Acronyms

VAE Virtual Acoustic Environment . . . 5
VSS Virtual Singing Studio . . . 9
RIR Room Impulse Response . . . 5
ISM Image Source Method . . . 6
TO Transition Order . . . 8
VBS Vector Based Scattering Method . . . 23
LR's Late Rays . . . 23

Contents

INTRODUCTION

2.1 - PROJECT OVERVIEW

The project described in this paper aims to extend the functionality of a currently implemented system called the Virtual Singing Studio situated in the listening space of the Audio Lab at the University of York. The system allows a user to hear themselves as though they are present in another acoustic environment. This project looks into extending this functionality to allow the user to move themselves around a digitally modelled version of Hendrix Hall, one of the lecture halls on the Universities campus by generation a large grid of synthetic room impulse responses using the room acoustic simulation software Odeon.

The aims of the project are as follows:

- Extend the Virtual Singing Studio to allow a user to select any position the want within the given virtual space
- Allow the user to feel as though they are freely moving around the space
- Investigate the plausibility of the produced system
- Investigate the perception of mobility given different Room Impulse Response densities provided.

2.2 - PROJECT MOTIVATION

2.3 - BACKGROUND

The following section covers material that forms the basis of the system produced as a result of this project.

2.3.1) Virtual Acoustic Environments

Virtual acoustics has been previously described [1] as follows:

> *"Virtual acoustics is a general term for the modelling of acoustical phenomena and systems with the aid of a computer"*

By this definition, a Virtual Acoustic Environment (VAE) can be thought of as an environment (such as a room) for which the acoustical phenomena have been either recreated or synthesised. To produce a VAE, prior knowledge regarding the room which is to be acoustically recreated must be known; how do all audible frequencies propagate around the room for a set sound source location and receiver location?

This information can be gathered by taking a Room Impulse Response (RIR) and used to recreate the acoustics of a room for the set sound source and receiver location.

2.3.2) Room Impulse Responses

In order to reproduce the acoustical phenomena of a room, an RIR must be obtained. This is done by exciting all audible frequencies within the room by using a sound source such as a loudspeaker, and recording the result using a receiver microphone.

There are a number of techniques used for exciting all audible frequencies. These include an **impulse** (such as a starter pistol) or an **exponentially swept sine** for which a sine wave is exponentially increased in frequency over a fixed period of time. Using a starter pistol means that no post processing is required as all the frequencies are excited at the same time and the impulse recorded at the receiver position can be used for convolution with an audio source. Using an exponentially swept sine requires post processing in order to time align all of the frequency dependent room reflections, thus producing an impulse response through the use of a deconvolution algorithm, however this method produces a greater signal to noise ratio thus is the desired method [2].

Though using an omni-directional microphone to record an RIR is a set standard [3], it is also possible to record RIR's using techniques such as Ambisonics to capture a three dimensional sound field.

2.3.3) Ambisonics

Ambisonics is a technique used to encode and decode three dimensional spatial audio information using just four audio channels. A three dimensional sound field can be recorded using a microphone known as a Soundfield microphone shown on the left in figure 1. These microphones contain four coincident capsules, one of which is an omni-directional capsule (W) and the rest of which are figure of 8 capsules used to record sound in the X (front and back), Y (left and right) and Z (up and down) direction illustrated on the right in figure 1. By combining these signals in what is known as a B-Format audio file, the sound field surrounding the microphone can be captured. By using a system specific Ambisonic decoder, the soundfield can be accurately reconstructed by replaying the B-Format file over a spherical loudspeaker array of arbitrary size.

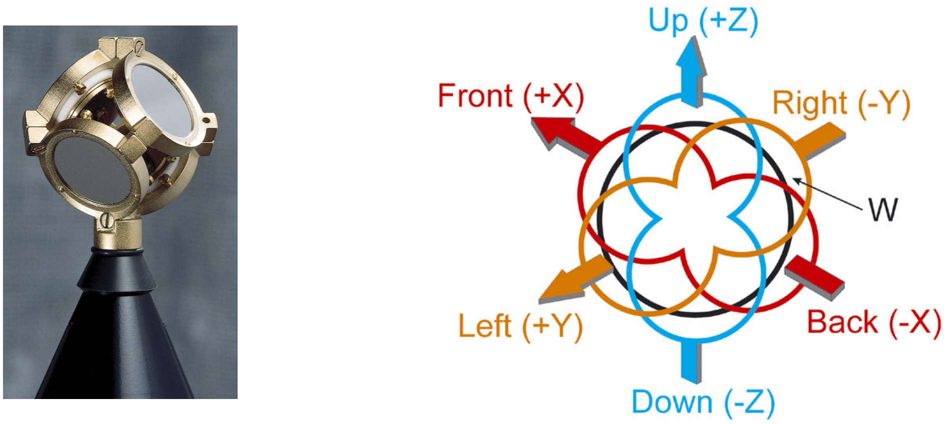


Figure 1: **Left**: Picture of a Soundfield microphone with coincident capsules exposed **Right**: Soundfield microphone polar pattern **soundFieldMic**

2.4 - Software Overview

2.4.1) ODEON

A method for measuring RIR's has been discussed, however there is also a way to synthesis RIR's by using room acoustic simulation software such as Odeon [4].

Odeon was designed to provide reliable predictions of room acoustics by using a hybrid of two geometric acoustic models, Ray-tracing and the Image Source Method (ISM) to synthesis RIR's. These methods model how sound would propagate around a room as though it were a straight line (ray). This inherently neglects wave phenomena such as phase and diffraction, properties that are negligible at high frequencies, however they are fundamental in describing low frequency wave behaviour [5]. Therefore geometrical methods are not accurate at modelling sound propagation for low frequency waves.

The reason for Odeon using a hybrid of the two method comes from the inherent problems encountered in each method.

2.4.1.1 Ray-Tracing

The ray-tracing method imitates a sound source by emitting a large number of particles in various directions from a single point [6]. These particles are then traced around the room, losing energy each time the particle encounters a surface according to the absorption coefficient assigned to that surface. The angle at which the particle is then reflected is determined by the scattering coefficient assigned to the surface of contact, ranging from a specular reflection to a completely random reflection [7]. For a specific receiver position, an area around said point is defined in which rays are collected and used to calculate the results.

Ray-tracing does not provide a completely accurate result as it is a risk that some rays may not pass close enough to the receiver to contribute to the final result. The outcome of the ray-tracing method is a statistical result rather than a complete one.

2.4.1.2 Image Source Method

The ISM can be used to find all possible specular reflections paths from the sound source to the receiver position. This is done by representing each specular reflection from a surface as a secondary source known as an "image source" [6], illustrated in figure 2. The advantage of the ISM is that each surface can be modelled by an image source which provides a great deal of data regarding the contribution to the received sound making this method more accurate than ray-tracing, however as each image source then emits more rays there is a possibility to produce a great deal more image sources. This means that the number of image sources grows exponentially for each new image source than is produced making this method much more computationally expensive than the ray-tracing method. This problem can be avoided by setting a *reflection order* which determines how many times a ray can reflect off a surface before calculations are stopped, thus preventing the creation of more image source though the calculates for the prediction of the rooms acoustics will be incomplete. This obviously then makes the ISM less accurate with a smaller reflection order.

Once all image sources have been calculated, a visibility check is run. This checks to see whether the image sources that have been produced can be seen by the receiver, therefore checking whether that image source needs to be used or not. If the image source is not in sight of the receiver, it therefore does not need to be used for calculations. If the receiver position is moved, the visibility check is run again. Overall, this saves time as the image sources do not need to be recalculated.

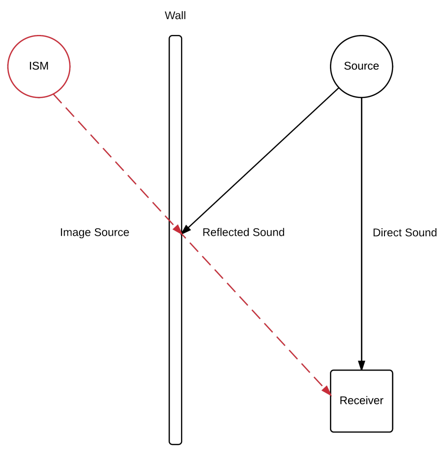


Figure 2: Illustration of the image source method (by the author)

2.4.1.3 Hybrid Method

Ideally, the ISM would be used to calculate all sound rays due to its accurate results, however due to the computational limitations, Odeon uses a hybrid method to provide a reasonable compromise between calculation time and accuracy.

The hybrid method first uses the ISM to calculate a number of image sources up until a specified reflection order determined by a Transition Order (TO). For example, if the TO = 2, the image source method will allow a ray to reflect twice which will produce a number of image sources, then it will switch to using the ray tracing method to calculate a statistical model of how the rest of the rays might propagate. This gives the user the choice between computation time and accuracy.

2.4.1.4 Scattering

Quote from Odeon Manual:

> *"In short, each time ODEON detects an image source (IMS), an inner loop of (scatter) rays (not visualised in the 3D Investigate Rays display) is started, taking care of the scattered sound which is reflected from this image source /surface"*

Reword that shit

2.4.1.5 Room Modelling

In order to predict the acoustics of a space, Odeon requires a geometry file in the form of a .par file. This file contains information regarding the room dimensions, object dimensions and positions. This can be produced within Odeon itself by using the built in 'Extrusion Modeller' which allows a user to use a script like language to describe the rooms geometry. It is also possible to use 3rd party applications such as Google SketchUp [8], a software which is described in section 2.4.2 Google SketchUp.

2.4.1.6 Material Selection

Selecting the appropriate absorption coefficients of the surfaces within a VAE is crucial to its accuracy. Odeon provides a material list with common materials that can be assigned to the surfaces of a model read from a geometry file. This material list can be extended by creating new materials and assigning absorption coefficients to the appropriate frequency bands.

2.4.2) Google SketchUp

Google SketchUp [8] is an easy to use 3-D modelling software that can be used to produce room models. Unlike Odeon extrusion modeller, it allows the user the draw surfaces with a mouse, easily duplicate structures and provides simple measuring tools and markers to enable accurate modelling. Plug-ins such as SU2Odeon ("SketchUp to Odeon") [9] enables the user to convert the model into a .par file for Odeon to use as a geometry file.

2.4.3) Max/MSP

Max/MSP (Max) is a visual programming language that can be used to easily route and manipulate audio signals through the use of simple objects and patches [10]. Max allows a number of external devices to be easily connected to the software such as smart phones and motion sensors through the use of plugins.

[More on Max]

2.4.3.1 Spat

Spat, developed at the research institute IRCAM [11] is designed for real-time spatialisation of sound signals in Max without having to touch any code. It provides an simple way to convolve B-format audio signals with impulse responses as well as decode the B-format signals for any given number of speakers in any arrangement by simply providing the number of speakers and angles at which they are placed.

2.5 - Project Description

2.5.1) The Virtual Singing Studio

The Virtual Singing Studio (VSS) is a loudspeaker based room acoustics simulator used as a tool for analysing the correlation between room acoustic characteristics and vocal performance parameters as part of Dr Jude Breretons PhD Thesis [12]. It is comprised of a head-mounted microphone used to capture a real-time audio input from a singer, a software patch that convolves the audio signal with a number of Ambisonic B-Format RIR's and finally a spherical array of 16 loudspeakers for which the convolved audio signal is decoded and fed to. In addition, a head-tracking device (an Oculus Rift [13]) is also used to track which direction the user is facing in the virtual space. A flow diagram of the system can is shown in figure **??**.

[Add VSS system diagram here]

The VAE used in the VSS was initially the National Centre for Early Music, a space in York frequently used for musical performance. Using a Soundfield microphone, 16 Ambisonic B-Format

RIR's were captured. Four positions were chosen and four RIR's facing in each direction (front, left, back, right) were recorded in each. These four directional RIR's are used to approximate the room acoustic phenomena that would occur if the user were projecting in that direction. This is done by using the data from the head-tracking device to amplitude pan the convolved signals before sending them to the spherical speaker array. Though this is a crude method for achieving such a state, it provides enough variation between the two directions to be considered convincing.

2.5.2) Project Motivation

The VSS addressed a problem faced when trying to research how musicians preform in different acoustic environments: having to travel to each performance space with musicians and researchers. This also indirectly provides a solution for performers wanting to rehearse is spaces that are often inaccessible and would otherwise be expensive to book and travel to. By obtaining an RIR of the desired location, the only time travelling will be necessary is to initially obtain said RIR. However, one limitation of using the VSS is the restriction of position. If a performer wanted to try and sing at another point in the room, an RIR would have to be taken in that position too. This could be done initially, taking a range of RIR's in a number of positions, however it cannot be guaranteed that all positions desirable to the performer will be available. Therefore, this project aims to implement a system that will allow the user of the VSS to feel as though thy can move around a VAE freely.

Previous projects have addressed mobility within VAE's before, such as [14]. The paper initially looks at two methods for providing mobility:

1. **Direct room impulse response rendering**: By producing a large grid of RIR's within the desired space, it is possible to allow a user to select a position in the room by interpolation between the two nearest neighbouring RIR's. This method does not provide truly accurate room acoustics for the selected location and also suffers from the fact that a large number of RIR's are required, thus the requirement for a large amount of storage space.
2. **Parametric room impulse response rendering**: By actually synthesising RIR's in real time for the given position of the user in the VAE, it is possible to provide an accurate RIR for the given location. This method avoids the need for a large amount of storage space and a system to retrieve the correct files, however, as will be seen through the rest of this report, post-processing of the obtained RIR's would not be possible with real time rendering.

As the intention was to build upon the existing VSS system, it was decided that the direct room impulse response rendering method would be used, as this method would allow for the modification of the existing patch used to real-time convolution.

As the intention was to build upon the existing VSS system which requires pre-recorded (and post-processed) RIR's, it was decided that the direct room impulse response rendering method was to be used and the effects of the issues related with the method studied.

2.5.3) Project Objectives

Given the project aims and the background provided, the following is a list of objectives that were set in order to complete this project:

1. Find an appropriate room to be modelled as a VAE
2. Digitally model the room using either Odeons extrusion modelling or 3rd part application
3. Import room model into Odeon to finish material selection and RIR settings
4. Produce a grid of RIR's that can be used to interpolate between to simulate user positions
5. Record real RIR's in locations that can be compared to synthetic RIR's
6. Extend upon existing software patch to accommodate new functionality
7. Preform user tests:
    - Test #1: Does the perception of distance change when using real or synthetic RIR's?
    - Test #2: How far does the user have to move in the given VAE before they notice they have moved
    - Test #3: How many RIR's are required for interpolation for the user to feel they are moving around the space freely

In order to test the perceptual differences when using synthetic RIRs (Test #1), real RIRs of the same space had to be taken.

Previous test in the VSS have required 'Plausibility' test, where the user must evaluate the VAE produced by the VSS without reference to the real venue. This is usually because testing a virtual environment against a real one (authenticity tests) requires travel which can be expensive and difficult. Therefore, by running test based on 'plausibility' a sense of how convincing the virtual room is can be obtained. In the case of other virtual reality systems where the virtual environment does not exist in the real world, only these types of test can be run. Though plausibility test are acceptable, an authenticity test gives more objective results by comparing the simulated VAE to the real thing. Therefore impulse responses of Hendrix Hall were taking, in the same format as the synthesised RIR's.

2.5.4) Project Overview

The following outlines the steps that were required to produce a system that allowed a user to feel as though they are freely moving around a VAE and the tasks necessary to investigate the plausibility of said system, thus completing the project aims and objectives set

2.5.5) Room Choice

The following room features were kept in mind when searching for a room to use as part of this project:

Size:	Large enough to be used as a singing space
Simplicity:	Simple enough architecture to be able to model with the time available
Accessibility:	The room had to be easily accessible to take multiple measurements to make blue prints and take RIR measurements

The room chosen was Hendrix Hall, a large lecture theatre on the University of York campus. The room contains retractable seating leaving the large space in the centre open and unobstructed. The room is architecturally simple being almost perfectly rectangular with the occasional wall indent. With it being located on the universities campus it can be booked for any time of the week meaning it can be accessed when necessary and for free.

2.5.6) User Test Planning

In order to perform the final user tests (test #1 and test #2), minimum distance between RIR's had to be decided.

IMPLEMENTATION

This section describes the steps taken from setting the project objectives to reaching the final implementation of the desired system.

3.1 - Room Modelling

3.1.1) Room Measurements

Before taking room measurements, a quick top-down map of the room was made, highlighting objects, wall indents or protrusions and where doors and windows existed. The dimensions of the room were then measured in meters and noted on the not to scale room map. The tool used for measurements was a DeWalt laser distance measurer [15]. This allowed for accurate measurements of distances that would otherwise not be accessible such as the distance between roof lighting and other fixtures. Once the basic layout had been mapped, maps of individual walls were made and features such as window and door dimensions, distanced between doors etc were noted. The example of an annotated blueprint can be seen in figure 37 in Appendix A.

Hendrix Hall measured at approximately 18.3m x 18.2m x 5.5m.

3.1.2) Designing the room

The blue prints were used to model the room in Google SketchUp starting with a hollow rectangle with the dimensions of Hendrix Hall. From this the wall indents and protrusions were modelled by using a push/pull tool. Figure 3 shows an early iteration of the SketchUp model where several wall protrusions have been modelled.

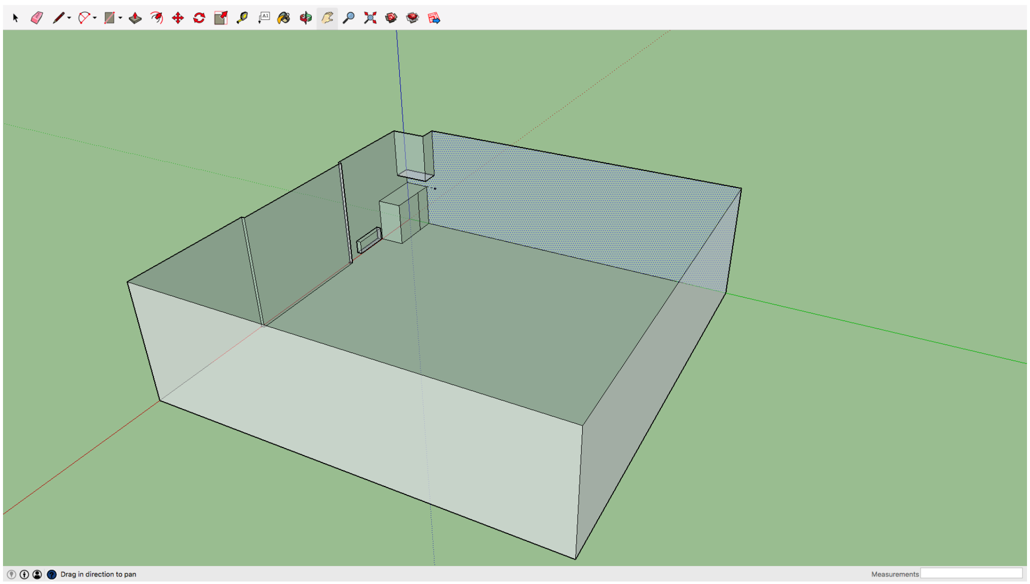


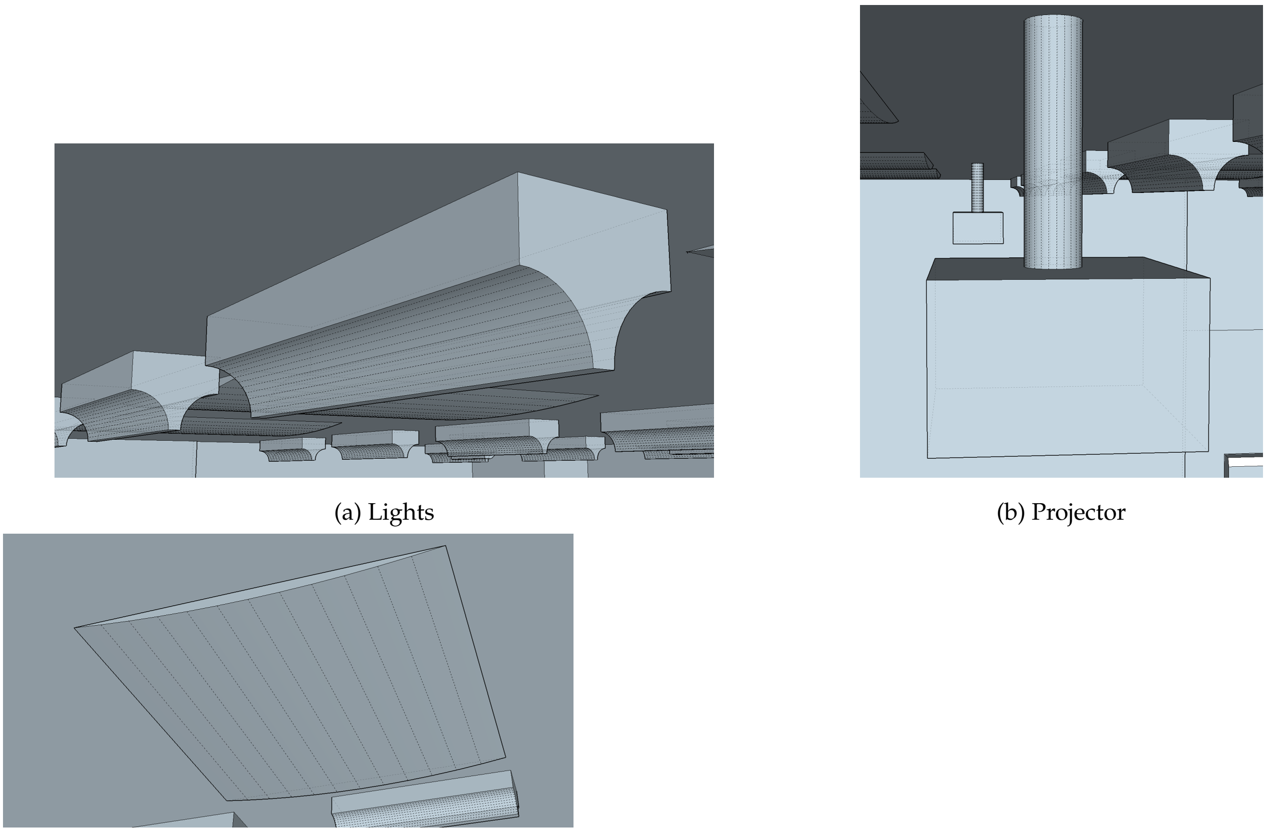
Figure 3: Early iteration of the Hendrix Hall SketchUp model with a few early wall protrusions being modelled such as the entrance door

The contents of Hendrix Hall are predominantly flat surfaces though some more complex surfaces include:

- Lights (concave curves)
- Canvas roof hangings (convex curves)
- Projector hangings (poles)
- Radiator, door and vent grills

These objects were initially designed to look as close to the real thing as possible, however the objects with curved edges posed a problem when importing the model into Odeon. This is because SketchUp uses a large number of short surfaces to represent curves as can be seen in Figure 4 which shows the initial models of the lights, projectors and roof hangings. According to the Odeon manual [7], when modelling a room for acoustic simulation purposes it is more accurate and less time consuming to keep the model simple and to add the appropriate scattering coefficients or materials in Odeon itself. This also applied to the objects (such as the radiators) that contained a grilled surface, as a specific 'grill' material could be selected from Odeons material list (see section Material Selection).

These objects, shown in figure 5 were then redesigned more simply where the roof hangings were represented as four joint slanting surfaces and lights and projector poles are simple rectangles.

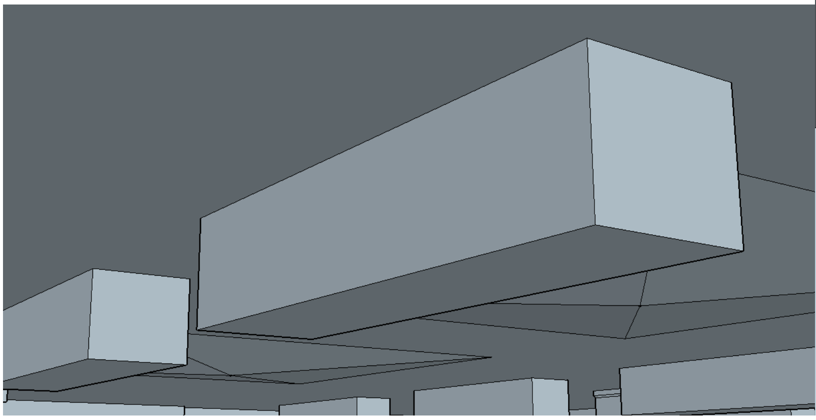


(a) Lights

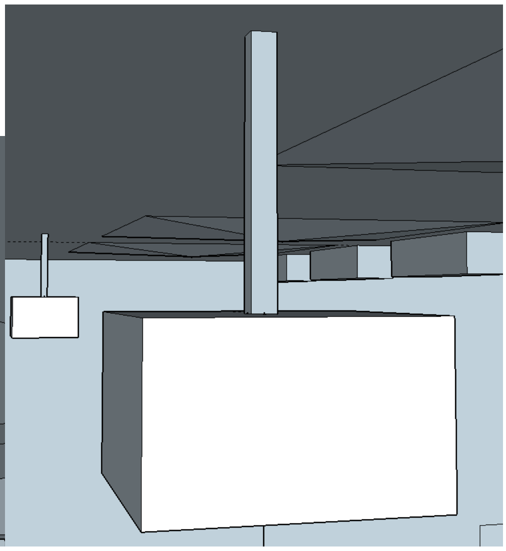
(b) Projector

(c) Roof Hanging

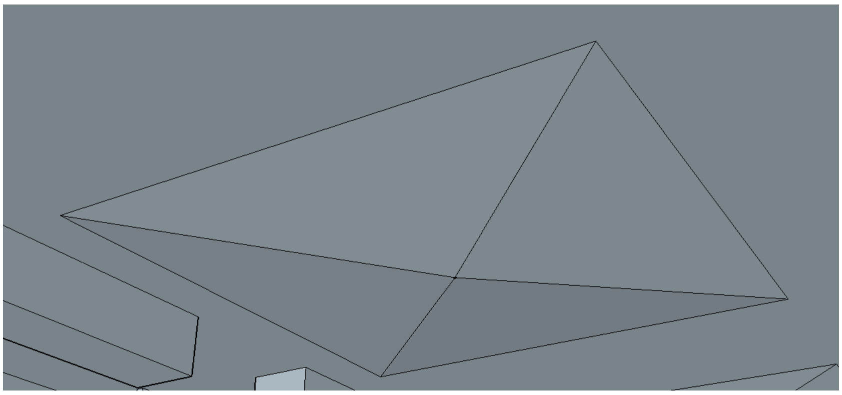
Figure 4: Initial models of the lights and projector hangings made of a large number of surfaces



(a) Simple Lights



(b) Simple Projector



(c) Simple Roof Hanging

Figure 5: Simplified models of the lights and projector hangings containing a smaller number of surfaces

3.2 - ODEON

3.2.1) Water Tightness Test

Once the SketchUp model had been exported as a .par file using the SU2Odeon plug-in [9] it could be opened in Odeon and checked to ensure that there were no gaps in the model for which rays to escape. If this was the case, the model would have to be fixed in SketchUp and reimported into Odeon. Odeon makes checking the model easy by running a '**water tightness**' check where a large number of rays are reflected around the room seeing whether any of them manage to escape. Figure 7 shows the Hendrix Hall model undergoing such a test. Once it was ensured that the model was fit for use, the surface materials could be assigned.

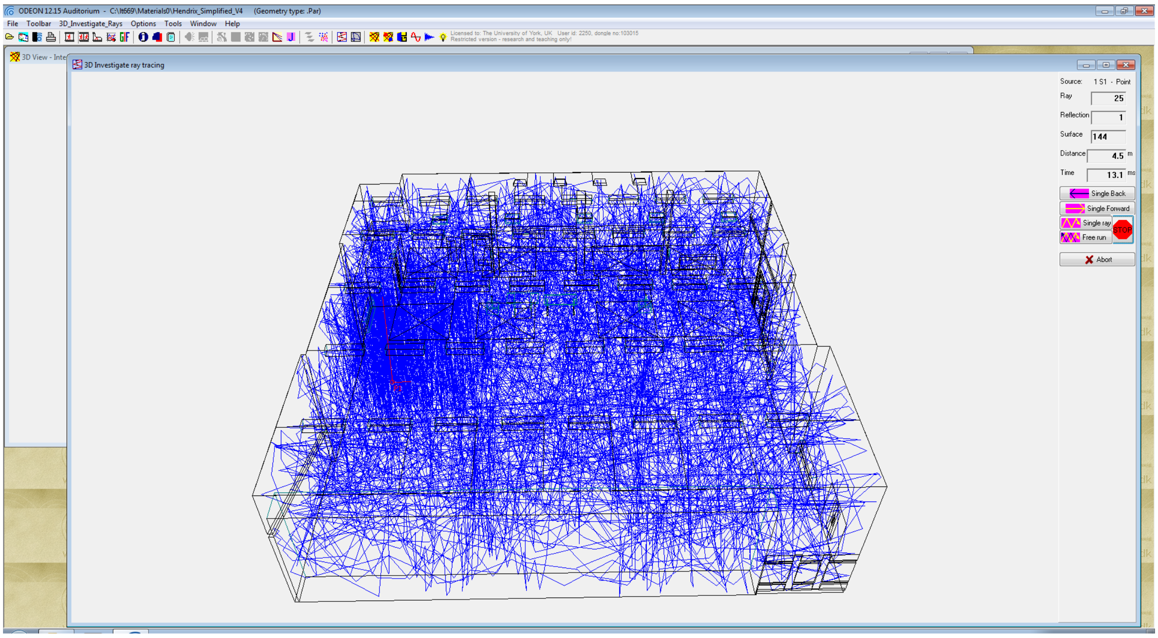


Figure 7: Hendrix Hall model undergoing a water tightness test in Odeon.

3.2.2) Material Selection

The surface materials within a room heavily influence how sound propagates around a room, effecting reverberation time and frequency content of said reverb. It is therefore imperative to assign materials as close to the real material as possible in order to produce an accurate representation of a known acoustic environment. For this, Odeon provides a list of common materials often found when constructing buildings.

3.2.2.1 Initial Materials

Odeon's material list appears to be designed for auralisation of structures with very basic interior. Due to a lack of choice, exact materials in the room could not be modelled accurately, however the closest match to what is thought to be the true material was selected as a replacement instead. In some cases, appropriate replacement materials were not available, therefore new materials had to be added to the material list. This can be done by finding a materials absorption coefficients, selecting '**Edit an existing material**' where new materials absorption coefficients can be entered and saved as a new material as shown in figure 8. Absorption coefficient values range from 0 - 1 (0% - 100%), indicating the percentage of attenuation applied to the selected frequency band upon a contact with the surface. Required materials for which an appropriate replacement could not be found in the material list are listed in table 2.

Material	Surface applied to
Hard Plastic [16]	Roof lights and projector covers
Mineral fibre [17]	Ceiling Tiles
Slate [1] [18]	Blackboard

Table 1: Table of materials for which absorption coefficients were sought and added to Odeon's material list.

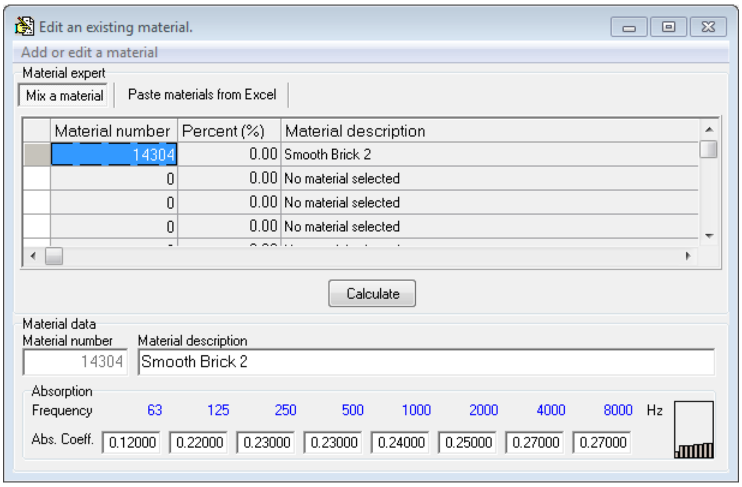


Figure 8: Absorption coefficient editing window in Odeon used to add unavailable materials.

3.2.2.2 Surface Types

For a number of surfaces it is appropriate to edit properties other than just their absorption coefficients. As explained in section Designing the room, the roof hangings are constructed of four joint slanting surfaces. By changing the individual surface types from **'Normal'** to **'Fractional'**, Odeon avoids erroneously calculating the diffraction caused due to each of the individual surfaces and treats them as a whole surface. For surfaces such as the contracted seating shown in figure 9 where gaps are present in the over all structure, it was possible to model this as one solid object and to set a **transparency** value. A transparency value of 0 means the surface is a solid where as a value of 1 makes a surface totally transparent which will let ways pass straight through. A transparency value of 0.3 was chosen as a reasonable estimate, the effects of which are shown in figure 10, showing how a ray can pass through the front of the object, propagate around the inside and eventually reflect back out again.

[1]The absorption coefficients provided were only available for the 125Hz to 4kHz octave bands, therefore absorption coefficients for the 63Hz and 8kHz octave bands were given a value of 0.1

Figure 9: Contracted seating in Hendrix Hall showing gaps in the structure.

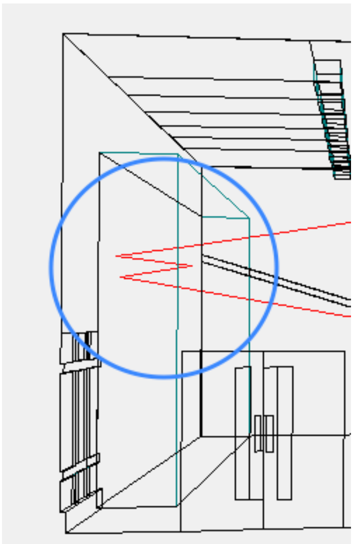


Figure 10: Blue circle highlights a ray penetrating the modelled seating area, reflecting 3 times and escaping the seating area due to the transparency value set.

As previously mentioned in section Designing the room, the lights were modelled as simple rectangles as opposed to the more complex objects made from a large number of surfaces with the intention of more accurately modelling the scattering effect (descripbed in section Scattering) due to their shape by altering the objects **Scattering Coefficient**. Odeon provides a table of initial indicators for possible scattering coefficients, shown in figure 11 which can be used to select the scattering coefficient for a surface that fits a similar description to the one given. A scattering coefficient of 0.2 was selected for the lights by assuming the effect would not be as sever as a bookshelf. **[Seriously, re-word this bit]**.

Material	**Scattering coefficient at mid-frequency (707 Hz)**
Audience area	0.6–0.7
Rough building structures, 0.3–0.5 m deep	0.4–0.5
Bookshelf, with some books	0.3
Brickwork with open joints	0.1–0.2
Brickwork, filled joints but not plastered	0.05–0.1
Smooth surfaces, general	0.02–0.05
Smooth painted concrete	0.005–0.02

Figure 11: A table provided in the Odeon Manual [7] that can be used to set an approximate scattering coefficient for surfaces similar to those described

3.2.2.3 RIR Comparison

Once the initial materials had been selected, the authenticity of the RIR's were checked by comparing their frequency content against that of a real measured RIR from Hendrix Hall (see section Real RIR Recordings). This was done by using Matlab to plotting the spectrograms of the omni-directional W channel of both RIR's, as can be seen in figure 12. The spectrogram shows the high frequency content in the real RIR attenuating in a smooth roll off fashion from 20kHz to about 8kHz starting from about 0.2 seconds in, where as the full audible spectrum appears to attenuate almost evenly in the Odeon produced RIR.

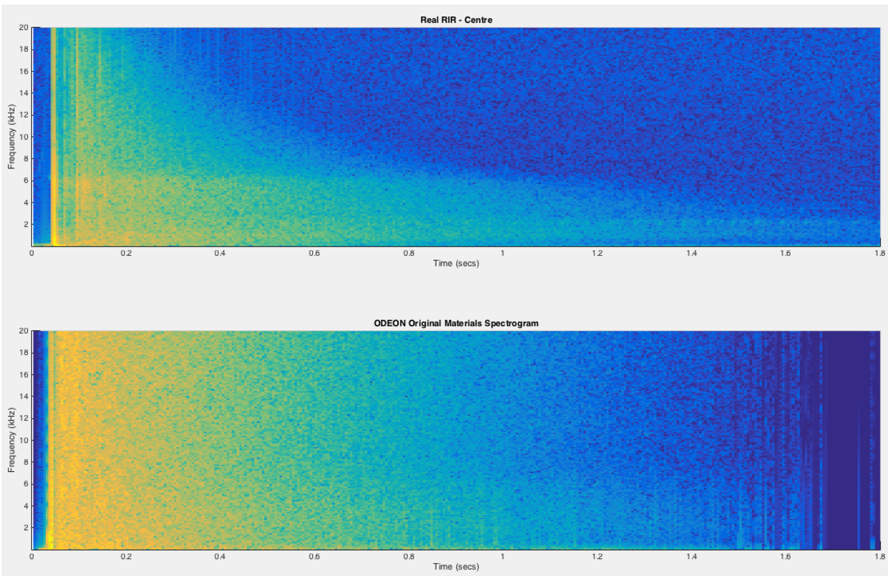


Figure 12: Spectrograms of real RIR (top) against spectrogram of first rendered RIR from Odeon.

In an attempt to produce a more realistic synthetic RIR the surface materials were edited several times until .

As the most obvious difference between the two is the difference in high frequency attenuation, large surfaces with low valued high frequency absorption coefficients such as the walls and ceiling were edited to absorb more of the high frequency content. Several iterations can be seen in figure 13 showing the spectrogram of the real RIR and several iterations of the RIR's rendered from Odeon. The Spectrogram at the bottom is that of an RIR produced in Odeon with the final material selections. The difference between the real RIR and the final Odeon RIR can be seen in figure 14. This was done by adding 10% onto the existing absorption coefficients for the 63Hz - 2000Hz octave bands, 20% to the 4000Hz band and 30% to the 8000Hz band. This also included edited the ceiling material coefficients by adding an extra 10% to the 8000Hz octave band.

The following table contains a list of audio samples and edits made to the material list in order to produce each one, with their corresponding spectrogram graph name which can be found in figure 13.

Audio file	**Graph Name**	**Absorption Coefficient Edits**		
		Material	**Addition**	**Octave band**
RealRIR.wav	Real RIR	NA		
OdeonOriginal.wav	ODEON Original	NA		
Odeon3RIR.wav	ODEON 3	Ceiling (Mineral Fibre)	+10%	8kHz
Odeon6RIR.wav	ODEON 6	Walls (Solid Brick)	+10%	All
Odeon7RIR.wav	ODEON 7	Walls (Solid Brick)	+10%	4kHz
			+20%	8kHz
		Ceiling (Mineral Fibre)	+10%	8kHz

Table 2: List of audio samples with corresponding spectrogram title in figure 13 and the edits made for each one.

By analysing figure 14 it can be seen that it was possible to produce a much more fitting RIR compared to the one produced using the original materials. By listening to the audio samples, it is clear that though the final Odeon RIR (Odeon7RIR.wav) is much more similar to the real RIR (RealRIR.wav) than the one produced using the original surface materials (OdeonOriginal.wav) in terms of reverberation time, they still sound very different in terms of their frequency content, where the RIR's produced in Odeon lack what the author considers *depth*. This is most likely due to the fact that the geometrical acoustic modelling methods used to produce the RIR do not accurately reproduce the low-frequency content, the reasons for which were discussed in section **??**

[Consider adding more detail of material selection process here]

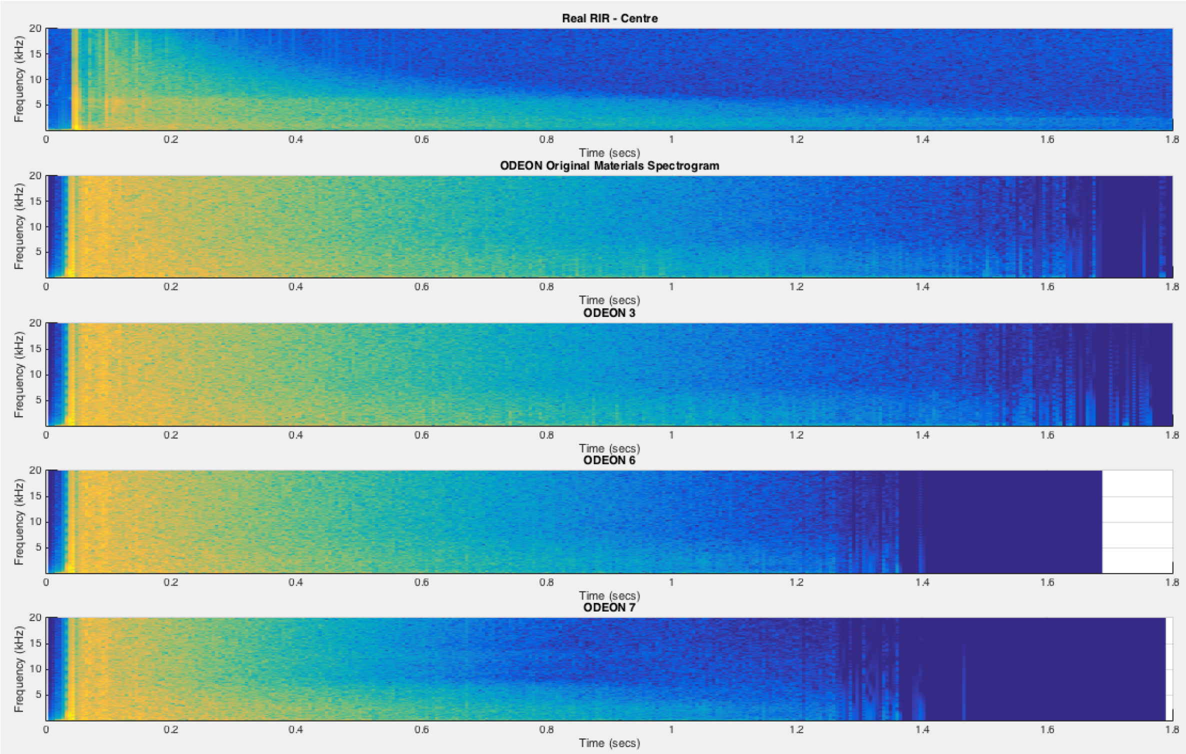


Figure 13: Spectrograms of real RIR (top) against spectrogram of several rendered RIR from Odeon with different material absorption coefficients where ODEON (bottom) shows the final RIR used.

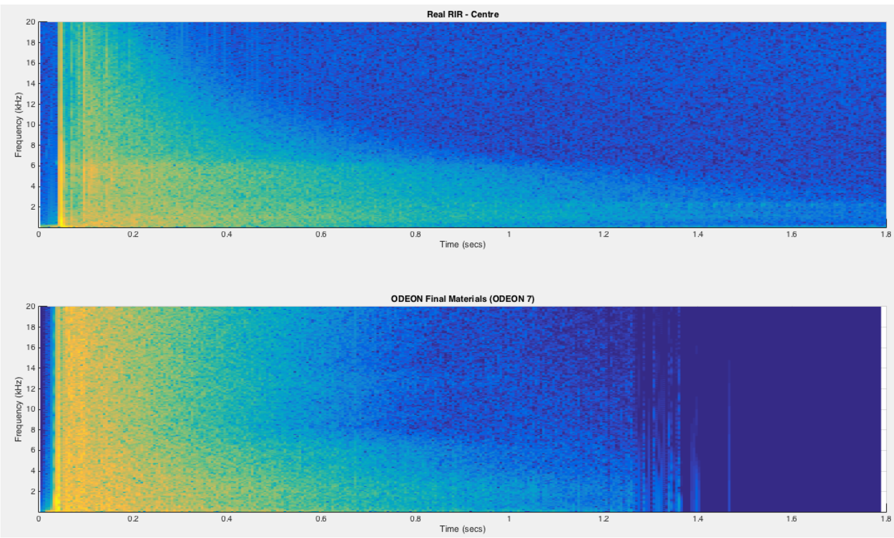


Figure 14: Spectrograms of real RIR (top) against Odeon RIR from Odeon with final material absorption coefficients.

A full material list is available in '`Material_List_V2.xlsx`', showing which materials are applied to each surface within the Odeon model.

It was later discovered (see section RIR Topology Problems) that the RIR's used to decide the final materials were incorrect, a result of which caused them to be much louder than they should have been. To ensure that the materials selected when using the incorrect RIR's were still appropriate for the new ones, two new RIR's were exported with the original materials and the final materials selected and plotted to compare against the same real RIR used before. As the new Odeon RIR's were so much quieter, the real RIR was reduced in level to match those produced by Odeon. The results are shown in figure 15. As it can be seen, the final materials used in the room model reduce the length of the reverb and attenuate high frequencies quicker than when the room model contained the original materials, the match is still not perfect.

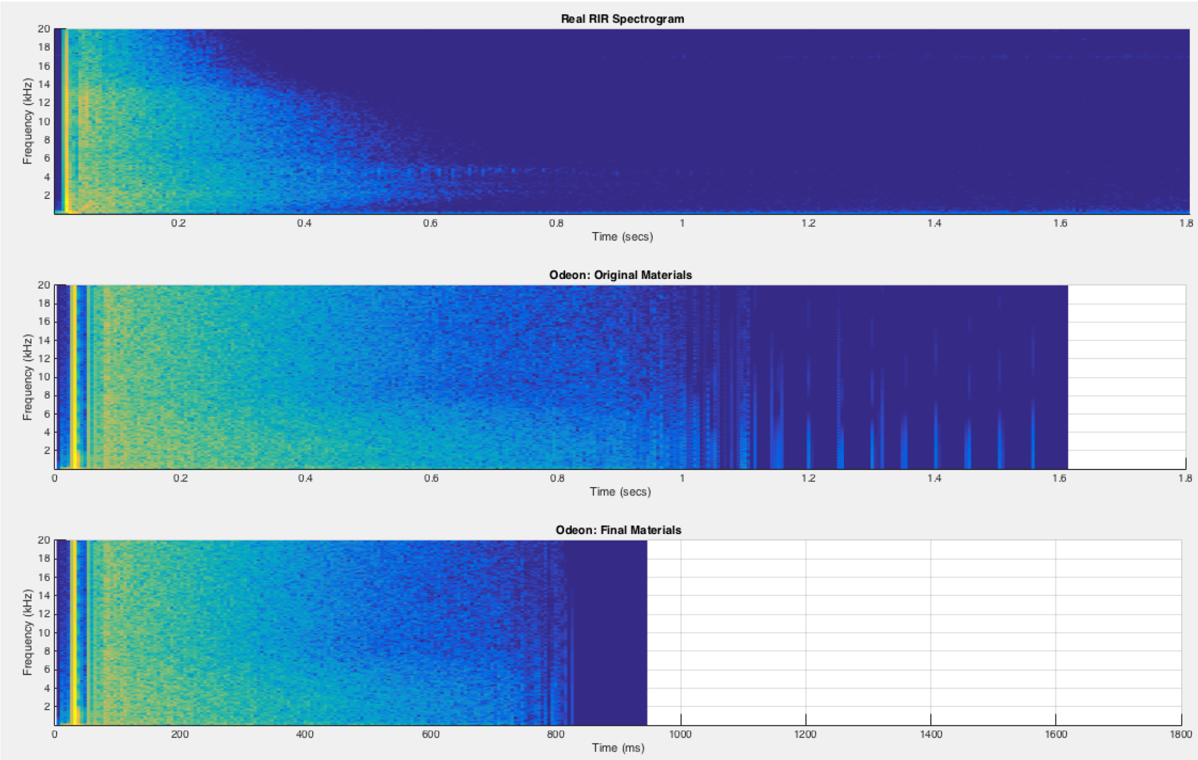


Figure 15: Spectrograms of correct Odeon RIR's with original materials (top) and final materials (center) to compare against real RIR that has been level calibrated (bottom).

NOTE: The reason the Real RIR looks so different to the previous one is because it is much quieter, therefore the spectrogram shows less energy much earlier on. However the roll off described the first time is still present and is highlighted, starting again from about 0.2 seconds in. Why do the Odeon RIRs look similar to before then?

3.2.2.4 Final Material Choice

The incorrect RIR's were used to calculate the room materials to begin with. Upon rendering new RIR's, three tests RIR's were rendered with the three main differences in materials selection in order to check that the spectrogram was close enough to the real RIR's

3.2.3) Odeon Output Settings

3.2.3.1 Ray Settings

Odeon provides the following options for RIR rendering:

Astop - The maximum possible attenuation of each octave band. **Apass** - Ripple of octave band filters in dB.

Band overlap - 100% overlap gives a smooth transition between the FIR octave band filters as they are not completely rectangular.

Maximum reflection order - Maximum allowed is 2000 as is the default. This means a ray can only reflect around a room 2000 times before the simulation is stopped. This prevents trapped rays from prolonging the impulse response if it never reaches the receiver.

Late Rays - These are emitted from the source and reflected according to the Vector Based Scattering Method (VBS) (described in section Scattering) taking into account scattering due to surface size and roughness.

Transition Order - As explained in section ODEON, a TO can be set to determine the number of early rays sent out to find a number of wall combinations for reflections using the ISM. After this number of rays has been reached, the ray-tracing method is used.

All settings apart from Late Rays (LR's) and TO are left at their default values as they are sufficient for accuracy and prevent the increase of computation time which will be needed given the large number of RIRs required for free movement.

Odeon suggests two possible modes, **Engineer** and **Precision** which both suggest using a different number of LR's which are 1285 and 20560 respectively. As both the TO and LR's value increase both the accuracy and computation time of the RIR's, different combinations were tried and RIR's were rendered. Figure 16 shows plots of the rendered RIR's with the name indicating the TO and LR's value. For example, 'TO_2_LR_1285' indicates the sample was produced where TO = 2 and LR's = 1285. Table 3 can be used to clarify the settings used for each RIR in figure 16 and indicates which audio sample can be listened to. For consistency, all RIR's were taken from the centre of the room.

Audio Sample	TO	LR's	Graph Name
TO_2_LR_1285.wav	2	1285	TO_2_LR_1285
TO_2_LR_20560.wav	2	20560	TO_2_LR_20560
TO_4_LR_1285.wav	4	1285	TO_4_LR_1285
TO_4_LR_20560.wav	4	20560	TO_4_LR_20560

Table 3: Table of audio samples with the corresponding settings information and graph name

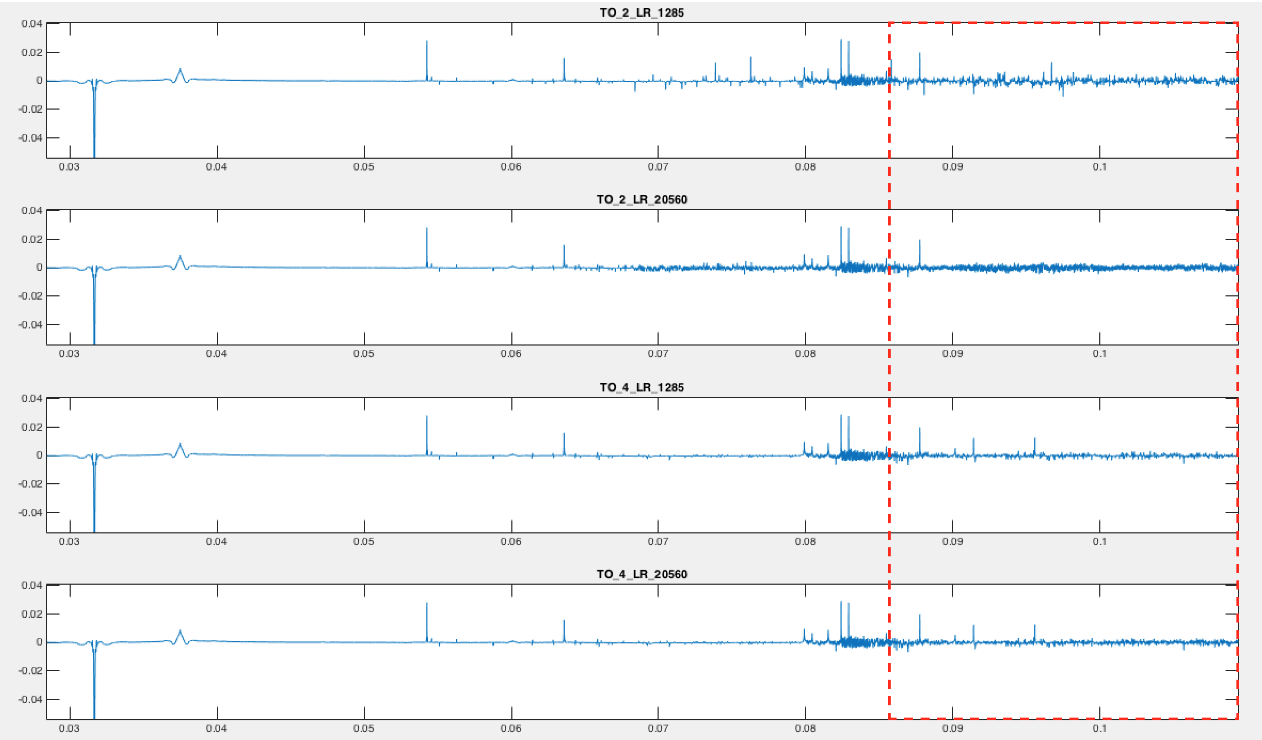


Figure 16: RIR's rendered using different TO and late ray values

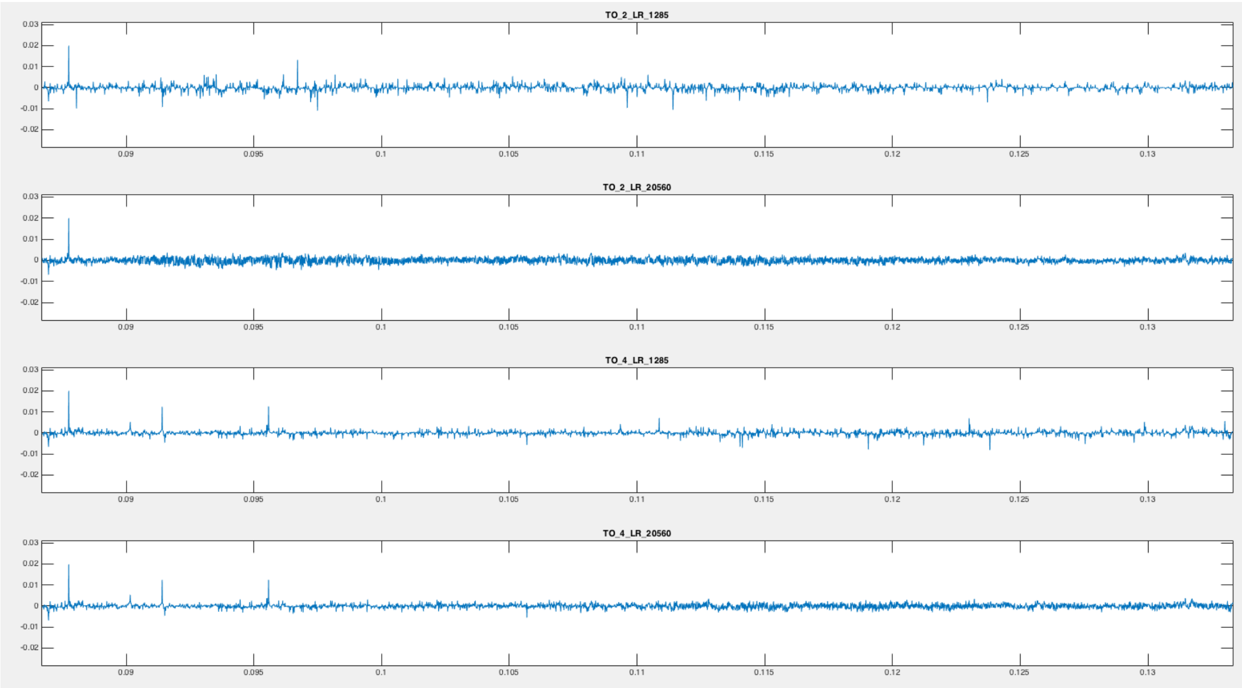


Figure 17: Zoomed in section indicated by the red dashed line in figure 16

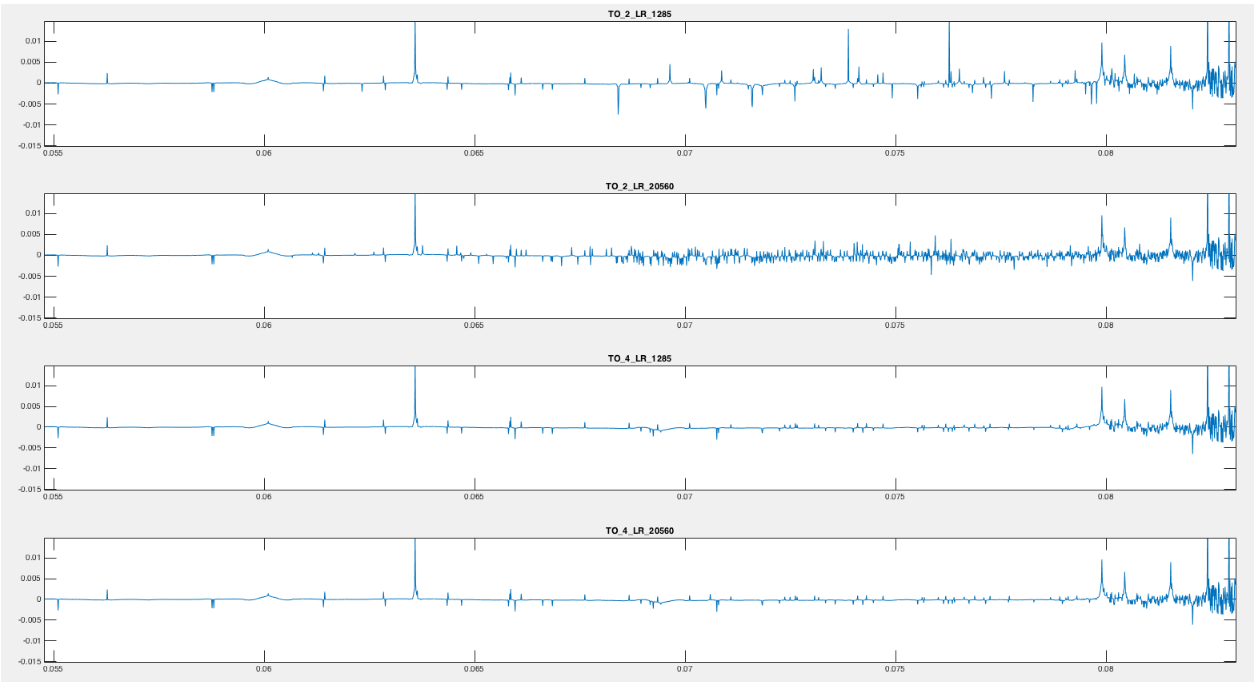


Figure 18: Zoomed in section showing the effects of using a high TO value

Audio samples: SupportingFiles/Audio/Odeon_Settings

Through aural analysis, it can be heard that the number of LR's used has an audible impact, where a lower LR's value produces an RIR that sounds more *grainy*. This can be seen in figure 17, showing the RIR's in figure 16 which have been zoomed to show the RIR from approximately 0.085s where a difference caused by the different setting combinations can be seen. The two RIR's with a high LR's value (2nd and 4th plot) show more densely packed reflections due to the fact that more rays are used. This produces a 'smoother' more natural sounding reverb tail.

Figure 18 shows an earlier part of the RIR's where the effect of using a high TO value can be seen. The two plots showing the RIR's with a TO of 2 show more random reflections earlier on where the ray-tracing method has been used, whereas the bottom two plots with a TO of 4 show less random reflections as the ISM will still be used. Though the difference in using a TO of 2 or 4 can be seen, is not audible.

The time difference taken to render the RIR's is also an important factor when choosing which of these settings to use. Given the potentially large number of RIR's required, a slightly more accurate output may result in a much greater calculation time. Odeon uses information gathered when previously rendering RIR's to speed up calculations for other RIR's being rendered in the same room, however there initial time taken to calculate each of the RIR's was noted and compared for reference.

Initially it was found that increasing the TO from 2 to 4 increased the time taken to render by approximately 20 seconds. However, It was later realised that inaccurate calulation times may have been given due to the fact the project was being stored on the University network, thus adding variable delay times. The project was later stored on a local C drive, eliminating any network speed issues and inconsistencies. After doing so, the difference in TO values made almost no difference to the calculation times. In hindsight, a higher TO value could have been used to produce more accurate RIR's.

It was found that the LR's value made quite a difference to the calculation time:

Transition Order	Late Ray Value	Calculation Time (s)
2	1285	11
2	20560	28
4	1285	11
4	20560	28

Despite the increase in calculation time by 17 seconds, the inaccuracy of RIR's produced when using the lower LR's makes using the higher value justifiable.

As the VSS requires B-format audio files, the B-format option was selected.

3.2.3.2 Directivity

Without selecting otherwise, Odeon uses an omni-directional point source as the sound source. This can be changed however by providing a .cf2 file. This file stores loudspeaker performance data and polar plots in what is known as a Common Loudspeaker Format, a standard used by loudspeaker manufacturers. By importing one of these files, Odeon simulates the directionality of the selected loudspeaker as the sound source. This can be used to attempt to more accurately recreate an RIR that would be taken in a real space, by finding the .cf2 file that corresponds to the loudspeaker used for the measurement.

It is also possible however, to input a custom directivity pattern to model a specific sound source. This provides an opportunity to create a more accurate sound source for a human head, something desirable when creating RIR's that will be convolved with the audio input from a singer. Techniques for recording the directivity of the human head have been reviewed and improved upon in [19], however only directivity data for the horizontal plane was recorded. A student from the University of York has recently investigated the directivity of a human head in both the vertical and horizontal planes ranging from the 63Hz to 8kHz octave band [20] **[CHANGE NAME]** and generously provided said data to be used in this project. Figure 19 shows the directivity patten editor window in Odeon with the input horizontal and vertical directivity patterns for the 8Khz octave band taken from [20].

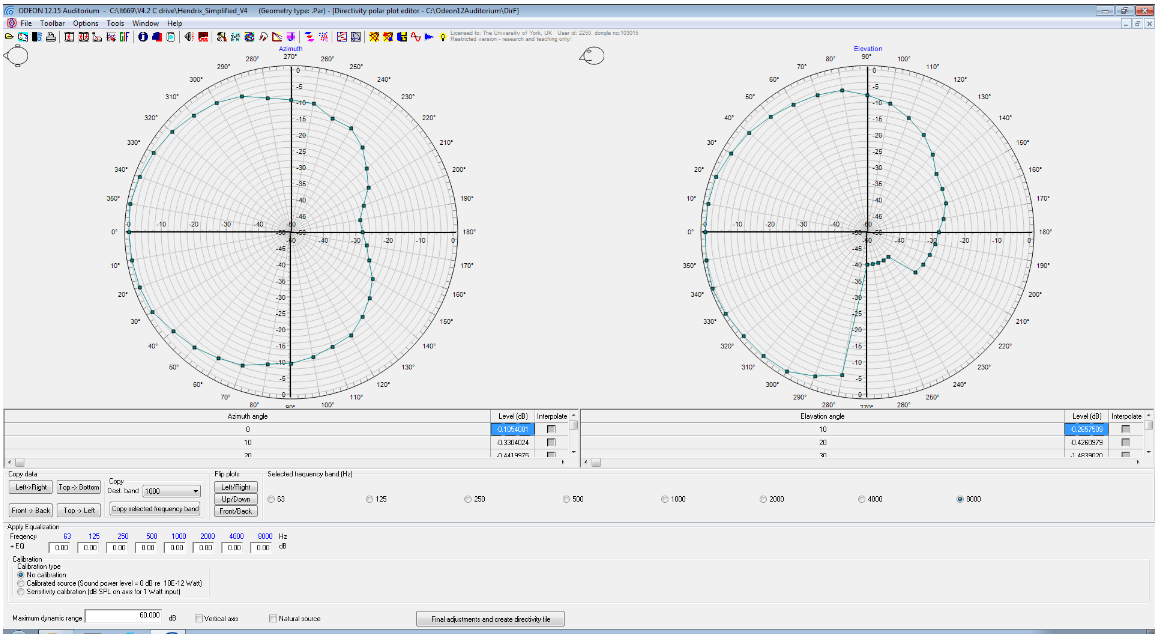


Figure 19: Directivity pattern editor window in Odeon showing the directivity pattern for the horizontal plane (left) and the vertical plane (right) for the 8Khz octave band using data from [20]

3.2.4) RIR Topology Problems

3.2.4.1 RIR Analysis

It has been noted that the RIR's produced will attempt to accurately resemble a human head. This involves placing the sound source below the receiver, to resemble the mouth (sound source) below the ears (receiver). When taking real RIR measurements it is often not possible to get the source and receiver close enough due to the physical dimensions of the equipment used and therefore result in an unnatural distance between them (see section Real RIR Recordings). However, with using software such as Odeon, where sound sources are calculated from a point source, it is possible to move the source and receiver much closer together. Initially, the sound source was placed 1m off the ground and the receiver was positioned 0.05m above the sound source. Three RIR's positioned along the centre of the room at varying distances from the left wall can be seen in figure **??**. **[Note where the grid positions are]**

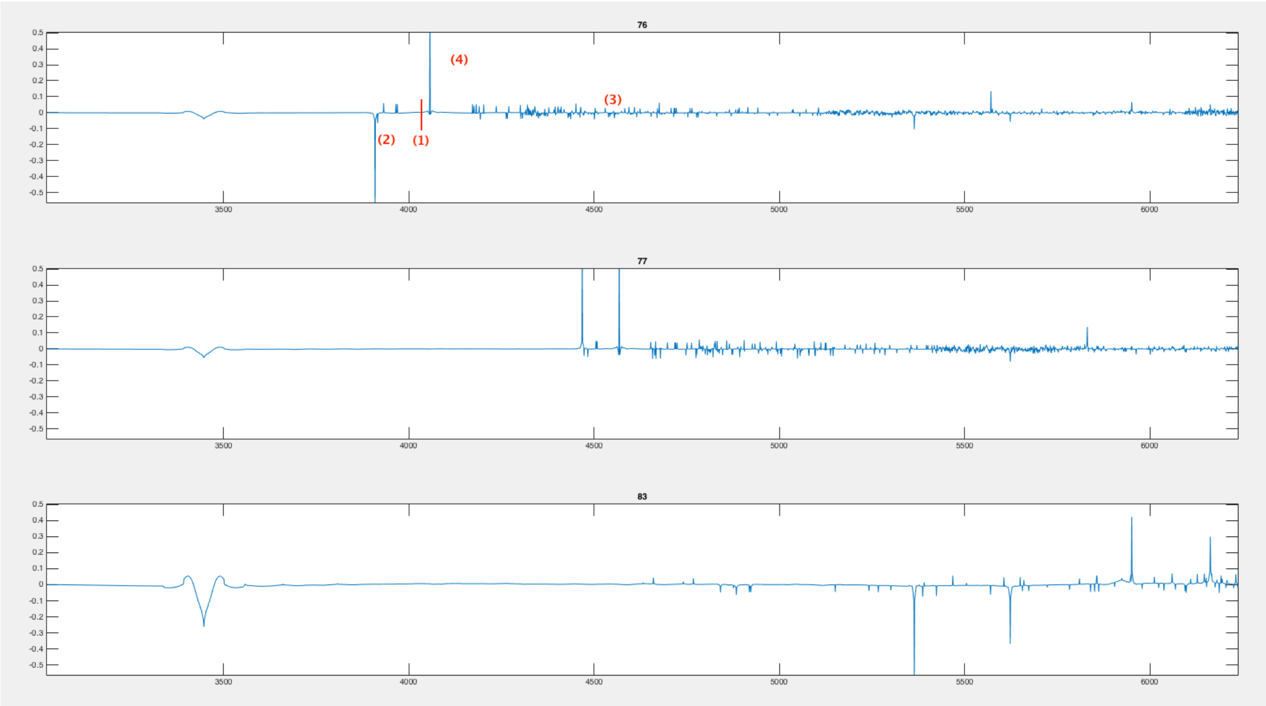


Figure 20: Initial RIR output with receiver 5cm above sound source showing RIR's from grid positions 76, 77 and 83

These RIR positions were chosen as the varying distance from the left wall can be used to estimate when to expect reflection other than that from the floor which will remain the same across each RIR. Each RIR shows a dip of varying amplitude at the same instance in time. Without further inspection, this could be assumed as the direct sound from the source to receiver, which would be in the same place in time as the source and receiver are kept the same distance across all RIRs, though the varying amplitude is questionable.

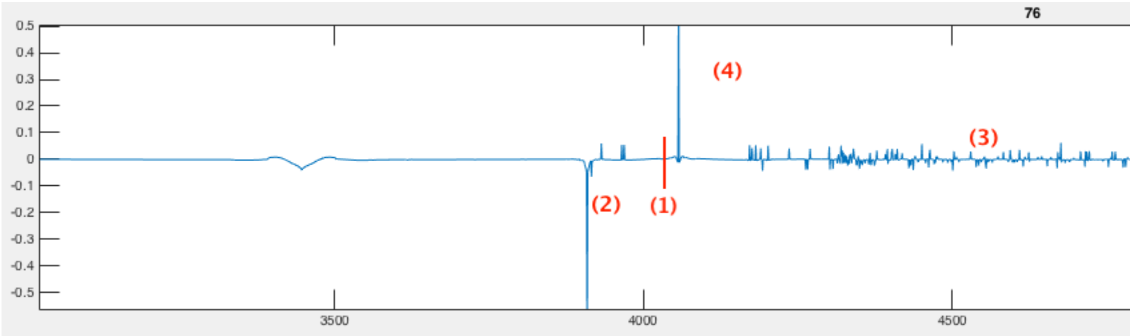


Figure 21: Initial RIR output with receiver 5cm above sound source from grid position 76

Figure 21 shows the first RIR from figure 20 which is used for analysis or early reflections. The RIR's can be validated by calculating where expected reflections should be using the following equation:

$$t_s = t_p - \frac{d}{c} \tag{1}$$

where:

t_s = Expected start of impulse
t_p = Time of first recorded reflection (first peak)
d = Distance between source and receiver
c = Speed of sound ($344m/s$)

Therefore, if this dip is assumed to be the direct sound which occurs at $t_p = 0.03448s$, where $d = 0.5$, then the start of the impulse can be calculated to be:

$$0.03448 - \frac{0.05}{344} = 0.03433s = t_s \tag{2}$$

This is indistinguishable from the direct sound peak, which is understandable given the source and receiver are 0.05m away. Now the expected start of this impulse is known, the expected first reflection due to the floor can be calculated:

$$\frac{2.05}{344} + t_s = 0.4029s \tag{3}$$

However, it can be seen that the first reflection occurs at 920, at a time of $0.039s$ suggesting that sound has travelled 1.6392m before reaching the receiver. However, the shortest available distance a sound could travel other than directly from the source to the receiver is 2.05m, which is the first floor reflection.

$$(0.039 - t_s) * c = 1.6382m \tag{4}$$

As the source and receiver are placed 1.85m away from the left side wall, a second reflection would be expected to occur at:

$$\frac{1.85 \times 2}{c} + t_s = 0.04509s \tag{5}$$

indicated by (3) in figure 21. However, it can be seen that no strong reflection occurs here.

After an email was sent to the technical department [21], it was suggested that the small dip may be a bug in the version of Odeon being used. It could therefore be assumed that the direct sound is actually captured at the peak shown at (2) in figure 21 at a time of 0.03908. From this the new expected time for the start of the impulse could be calculated:

$$0.03908 - \frac{d}{c} = 0.03893 = t_s \tag{6}$$

It could then be assumed that the first reflection caused by the floor is the peak at (4) in figure 21 which occurs at a time of 0.04058s. However, this would suggest that sound has travelled 0.05532m:

$$(0.04058 - t_s) \times c = 0.05532 \tag{7}$$

meaning that it has reflected off a surface approximately 0.28m away, of which there is no surface.

It is also apparent that if the first peak other than the dip at the beginning of the file is taken to be the direct sound, it can be seen across impulse responses that these would occur at different times depending on room position. This is also incorrect as it has been mentioned that the source and receiver distance is kept constant across RIR's. The first strong reflection appears to be delayed in time the further away from the left side wall the RIRs are taken which would be accurate, however the timings are incorrect.

3.2.4.2 Experiment

After some further investigation, it was found that the Odeon Manual [7] states in the section **Minimum distance from receiver to closest surface**:

> *"If a receiver is placed very close to a surface then results will be sensitive to the actual position of the secondary sources generated by ODEON's late ray method. If such a secondary source happens to be very close to the receiver, e.g. 1 to 10 centimetres, this may produce a spurious spike on the decay curve, resulting in unreliable predictions of the reverberation time."*

Though this is a problem said to be caused by secondary sources, it can be assumed that this issue could be caused to primary sources. This was investigated by producing RIRs with the source in a stationary location with receivers varying in distance above the source. Figure 22 shows six RIR's produced using different distances between the source and receiver where the numbers 1 - 6 indicate the following:

Graph Number	Distance from floor	
	Source	Receiver
(1)	1m	1.05m
(2)	1m	1.15m
(3)	1m	1.30m
(4)	1m	1.50m
(5)	1m	1.60m
(6)	1m	1.70m

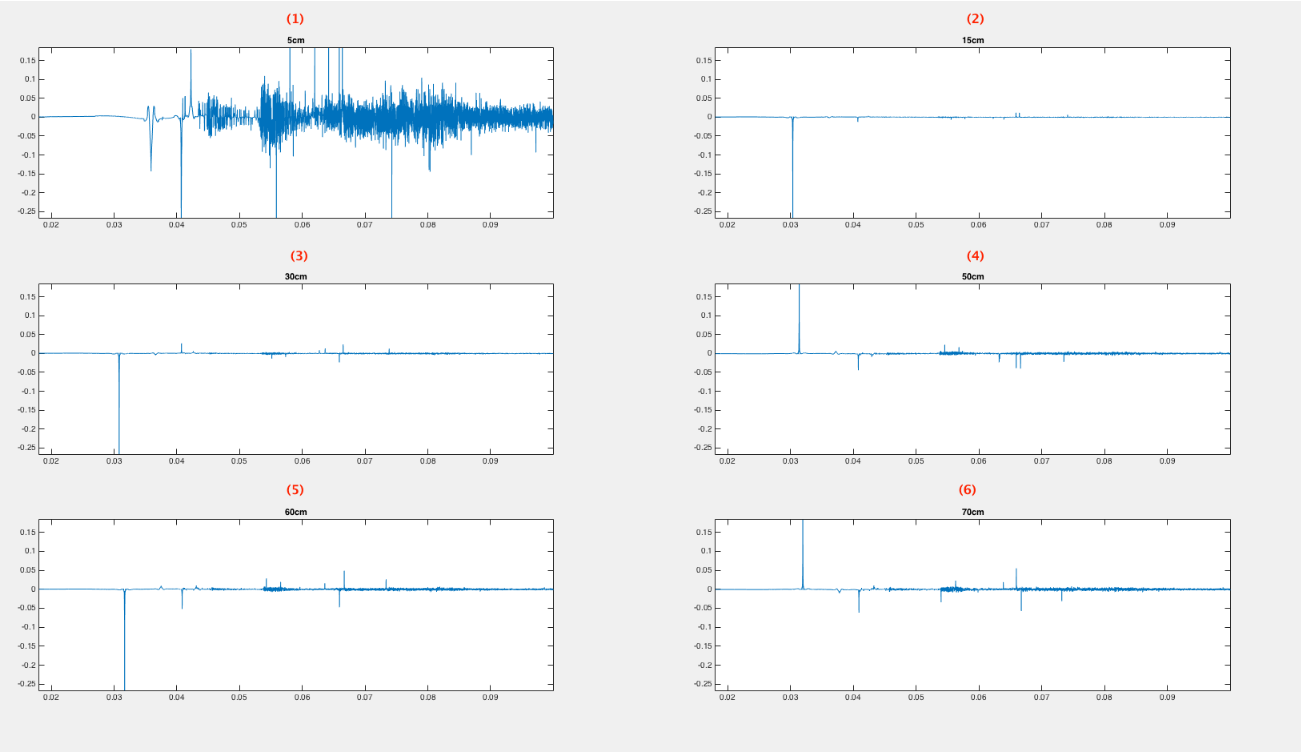


Figure 22: Plots of RIR's produced with the receiver placed at varying distances from the source, indicated by the title of each plot where **(1)**: 5cm **(2)**: 15cm **(3)**: 30cm **(4)**: 50cm **(5)**: 60cm **(6)**: 70cm

The following numbers (1) - (6) refer to the individual plots in figure 22.

A 5cm distance between the source and receiver shown in (1)[1] was originally used. This plot can be seen to be greatly different from the rest. It is in this RIR that the 'dip' found in the previously investigated RIR's is contained, whereas the RIR's with a greater distance between source and receiver (2) - (6) do not.

[1]***Note:*** *The reason the RIR shown in (1) looks different to previously investigated RIR's such as those in figure 20 is due to (1) being zoomed in, causing it to look greater in amplitude. This was done so it could be compared to (2) - (6) whilst being on the same*

3.2.4.3 Results

It is suggested in the Odeon manual that source and receiver be at least between 30cm and 50cm apart. Though a distance of 15cm (2) shows an RIR similar to the rest of the assumed correct ones, it is obvious that there is one strong direct sound, however it is difficult to see other reflections, whereas when the receiver is moved further away to a distance suggested by the Odeon manual, the early reflections become more obvious, with clarity of amplitude being more consistent through (4) - (6). The peaks in the RIR (5) (60cm) were investigated to assure they correctness, shown in figure 23. This RIR was chosen as the distance between the source and receiver for the real RIR's taken in Hendrix Hall was also 60cm. As the aim of achieving a more accurate human head topology is not possible due to this issue caused by Odeon, it was decided that for now, consistency is the next best thing.

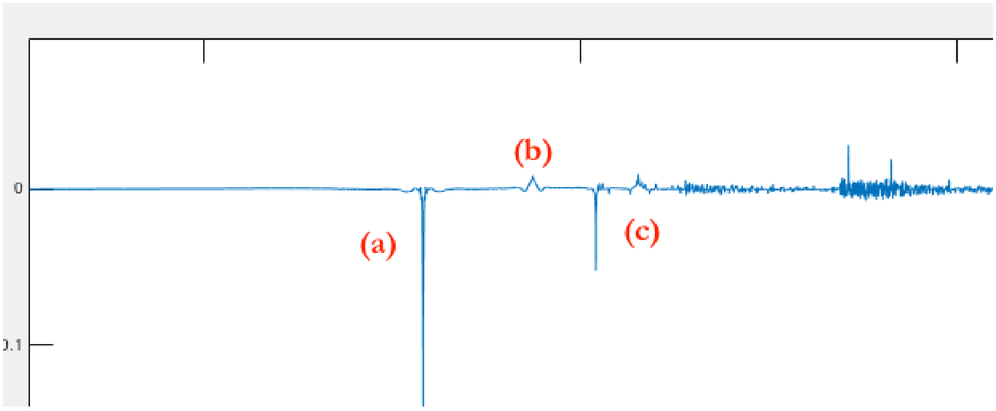


Figure 23: RIR produced when the receiver is places 60cm above the source.

The first peak (a) can be assumed to be the direct sound, present at $0.03168s$ meaning the start of the impulse can be calculated using equation 1, where now $d = 0.6$:

$$0.03168 - \frac{0.6}{c} = 0.02994s = t_s \tag{8}$$

From this, it can be calculated that the peak at (b), which occurs at $0.0375s$ had been captured after sound has travelled 2.6m:

$$(0.0375 - t_s) \times c = 2.6m \tag{9}$$

This is the reflection from the floor which is 1.6m below the receiver, causing the sound to travel 1m to the floor and then a further 1.6m to reach the receiver. This is shown in **Path 1** in figure 24. The peak at (c), occuring at $0.04084s$ can be calculated as having travelled 3.75m which is approximately the distance travelled by a reflection caused by the left side wall shown as **Path 2** in figure 24.

scale.

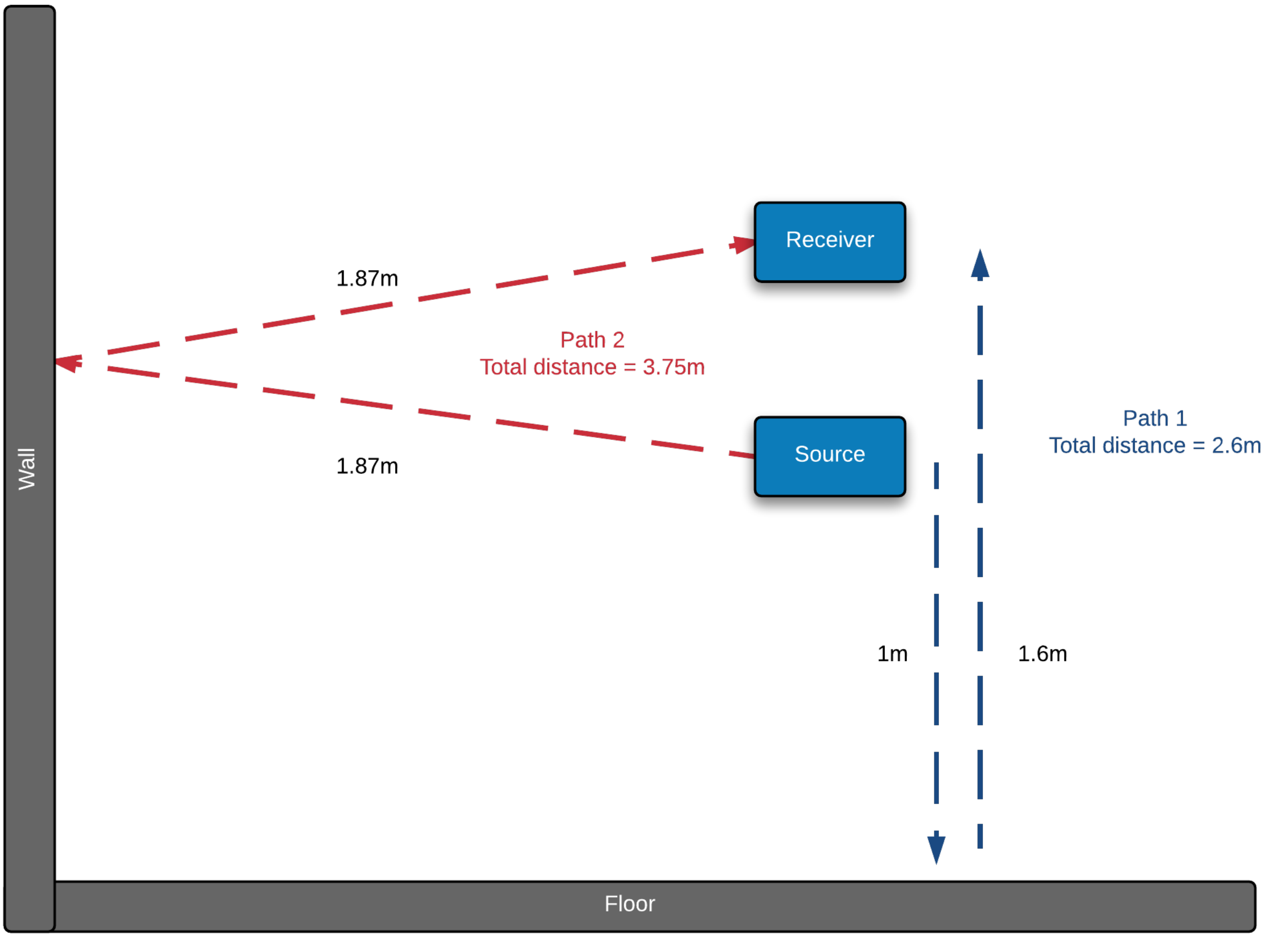


Figure 24: Illustration of the two early reflection paths due to the floor and left side wall when the receiver is placed 60cm above the source.

It can therefore be noted that the RIR's with a distance of 60cm between the source and receiver are accurate and can be used to produce a mass of RIR's.

3.2.5) RIR Locations

As discussed in section User Test Planning, RIR's were to be taken every 1m starting from the centre of the room and should form a grid (rectangular or square). However, there were some positions within the room that would cause some RIR positions to be invalid. These positions were location anywhere the sound source and receiver were placed closer than 0.75m next to a surface. This is because the loudspeakers in the VSS are placed 1.5m away from the centre, where the user will be stood meaning that any sound that has to travel less than 1.5m the reach the receiver will be impossible to reproduce at the correct time within the VSS. If a source and receiver were to be placed placed closer than 0.75m from a wall, sound would have to travel less than 1.5m before reaching the receiver. When taking this into account, figure 25 shows the positions that would be invalid due to them being placed too close to a surface, marked as U, V, W and Z. W represents a large space of invalid RIR positions and they would all be too close to the table surfaces.

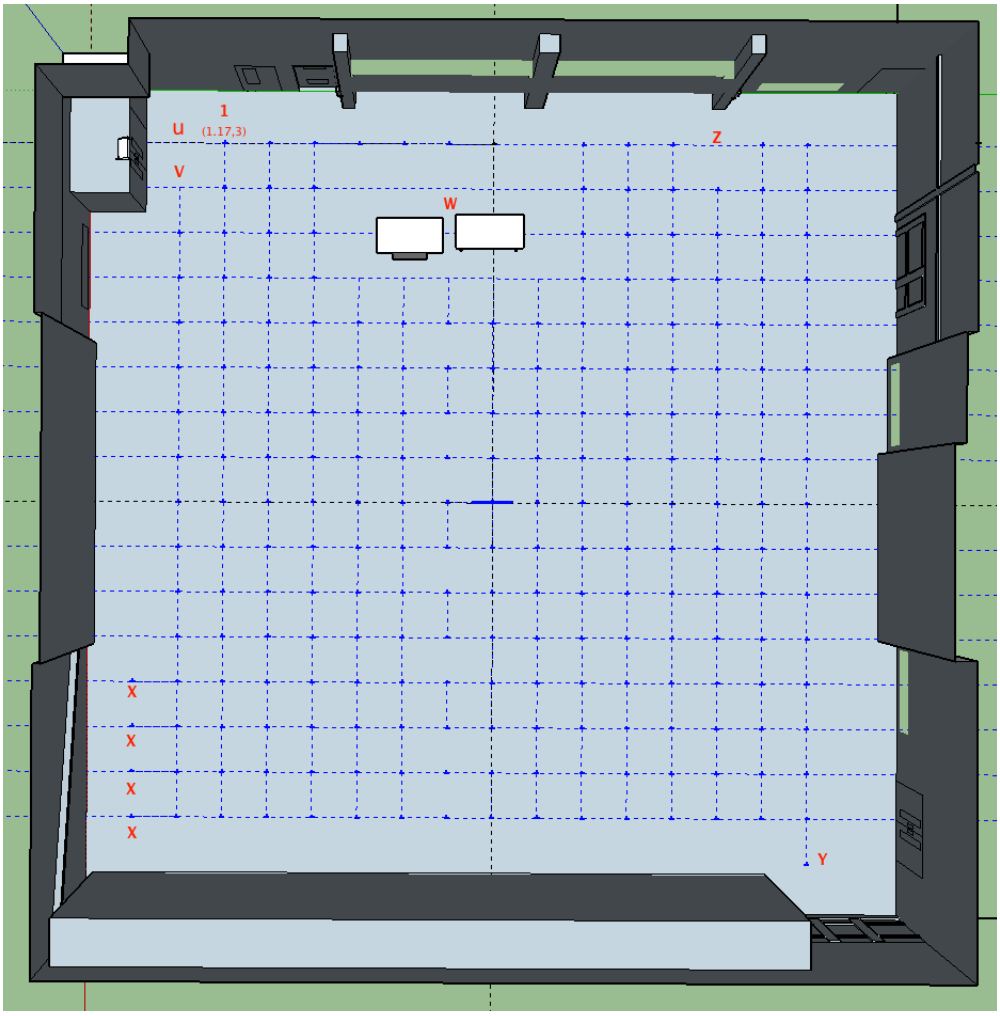


Figure 25: Top down view of the Hendrix Hall model indicating the RIR positions that are invalid due to being too close to a surface (U, V, W and Z), where other RIR's that are valid are not included as it would cause inconsistency (X and Y)

It was decided however that these RIR's would still be produced and used as part of the final system. If they were not included, there would be large gaps in the space where the user would have to jump between, thus be unable to produce what what feel like an consistent path.

It was also later discovered (see section **??**) that the latency due to the software used would mean that the beginning of the RIR's would have to be trimmed anyway, avoiding this issue.

There are several RIR's in figure 25 labeled X and Y. These positions would still produce valid RIR's, however they were not included in the final grid as they would cause inconsistencies in the positions available to the user. All other blue crosses in figure 25 were also included in the final RIR grid, producing a grid 15x16, totalling 240 available positions. Figure 26 shows the final maximum avilable RIR's locations.

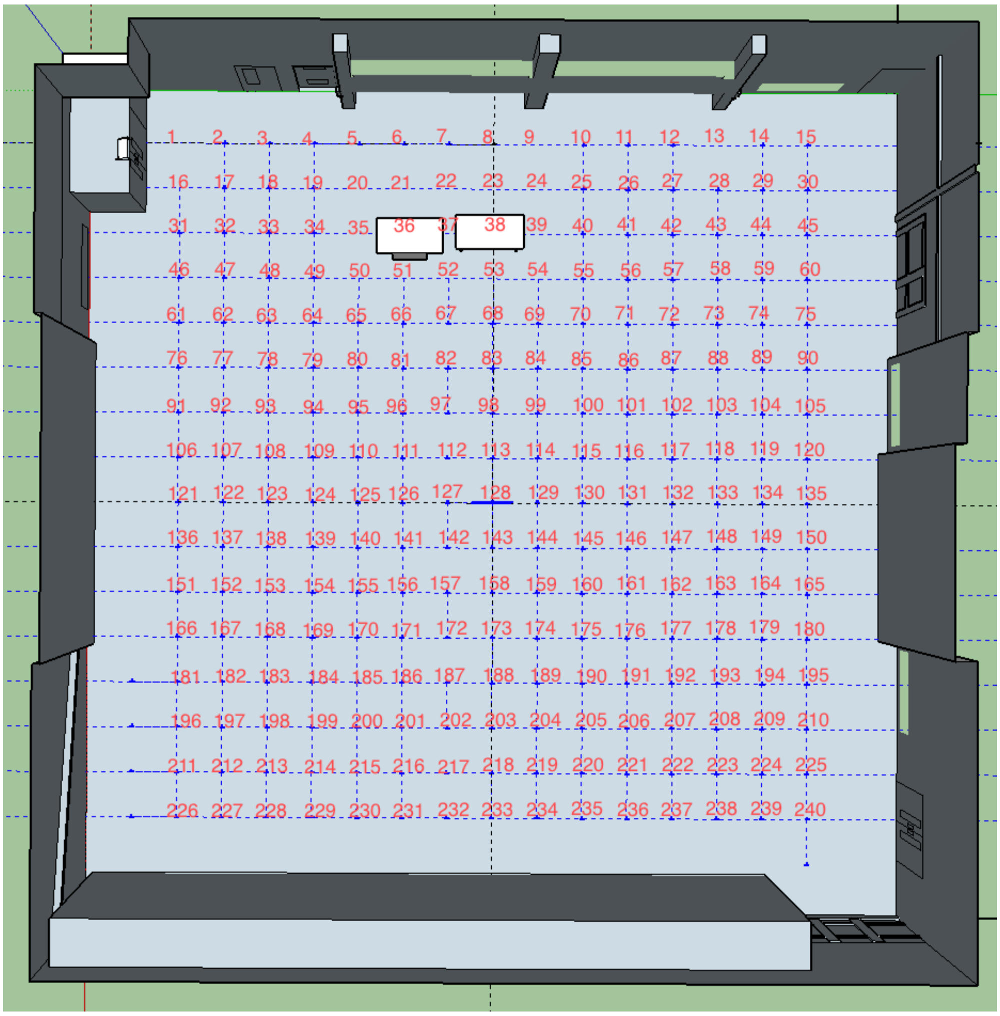


Figure 26: Top down view of the Hendrix Hall model showing the maximum available RIR locations

The tables present within the room so not cause an issue

3.2.6) RIR Rendering

Once all of the source and receiver positions had been decided, they must be included in Odeons

The source and receiver positions can be set using a .SouRecScript file, which contains information regarding the Cartesian coordinates of both sources and receivers within the room as well as the sound source directivity patter and the direction in which the source is facing. This made producing 240 RIR's almost effortless. However, one the script had set to locations of the source and receivers, each source and receiver had to be paired up individually. This is because Odeon provides a **Job List**, where any number of RIR's can be rendered one after. For this to work, the user must tell Odeon which receiver to use for each sound source. Figure 27 shows the job list menu where each source has to be connected to each receiver, using a series od drop down menus and scroll bars. This part of Odeon is clumsy and in no way designed to make the process fast and simple. This had to be done for all 240 RIR's and then again for the other three sets of RIR's used for facing different direction within the room, totalling 960 RIR's.

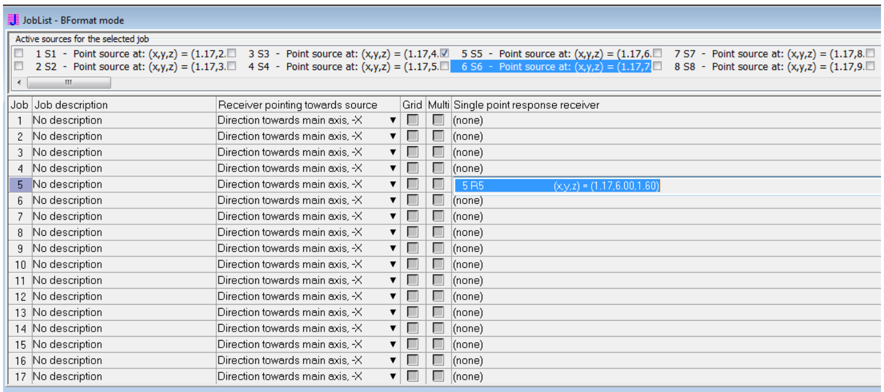


Figure 27: Source and receiver pairing window in Odeon

A naming system was used for the RIR files to indicate which grid positions they were taken from and which direction they are facing. For example: '008_0.wav' is is an RIR from grid position 8 with 0° rotation (facing the blackboards) where '128_90' is taken from grid position 128 and has a 90° rotation (facing the left side wall).

Files available: SupportingFiles/sourecscripts/

3.3 - Software

As a software patch was already in use with the VSS, the idea was to extend this software patch to accommodate the newly proposed functionality.

This section will first give a quick overview of the original max patch and then an overview of the newly produced software with a simple explanation of how it works, followed by a more detailed explanation of how the two main parts of the software work.

3.3.1) Software Overview

3.3.1.1 The Original Patch

The original max patch was used to convolve a real time audio signal with a set of four RIR's simultaneously, allowing the user to turn their head in the VAE through the use of an Oculus Rift as a head tracking device. Four positions within the VAE were available. To select one, the user (or an operator) selected an 'open' button which prompted a file navigation window. The RIR files then had to be found (in the correct order) and opened one at a time, with a new file window opening after each file had been selected. Figure 28 shows this.

This was the primary aspect of the original patch that needed extending to make the process automatic.

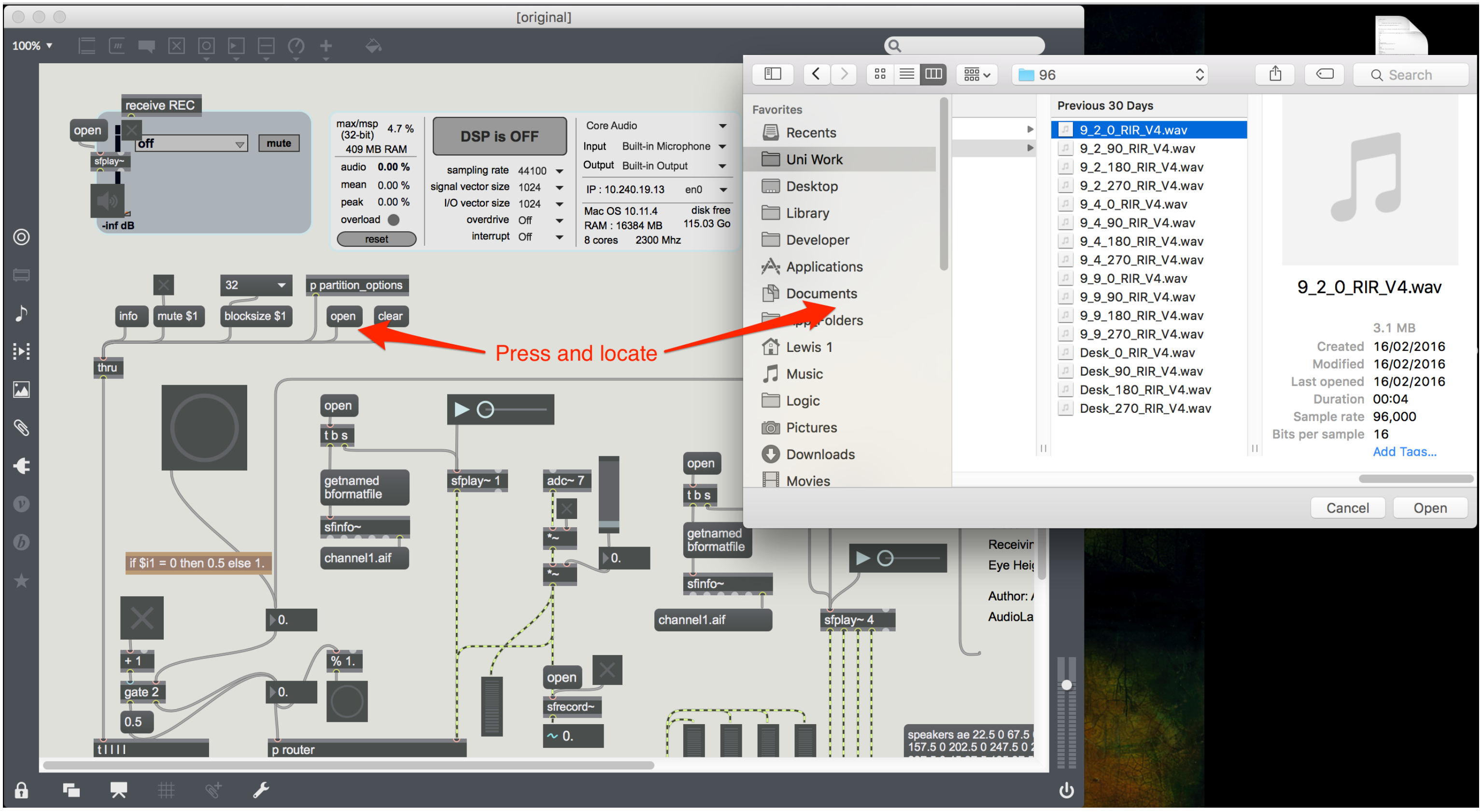


Figure 28: Original Max patch used in the VSS showing the file location window that pops up 4 times.

3.3.1.2 Extended Patch Overview

Figure 29 shows an annotated top level view of the Max patch produced to take a user input, load the appropriate RIR files and convolve with a real time audio input. The annotated sections can be described as follows:

1: Two buttons used to reset the system and start the timer used when moving around the space. The settings patch extends to provide a range of options including the density of

2: Here, an audio file can be loaded into the system and used instead of a real time audio input. This is used in user test #3.

3: A timer used to load new RIR files when appropriate.

4.1: Patches that send a user interface to an iPad which allows a user to select a location within the VAE. User interaction is monitored (in the form of screen coordinates) and sent to **4.2**.

4.2: Takes the coordinates of the user input and calculates which (if any) RIR file should be loaded into the system, in section **4.3**.

4.3: Three patches (extended versions of the max explained in section The Original Patch) used to simultaneously convolve an audio signal (real time or audio file) with four directional RIR files. While one loads the next necessary RIR file the other two are used to simulate the movement of the user by panning the real time audio between the two currently running convolutions. This is how the user is moved along a path, explained in section Iteration 3.

5: An extended version of the real time head-tracking system used in the original patch, used to pan between the four directional RIR's to simulate head movement in the VAE.

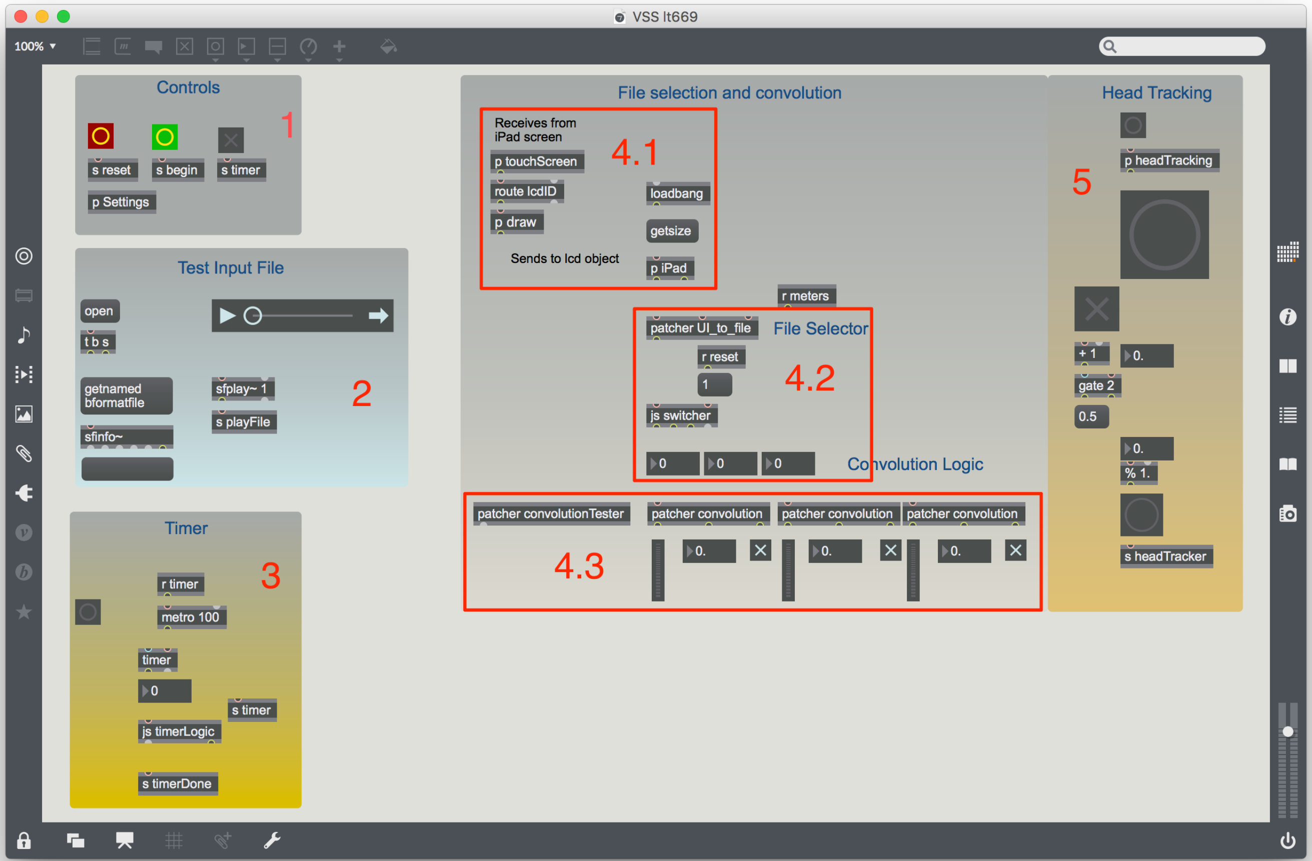


Figure 29: Top level Max patch

The software implementation was split into two main sections: **Location Selection** consisting of section **4.2** and **Mobility** mainly consisting of section **3 and 4.3**.

3.3.2) Location Selection

In order to allow the user to move themselves around the room, they had to be presented with a means of doing so. This involved presenting the user with an interface that resembled the space available to them on which they could select their location. This then had to output the coordinates selected by the user and convert them into a format which can be interpreted as a file name indicating which RIR in the available grid to use. This could then be passed to the rest of the system to load the appropriate files. A simple block diagram of the user interface part of the system is shown in figure 30.

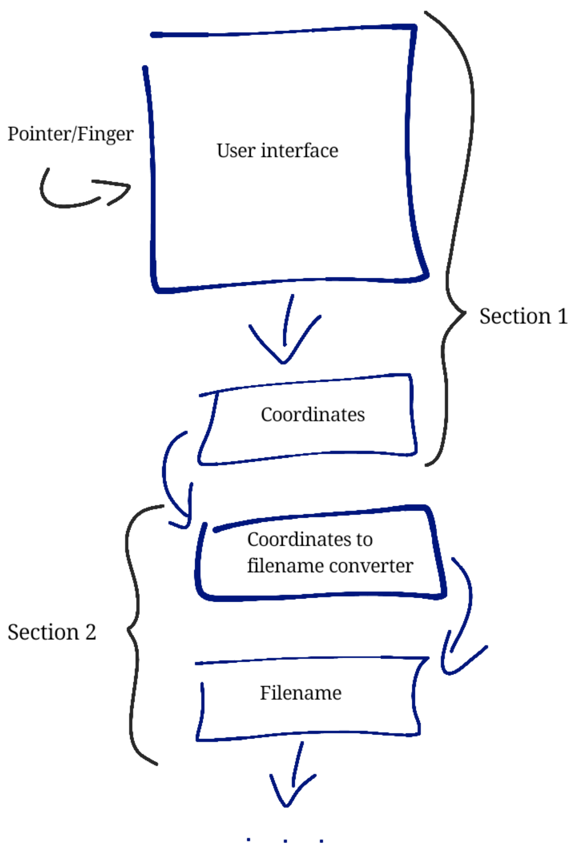


Figure 30: Flow diagram of the location selection software design. **Section 1** indicates the user interface section where coordinates are recorded and **Section 2** takes these coordinates and finds the appropriate RIR file.

In Max, the 'lcd' object is used for this function. This object presents a quadrilateral of variable length and height with the ability to output its dimensional information by sending a 'getSize' message to its input, as well as output the coordinates of a mouse click/drag. Figure 31 shows the lcd object with its inputs and outputs represented by section 1 in figure 30.

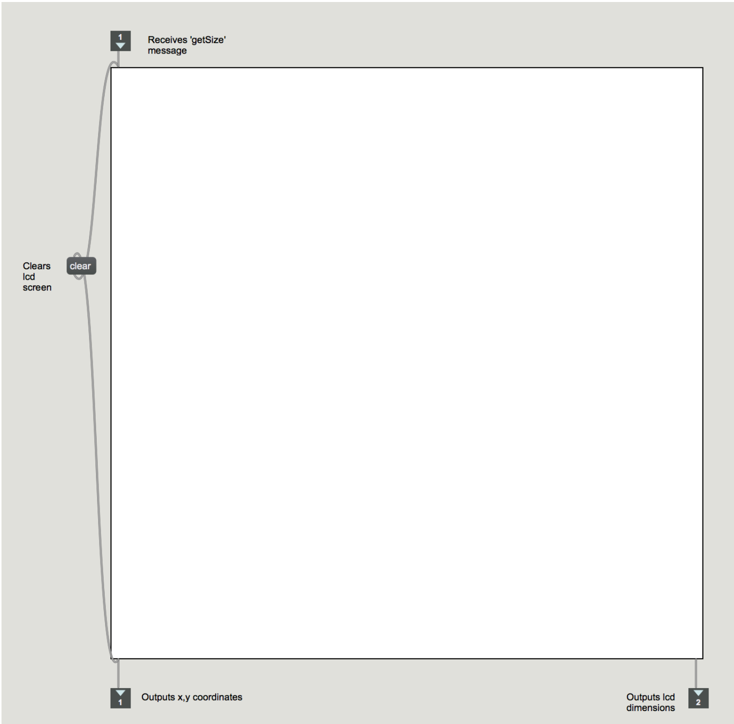


Figure 31: Flow diagram of the location selection software design

The outputs from the lcd object are sent to the patch 'UI_to_file' which contains a JavaScript file called 'loadFilesLogic'. This JavaScript file converts the (x,y) coordinates into an appropriate filename, by taking into account the size of the lcd screen and how many RIRs there are per meter.

The javascript takes 5 inputs: UI (x,y) coordinates , (x,y) lcd dimensions and a number representing how many sections the lcd grid should be split into. This last value is calculated by sending a number from 1-5 (distance per RIR) to the third inlet of the UI_to_file patch (on the far right). As the maximum number of RIRs that will be available per length of the room is 15, the input is divided by 15 and rounded to the nearest value, giving the number of sections the lcd screen should be split into along the x-axis (the number of y-axis sections is calculated later). This information can then be used to determine in which *'section'* the user is currently located based on their coordinates, allowing it to load the appropriate RIRs that are available in that section. Figure 32 shows the section of 'UI_to_file' highlighting each section.

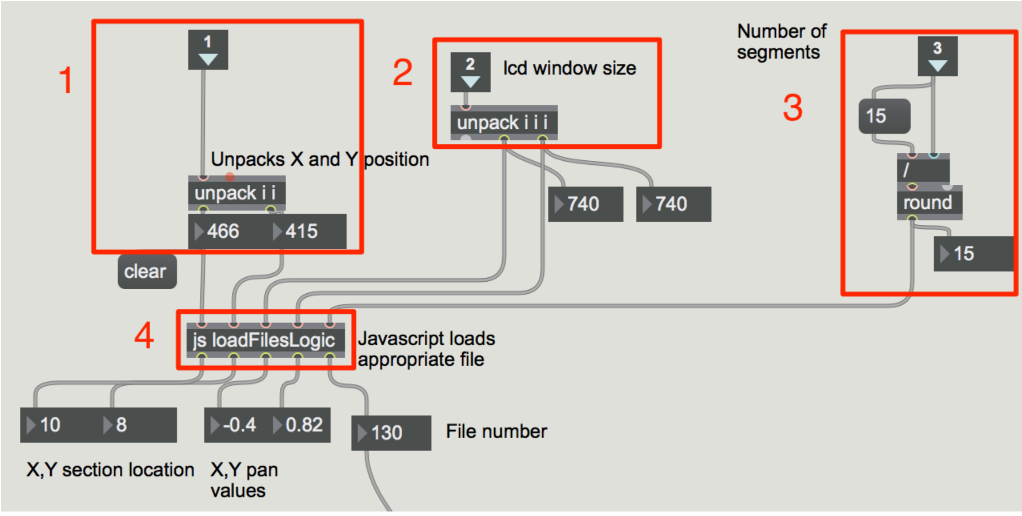


Figure 32: Screen shot from Max showing a section that converts the user interface coordinates into the appropriate file name. 1) Coordinates from lcd object 2) Dimensions of lcd object 3) Number of segments the lcd object should be split into due to the number of RIR's per meter 4) JavaScript file take produces an appropriate file name given the input data.

3.3.2.1 File name JavaScript: 'loadFilesLogic'

```
function msg_int(input){
 if(inlet == 0){
  xPos = input;
 } else if (inlet == 1){
  yPos = input;//Add off set to start at (0,1)
 } else if (inlet == 2){
  windowSize [0] = input;
 } else if (inlet == 3){
  windowSize [1] = input;
 } else if(inlet==4){
  numberOfMeters = input;
 }

```

The first section in the JavaScript files simply stores the data from different inputs to different variables that are used throughout the rest of the code.

The code in figure 33 calculates which section the user is closest to. The numberOfMeters variable is the 5th input to the js object which can be used to determine how many rows and columns of RIRs there are going to be. When the RIRs are separated by 3m or 5m, there are the same number of rows as there is columns (3m: 5x5) (5m: 3x3), whereas (1m: 15x16) the other distances produce grids of RIRs with one more row than there are columns. The if statement from 31 - 39 ensures that the lcd screen is split into the correct amount of sections in order to correctly load the appropriate RIR.

MMMMMMMBITCH

```
    //Round to nearest value
    xSection = Math.round(xPosition);
    ySection = Math.round(yPosition);

    //Start the lcd grid sections from column 1 row 1 instead of column 0 row 0
    if(xSection == 0){
     xSection = 1;
    }
    if(ySection == 0){
     ySection = 1;
    }

```

3.3.3) Mobility Implementation

3.3.3.1 Iteration 1

3.3.3.2 Iteration 2

3.3.3.3 Iteration 3

3.4 - Head-Tracking

YEI Sensor was not good.

3.5 - Latency Test

3.6 - RIR Trimming

3.7 - Real RIR Recordings

3.7.1) RIR Measurement Setup

For the VSS it is desirable to obtain RIR's that can be used to represent the topology of a singer, i.e mouth (sound source) bellow the ears (receiver). To achieve this, a Genelec 8040B [22] loudspeaker was used as a directional sound source and a Soundfield St450 MKII microphone [23] was used as the receiver to record the three dimensional sound field in Ambisonic B-format.

Figure 34 shows an image of the human head topology sound source and receiver set up. The Genelec is placed 1m above the ground and the Soundfield microphone places 0.6m above the sound source. Ideally the receiver would be placed closer to the sound source to more accurately represent the distance between the ears and the mouth, however due to the physical dimensions of the equipment being used this was not possible. The sound source was placed 1m off the ground simply due to the limitations set by the maximum height of the microphone stand.

Figure 35 shows the overall set up used for recording the RIR's as follows: **(1)** the Genelec and Soundfield microphone in the above mentioned source and receiver set up with a special soundfield cable running into **(2)**, the soundfield unit used to output a 4 channel B-format signal through 4 XLR to jack cables to **(3)** a Fireface UXC audio interface plugged into **(4)** a Mac running the digital audio workstation Reaper. Reaper was used to record directly to 4 channel tracks where channels 1 - 4 recorded the W, X, Y, Z channels respectively and to simultaneously output a 15 second long sinusoidal sweep to the Genelec. This method avoided synchronisation issues often faced when using separate devices to output a signal and to record a signal.

3.7.2) Positions

Four positions within the room were chosen and marked with tape for the RIR positions shown in figure 36 where:

Position	(1)	(2)	(3)	(4)
Coordinates [x(m), y(m)]	[9,9]	[4.5,9]	[2,9]	[10.13,1.46]

For each position an RIR was taken in four different directions starting at 0° and rotation anti-clockwise 90° (anti-clockwise rotation is a standard in Ambisonics) by rotation the sound source and keeping the receiver facing the same direction. This ensures that when the user turns their head in the VSS the sound field does not rotate as they do. Once the four directional RIR were recorded, the source and receiver were moved to the next marked location with care being taken to make sure the receiver is placed the same distance away from the sound source each time.

3.7.3) Measurements

The sound source signal used was a 15 second long exponentially swept sinusoid ranging from 20Hz to 20kHz with an 8 second padding to ensure that the room could be vacated before the sweep began. The sweep was produced using the Matlab function `generatesweep.m` taken from the departmental website [24] which produced both the sinusoidal sweep and an inverse sinusoidal sweep used for deconvolution at a later stage.

3.7.4) RIR Analysis

Once the measurements had been recorded, the first 8 seconds of the files were muted in order to remove noise such as door slams as the room was vacated. Then the signals were deconvolved with an inverse of the sinusoidal sweep to time align the frequency dependant room reflections. This was done using the deconvolution Matlab function `deconvolve.m` again taken from the departmental website [24].

3.7.5) Issues

Several issues arose when taking the initial RIR's. Simply connection the outputs from the Soundfield converter to the incorrect inputs on the Fireface audio interface meant that the initial recordings contained tracks that were not necessarily recorded in the correct order, i.e the tracks were not recorded as [W, X, Y, Z] but could have been recorded in a possible 24 combinations. This meant that when used to convolve with an audio source, the localisation of the sound source would be incorrect.

Through observation it was possible to narrow down which order the channels might have been recorded in, however as it could not be 100% certain the measurements were taken again.

3.8 - RIR Finalisation

Once all of the RIR's required for the final system and testing and been rendered/recorded, some final touches had to be made before they could be used.

User Testing

4.1 - User Testing Aims

4.2 - Testing Procedure

4.2.1)

Appendices

Appendix A

Appendix B

```
inlets = 5;
outlets = 5;

//Create arrays to store previous positions
var xArray = new Array(2);
var yArray = new Array(2);

var windowSize = new Array(2);

//Variables to use for file searching
var fileX, fileY, search;

```

```
//Defines how to split up the grid
var numberOfMeters;

//Loads appropriate files given users finger coordinates
function msg_int(input){
 if(inlet == 0){
  xPos = input;
 } else if (inlet == 1){
  yPos = input;//Add off set to start at (0,1)
 } else if (inlet == 2){
  windowSize [0] = input;
 } else if (inlet == 3){
  windowSize [1] = input;
 } else if(inlet==4){
  numberOfMeters = input;
 }

 //Split into sections
 if(numberOfMeters == 3 || numberOfMeters == 5){
  //Even grid for 3m and 5m
  xPosition = (xPos/windowSize[0])*(numberOfMeters);
  yPosition = (yPos/windowSize[1])*(numberOfMeters);
 } else if (numberOfMeters == 4 || numberOfMeters == 8){
  //4m separation requires different x,y coordinate scaling
  xPosition = (xPos/windowSize[0])*(numberOfMeters-1);
  yPosition = (yPos/windowSize[1])*(numberOfMeters);
 } else{
  //Extra row for others
  xPosition = (xPos/windowSize[0])*(numberOfMeters);
  yPosition = (yPos/windowSize[1])*(numberOfMeters+1);
 }

 //Round to nearest value
 xSection = Math.round(xPosition);
 ySection = Math.round(yPosition);

 //Start the lcd grid sections from column 1 row 1 instead of column 0 row 0
 if(xSection == 0){
  xSection = 1;
 }
 if(ySection == 0){
  ySection = 1;
 }

 //Distance in % away from center of section
 xBetween = 2*(xPosition - xSection); //x2 to get 100%
 yBetween = 2*(yPosition - ySection);

 //Which RIR to load in centre location
 outlet(0,xSection);
 outlet(1,ySection);

```

```
//Output panning values
outlet(2,xBetween);
outlet(3,yBetween);

//Store current location
xArray[0] = xSection;
yArray[0] = ySection;

//If either coordinate is changed search for new files
if(xArray[0] != xArray[1] || yArray[0] != yArray[1]){

 if(xArray[0] != xArray[1]){
  //Store previous value
  xArray[1] = xArray[0];
  X = xArray[0];
 }

 if(yArray[0] != yArray[1]){
  yArray[1] = yArray[0];
  Y = yArray[0]
 }

 //Output user location within grid
 if(numberOfMeters == 4 || numberOfMeters == 8){
  fileNumber = X + ((numberOfMeters-1)*(Y-1)); //Requires different algorithm for 4m
    due to different grid shape
 } else {
  fileNumber = X + ((numberOfMeters)*(Y-1));
 }
 outlet(4,fileNumber);
}
}
```

Sections/Appendix/AppendixA/Code/loadFilesLogic.js

REFERENCES

[1] J. Huopaniemi, L. Savioja, T. Lokki, and R. Väänänen, "Virtual acoustics - applications and technology trends," *Proc. European Signal Proc. Conf. (Eusipco'2000)*, pp. 2201–2208, 2000.

[2] G. Stan, J. Embrechts, and D. Archambeau, "Comparison of different impulse response measurement techniques," *Journal of the Audio Engineering Society*, vol. 50, no. 4, pp. 249–262, 2002, ISSN: 1549-4950. [Online]. Available: http://orbi.ulg.ac.be/handle/2268/34825 (visited on Apr. 25, 2016).

[3] Iso 3382-1, "Acoustics - measurement of room acoustic parameters - part 1: performance spaces," vol. 3, 2009.

[4] (2016), Odeon, [Online]. Available: http://www.odeon.dk/pdf/ODEONManual.pdf (visited on Apr. 25, 2016).

[5] S. Siltanen, T. Lokki, and L. Savioja, "Rays or waves? understanding the strengths and weaknesses of computational room acoustics modeling techniques," *Proceedings of the International Symposium on Room Acoustics, ISRA 2010*, no. August, pp. 1–6, 2010. [Online]. Available: http://www.acoustics.asn.au/conference%7B%5C_%7Dproceedings/ICA2010/cdrom-ISRA2010/Papers/O5a.pdf (visited on Apr. 25, 2016).

[6] J. H. Rindel, "Computer simulation techniques for acoustical design of rooms," *Acoustics Australia*, vol. 23, pp. 81–86, 1995. [Online]. Available: http://www.odeon.dk/pdf/AustralAc_1995_Rindel.pdf (visited on Apr. 25, 2016).

[7] C. L. Christensen and G. Koutsouris, *Odeon user's manual*, Scion DTU Diplomvej Kgs. Lyngby Denmark, 2015. [Online]. Available: http://www.odeon.dk/pdf/ODEONManual12.pdf (visited on Apr. 25, 2016).

[8] Google. (2015). Google sketchup, [Online]. Available: http://www.sketchup.com/ (visited on Apr. 25, 2016).

[9] (2016). Su2odeon, Odeon, [Online]. Available: http://www.odeon.dk/su2odeon-plugin-trimble-sketchup (visited on Apr. 25, 2016).

[10] (2016). Max/msp, Cycling74, [Online]. Available: https://cycling74.com/products/max/ (visited on Apr. 25, 2016).

[11] (2016). Spat, IRCAM Centre Pompidou, [Online]. Available: http://forumnet.ircam.fr/product/spat-en/ (visited on Apr. 25, 2016).

[12] J. S. Brereton and B. A. Hons, "Singing in space (s): singing performance in real and virtual acoustic environments — singers ' evaluation , performance analysis and listeners ' perception .," no. August, 2014.

[13] (2016), Oculus, [Online]. Available: https://www.oculus.com/en-us/ (visited on Apr. 25, 2016).

[14] L. Savioja, J. Huopaniemi, T. Lokki, and R. Väänänen, “Creating interactive virtual acoustic environments,” *J. Audio Eng. Soc*, vol. 47, no. 9, pp. 675–705, 1999. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=12095 (visited on Apr. 25, 2016).

[15] *Dewalt laser distance measurer manual*, DeWalt, 2016. [Online]. Available: http://service.dewalt.co.uk/PDMSDocuments/EU/Docs//docpdf/dw03101-typ1_en.pdf (visited on Apr. 25, 2016).

[16] (2016). Absorption coefficients, AcousticTraffic, [Online]. Available: http://www.acoustic.ua/st/web_absorption_data_eng.pdf (visited on Apr. 25, 2016).

[17] (2016), BemBook, [Online]. Available: http://www.bembook.ibpsa.us/index.php?title=Absorption_Coefficient&action=edit (visited on Apr. 25, 2016).

[18] P. Kovalchik, R. Matetic, S. Peng, and G. Cole, “Measurement method for determining absorption coefficients for underground mines,” pp. 1–4, [Online]. Available: http://www.cdc.gov/niosh/mining/userfiles/works/pdfs/ammfd.pdf (visited on Apr. 25, 2016).

[19] B. B. Monson, E. J. Hunter, and B. H. Story, “Horizontal directivity of low- and high-frequency energy in speech and singing.,” *The Journal of the Acoustical Society of America*, vol. 132, no. 1, pp. 433–41, 2012, ISSN: 1520-8524. DOI: 10.1121/1.4725963.

[20] C. Armstrong, “Noisy fire hazard,” (Unpublished Masters Dissertation, University of York, May, 2016).

[21] L. Thresh, *Impulse response analysis question*, Personal Email, Odeon Support, Mar. 2016.

[22] *Genelec 8040b active monitoring system*, Genelec, 2016. [Online]. Available: http://www.genelec.com/sites/default/files/media/Studio%20monitors/8000%20Series%20Studio%20Monitors/8040B/genelec_8040b_datasheet.pdf (visited on Apr. 25, 2016).

[23] (2016). St450 mkii soundfield portable, TSL Products, [Online]. Available: http://www.tslproducts.com/soundfield/st450-soundfield-portable/#spec-tab (visited on Apr. 25, 2016).

[24] (2016). Sine sweep matlab script, University of York, [Online]. Available: http://www.elec.york.ac.uk/internal_web/gsp/amt/modules/Audio_Acoustics_and_Applications/ (visited on Apr. 25, 2016).

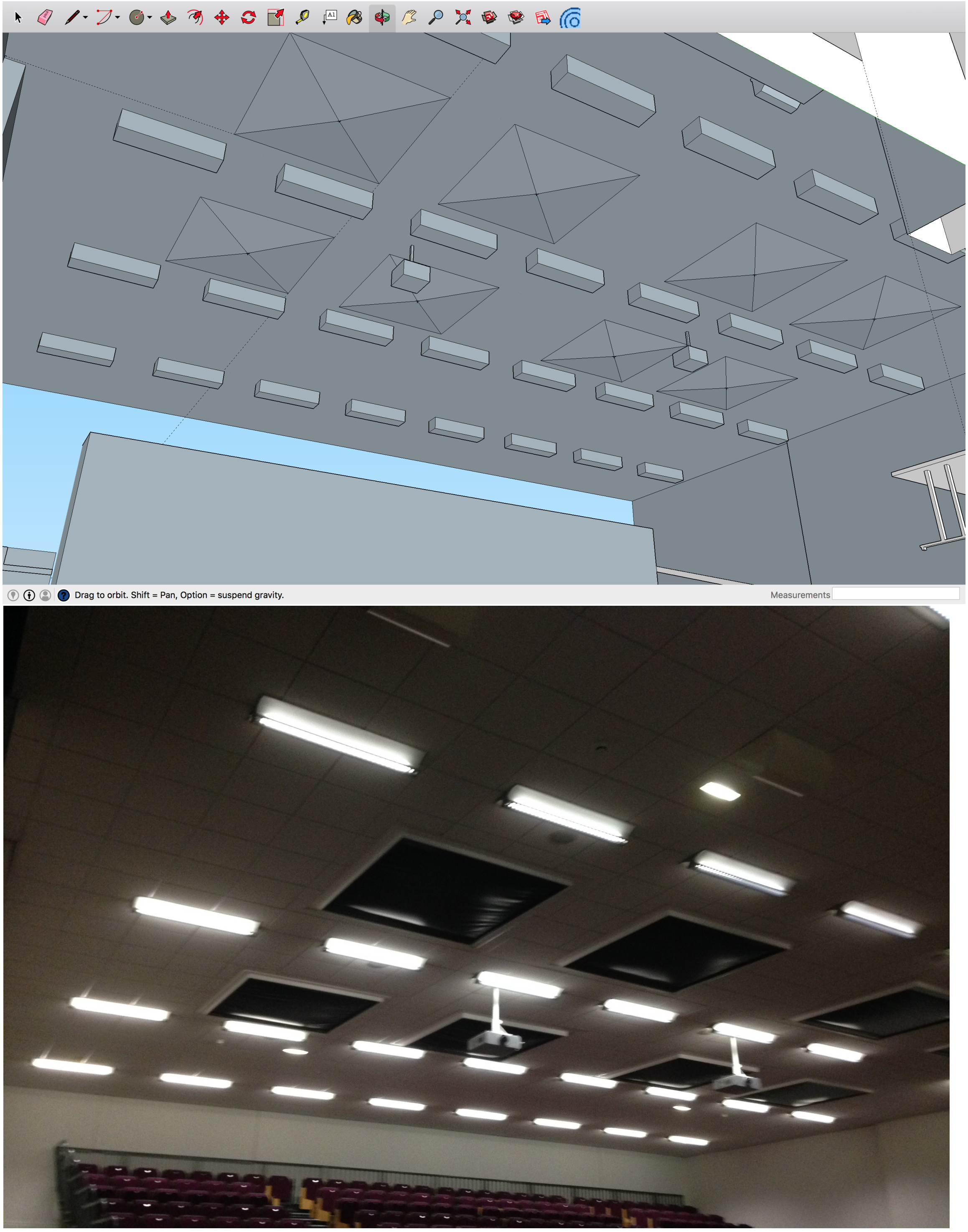


Figure 6: Comparison of the final simplified model and a picture of the real Hendrix Hall

```
//Split into sections
if(numberOfMeters == 3 || numberOfMeters == 5){
 //Even grid for 3m and 5m
 xPosition = (xPos/windowSize[0])*(numberOfMeters);
 yPosition = (yPos/windowSize[1])*(numberOfMeters);
} else if (numberOfMeters == 4 || numberOfMeters == 8){
 //4m separation requires different x,y coordinate scaling
 xPosition = (xPos/windowSize[0])*(numberOfMeters-1);
 yPosition = (yPos/windowSize[1])*(numberOfMeters);
} else{
 //Extra row for others
 xPosition = (xPos/windowSize[0])*(numberOfMeters);
 yPosition = (yPos/windowSize[1])*(numberOfMeters+1);
}
```

Figure 33: Code

Figure 34: Human head topology RIR measurement set up with a Genelec 8040B sound source placed 1m off the floor 0.6m below a Soundfield microphone used as a receiver

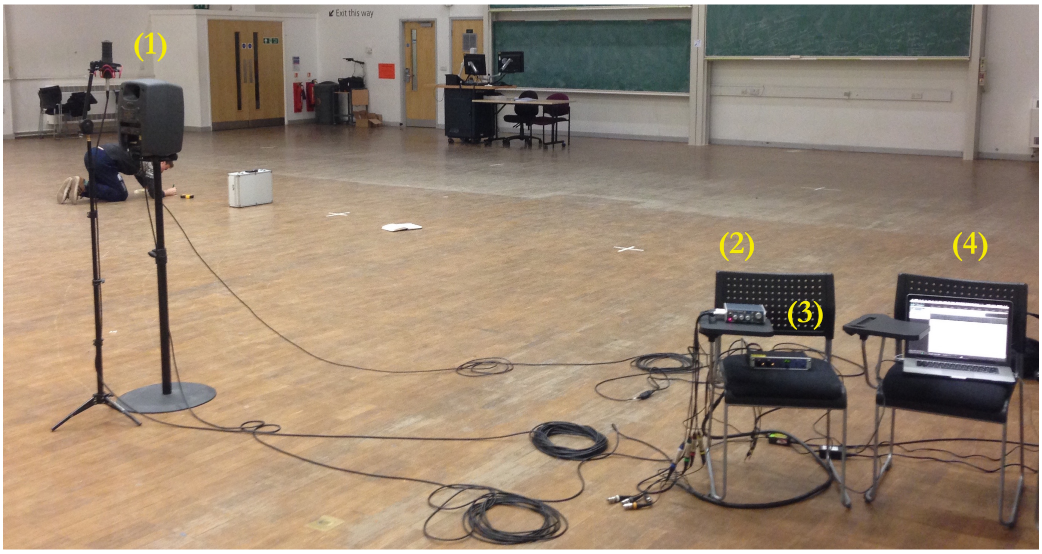


Figure 35: Real RIR measurement set up. **(1)**: Sound source and receiver set up **(2)**: Soundfield interface **(3)**: Fireface UXC audio interface **(4)**: Mac running Reaper to output sinusoidal sweep and record B-format input

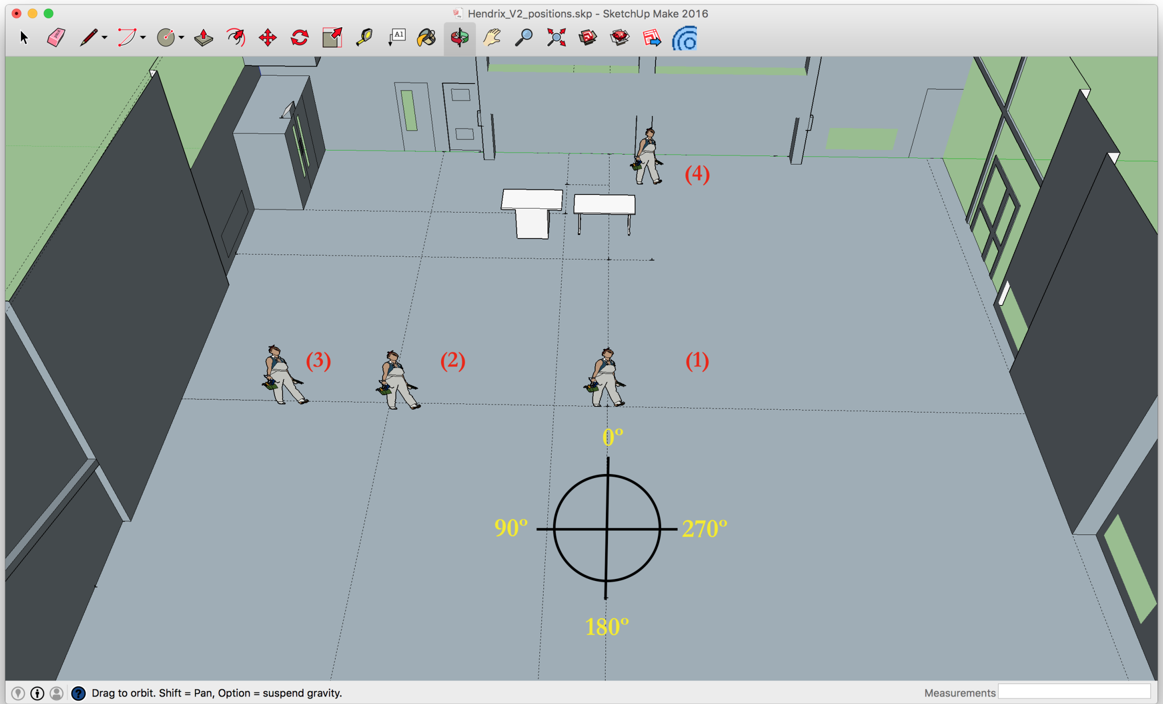


Figure 36: Google SketchUp model showing the positions of where the real RIR's were taken

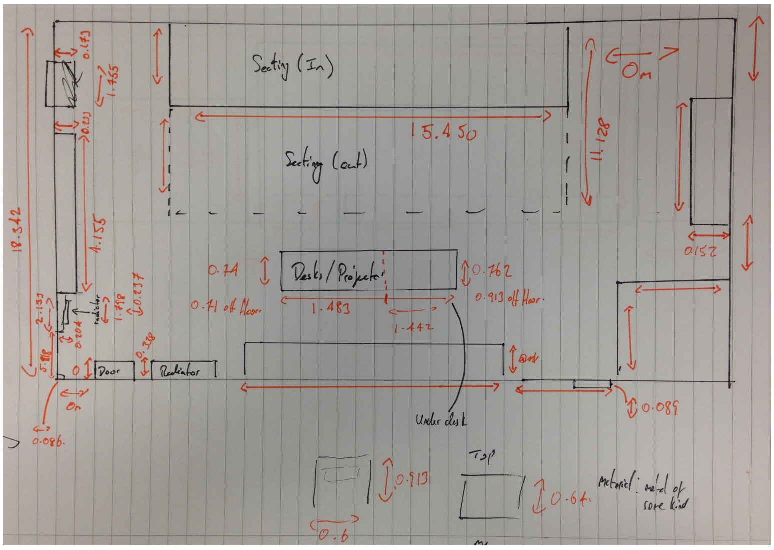


Figure 37: Annotated blueprint of Hendrix Hall from a birds-eye view

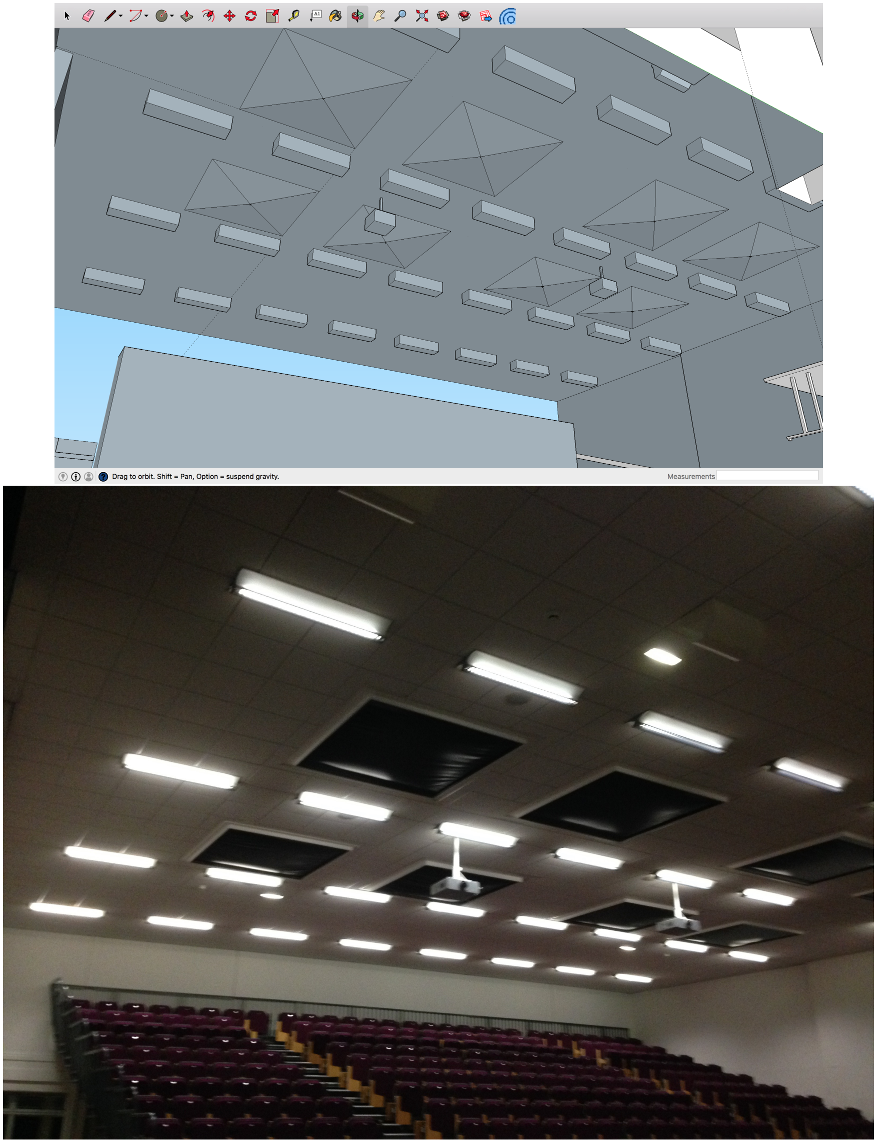


Figure 38: Real Vs SKU Roof

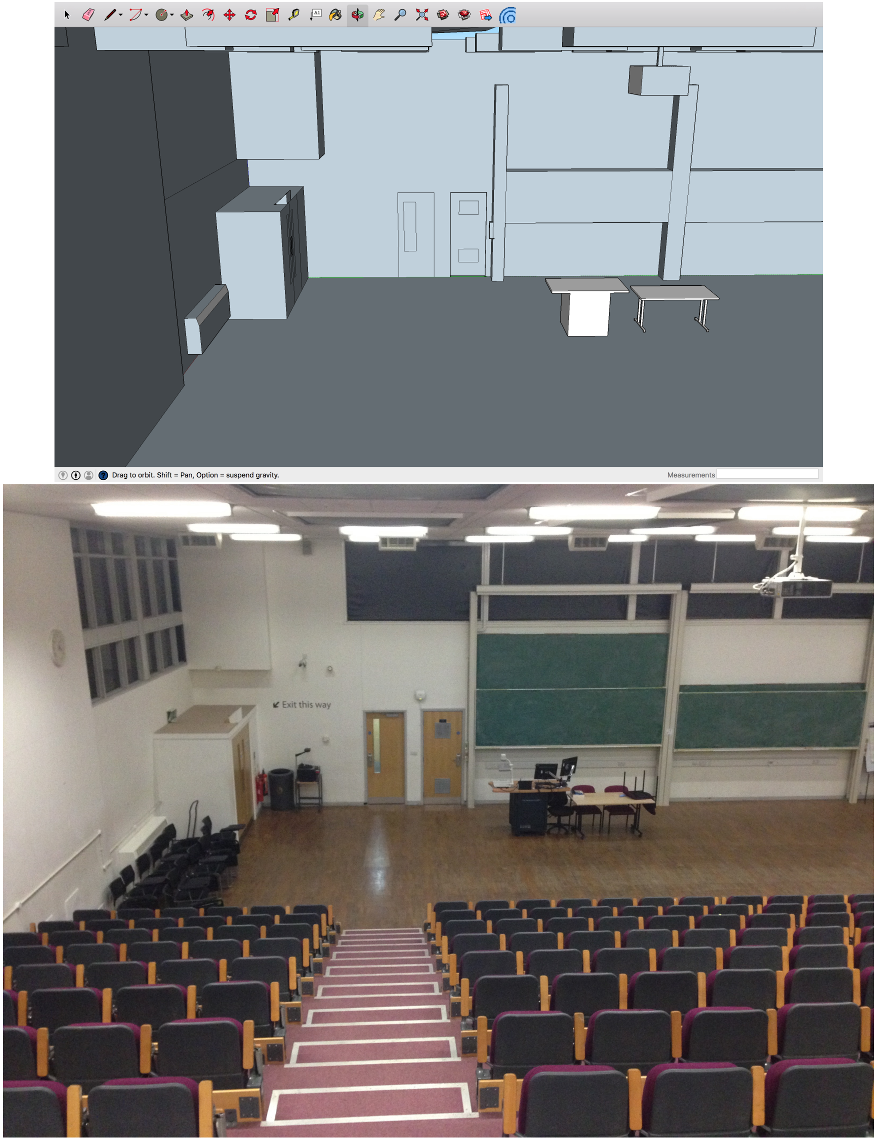


Figure 39: Real Vs SKU Roof

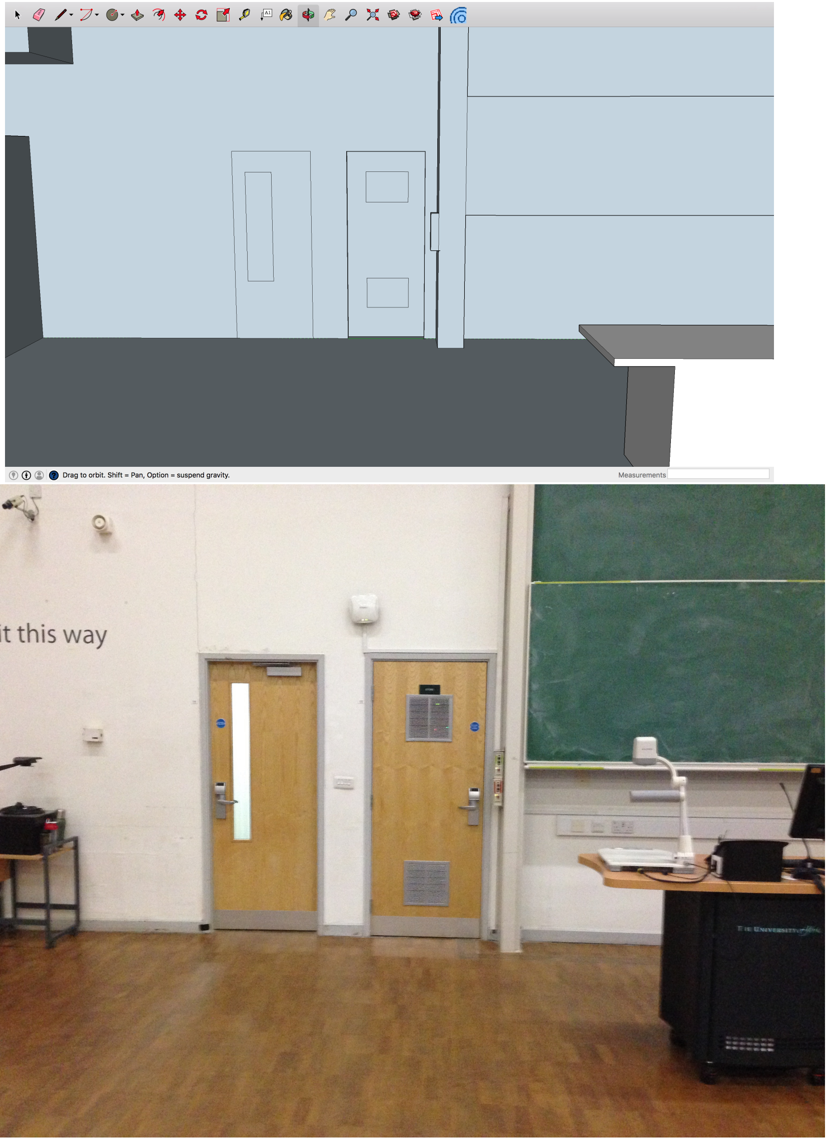


Figure 40: Real Vs SKU Roof