate.

4.2.1 ***Player Actions***. It was defined that the actions a player could execute in the game prototype include collecting health kits, picking up ammunition, firing and reloading a equipped weapon.

To perform the described actions, the player simply has to press the button assigned for each action defined in the controller scheme.

4.2.2 ***Player Challenges***. Each section has different objectives, thereby posing unique challenges for the player. Throughout the level, the player will be tasked with identifying and collecting a certain number of health kits, identifying and collecting a specific amount of ammunition, overcoming a target shooting challenge, and finally, eliminating all the *Nauds* that are attacking their space station in order to complete the level. When firing the weapon, the player must consider the weapon's range, as it only reaches enemies up to a medium distance. If the enemies are further away, they won't be hit by the shot.

4.3 Prototype Development

The prototype was developed iteratively, following a phased approach. The development stages included the implementation of player mechanics, object pickup, controller scheme, lane system, object spawning and picking, enemy design and level creation.

4.3.1 ***Audio Feedback System***. In order to enhance the players' ability to identify and locate objects in the game environment, a diverse range of audio clips was used. These clips were sourced from a combination of open-source platforms and custom-crafted using *Audacity*[15] software. By incorporating a wide variety of audio cues, the audio system aimed to provide distinctive and recognisable sounds that would assist players in discerning the placement and presence of different objects within the game.

To establish the audio feedback system, each object emitting audio feedback, such as the player, enemies, or health kits, was structured with an **Audio** *GameObject*. These *GameObjects* consisted of child *GameObjects*, each representing a specific sound effect. For example, the **Footsteps** *GameObject* was responsible for player footstep sounds. Each child *GameObject* was equipped with an *AudioSource* component, which outputted to an *AudioMixer*.

The *AudioMixer* played a crucial role in adjusting various audio elements such as volume, pitch, and effects. It created audio structures by directing and grouping sources to specified buses, applying effects, and routing them to the game's output.

To control the audio playback, each **Audio** child *GameObject* included an *AudioController.cs script*. This script contained methods like *PlayAudio()* and *StopAudio()*, allowing for precise control over when and how the audio clips were played. Some *GameObjects* also had additional methods, such as selecting audio clips randomly from an array.

To help the player determine the location of the sound sources and consequently, the lane in which the objects were positioned, each child *GameObject* usually included a *ResonanceAudioSource* component that enabled features like HRTF.

Overall, this approach ensured a diverse range of audio feedback, precise control over audio playback and the utilisation of spatial audio features to enhance the player's immersion and ability to locate objects within the game environment.

4.3.2 ***Player***. The **Player** is represented by a *Prefab*, which consists of child *GameObjects* that contribute to its functionality.

To manage the Player's behaviour, a script called *PlayerController.cs* was created. This script handles informative and control variables related to the Player, such as tracking life status and implementing interaction methods tied to the input scheme. Examples of interaction methods include picking up objects and shooting weapons.

The development of the **Player** also involved implementing a health system that incorporates elements like heart rate, breathing, life recovery, and death. Additionally, mechanics for footstep

[15] https://www.audacityteam.org/ (as consulted in May 2023)

sounds and weapon handling were integrated into the **Player**'s functionality.

4.3.3 *Pickable Objects*.

As for the pickable objects, namely health kits, ammunition, and a pistol, each one was developed with distinct scripts and attached with a *ResonanceAudioSource* for spatial audio features. **Health kits** *GameObjects*, controlled by *HealthKitController.cs*, replenish player's life and trigger appropriate sound effects. The **Ammunition** *GameObject*, managed by *AmmoController.cs*, reloads the weapon and trigger appropriate sound effects.

The development of the pistol in the game posed a unique challenge. It involved creating a **Pistol** prefab, as well as **InvisibleGun** prefabs, to enable shooting in three different directions: left, right and front. This implementation aimed to enhance the spatial audio features and provide clear auditory cues to the player regarding the direction of the shots.

Using controller scripts, mechanisms were established to control firing, manage ammunition and play corresponding sound effects. Ray casting technique was used for target hit detection with a fixed range.

4.3.4 *Lanes, Object Spawner and Picker*.

In this phase of development, the creation of a mechanism for sliding objects through the game scenario was required, resulting in the introduction of lanes. This led to the development of an object spawner and an object picker to aid player-object interactions in the game environment.

Lanes were implemented as a conveyor, transporting objects and enemies towards the player. These lanes were designed with collision detection mechanics ensuring objects move along them. To optimise game performance, unpicked objects were programmed to self-destruct once they exited the designated lanes. This approach helped maintain game efficiency and prevent unnecessary object accumulation.

An object spawner was designed to generate pickable objects randomly within each lane. It uses spawning points within lanes, randomly selecting a lane and an object for instantiation.

Finally, an object picker was developed to handle picking object interactions, easing the load on the **Player**. This picker uses ray casting technique to detect interactions with health kits and ammunition in each lane and trigger corresponding methods. The weapon, is instantly picked up upon instantiation, accompanied by a reload sound effect.

4.3.5 *Nauds*.

At this point the prototype required the implementation of an Naud enemy for the player to fight. The enemy was designed to be instantiated similarly to pickable objects, having less health than the player but moving faster. If the enemy reaches a certain proximity to the player without being defeated, it self-destructs, causing significant damage to the player.

The enemy sound effects were enhanced by gradually increasing the volume of footsteps as the enemy approaches the player. Additional audio components were included to provide sound effects for the enemy's roar, attack and death.

A technical issue with the sound effects was identified due to the *ResonanceAudio* framework's limitations with moving sound sources. To address this, the audio framework was switched to *Steam Audio*, which offers improved handling of moving sound sources. This change required replacing *ResonanceAudioSource* with *SteamAudioSource.cs* across all objects.

4.3.6 *Level Design*.

The next stage of the prototype involved design a level for formative usability testing, consisting of four distinct rooms, each with unique objectives as depicted in Figure 6.

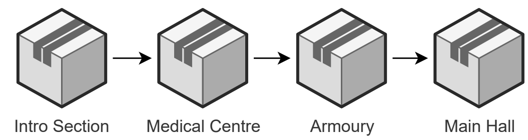

Figure 6: Lanes Layout.

Room 1, termed *Under Attack*, functions as an introductory space. The player starts with reduced health, enabling them to familiarise themselves with audio cues such as footsteps, breathing, heart rate and ambient music. Progression to the next room is achieved through interaction with a **Door** prefab.

In Room 2, the *Medical Centre*, health kits are randomly spawned to help players understand their sound, purpose and usage. The door to the next room only appears once the player has fully regained their health by collecting four kits.

Room 3, the *Armoury* was designed to teach the player about weapon use and ammunition interaction. The room is divided into three phases: weapon pick-up, ammunition collection and a target shooting challenge against static dummy robots.

The final room, *Main Hall*, allows the player to interact with all the game's resources. The goal here is to defeat all *Naud* enemies. After all enemies are defeated, the level is completed.

With the level implementation it was necessary to create new *prefabs* and modify existing ones. The **Object Spawner** *prefab* was significantly altered to accommodate the instantiation of objects as per the room objectives. Additionally, a **LevelManager** *prefab* was developed to manage level setup and transitions. This *prefab* also houses a *GameObject* responsible for playing ambient music in each level.

4.3.7 *Training Level*.

In addition to the main level, a dedicated *Training Level* was designed to instruct players on the use of the *Spatial Tracker* mechanism. To help players comprehend the discretisation of sound and haptic stimuli, a new scene was created using *Unity*, featuring a player and nine new *GameObjects*, each representing a cell of a grid.

A new *prefab*, named **TrainingSpawner**, was developed to randomly select a cell, instantiate an enemy within it, and play corresponding *Spatial Tracker* mechanism stimuli. Depending on the chosen cell, distinct sounds and vibrations were triggered as can be observed in Figure 7.

5 EVALUATION

To validate the effectiveness and usability of the developed *Spatial Tracker* mechanism, user tests were conducted. These tests aimed to assess the users' understanding and utilisation of the mechanism, as well as their level of immersion and enjoyment during game play. Additionally, experimental tests were conducted to evaluate the impact of the mechanism on players' performance in locating

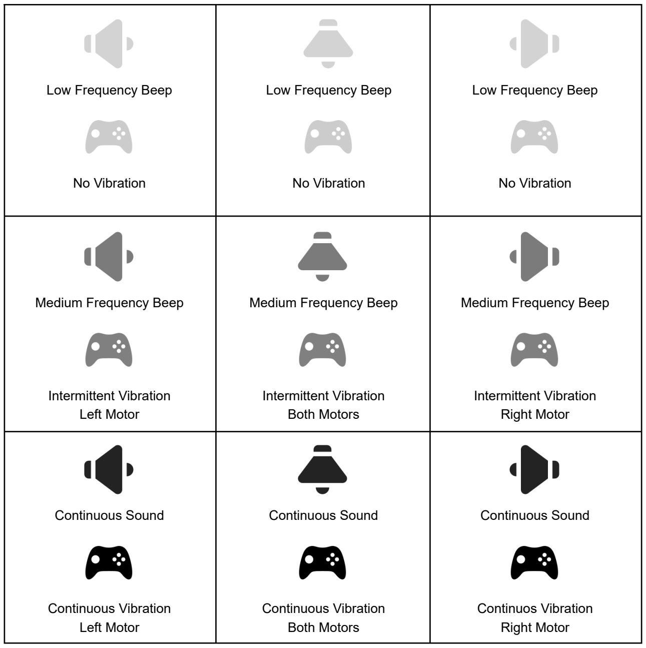

Figure 7: Stimuli played at each cell

objects in the game environment. The results of these tests, provide valuable insights into the strengths and weaknesses of the mechanism and offer potential areas for future improvements.

Several key metrics were captured throughout the testing phases and recorded in log files for cross-referencing with participant feedback. This data includes information on player's current level section, player's transitions between sections, object generation events, health fluctuations, firing actions, enemy hit occurrences, and enemy termination instances. This extensive data collection provides a rich foundation for analysing the spatial tracking mechanism's functionality and user reception.

5.1 Formative Usability Tests

The usability tests focused on key interactive steps, including identification and collection of health kits, weapon comprehension, ammo box collection, target shooting challenge completion, and successful navigation of the main hall section.

13 individuals, aged 23 to 62 years old, participated in these tests, comprising 8 men and 5 women. A notable portion of the participants (61.5%) engaged in casual gaming one to three hours per week [22], and over half (53.8%) were familiar with the *Rail Shooter* genre.

The usability testing procedures were conducted individually for each participant in a dedicated room with the necessary equipment.

The testing process consisted of three distinct phases. In the first phase, participants played the game without the spatial tracker mechanism enabled. Then, in the second phase, participants received training on how the mechanism worked and its activation. In the third and final phase, participants played the prototype with the spatial tracker mechanism enabled.

Table 1: Problems and Solutions from Usability Tests

Problem	**Solution**
The end of the game is abrupt there is nothing indicating that the player has returned to the main menu.	Add an auditory message to indicate the end of the level.
There is no feedback on the player's actions, such as when attempting to shoot without a weapon, or trying to pick up objects when there are none in the scene.	Add sound effects for when the player attempts to shoot without a weapon and pick up objects when they are not present in the scene.
Difficulty in understanding that it was necessary to pick up health kits until health was fully restored.	Reduce the number of health kits to be picked up in the medical centre section from 4 to 2.
Bug in which the third dummy robot was not instantiated, creating a loop where the player could not progress in the game	Fix the bug that occurred due to a *Null Pointer Exception*
Difficulty in identifying which section the player is in	Add an auditory message for each section to identify it

After completing the testing process, demographic data and questions related to usability were collected.

Test results highlighted some challenges, such as difficulty in understanding the need to collect more than one health kit, lack of feedback upon object pick-ups, and abrupt ending of the main hall section. Additionally, a bug was reported where the third dummy robot was not generated, causing confusion during the armoury section.

Solutions included creating new audio messages using Text To Speech (TTS) technology to indicate the start of a new section and the level's completion. A sound effect was also implemented to signal whenever the player attempted an action they couldn't currently perform. The bug with the dummy robot was resolved, the number of health kits in the medical centre was reduced. For more details refer to the Table 1.

5.2 Summative User Tests

These summative studies intended to evaluate participants' performance and the *spatial tracking* mechanism's impact on locating enemies in the main hall section.

The participant selection criteria for these studies included being over 16 years old, having been blind or having reduced visibility for at least five years, and having no hearing problems. The study included ten participants introduced by the Raquel and Martin Sain Foundation[16], half men and half women, aged between 26 and 60 years old. The participants' gaming habits varied, with 50% not

[16] https://www.fundacao-sain.pt/ (as consulted in May 2023)

playing regularly, 40% playing one to three hours per week, and 10% playing seven or more hours per week [22].

The experimental tests occurred over two sessions on October 27^{th} and 28^{th}, 2022, at the Raquel and Martin Sain Foundation facilities. Similar to the formative usability tests, the summative user tests began with an introductory phase for participants to become familiar with the prototype. Participants were allowed to interact with the prototype and clarify any questions.

Following the introductory phase, the participant tested the prototype with the *spatial tracking* mechanism deactivated. After the initial test, they moved onto the next phase where they learned how the *spatial tracking* mechanism worked. Finally, the participant tested the prototype with the active mechanism.

Data was collected during the summative user tests and processed. The metrics collected included the number of shots on target (**Shots**), the number of attempts to shoot without ammo (**Ammo**), the number of enemies killed (**Killed**), level completion status (**Level**), the time taken to complete the test (**Time**) and finally the (**Score**) which is calculated based on the number of accurate shots (20%), enemies killed (20%), amount of ammunition collected (20%), number of health kits picked up (20%), and whether the level was completed (20%) or not. The maximum score is 100 points and the minimum is 0 points.

In the phase without the mechanism, the completion times ranged between 48 and 503 seconds, averaging at 285.1 seconds ($M = 285.1, \rho = 120.61$). In the phase with the mechanism active, the times ranged from 93 to 490 seconds, averaging at 211.2 seconds ($M = 211.2, \rho = 102.47$).

Normality tests, specifically the Kolmogorov-Smirnov and Shapiro-Wilk tests, were applied to the collected metrics to check if they followed a normal distribution. Due to the fact that not all metric samples exhibited normal distribution and the sample size was less than 30, non-parametric tests were used [23].

Further analysis using the Wilcoxon Signed-Rank Test showed no significant difference in the scores for *Shots*, *Killed* and *Score* before and after the mechanism was used. Similarly, the McNemar test did not reveal a significant difference in the proportion of participants who completed the level before and after the use of the mechanism.

In addition, Spearman tests revealed no significant correlation between the *Ammo* variable and either the *Level* or *Killed* variables. This analysis aimed to evaluate whether any design issues, particularly related to the weapon reloading mechanics, could have had an impact on player performance.

The study also investigated the potential influence of participants' profiles on the results. Despite filtering the data to include only participants who were experienced with playing video games and conducting retests using the Wilcoxon Signed-Rank Test and McNemar Test, no significant differences were found for the variables examined. This suggests that participants' prior gaming experience did not have a notable impact on the observed outcomes.

Lastly, a satisfaction questionnaire administered to participants suggested a generally satisfactory interaction with the prototype. Feedback showed that most participants felt emotionally connected to the game and all were willing to replay it. A One-Sample T-Test confirmed that the average response to the satisfaction questionnaire was statistically equivalent to an agreeable response, further affirming that the prototype interaction was satisfactory for the participants. However, it was noted by three participants that the mechanism introduced additional inputs that caused confusion, with one participant suggesting a reduction in the haptic stimulus.

5.3 Results and Findings

The study aimed to validate the effectiveness and usability of the developed *Spatial Tracker* mechanism through a series of user tests. However, based on the analysis of the experimental results, it was not possible to confirm the research goal of improving the process of locating objects in a virtual environment for individuals with visual impairments using the mechanism.

To accurately interpret the results, it is essential to consider the effectiveness of the baseline prototype (without the *Spatial Tracker* mechanism). If the baseline prototype already provided satisfactory performance in locating objects, the addition of the tracking mechanism may not have resulted in significant improvements. This possibility highlights the importance of further analysis and evaluation.

Qualitative feedback from participants regarding their experience with the baseline prototype could provide valuable insights and suggestions for improvement in future iterations.

One possible reason for the absence of a significant impact from the *Spatial Tracker* mechanism could be justified by insufficient training. It is possible that the participants did not have enough practice or exposure to the mechanism to use it effectively. The number of training sessions required to learn a new tool varies depending on factors such as its complexity, the individual's previous experience and skills, learning style, and the type of training provided.

The Ebbinghaus[24] forgetting curve suggests that learning is better retained when sessions are repeated over time with breaks in between. This means that the training phase with the mechanism should be spaced out rather than condensed into a single intense session. A study conducted by Donovan and Radosevich[25] supports this theory, as they found that a moderate level of spacing between training sessions was most effective for retention.

Considering these factors, it is possible that the effectiveness of the *Spatial Tracker* mechanism could be improved by implementing a training schedule that includes repeated sessions distributed over time. This approach would provide participants with more opportunities to learn and retain the necessary skills for utilising the mechanism effectively.

Another reason that may justify the absence of an impact is the sample size. A small sample may affect the reliability of the results, as there is a higher risk that the sample is not representative of the general population. This can result in very high confidence intervals for the statistical data obtained in the experimental tests, making it more difficult to identify significant differences in the specified metrics. The Central Limit Theorem[26] supports this idea, stating that larger sample sizes tend to have sample means that follow a normal distribution around the population mean, making them more representative of the population. Conversely, small samples may not adequately represent the population, leading to larger confidence intervals and less reliable results[27]. According to the 2001 Portuguese census, there were approximately 163, 569 visually impaired individuals, constituting about 1.6% of the total resident population. As of 2021, the World Health Organization (WHO) estimated that at least 2.2 billion people worldwide have a visual impairment or blindness, which equates to approximately 28% of the global population. These figures highlight the challenge of obtaining significant sample sizes for this demographic group.

The order in which the tests were conducted, starting with the test without the *Spatial Tracker* mechanism and then introducing the mechanism, may have introduced a learning effect issue [28]. Participants' performance could have improved simply due to repeated exposure to the tasks involved, rather than as a result of the specific conditions being tested. To counterbalance this potential bias, a random selection of which option (with or without mechanism) to test first could have been implemented.

The design of the prototype itself may have influenced the results. Issues such as participants running out of ammunition and the potential impact of automatic ammunition loading mechanics on performance were identified. However, no correlations were found in this regard.

In conclusion, further analysis, evaluation and investigation are needed to determine whether the *Spatial Tracker* mechanism was ineffective or if the baseline prototype already provided a high level of efficiency.

6 CONCLUSION

This document presents a use-case aimed at helping individuals with visual impairments locate objects in a game environment. The research goal was to determine if a *Spatial Tracker* mechanism could improve the process of object location for visually impaired individuals in virtual environments.

To address this question, a game prototype was developed, incorporating the *Spatial Tracker* mechanism. Once the first stable version of the prototype was completed, usability tests were performed to identify possible usability problems. The experiment consisted of three phases: a baseline phase without the mechanism, a training phase with the mechanism and a final phase with the mechanism enabled.

The evaluation with visually impaired users involved 10 participants, and the results indicated that the developed mechanism did not have a significant impact on object location. It is suggested that the baseline prototype may have already provided satisfactory performance in locating objects, making the addition of the tracking mechanism less impactful. Qualitative feedback from participants indicated potential confusion with the tracker mechanism, and it was also noted that participants may not have received sufficient training to effectively use the *Spatial Tracker* mechanism. On the other hand, the small sample size may have affected the reliability of the results, as it may not be representative of the general population.

6.1 System Limitations and Future Work

During the experimental evaluation, a consideration regarding the game design choices, specifically the weapon reload mechanic, arose. The question was raised whether there would be better performance if the pistol reload system was automatic. The introduction of an automatic reload mechanic could potentially relieve cognitive and task burden for the players, allowing them to focus more on game play and interaction within the game environment.

Another question raised was whether introducing phased training with the mechanism over time could enhance participants' performance. The idea behind phased training is that gradual exposure allows participants to familiarise themselves with the mechanism, understand its functionalities and adapt to interacting with the game environment. While there is no specific data supporting this hypothesis, it is reasonable to assume that gradual exposure could lead to better adaptation and mastery of skills. However, further empirical studies are needed to confirm this hypothesis and compare participant performance across different training modalities, providing concrete evidence of the benefits of phased training with the *Spatial Tracker* mechanism.

Although the experimental results did not show a significant impact of the developed mechanism, there are interesting topics that have emerged and could be further explored. For example, studying the interaction of participants, both with and without the mechanism, using different devices like mobile devices. This approach could offer advantages such as improved accessibility, portability, lower cost and leveraging existing development ecosystems. It would enable the functionality to reach a wider range of users while utilising available tools and resources.

To further evaluate the effectiveness of the mechanism, it would be important to assess its performance in completing a predetermined set of tasks with varying levels of complexity. For example, creating a variant of the training level where participants need to locate multiple objects in the grid, or increasing the complexity by introducing additional challenges. This would allow for a more thorough evaluation of the mechanism's ability to handle increasingly difficult tasks and determine its impact on participants' performance. By conducting assessments across a range of tasks, a better understanding of the mechanism's capabilities, limitations and areas for improvement can be gained.

Another idea that was considered during the design phase of this project was the use of multiple spatial tracking mechanisms, where each mechanism would have a specific functionality associated with it, and it would be up to the player to decide when and which mechanisms would be active at any given moment.

In addition to the already mentioned areas of exploration, it would be worthwhile to evaluate the integration of ambisonic sound effects with the *Spatial Tracker* mechanism. By incorporating ambisonic sound effects, the interaction with the mechanism could become more immersive, enhancing the participants' spatial perception and object location abilities in the game environment. Assessing the impact of ambisonic sound effects on the mechanism's effectiveness could provide insights into its influence on accuracy, speed and ease of object location. This investigation could further enhance the overall user experience and potentially optimise the performance of the *Spatial Tracker* mechanism.

In summary, although the current study did not yield significant results in enhancing the object location process in the virtual environment, the development of the *Spatial Tracker* mechanism sets the stage for future investigations. It represents progress in the research of interface usage in video games for individuals with visual impairments and contributes to improving accessibility for this population. The mechanism serves as a foundation for further improvements and invites additional studies to assess its effectiveness in various contexts. Despite the initial findings, the potential impact of this work is substantial and points towards an exciting future in the field.

REFERENCES

[1] G. R. White, G. Fitzpatrick, and G. McAllister, “Toward accessible 3d virtual environments for the blind and visually impaired,” in *Proceedings of the 3rd International Conference on Digital Interactive Media in Entertainment and Arts*, DIMEA ’08, (New York, NY, USA), p. 134–141, Association for Computing Machinery, 2008.

[2] S. Maidenbaum, S. Levy-Tzedek, D.-R. Chebat, and A. Amedi, “Increasing accessibility to the blind of virtual environments, using a virtual mobility aid based on the "eyecane": Feasibility study,” *PLOS ONE*, vol. 8, 08 2013.

[3] S. M. Trewin, M. R. Laff, A. Cavender, and V. L. Hanson, “Accessibility in virtual worlds,” in *CHI ’08 Extended Abstracts on Human Factors in Computing Systems*, CHI EA ’08, (New York, NY, USA), p. 2727–2732, Association for Computing Machinery, 2008.

[4] S. Maidenbaum, S. Abboud, G. Buchs, and A. Amedi, “Blind in a virtual world: Using sensory substitution for generically increasing the accessibility of graphical virtual environments,” in *2015 IEEE Virtual Reality (VR)*, pp. 233–234, 2015.

[5] N. I. Ghali, O. Soluiman, N. El-Bendary, T. M. Nassef, S. A. Ahmed, Y. M. Elbarawy, and A. E. Hassanien, *Virtual Reality Technology for Blind and Visual Impaired People: Reviews and Recent Advances*, pp. 363–385. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[6] M. Piçarra, A. Rodrigues, and J. a. Guerreiro, “Evaluating accessible navigation for blind people in virtual environments,” in *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI EA ’23, (New York, NY, USA), Association for Computing Machinery, 2023.

[7] B. Yuan, E. Folmer, and F. C. Harris, “Game accessibility: a survey,” *Universal Access in the Information Society*, vol. 10, no. 1, pp. 81–100, 2011.

[8] C. Mangiron and X. Zhang, *Game Accessibility for the Blind: Current Overview and the Potential Application of Audio Description as the Way Forward*, pp. 75–95. London: Palgrave Macmillan UK, 2016.

[9] L. Evett, D. Brown, S. Battersby, A. Ridley, and P. Smith, “Accessible virtual environments for people who are blind: Creating an intelligent virtual cane using the nintendo wii controller,” *Proceedings of the Seventh International Conference on Disability, Virtual Reality and Associated Technologies*, 01 2008.

[10] V. Nair, J. L. Karp, S. Silverman, M. Kalra, H. Lehv, F. Jamil, and B. A. Smith, “Navstick: Making video games blind-accessible via the ability to look around,” in *The 34th Annual ACM Symposium on User Interface Software and Technology*, UIST ’21, (New York, NY, USA), p. 538–551, Association for Computing Machinery, 2021.

[11] M. Lumbreras and J. Sánchez, “Interactive 3d sound hyperstories for blind children,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI ’99, (New York, NY, USA), p. 318–325, Association for Computing Machinery, 1999.

[12] B. Yuan and E. Folmer, “Blind hero: Enabling guitar hero for the visually impaired,” in *Proceedings of the 10th International ACM SIGACCESS Conference on Computers and Accessibility*, Assets ’08, (New York, NY, USA), p. 169–176, Association for Computing Machinery, 2008.

[13] W. Yost, *Fundamentals of Hearing: An Introduction: Fifth Edition.* Brill, 2021.

[14] J. C. Middlebrooks and D. M. Green, “Sound localization by human listeners,” *Annual Review of Psychology*, vol. 42, no. 1, pp. 135–159, 1991. PMID: 2018391.

[15] J. Blauert, *Spatial Hearing: The Psychophysics of Human Sound Localization.* The MIT Press, 10 1996.

[16] J. W. Strutt, “On the perception of the direction of sound,” *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, vol. 83, no. 559, pp. 61–64, 1909.

[17] T. Letowski and S. Letowski, “Auditory spatial perception: Auditory localization,” 05 2012.

[18] A. D. Musicant and R. A. Butler, “The influence of pinnae-based spectral cues on sound localization,” *Journal of the Acoustical Society of America*, vol. 75, pp. 1195–1200, Apr 1984.

[19] D. R. Begault, E. M. Wenzel, and M. R. Anderson, “Direct comparison of the impact of head tracking, reverberation, and individualized head-related transfer functions on the spatial perception of a virtual speech source,” *Journal of the Audio Engineering Society*, vol. 49, pp. 904–916, Oct 2001.

[20] R. Urick, *Principles of Underwater Sound.* Peninsula Publishing, 1983.

[21] T. Fullerton, *Game Design Workshop: A Playcentric Approach to Creating Innovative Games, Fourth Edition.* A K Peters/CRC Press, 4 ed., 2018.

[22] E. S. Association, “2022 essential facts about the video game industry,” 2022.

[23] P. Royston, “An extension of shapiro and wilk’s w test for normality to large samples,” *Applied Statistics*, vol. 31, no. 2, pp. 115–124, 1982.

[24] H. Ebbinghaus, *Memory; a contribution to experimental psychology.* Teachers College, Columbia University, 1909.

[25] J. J. Donovan and D. J. Radosevich, “A meta-analytic review of the distribution of practice effect: Now you see it, now you don’t,” *Journal of Applied Psychology*, vol. 84, no. 5, pp. 795–805, 1999.

[26] S. G. Kwak and J. H. Kim, “Central limit theorem: the cornerstone of modern statistics,” *Korean journal of anesthesiology*, vol. 70, no. 2, pp. 144–156, 2017.

[27] G. M. Sullivan and A. R. Artino Jr, “Analyzing and interpreting data from likert-type scales,” *Journal of graduate medical education*, vol. 5, no. 4, pp. 541–542, 2013.

[28] E. B. Goldstein and J. R. Brockmole, *Sensation and Perception.* Cengage Learning, 10 ed., 2016. Chapter 1.

ACRONYMS

BATE Beyond Accessibility To Efficiency
API Application Program Interface
FPS First Person Shooter
HRTF Head-Related Transfer function
ILD Interaural Level Difference
ITD Interaural Time Difference
SONAR Sound Navigation and Ranging
TTS Text To Speech
WHO World Health Organization