



---

# Door To Door

Jordan Wu

---

## Product Description

*Door to Door* is an interactive explorable three-dimensional environment that is experienced through a virtual reality headset in order to fully immerse the audience in various eerie locations featuring strange and uncanny themes.

The environment is controlled using the handprints on the table which switch between the scenes inside the oculus headset. This is done through the

use of conductive paint which sends signals to an Arduino which in turn sends a signal to the unity game engine, displaying it on the Oculus headset.

---

## **Intended Audience**

*Door To Door* was created with the intention of being an installation piece that would be displayed in a digital arts gallery. With this in mind the content of the environment would have to be suitable for any members of the public and artists who would be likely to enter a digital arts exhibition. This could range from around mid teens to thirty years of age. Viewers in the exhibition would also be drawn to the piece because of the use of virtual reality which, even with its recent surge in popularity, is still a niche product. Nonetheless, the work was built with a more mature audience in mind, who would be able to appreciate the themes and structure of the environment in comparison to the conventional fast-paced video game action.

---

## **Initial Ideas And Intended Outcomes**

Before doing extensive research into level and design structure, the piece was designed to be a fully explorable world with a controllable first person camera in which movement and character placement would affect the structure of the world around you such as in video games like *DMC: Devil May Cry*:

The effects and actions would be based around the themes of distortion and discord, as if the environment was in an unstable state with the user being directly responsible. Creating a composition alike to this would have taken excessive amounts of time to produce, especially creating something that would be as effective and well designed as the video above. I feel the piece would also have no objective and be labelled as just another game rather than something that has a deeper meaning.

However, I still went on to created a virtual environment with eerie themes but the controls in the final product were minimal and bound to only head movement and teleporting scenes, so that the user would not be too distracted by the interaction of the piece and could focus more on the themes and elements of the places displayed. With this in mind I still wanted the position and view that the camera displayed to

affect the objects displayed on screen. I wanted to do this through ray casting and making certain elements disappear or reappear using this method.



### *Level concept design*

The idea was to recreate the room in such a way that it would be both similar and different to the original, with the actual room being a bit blander than the virtual space. The wooden table would be in both the virtual and the real world. The only difference would be that there would be people sitting on the chairs inside the virtual space. This adds an ominous feeling that there are people there with you but simultaneously there isn't anyone there. Along with a séance like theme the piece would put people in a state of unease through subtle small changes. Although it is not clear in final product, the people sitting round the table are all staring at one another including yourself in different scenes. There was also originally a sequence between the levels but this idea was scraped.



*Level Sequence*

---

## **Background Research**

My background research focused on different artists who had themes of surrealism and disruption in their works. I began researching Surrealist artists such as Max Ernst's landscape pieces and Vladimir Kush, taking on board the symbolism and general integration between two separate themes in the pieces. After researching these surrealist artists, I discovered photo manipulations and further studied artists such as Nancy Fouts and Erik Johansson, trying to find ways of incorporating their ideas into my work.





Later on in my research I also studied the *Uncanny Valley*. This is a concept in which an image or object has a slight resemblance to that of a human, because of this the viewer both recognises and partially understand it but at the same time is repulsed and experiences unease making the image or object *weird* or *creepy*. I kept this concept in mind throughout the design process of the scenes, as I wanted to create an eerie atmosphere without using the cliché horror techniques such as jump scares and monsters. I based these areas around the AI used in the game *Sherlock Holmes Nemesis* in which the NPC *Watson* will teleport in front of you when he moves off screen. As displayed in this video, alongside a dramatic soundtrack, which was also incorporated in my piece, we can see that there is a unnerving feeling every time he appears around you.

Another area of interest that helped me shape my ideas were the warped environments created in video games such as the ones designed by Alessandro Taini in *Enslaved: Odyssey to the West*, *DMC:Devil May Cry* and *Heavenly Sword*. From games like these I briefly looked into video game level design, looking at the methods artists use to coerce the gamer into going in specific directions. An example of this would be leaving a open space thats more brightly lit than other paths so the player instinctively moves down that one. A more basic example would be making items glow so you know to interact with them.



---

### **Creative Research:**

<http://igor.gold.ac.uk/~skata001/hiveMind/2015/10/30/freedom-through-surrealism/>

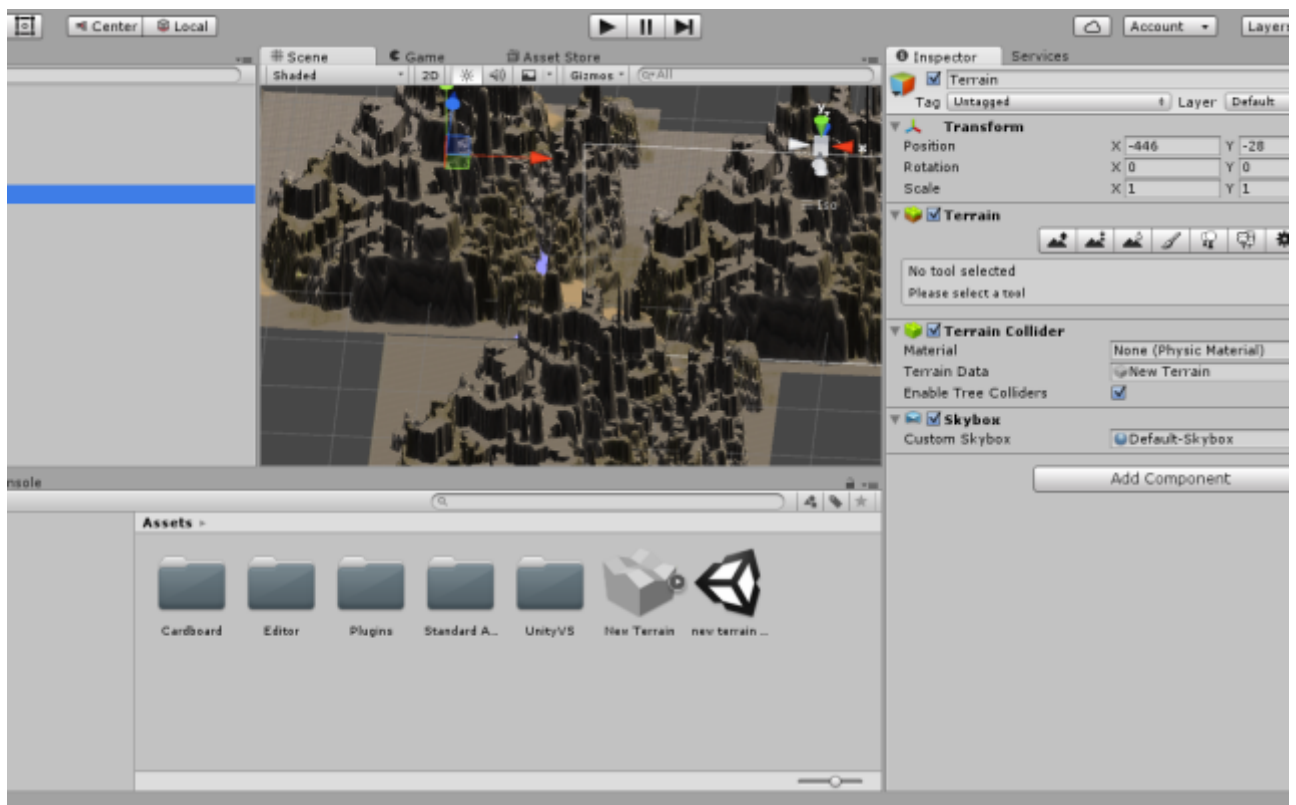
---



## Building and Design Process

Even with the idea in mind the process of creating this piece took on many forms and many ideas were scrapped. As mentioned above, the piece was originally meant to be a fully explorable setting with controls and a place that was procedurally generated. The world would have been designed and created using *openFrameworks* with libraries such as *ofxRules* and *AssimpLoader* alongside uses of shaders and a first person camera library.

Before the use of the Oculus the composition was originally planned to be developed on the more affordable *Google Cardboard*. I went on to create a working prototype using the *Unity game engine*. This allowed me to create my first testable *Google Cardboard* environment on an android mobile. However, the piece was basic and not procedurally generated.



After I understood the scope of the project I realised that the piece was a bit too ambitious with the time constraints and attempted to rework the concept. Without abandoning the idea I went on to narrow it down, removing features such as player control and the procedural generation to focus on more important aspects of the art

piece such as the principles of disorder and strangeness and how these would be presented and interpreted by the viewer.

During the development period I did not consider using *Unity* as the main system because of the lack of C++ support. My original plan was to create an environment using OpenFrameworks and export the file to a mobile device. The problem however was the issue of exporting an *Xcode* file as a executable for android or iPhone that would work with *Google cardboard*. Creating an android application this way would require a wide range of programs such as *Eclipse* and additional plugins and coding. The other programming options at the time were *Unreal Engine* and *Unity*. Unlike *Unity*, *Unreal Engine* had supported use of C++ while *Unity* ran on C# scripts.

Because of the lack of C++ support I began creating my project on *Unreal Engine*. Using the engine I was able to create a realistic model of the exhibition room using photos as sources. I made sure to get the dimensions correct and the positioning of the camera in the location in which the user would experience the piece. When recreating the room, I had to keep in mind different textures and objects that were both available and essential in creation. When people see something that they recognise they are drawn to small distinctive features. I had to pinpoint specific features around the room to make it recognisable to the viewer. Even though this was the main point of the table other features also needed to be present. Objects such as the wiring poles along the walls, the computer desk and the TV were also included as static but distinguishable objects in the real world. Further detail was added such as the *Goldsmiths* logo and the same clock.



*Exhibition Room Panorama*





### *Early UE4 Room Structure*

During the process of creating the basic prototype in *Unreal Engine* I had decided at this point that using the *Oculus Rift* was the most ideal way of immersing the viewer into my environment. At this point I had already scoured the internet for ways in which I could develop a *Google Cardboard* application from *Unreal Engine*. Unfortunately, after much research and posting on forums I realised that it was not possible and I was more likely to have luck with the *Oculus Rift*. And even if I was able to develop my own *UE Google Cardboard* library there would more that likely be bugs and other problem present, alongside the fact that I also had to create my world.



*Original Physical Design (before exhibition room)*

The design concepts for *Door To Door* included the uses of a pressure pad or some kind of touch motion on the table. Using an *Arduino* was the next step in order to add physical interaction to piece. After researching many methods in *Unreal Engine* plugins I struggled to find a suitable way of conveying information between the *Arduino* and *Unreal Engine*. I then tried to send OSC signals between the two programs, trying to use it as a link.

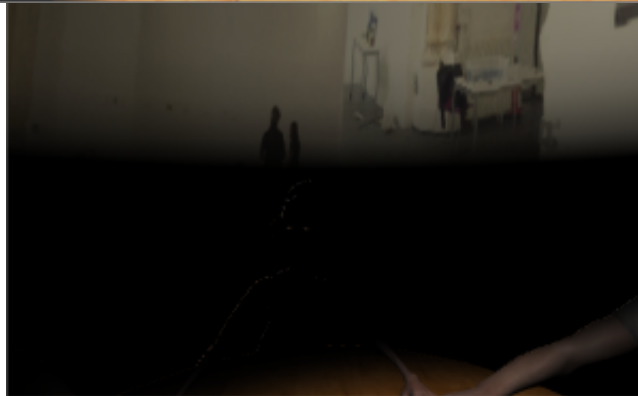
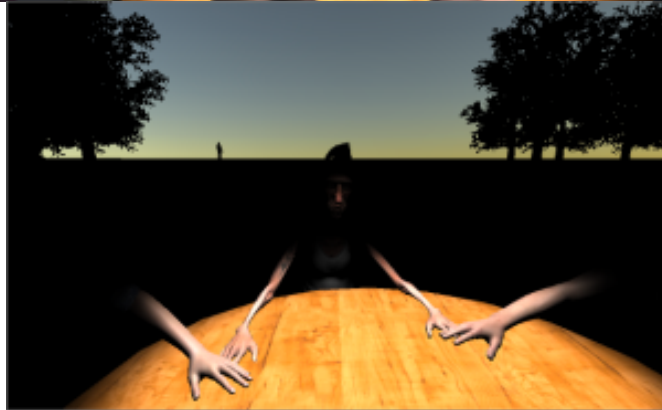
Along with the steep learning curve of using a new game engine I was also unable to successfully find a means to displaying the scenes created in an *Oculus Rift*. With only a few weeks remaining and the project at a standstill I decided to, once again, switch to a new program. This time I wanted to use Unity. Although it does not support C++, Unity is able to run C# scripts and is known to be easily compatible with the *Oculus Rift*, fixing both my problems together.

The room created in *Unreal Engine* was firstly exported a .FBX file so that it could then be imported into *Unity* as an asset. The model needed to be scaled down and textures had to be applied. After creating a suitable terrain to place the model in I then used photos taken inside the room to re-create it.

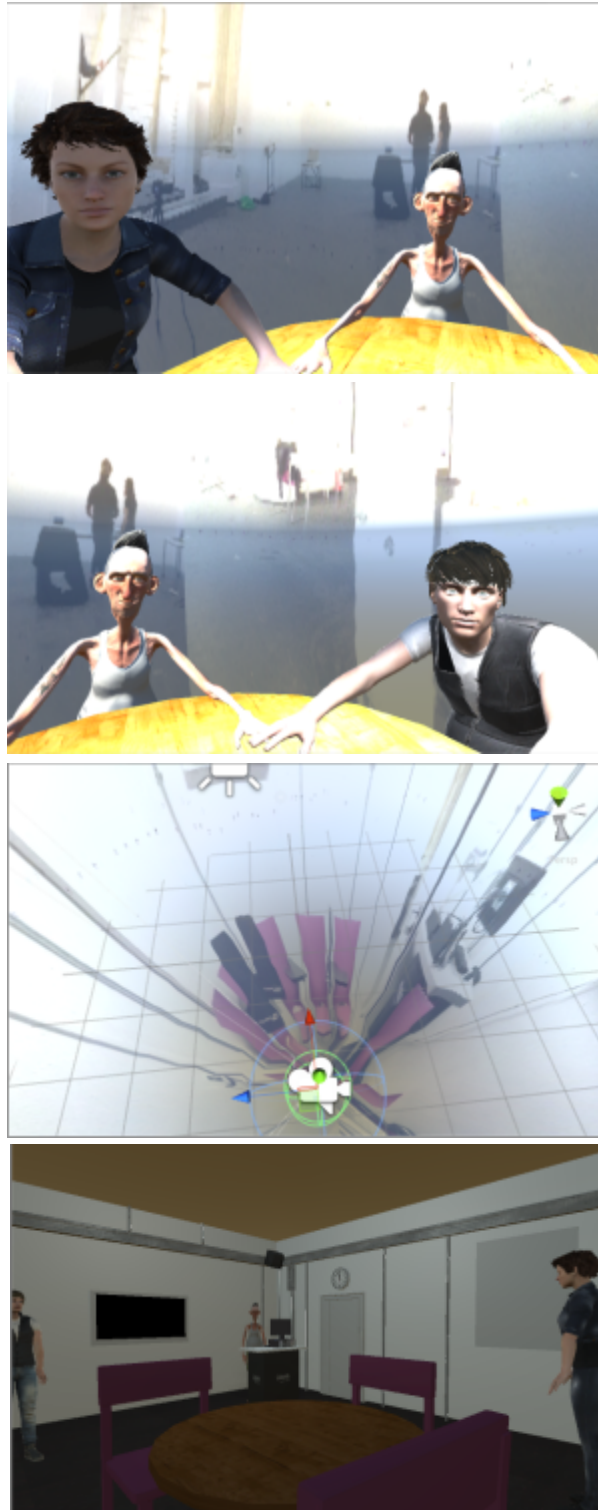
## *Unity Room Recreation*

Using the Oculus Rift for test runs was also a success. This led me to create more scenes with different variations, some not included in the final piece.









Another essential step was to include interactivity into the piece. Instead of using a pressure pad for the hand controls I decided to use Conductive paint. Building the conductive paint into a simple Arduino circuit, every time the paint was touched the resistance in the circuit would increase and the output would be displayed on the program. using an if statement I used this to send a number to *Unity* using serial

ports. This would be received using a C# script in *Unity*. The video below shows me using the conductive paint to move an object in the room.

<https://www.youtube.com/watch?v=s9abuCnoimg>

Using the same code from above I then modified it so that instead of moving the cube a new level would be loaded. The shape of the 'buttons' on the table were also important. Instead of having a regular button or small area of touch with one finger hand prints were used instead. This way the user feels more connected to the characters in the scene, as if they really are taking part in a séance and not just playing another video game where joysticks control their movement. I believe that this is one of the aspects that separates my piece from being another video game experience.

To add to the atmosphere of the setting, audio was an important aspect of the piece. I began experimenting with using cymbals being played backwards and low humming sounds as well as suitable transition sounds like a camera shutter, nothing too abstract or distracting. With very limited knowledge in creating audio I used *GarageBand* to manipulate and edit the sounds.

For the background music the soundscape synthesizer was used. I changed instrument used depending on the scene but kept the notes that were playing the same so there was a low humming noise in the background throughout. The only scene without the background music was the original empty recreation of the exhibition room. I felt that this needed to be silent as it gave the scene its own uneasy atmosphere, as if the user is waiting for something to happen.

The scene transition sounds, although they may sound like a camera were actually based on my work from last year. My piece *Recursive Flaws* featured various sounds of typing, rerecorded again and again. I used one of these sound files and reversed the sound and shortened it to create an eerie mixture of backwards typing as well as a shutter like sound.

Setting up the exhibition was not too difficult. The only challenges were hiding the laptop and the *Arduino* away from public view. The laptop was placed inside a plinth in the corner of the room and the *Arduino* was glued and placed inside a container under the table. The Conductive paint was applied directly onto the table.



*Arduino Placed Under The Table*



*Conductive Paint Applied To The Table*





*Exhibition Setup*

---

***Door To Door***







<https://www.youtube.com/watch?v=rFAuBjWxoko>

<https://www.youtube.com/watch?v=el8Ijy2sZiQ>

<https://www.youtube.com/watch?v=-Pdm5aEn6Kk>

---

## Encountered Problems

1. The first issue was the constant change in different programs, first being *OpenFrameworks* and the *Unreal Engine*. This would have been a bigger problem but swift action and converting the work (such as the models) over to Unity made the process a success. Even though the lack of coding in the final piece was also a problem, I had still been coding throughout the project even if it was not present in the final product.
2. The conductivity of the paint varied depending on the material that it was placed on. Originally there was a mirror plate covered in the conductive paint for the hand pads. The plate did not work properly and caused more errors in the code. This was amended by putting the paint directly on the table as wood seemed to be the most successful material.
3. Problems with the serial ports and the information send and received from the *Arduino* caused more errors than needed. They caused the scenes to run through continuously. At first I thought that there was a problem with the coding and Unity receiving too much information. I spent a lot of time working on a script that had a delay time between receiving signals however the problem persisted. By the time I realised that the problem was with not only the conductive paint and mirror plate but also wiring problems it was too

late to adapt the code. I used my C# failsafe which was not perfect but fixed the problem of the *Oculus* running through all the scenes consistently.

4. The next problem branches from the last one. The loading times between scenes caused confusion for the viewers, who had thought the program had crashed. The sensitivity of the conductive paint also affected the interactivity of the piece. The fact that people could not hold it down even though there were hand prints on the table confused them. I had to manually explain to them to touch the paint only once.
5. The calibration in the program was slightly off. This caused some scenes to be slightly off angle. Although this was fixable manually many of the viewers enjoyed this happy accident and said that it added to the sense of confusion and disorientation.

---

## Evaluation

Although the exhibition piece was not exactly how I imagined it to be when I first came up with the concepts I believe that it is was a success but not without its weaknesses and areas of improvement. Even from the second overhaul of the idea, there was a lot missing such as the ray-casting techniques and animated moving AI.

On the other hand, with the three changes in the programs used and the time constraints, I think that being able to incorporate these elements would have been highly unlikely with my amateur skills in using these programs. The original concepts that I developed were also a bit too over ambitious. Creating a whole gaming world itself is a lot of work, in both designing it as well as working on it, not to mention the original plan of also decorating the physical surrounding to a new room inside the exhibition room. In the future I will consider the time we are allocated to produce the work as well as the level of skill that I have in certain programs when designing new pieces.

The amount of coding in the final product was also of a small amount. To avoid a situation like this I must do more thorough research into both my ideas and the programs I am planning to use before diving head first into the project. The errors and problems with the interaction were one of the major weaknesses in my project. Although some of them turned out for the better I need to be able to find and correct these problems before it is too late. Whether that is more viewer feedback and tests or setting up earlier. Problems such as the loading times could potentially be fixed (such as putting all the scenes in one instead of loading them individually) with experimentation and time.

On the other hand, the audience were mainly impressed with the piece and were fascinated by the evolution and realness of the program, saying that they almost forgot that they were sitting in a room in *Goldsmiths*. This was a pleasant outcome as taking the viewer away from the current reality was one of my goals. People were also intrigued by the themes and structure of the piece. Many of them asked if there was a relevance to the order in which the scenes showed and who or what was happening.

They seemed to enjoy trying to decipher a meaning or find an understanding. The success was reflected by the queues of people waiting to experience it. The piece also gave people an opportunity to experience virtual reality which, without a lot of money or enthusiasm, they would not normally have encountered. The experience for them in itself was enough to satisfy. It has also shown me the importance of interactivity in a gallery art piece. People are more likely to be engaged if they feel a part of the piece and not just controlling someone else like in a video game.

Overall I feel that *Door To Door* was a success. It definitely has its faults and has much room for improvement but it has taught me valuable skills in not only programming and coding but also of the audience in galleries. There were many challenges throughout the project but the reactions of the spectators showed its strength as a piece.

---

## References

### *Programs Used:*

Unity3D

Arduino

*Used in Research:*

OpenFrameworks

Unreal Engine

*Addons in Research:*

ofxRules

*Videos and Images:*

ofxRules video:

<https://youtu.be/M4YJe86CLc4>

Sherlock holmes Video:

<https://youtu.be/13YlEPwOfmk>

DMC: Devil May Cry:

<https://youtu.be/6waRMnKApVo>

AlesandroTaini:

[http://www.cgsociety.org/cgsarchive/newgallerycrits/g40/266440/266440\\_1290781090\\_large.jpg](http://www.cgsociety.org/cgsarchive/newgallerycrits/g40/266440/266440_1290781090_large.jpg)

Erik Johansson:

[http://igor.gold.ac.uk/~skata001/hiveMind/wp-content/uploads/2015/10/a38f2c050c93b3256f8b17ef5f77a7f33028339a\\_800x600.jpg](http://igor.gold.ac.uk/~skata001/hiveMind/wp-content/uploads/2015/10/a38f2c050c93b3256f8b17ef5f77a7f33028339a_800x600.jpg)