# 3D Localisation of Sound Sources in Virtual Reality

Edvinas Danevičius(✉), Frederik Stief, Konrad Matynia, Morten Læburgh Larsen, and Martin Kraus

Department of Architecture, Design, and Media Technology, Aalborg University, Fredrik Bajers Vej 7K, 9220 Aalborg Øst, Denmark
{edanev19,fstief14,kmatyn16,molars15}@student.aau.dk, martin@create.aau.dk

**Abstract.** This paper presents a comparison of 3D localisation of sound sources using various 3D audio engines for Virtual Reality (VR) environments. An experiment was created with the Oculus Spatializer, Unity Default engine, Unity Reverb engine and the AM3D Spatializer. These four engines were tested against each other in a Virtual Reality setting, where the tester was tasked with the localisation of invisible audio sources present in the virtual room. The evaluation of the experiment showed that there were statistically significant differences between the four engines under specific circumstances.

**Keywords:** 3D audio · Virtual reality · Sound localisation · Audio experiment · Spatializer plugin

## 1 Introduction

Immersive audio effects, ambience and music in virtual applications comes from panning audio from side and rear of the listener. This is used to widen the perceived dimensions of the scene, by extending what is being seen on a display. The ability to work in an environment where audio can be relayed from any direction around the listener greatly expands the acoustic space in which audio engineers can work [11].

3D Audio is an essential aspect of most Virtual Reality (VR) applications. It is used to enhance the experience, and sometimes even serves as a key gameplay element that guides the progression of video games. However, the three dimensional aspect of audio usually has been implemented as simple stereo panning. Audio is fed to either one of the headphones' speakers to represent the position of the source in the virtual environment and guides the attention of the player. A realistic 3D audio implementation involves the tracking of the audio source and simulating the environment surrounding it. Mainly, they improve the player's immersion and, in addition, enhance the localisation of audio objects [3].

In order to test this localisation of audio objects, a simulation comparing four 3D audio engines was created. This includes the AM3D Spatializer, Oculus


A. Brooks et al. (Eds.): ArtsIT 2020, LNICST 367, pp. 307–319, 2021.
https://doi.org/10.1007/978-3-030-73426-8_18

Spatializer, Unity Default engine and Unity Reverb. Unity Default being Unity's built-in audio engine and Unity Reverb a different configuration of the same engine, enabling the Reverb feature. The simulation is a VR experience, which is based on sound localisation using the four different engines. The results of each test and engine are analysed based on the accuracy, which is represented by the distance from the tester's selection to the actual position of the audio source. With this in mind, the following hypothesis was created:

*There is a difference in accuracy when locating 3D audio between the four investigated engines.*

To test this hypothesis, relevant research was conducted, the simulation designed, implemented, evaluated and analysed.

The research began as a collaboration agreement between the authors and the company Goertek Europe. As the interest of the authors was to investigate the importance and impact of 3D audio in games and virtual environments, Goertek Europe suggested testing and evaluating using their engine AM3D Spatializer, which they supplied for the test. We signed a Non-Disclosure Agreement (NDA) form, which prevents us from describing the details of the engine, thus it will not be explained in detail in the later chapters.

## 2 Background

### 2.1 Sound Localisation

Sound localisation is the process of identifying sounds' actual or perceived positions in terms of direction and distance relative to the listener [15]. People can identify sounds all around them, but are less accurate when the sounds are coming from the sides or behind their head. Binaural cues are used to localise sounds and while many accumulative factors impact a sound before it reaches the ear, the factors can be simplified and represented as a filtering operation based on the difference between the signal received by the left and right ear [3,11]. This can be utilised by using Head-Related Transfer Functions (HRTFs), which are functions that describe how the ears receive a sound after it interacted with objects. The result is a binaural sound that contains localisation information, which is used to pin point the origin of the sound [15].

### 2.2 Reverb

The height dimension contains useful acoustic data for enhancing the experience. Reverb is useful when trying to access height information [11]. Reverberation, or reverb, is an acoustic phenomena that occurs in enclosed spaces. When sound is produced in these spaces, it does not disappear immediately but will gradually decrease in loudness while bouncing off surfaces [8]. The time it takes for the sound to go from audible to silent depends on the sound's characteristics as well as the size and material of the space [14]. The reverb can be examined with an impulse response. The impulse response of a space can be divided into three parts: the direct sound, early reflections and the reverberation.

The first sound to reach the ears is the direct sound. Afterwards, the early reflections will reach the ears after being reflected once or twice from surfaces like walls, ceilings or floors. The last part to reach the ears is the reverb tail or late reflections. The early reflections can also be described as echoes. Since early reflections are loud and arrive only 50 ms after the direct sound it tends to create an echo environment [3].

### 2.3 Related Work

Multiple ways of detecting the location of sound sources have been investigated in the past. For example, evaluation in audio localisation has been done by placing participants in virtual MCRoomSim scenes. Movement sampling was done by sampling the participants head positions and orientations using Vicon camera tracking system [12].

Other studies about localising sound sources in a virtual environment include the "The Binaural Navigation Game". This game was made for both normal sighted and visually impaired individuals. The objective was to test and train the user's ability to localise sounds. This is done by utilising how 3D binaural sounds are perceived and implementing HRTFs [2].

Some of the findings include the listener not being very good at determining the distance to the audio source. Other findings include a slight increase in accuracy when the test participant points towards the audio source and is able to turn their head [10].

## 3 Test Environment Design

To compare the aforementioned engines, a VR simulation was created. In the simulation, the tester is placed in the middle of an empty, light-grey room. Sounds from different engines are played in different positions of the room, one after another. The tester is then supposed to pinpoint the exact position they believe the sound is playing from by using a virtual laser pointer device. The audio sources are invisible, so the tester has no visual aid. Furthermore, the room is kept as minimalistic as possible, to not distract from the task. The only details added are an ambient occlusion effect to darken corners and a tile pattern for the floor, to ease the estimation of the depth of the room. This helps pinpointing the sources and also alleviates problems with motion sickness.

### 3.1 Audio Source

Localisation of sound becomes easier for people if it is a familiar sound. A typical example for this is the sound of a telephone ringing. Telephone ringtones are easily recognisable and also in the real world the sound cue signalises a person to find the source, in order to answer it [9]. Therefore, the iPhone Marimba ringtone has been chosen. The audio is played omnidirectional to enable the production of as many reflections as possible, opposed to a limited degree emitting source.

### 3.2 Room Setup

The simulation is comprised of two rooms. The first room acts as a tutorial room to familiarise the tester with the surroundings and controls. It contains a button in the middle, which loads the second room and starts the test. This allows the tester to start the test whenever they are ready. Both of the rooms have the same dimensions of $10 \times 10 \times 3$ m (width, depth, height). In both of the rooms, the tester is placed in the middle and can move around in a small area indicated by a blue square on the floor. Apart from the button in the tutorial room they are identical.

The ability to move as well as tilting the head enables the person to pinpoint the audio source more precisely by listening to and analysing how the sound changes in different head positions. In addition to that, a 45° turn can be executed by tilting the left thumb-stick left or right. The tutorial room can be seen in Fig. 1a.

The wall properties of the room that alter the intensity of the reflections and reverb have been kept to default or slightly adjusted to even out the differences in the engines as much as possible.

### 3.3 Laser Pointer

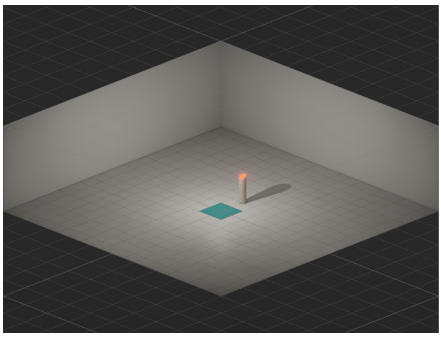
Both of the rooms allow the tester to use a virtual laser pointer, which is emitted from the right controller. The laser pointer is activated by clicking and holding down the right grip button. Additionally, while the laser is activated, the right thumb-stick can be used to change the length of the laser by tilting it forwards or backwards. In order to confirm the position of the laser, the right trigger button must be pressed while the right grip is held. Once the position is confirmed, haptic feedback is relayed to the right controller. In addition to haptic feedback, an audible click sound is played, to inform the tester that the confirmation action was executed successfully.

The selection is visualised by intersecting coloured lines for the x, y and z-axis. The purpose is to help the user pinpoint the location of the audio source as precisely as possible in a 3D virtual environment.

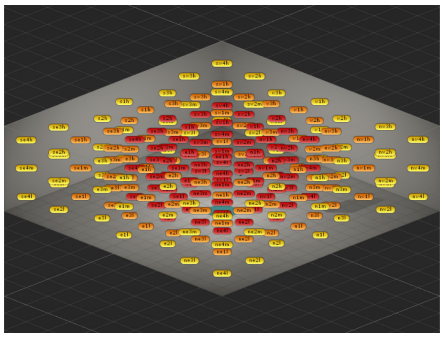
### 3.4 Audio Source Positions

We defined in total 216 scattered positions where audio can play. These positions are split into three groups; near field, mid field and far field, which are based on the distance from the centre. Each field consists of 24 positions on 3 different height settings, totalling 72 positions per field. The layout of these positions can be seen in Fig. 1b.

The positions have been scattered as evenly as possible to have sounds playing all over the room, while still having varying distances to the walls as to experiment with different distances for early reflections. Furthermore, all positions get randomly moved up to 0.45 m upon start of each testing session.



(a) The tutorial room.



(b) Audio positions. Red - near field, orange - mid field and yellow - far field.

**Fig. 1.** Tutorial room and testing room with visible audio source positions.

### 3.5 Experimental Procedure

The testing was performed by one of the authors. For each test, no time limit is set for locating each audio source as the focus was on accuracy. After the test session, the tutorial room is loaded once more, where the tester starts the next test when ready.

Multiple tests were conducted in a row with breaks in-between. The testing sessions were spread out throughout the week, each day consisting of 15 sessions at most. In total, 54 test runs were carried out, consisting of 24 selections each.

## 4 Experimental Setup

The simulation was created using Unity 2019.3 [13] and made use of the Oculus Integration 15.0 [4] asset for Head Mounted Display (HMD) tracking. In addition to that, the Oculus Audio Spatializer 15.0 [6] and AM3D Spatializer 1.1.9.0.0 [7] plugins were used for relaying 3D audio. The simulation can run without a VR headset in desktop mode, which can be used for internal testing.

In terms of hardware, Oculus Rift [5] with two base stations and Oculus Touch controllers were used. For audio, SONY MDR-7506 [1] professional headphones were utilised.

### 4.1 Scenes and Data Collection

The tutorial and testing rooms were implemented using Unity's scene feature. Each scene acts as a container for said room. The most important room however is the second room, where the tester is required to use the laser pointer in order to locate audio sources. If the laser pointer's position is confirmed while an audio source is playing in the room, information is collected, aggregated and saved in a in-memory buffer. The parameters which are being saved are: distance to the audio source, time taken to find the audio source, linear and angular movement of the HMD.

### 4.2 Engine Setup

The two audio sources that were set up to work without a spatializer plugin were `UnityReverbAudioSource` and `UnityAudioSource`. The `UnityReverbAudioSource` was configured to use Unity's built-in reverb functionality while `UnityAudioSource` is set to use no reverberation. This was achieved by adjusting the `Reverb Zone Mix` value between 0.0 (minimum) and 1.0 (maximum). In addition to that, the `Output` value was set to different mixer groups: (`UnityReverb` for `UnityReverbAudioSource` and `Unity` for `UnityAudioSource`). Other settings for both audio sources were the same.

In order for reverb to work when using `UnityReverbAudioSource`, a `GameObject` (centred at world origin) with `AudioReverbZone` component was added. `AudioReverbZone` component's min and max distances were configured so that they cover the entire testing room. In addition to that, the `Room` reverb preset was used.

The remaining two audio sources `AM3DAudioSource` and `OculusAudioSource` require the additional components `AM3DAudioSourceSettings` and `Oculus Spatializer Unity`. Settings of `AudioSource` component are the same as in `UnityAudioSource`, except different mixer groups were assigned: `Oculus` for `OculusAudioSource` and `AM3D` for `AM3DAudioSource`.

Since these audio sources are linked to different mixer groups, the attenuation of mixer groups was adjusted so that the loudness of each audio source was similar. All groups except `Oculus` have been tuned -13 dB. In addition to that, `Oculus` and `AM3D` require mixer effects to fully utilise spatialization features: `OculusSpatializerReflection` for `Oculus` and `AM3D Spatializer Room Processor` for `AM3D`.

The `OculusSpatializerReflection` effect allows to configure the reflectivity and reverb settings of each wall in the room. The value 0.7 was chosen for each surface, which was recommended by Goertek for surfaces made out of concrete. Room dimensions are only specified for consistency, as `OculusSpatializer` determines the size of the room dynamically. The `AM3D Spatializer Room Processor` mixer effect only allows to adjust gain of direct sound and reverberations, which were set to 1.00 as volume is attenuated via the `Attenuation` mixer effect.

In order to configure reflectivity settings of AM3D Spatializer further, a `GameObject` (centred at world origin) with `AM3DAudioRoom` component was added. The `AM3DAudioRoom` component was configured to have the same room dimensions as the testing room and uses the same reflectivity settings as `OculusSpatializerReflection` mixer effect.

### 4.3 Engine Selection

When the testing room is loaded, 24 positions are randomly picked from the `NearField`, `MidField` and `FarField` groups (8 from each) and displaced on x, y

and z coordinates by a random offset within [−0.45, 0.45] range. Each position is then randomly assigned an audio source so that each engine appears twice in each group, which results in each engine being used exactly 6 times throughout the test. For randomisation, the C# class `Random` is used, which is initialised using a `Guid` value. This ensures that each test run is unique.

The sounds to be played appear in the aforementioned 24 positions. As the order of the pool is randomised, positions are picked incrementally starting from index 0. Once a position is chosen, the appropriate engine is enabled and the audio is played using the assigned audio source. Engine switching is performed at run-time, right before playing each audio source. This is achieved by leveraging Unity's `AudioSettings.SetSpatializerPluginName(string)` function. However, one downside to this is that the project cannot be built and must be used within the Unity editor.

## 5 Results

The tester performed 50 test runs and 1200 data points were collected in total.

### 5.1 Means

In Fig. 2 the data is visualised as histograms. Only small differences in accuracy are visible for each engine.

To gain an understanding of the data and make it easier to compare, the mean of the distances from the participant's selections to the actual position of the sound sources for all engines is calculated.

Examining the means of the four engines in Table 1 show that the participant was marginally more precise with the Oculus Spatializer engine, with a mean distance of 1.51 m. This was closely followed by the Unity Default and AM3D Spatializer engines with scores of 1.52 m and 1.55 m respectively. The Unity Reverb engine had the highest distance mean of 1.63 m.

**Table 1.** Distance means for each engine.

| Engine | Mean |
|---|---|
| Oculus Spatializer | 1.51 m |
| Unity Default | 1.52 m |
| AM3D Spatializer | 1.55 m |
| Unity Reverb | 1.63 m |

### 5.2 Scatter Plots

A trend is showing when presenting the data of the participant's selections as well as the actual locations of the sounds in all engines combined in 3D scatter

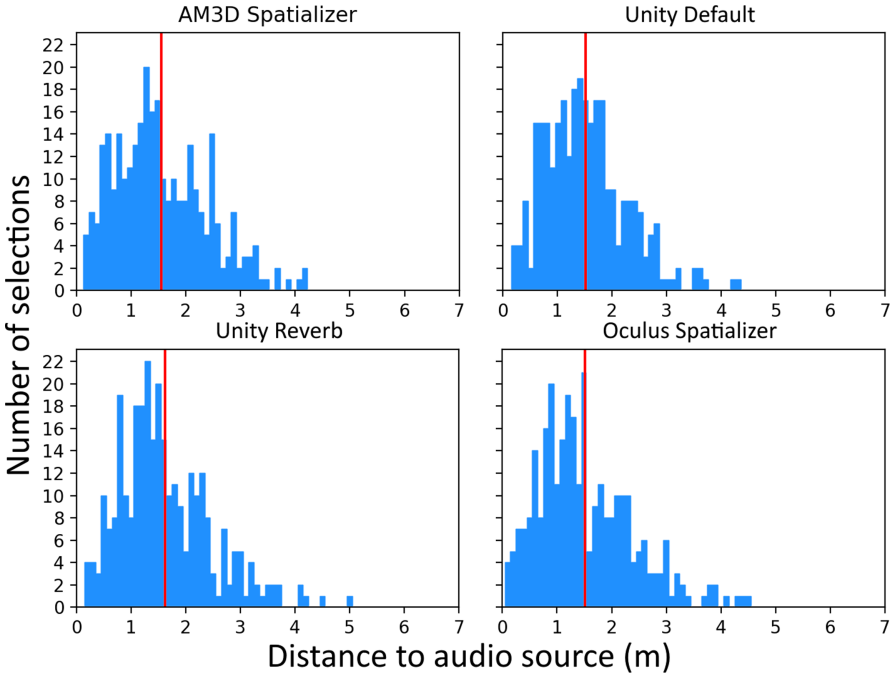



**Fig. 2.** Histograms of distances and means for each engine.

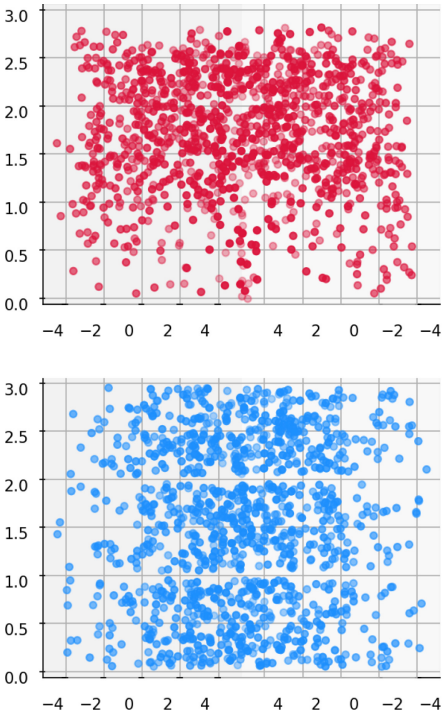
plots. An example of this can be seen in Fig. 3a. The x-axis is the width of the room and the y-axis the height. The participant's selections are displayed as red dots and the actual positions as blue dots.

The Unity Default engine gives a very good idea of the general direction of the sound, which is proven by the means, but no good indication of the height. This can be seen when looking at Fig. 3b, showing only the distributions of the Unity Default engine, compared to the sum of all engines in Fig. 3a.

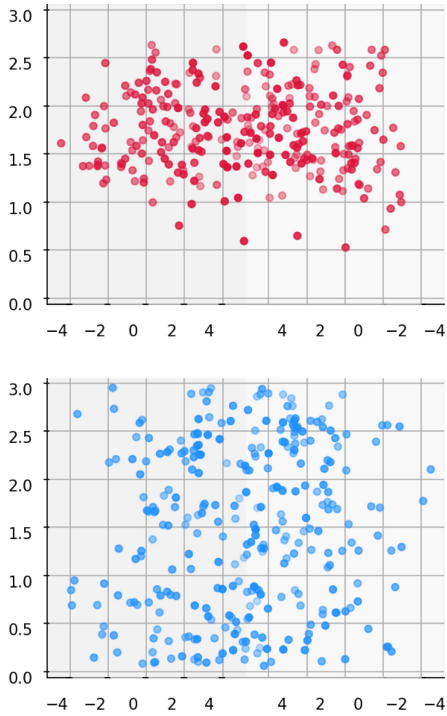
In terms of distance to the sound, there are no definitive trends, as the participant tended to put selections further away than the actual positions. This can be seen in Fig. 4.

### 5.3 Friedman Test

To detect statistically significant differences between the engines, the Friedman test was used, which is a non-parametric statistical test. A p-value below 0.05 obtained this way is expected to reject the null hypothesis. The p-value for the internal participant's data was 0.44. This shows that there are no statistically significant differences when comparing the engines against each other using all the data.




(a) Selections for all engines.




(b) Selections for the Unity Default engine.

**Fig. 3.** Side view of internal tester's selections (red) and actual positions (blue). (Color figure online)

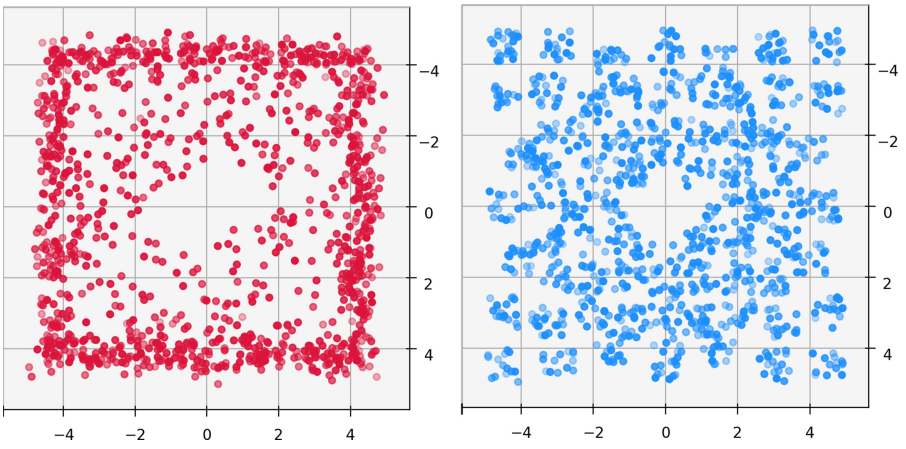



**Fig. 4.** Top down view of all participant's selections (red) versus actual positions (blue). (Color figure online)

However, when splitting the testing data up into the near, mid and far fields, a different image emerges. Focusing only on the far field data yields a p-value of 0.002. In this scenario, the distance mean of AM3D Spatializer was 25–35% lower than the other engines. The means can be seen in Table 2.

**Table 2.** Distance means of far field data.

| Engine | Mean |
|---|---|
| AM3D Spatializer | 0.93 m |
| Unity Reverb | 1.17 m |
| Oculus Spatializer | 1.19 m |
| Unity Default | 1.25 m |

## 6 Discussion

The main evaluation was supposed to include at least 50 participants, since the main objective of this study was to evaluate how the engines compare against each other. Due to the simulation being in a virtual reality environment, participants would have to share one HMD, which is neither possible nor responsible in the current COVID-19 situation. Given the NDA signed with Goertek regarding the use of their plugin, it was not possible for us to send the simulation out to test participants who have their own HMD available at home.

Evaluating with only an internal tester was made possible by randomising most aspects of the test. The audio positions are randomly picked from a large pool of predefined locations and displaced by the earlier specified margin. The goal of this was to hinder the learning effect by not repeating exact same locations over and over, while still keeping an even distribution of possible locations all over the room. After all locations have been picked, they are scrambled so the appearance of the different fields is random as well. The plugins used for the locations are spread evenly, with each of the four plugins appearing in six different locations each test run: two in the near field, two in the mid field and two in the far field. However, their order of appearance was randomised as well, so the tester does not know which engine was currently playing and which one is coming up next.

The tester did not see the actual location of the sound after the test nor the results, so the tester did not know if their results get better or worse. All of this should hinder the tester's learning effect as much as possible. By analysing accuracy over time there was no sign of any learning effect occurring as no improvement was found in the distance means, which can be seen in Fig. 5 by looking at the red dotted trend line. It is however possible that the tester had already been primed during the development of the simulation.

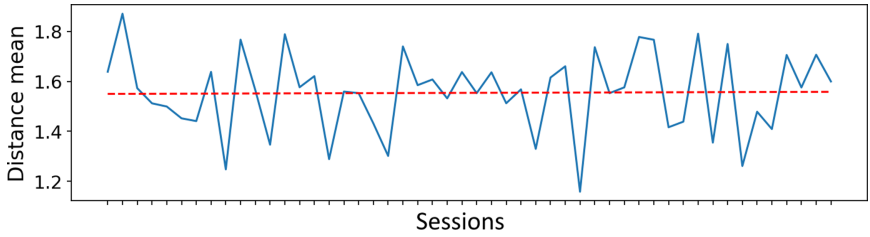



**Fig. 5.** The learning curve for the tester. (Color figure online)

All of the design choices regarding the room and sounds are as simple and minimalistic as possible, to make this test easily repeatable and achieve a higher reliability. The focus was on functionality, instead of distracting the users with aesthetics. Having no time limit during the tests means that the tester had the opportunity to fine-tune his selections on the audio sources. This could result in increased accuracy, compared to a time limited approach.

The choice of an instrument based marimba sound bears a lot of overtones and harmonic content in the sound, which could have impacted the perception of sound, compared to simpler sound sources, such as a white noise.

The tests were conducted with the headphones provided by Goertek (SONY MDR-7506), which they also recommended. Each headphones' frequency response is different, which could result in the sound being perceived differently. Four different headphones were tested internally prior to the evaluation and SONY's chosen as subjectively best.

The main study comparing the four engines in all three fields did not result in a statistically significant difference, although the outcome might have differed if the study was conducted with more than one tester.

Lastly, it is also important to note that hearing is subjective and differs from person to person. This has an influence on how we hear and perceive sound volume, distance and direction, amongst other things. This means that the origin of the sound will be perceived differently by each tester with individual differences becoming less apparent in a bigger sample size.

## 7 Future Work

During internal testing it was noticed that HMD tracking malfunctions when facing away from the base stations. Due to this, additional controls had to be added in order to facilitate all angles. This issue could be alleviated by using more base stations or a HMD that supports hand tracking via mounted cameras. For example, an Oculus Rift S would be suited better for such a test.

When analysing collected session data, most selections are directed towards the ceiling. This could be the result of improperly chosen reflectivity coefficients for each engine. In addition to that, the audio file that was played during the

test sessions could be changed, to see if different audio combinations yield different results. More internal testing is needed to determine the most suitable configuration.

Having a symmetrical room increases the likelihood of perceiving distant sounds as having higher attenuation due to the sound reflections converging in the middle of the room. Different room configurations should be tested as well, having rooms of different sizes and asymmetrical layouts. The placement of the tester could be randomised as well instead of always placing them in the middle. This changes the way the sounds would reflect and when the tester perceives it, but also increases the complexity of the test, since it introduces more variables.

Increasing the test participant's freedom of movement could impact the results. Should the tester be allowed to move further within a perimeter or even freely, another variable would have to be taken into account. The attenuation of sound while approaching the source in the VR setting would increase, so the importance of reverb and early reflections might become less apparent.

## 8 Conclusion

Overall the four engines only showed significant differences in performance when investigating the far field data. The initial interest in early reflections and reverb as key components to reproduce realistic 3D audio sensation was then analysed and interpreted. Reverb has made the biggest impact on the tester in the aspect of localising the sounds' altitude. When analysing the results of the Unity Default engine for example, the issue with the attenuation of sound when moving one's head around in the VR environment becomes apparent. The difference in volume in the left and right channels seems more important than the effects of reverb, as the gain changes alone serve as a good basis for localising the sound sources' general direction.

As for the early reflections, which were the main point of interest with Goertek's AM3D Spatializer plugin, the results from the analysis of the far field sources portion of the test indicate an advantage. From our observations, the task of localising the sound sources in the far field was more accurate with AM3D's early reflections the closer the sources are to walls. This is likely caused by the principle of how early reflections are calculated. The reflections of sound are simulated, thus the surfaces closest to the source are taken into account. This also impacts the decision of restricting the testers from moving around to a minimum during the experiment.

While the experiment as a whole only showed significant differences when investigating the far field, given different circumstances and more participants to test with, the results might have shown more differences.

**Acknowledgements.** We would like to thank Goertek Europe for supplying us with their AM3D spatializer plugin for Unity, an Oculus VR HMD and Sony MDR-7506 as well as suggestions for this study.

## References


1. Sony Electronics Inc.: SONY MDR-7506. https://pro.sony/ue_US/products/headphones/mdr-7506. Accessed 22 Apr 2020
2. Balan, O., Moldoveanu, F., Moldoveanu, A., Butean, A.: Developing a navigational 3D audio game with hierarchical levels of difficulty for the visually impaired players. In: RoCHI, pp. 49–59 (2015)
3. Cervera, A.S.: Effects of room acoustics on players' perceptions in audio games. B.S. thesis, Universitat Politècnica de Catalunya (2017)
4. Facebook Technologies, LLC: Oculus Integration 15.0. https://developer.oculus.com/downloads/package/unity-integration. Accessed 22 Apr 2020
5. Facebook Technologies, LLC: Oculus Rift. https://www.oculus.com/rift. Accessed 22 Apr 2020
6. Facebook Technologies, LLC: Oculus Spatializer 15.0. https://developer.oculus.com/downloads/package/oculus-spatializer-unity. Accessed 22 Apr 2020
7. Goertek Europe ApS: AM3D Spatializer. http://goertek.eu/solutions/am3d-software-suite/spatial-audio. Accessed 22 Apr 2020
8. Kuttruff, H.: Room Acoustics. CRC Press, Boco Raton (2016)
9. Mendoza, M.L.D.: Towards measuring and improving human sound localization and physical response to perceived sound through auditory conditioning (2016)
10. Middlebrooks, J., Green, D.: Sound localization by human listeners. Ann. Rev. Psychol. **42**, 135–159 (1991). https://doi.org/10.1146/annurev.ps.42.020191.001031
11. Roginska, A., Geluso, P.: Immersive Sound: The Art and Science of Binaural and Multi-channel Audio. Taylor & Francis, Milton Park (2017)
12. Rudrich, D., Zotter, F., Frank, M.: Evaluation of interactive localization in virtual acoustic scenes. 43. Jahrestagung für Akustik (DAGA 2017), pp. 279–282 (2017)
13. Unity Technologies: Unity 2019.3. https://unity.com/releases/2019-3. Accessed 22 Apr 2020
14. Vorländer, M.: Auralization: Fundamentals of Acoustics, Modelling, Simulation, Algorithms and Acoustic Virtual Reality. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-48830-9
15. Zhong, X., Xie, B., Glotin, H.: Head-related transfer functions and virtual auditory display. In: Soundscape Semiotics-Localization and Categorization, vol. 1 (2014). https://doi.org/10.5772/56907