

Wright State University

CORE Scholar

---

[Browse all Theses and Dissertations](#)

[Theses and Dissertations](#)

---

2019

## Virtual Reality and Analysis Framework for Studying Different Layout Designs

Madison Glines  
*Wright State University*

Follow this and additional works at: [https://corescholar.libraries.wright.edu/etd\\_all](https://corescholar.libraries.wright.edu/etd_all)



Part of the [Computer Engineering Commons](#), and the [Computer Sciences Commons](#)

---

### Repository Citation

Glines, Madison, "Virtual Reality and Analysis Framework for Studying Different Layout Designs" (2019).  
*Browse all Theses and Dissertations*. 2130.  
[https://corescholar.libraries.wright.edu/etd\\_all/2130](https://corescholar.libraries.wright.edu/etd_all/2130)

This Thesis is brought to you for free and open access by the Theses and Dissertations at CORE Scholar. It has been accepted for inclusion in Browse all Theses and Dissertations by an authorized administrator of CORE Scholar. For more information, please contact [library-corescholar@wright.edu](mailto:library-corescholar@wright.edu).

VIRTUAL REALITY AND ANALYSIS FRAMEWORK FOR  
STUDYING DIFFERENT LAYOUT DESIGNS

A thesis submitted in partial fulfillment of the  
requirements for the degree of  
Master of Science

By

MADISON GLINES  
B.S.C.S., Wright State University, 2017

2019  
Wright State University

WRIGHT STATE UNIVERSITY  
GRADUATE SCHOOL

May 2, 2019

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY  
SUPERVISION BY Madison Glines ENTITLED Virtual Reality and Analysis  
Framework for Studying Different Layout Designs BE ACCEPTED IN PARTIAL  
FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF Master of  
Science.

---

Thomas Wischgoll, Ph.D.  
Thesis Director

---

Mateen Rizki, Ph. D.  
Chair, Department of  
Computer Science and Engineering

Committee on Final Examination:

---

Thomas Wischgoll, Ph.D.

---

Pratik J. Parikh, Ph.D.

---

John Gallagher, Ph.D.

---

Barry Milligan, Ph.D.  
Interim Dean of the Graduate School

## **ABSTRACT**

Glines, Madison. M.S. Department of Computer Science, Wright State University, 2019. Virtual Reality and Analysis Framework for Studying Different Layout Designs.

This thesis describes the tools for studying different design prototypes. The goal was to develop effective tools to study these designs using a data-driven approach. “Proof of concept” experiments were conducted, in which participants were allowed to interact with a virtual environment depicting different designs as data pertaining to their virtual location and orientation was recorded for later analysis. The designs included “flat” store racks, as opposed to racks with more varied shapes, as well as “curved” racks. Focus of the design studies was to assist in identifying optimal locations for different product types. The automated data collection mechanisms required specialized data analysis tools which were also developed throughout this work. For the study, mostly full-scale walkable VR displays were used. However, this thesis also includes some discussion on potential avenues for expanding this project, such as the use of VR headsets.

## TABLE OF CONTENTS

1 Introduction.....	1
1.1 Why Use Virtual Reality? .....	2
1.1.1 What Technology are Other Projects Using? .....	2
1.1.2 What Technology are we using? .....	5
1.2 Data Collection.....	8
1.3 What is a Rack?.....	10
1.4 What is a layout? .....	13
2 Phase One - Flat Racks .....	15
2.1 Questions .....	15
2.2 Experiment Setup .....	17
2.3 My Contribution .....	21
3 Phase Two - Curved Racks .....	21
3.1 Questions .....	21
3.2 Experiment Setup (Phase 2) .....	22
3.3 My Contribution .....	25
3.3.1 CurBoxObject.....	25
3.3.2 Unique issues with CurBoxObject .....	27
3.3.3 Playback Overhead View .....	27
4 Phase Three - Fully Populated Racks .....	29
4.1 Questions (Phase 3) .....	29
4.2 Experiment Setup (Phase 3) .....	30
4.3 My contribution.....	32

4.3.1 Increasing image capabilities .....	32
4.3.2 New Control System.....	33
4.3.3 Creating images .....	35
5 Additional Research.....	37
5.1 Why Head Mounted Displays? .....	37
5.2 The Vive .....	38
5.3 The Oculus Right First Dev Kit .....	39
5.4 Future of HMDs with the Experiment.....	41
6 Conclusion and Future Research .....	41
6.1 Summary of Contribution.....	42
6.2 Phase Four and Beyond .....	43
References.....	45

## TABLE OF FIGURES

Figure 1. Tesco shelf at the “Virtual Store” [4].	4
Figure 2. A young woman shopping with her smartphone [4].	4
Figure 3. View of human in the DIVE with a ( $\theta = 90^\circ$ ) layout on display.	6
Figure 4. The Logitech F710 Wireless Gamepad with reflective spheres attached.	7
Figure 5. Samsung SSG-5150GB 3D active glasses with reflective spheres.	7
Figure 6. Schematic describing the overall setup of the DIVE system, including displays and computers involved in driving the system. [6]	8
Figure 7. Excerpt of output file	10
Figure 8. Head Rotation (horizontal) for single participant run [8].	10
Figure 9. Excerpt from a ( $60 = \theta^\circ$ ) right side rack with targets.	12
Figure 10. ( $\theta = 45^\circ, 90^\circ$ , and $135^\circ$ ) Overviews.	13
Figure 11. Excerpt from a ( $\theta = 60^\circ$ ) with floor.	14
Figure 12. Excerpt from a ( $60 = \theta^\circ$ ) right side rack with targets.	15
Figure 13. Exposure/intensity at step y [1].	16
Figure 14. ( $\theta = 30^\circ$ ) Overview and ( $\theta = 150^\circ$ ) Overview.	20
Figure 15. Before ( $\theta = 90^\circ, \alpha = 30^\circ$ ) and Before ( $\theta = 135^\circ, \alpha = 90^\circ$ ).	22
Figure 16. After ( $\theta = 90^\circ, \alpha = 30^\circ$ ) mirror and after ( $\theta = 90^\circ, \alpha = 30^\circ$ ).	22
Figure 17. After ( $\theta = 135^\circ, \alpha = 90^\circ$ ) mirror and after ( $\theta = 135^\circ, \alpha = 90^\circ$ ).	23
Figure 18. ( $\theta = 135^\circ, \alpha = 0^\circ$ and $30^\circ$ )	24
Figure 19. Example of half a column of a shelf.	33

Figure 20. Sample Training layout side view. ....	35
Figure 21. Basic shelf template with and without a product image. ....	36
Figure 22. Example of a slot shown with different stacking options.....	37
Figure 23. Vive HMD [9]. ....	39
Figure 24. Oculus Rift First Devkit HMD. ....	40



## 1 Introduction

The goal of the experiment was to take the field of view (FOV) model and compare it to human performance, to compare the reactions of the participants to different virtual environments, and to use this data to improve the model's reflection of human performance. This model applies a provided model of the FOV to a model of a rack to quantify what should be seen by a shopper.

To test this, we decided to design an environment to simulate a grocery store environment with racks set at various angles, curvatures, and heights. We would then analyze which conditions allowed for customers to see the most information easily. Our team of researchers from three different areas of expertise (psychology, industrial systems engineering (ISE), and computer science) set out to answer the seven questions stated below.

“Q1. How can exposure be quantified in 3D considering human field of regard?

Q2. What effect does rack orientation, curvature, and height have on exposure?

Q3. How sensitive are these findings to shopper traffic and scanning patterns?

Q4. What is the trade-off between exposure and floor space?

Q5. How can increases in exposure quantitatively be connected to gains in impulse profit?

Q6. What is the optimal rack configuration that maximizes impulse profit?

Q7. How sensitive is the optimal solution to the product location strategy, shopper volume, travel direction, floor space cost, and maximum aspect ratio?” [1]

There are three main phases to this experiment, with the first two phases focused on questions one through four. These include racks set at various angles, or “theta” values ( $\theta$ ), and was expanded upon to include curvatures to the racks, or “alpha” values ( $\alpha$ ). The third phase of

the experiment began to explore the final three questions by fully populating racks with product images.

## **1.1 Why Use Virtual Reality?**

Virtual reality (VR) was chosen due to significantly reduced space requirements, freedom to experiment with rack types and ability to record data required for analysis. If a company wanted to carry out this study in a non-virtualized environment, it would require either a large space to house different rack configurations or a lot of time to have one layout displayed, run a test, and then rearrange the whole area into the next layout to run another test. Additionally, they would need to have working prototypes of all types of racks they would want to try. This would require a lot of time and money; without the guarantee the rack type would be used.

Unlike the real world, a virtual environment (VE) has a significantly reduced space requirement and can house multiple layouts. We are free to build whatever shape of rack we would like without the same cost investment. VEs offer complete reproducibility (i.e. every participant can experience the exact same environment) and automatic records user data such as head movement.

### **1.1.1 What Technology are Other Projects Using?**

VR has been used to simulate grocery stores for many projects aside from this one. The first example of this is the use of virtual environments for the purpose of rehabilitating veterans who are recovering from mild traumatic brain injuries and PTSD. Through their research, the group had determined that a grocery store would be the ideal simulated environment to achieve their goals [2]. The primary focus in the design of this system was the realism of the virtual environment. Other goals included the capabilities for the therapist to set tasks for the participant to complete and to control pre-recorded voice prompts. The system also needed to

allow for the participant to interact with products on the shelf in various ways, such as picking an item up and moving it around [2]. The team later chose to expand their system to use an Xbox controller, based on feedback received from their participants. In consideration of their target demographic, they decided to avoid using head mounted displays (HMD) due to risks of motion sickness and other side effects which might result in anxiety. They also decided on the use of a simple television display [2].

In another experiment, researchers wanted to determine how the appearance of fruits and vegetables affect shopper's purchases. They chose to use a VR simulation due to the shelf life of the products and began by creating virtual models based on a range of fruit and vegetable samples, from those with "normal" shape to those which would be considered "severely misshapen" [3]. The use of VR also allows for the reuse of the same visual models for different participants, reducing the potential variability that might occur in using multiple different sets of physical fruits and vegetables. The display they chose to use was an HMD (Oculus Rift DK2), to not only increase the environment's ability to immerse the use. This experiment also uses an Xbox controller. The system was designed such that the participant could pick up and rotate items, as one may do when shopping for fruits and vegetables. Their VE was designed to "mimic the Audencia Business School IN SITU store laboratory, where consumer studies are carried out on real (nonperishable) food products" [3]. This experiment used the optical camera from the DK2 for head tracking [3].

In 2011, Tesco had opened a virtual store in the Seolleung underground station in Seoul, South Korea. [4] In this case, the displays for the virtual environment are not on monitors or HMDs, but rather on the user's smartphone screen. An example of a shelf from this store can be seen in Figure 1. Smartphones were integrated by instructing the shopper to scan the QR code or

barcode for the product being displayed as seen in Figure 2. After this, the shopper would place the order from their home, which would be promptly delivered by Tesco. Tesco went as far as to guarantee that if the order was placed by 1:00 PM, the order would arrive before the end of the day [4].



Figure 1. Tesco shelf at the “Virtual Store” [4].



Figure 2. A young woman shopping with her smartphone [4].

Besides just creating stores which implement smartphones, companies such as Kimberly Clark and Procter & Gamble have created virtual test stores as an alternative to a physical test environment [5]. This news prompted a team of researchers from France and the United States

to study behavioral differences of customers between physical and virtual stores. For this study, they first created a physical mock-up of a store, which was then recreated in a virtual environment. They designed a similar setup as the grocery store created for rehab, although they used a mouse as the controller and keyboard arrow keys for movement, and also allowed for the participant to zoom in on products they picked up. They collected data in three forms: soliciting demographic information and shopping habits, video of participants' activities within both environments, and having participants answer questions about the target brand after the experiment [5]. Their results show that when it comes to testing out the first four questions (which are focused more on the structure of racks and their layouts) that there should not be a significant difference between a virtual or physical environment. When it comes to a virtual store, participants were more likely to pick brands they were already loyal to, and to generally retain less information such as price and products' presence [5].

### **1.1.2 What Technology are we using?**

To run the VE for the experiment, we were used the Distributed Interactive Virtual Environment (DIVE) [6], which is set up to resemble a typical Cave Automatic Virtual Environment CAVE [7]. The DIVE was chosen because it met all the requirements of field of view (FOV) of more than 180°, had the ability to record the user data, and could move the user through the VE to make the comparison to the theoretical model.

The DIVE is set up with 3 walls with a 3 by 3 tilted stereoscopic display configuration. This system has a 12' by 12' of walking space and provides a height of about 87". It allows for a FOV of 270° [6]. In order to track and record the participants movements in the VE, we used 11 Optitrack cameras placed around the frame, which would detect the reflective spheres mounted to the Logitech F710 Wireless Gamepad and Samsung SSG-5150GB 3D Active Glasses. Figure

3 shows the DIVE in use, and Figures 4 and 5 shows the controllers used with the DIVE. SSG-5150GB 3D Active could be replaced with any active 3D glasses, as long as the reflective spheres are mounted to them. If the Logitech F710 Wireless Gamepad is broken, it too could be easily replaced, although if it were to be replaced with a different type of controller, adjustments to the code for the sake of compatibility would be required.



Figure 3. View of human in the DIVE with a ( $\theta = 90^\circ$ ) layout on display.



Figure 4. The Logitech F710 Wireless Gamepad with reflective spheres attached.



Figure 5. Samsung SSG-5150GB 3D active glasses with reflective spheres.

Some drawbacks of the system were that the DIVE does not have a fourth wall behind the participant, and also lacks a ceiling or floor screen. This results in some of the participant's view of the virtual environment being "cut off". This was not a major issue but had potential to sometimes obscure targets too early. The screen boundaries between tiles result in a slight distortion of the view, which is especially significant at the corners formed by two walls. Additionally, the use of more graphically intense environments and high-quality textures can

greatly increase loading times, as they are a result of multipath rendering on four different clients which load the data over a network from the same server. Figure 6 demonstrates the configuration of the computational hardware used within the DIVE. [6]

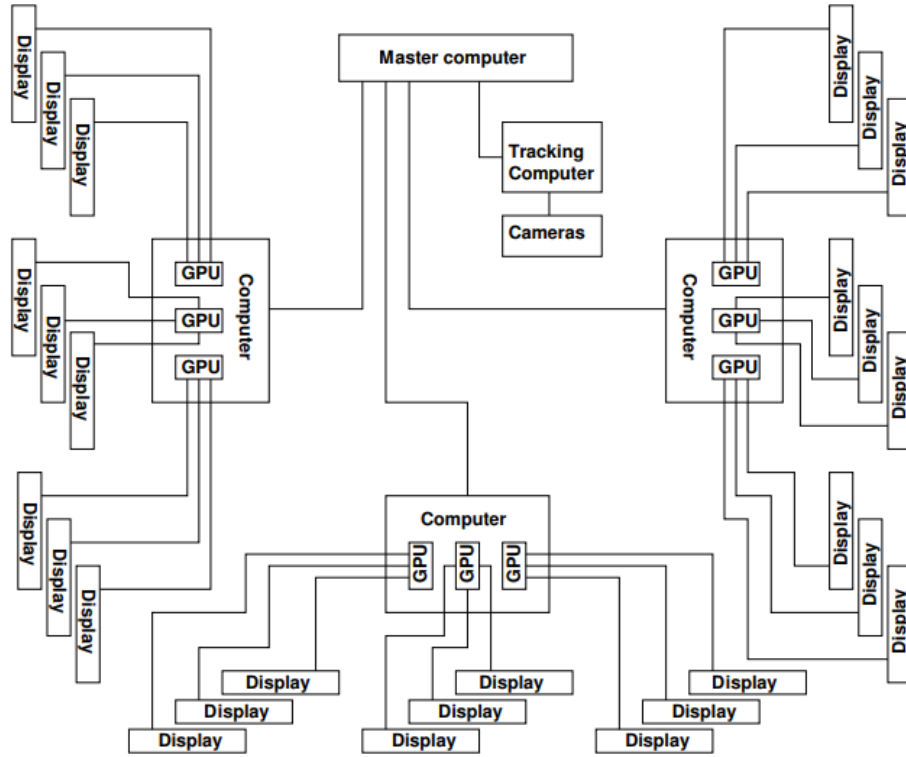


Figure 6. Schematic describing the overall setup of the DIVE system, including displays and computers involved in driving the system. [6]

## 1.2 Data Collection

A significant advantage of the existing setup is that the environment is generated from a parser which reads a separate configuration file. The configuration files are not required to be compiled between changes, which allows for quick adjustments to be applied to various layouts. The parser not only generates the VE, but also updates the user's location within the VE and gathers the data from the tracking system, which is written to the output file. In second the phase of the experiment, the parser also houses the code to run a "play back" of a user's movement within a previous run in the VE.



Initially, the controller class worked with the parser class and would pass on messages indicating button presses. The red “B” button was used to initialize the recording of data, and the left and right shoulder buttons were used for verifying when a target was spotted which would be used to compare to the theoretical model. In the third phase of the experiment, the class expanded to read in the input from the right joystick, along with being the class to process the updates to the location of the user. The parser is still the class to read and write the data to the output file.

The DIVE system records the participant’s head data, allowing us to further analyze the effects different layouts have on the participant. Figure 7 displays how the system records the head x-y-z coordinates and the second line shows how a button press is recorded within the data. The specifications for the starting location was that the participant must be given enough time to hold their head still to record initial head data. In order to interpret the data, our team needed to get a baseline of where a given participant’s head was located in the virtual environment. Figure 8 displays how the data can be used to recreate the scanning pattern a participant used during the layout.

**Configuration:** racks90-dist2.cfg, **Date:** 3-9-2017, **Time:** 14:40:9

**Time:** 0.000000 Button B pressed

**Time:**0.000013,**Head**,-0.000636,862.737915,76.599678,**ViewDirection**,0.019196,0.217598,0.975850,**Translation**:,0.000000,80.000000,-800.000000,**RotationalAxis**,1.000000,0.000000,0.000000,**Angle**,1.570000,**Scaling**,1.000000,**PhysicalHead**,-0.000636,4.087322,62.798636,**PhysicalViewirection**,0.019196,-0.975676,0.218375

**Time:**0.016732,**Head**,-0.001152,860.534789,77.416487,**ViewDirection**,0.019142,0.217780,0.975810,**Translation**:,0.000000,80.816658,-797.798405,**RotationalAxis**,1.000000,0.000000,0.000000,**Angle**,1.570000,**Scaling**,1.000000,**PhysicalHead**,-0.001152,4.085418,62.797756,**PhysicalViewirection**,0.019142,-0.975636,0.218557

Figure 7. Excerpt of output file

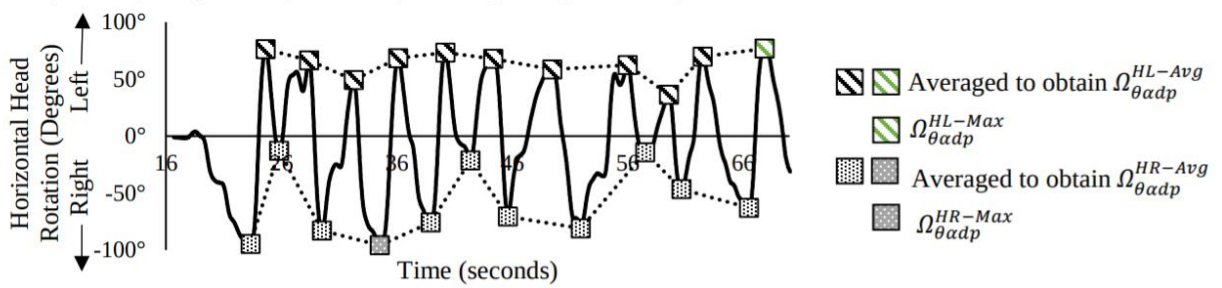


Figure 8. Head Rotation (horizontal) for single participant run [8]

### 1.3 What is a Rack?

The system consisted of three main objects: a PlaneObject (used for the floor), and a BoxObject (used for both the racks and the signs which display aisle numbers). The constructor for a BoxObject requires five parameters: x (length), y (height), and z (depth), and two tuples which specify the number of rows and columns. The first of these tuples (markerRaster) determines the number of rows to have on the front and back sides of the box, while the second

(markerSideRaster) determines the number of rows and columns on the left and right sides. The right, top and bottom sides of a BoxObject are all 2-dimensional planes. The size of the squares on this object is determined by taking the length of the box and dividing it by the x coordinate from markerRaster, and the height with the y coordinate from markerRaster. For the squares on the sides of the BoxObject, the same calculation is performed, using the z and y coordinates from markerSideRaster. In order to place a target, I needed to enter the corresponding markerRaster x and y and with the image number. This can be seen in Figure 7 on lines 400 and 413 set that there will be 150 columns and 6 rows on the front and back faces, and lines 401 and 414 set 7 columns and 6 rows on the side face of the BoxObject. On line 415, "TextureBackID 0" indicates the target image will be on the back face, and line 416 "MarkerBackPosition 50 4" these determines the exact x-y coordinates on the back face. In order to specify the image to render for the target (a red dot on a grey background), the index number 0 would be used as can be seen in Figure 9. The only requirement for using an image on a BoxObject is that only one image may be used on each face. For this reason, a BoxObject could have at most three images attached to it in early stages of development. The third phase increases the number of images that could appear on any face of the object.

```

393 # row 7 right
394 OpenGLObject
395     BoxObject 1260 78 58.8
396     Rotate 120.0 0.0 1.0 0.0
397     Offset 0 0 325.6
398     #Scale 0.01
399     #Color 0.7 0.7 0.0
400     MarkerRaster 150 6
401     MarkerSideRaster 6 7
402     TextureBackID 0
403     MarkerBackPosition 1 4
404 OpenGLObjectEnd
405
406 # row 8 right
407 OpenGLObject
408     BoxObject 1260 78 58.8
409     Rotate 120.0 0.0 1.0 0.0
410     Offset 0 0 325.6
411     #Scale 0.01
412     #Color 0.7 0.7 0.0
413     MarkerRaster 150 6
414     MarkerSideRaster 6 7
415     TextureBackID 0
416     MarkerBackPosition 50 4
417 OpenGLObjectEnd

```

Figure 9. Excerpt from a ( $60 = \theta^\circ$ ) right side rack with targets.

The third main object the CurBoxObject (short for “curved box” object) was introduced during the second phase of the system. The constructor for the CurBoxObject would accept all the same values as the standard BoxObject, with the addition of length, height, depth, and a value denoted as  $\sigma$ . The  $\sigma$  was used to define how much curvature the rack would have. The other difference in this implementation was that the x markerRaster and y markerSideRaster parameters no longer controlled the width of squares. The code had been implemented such that the length of the rack was equivalent to a diameter or slice of a circle, depending on the  $\sigma$ , and the circumference needed to be calculated by the system by the given length. This object uses the

same coordinate system to place targets and as of now has not fully been converted to fully accommodate more images.

#### 1.4 What is a layout?

The system initially had a layout with  $\theta$  values of  $30^\circ$ ,  $60^\circ$ , and  $90^\circ$ , but needed to be extended to allow layouts with  $\theta$  values of  $45^\circ$ ,  $135^\circ$ , and  $150^\circ$ . A “layout” consists of 20 rows with two racks on each side, positioned along the sides of the main aisle which the participant is guided through. Figure 10 below shows a sample of three different layouts.

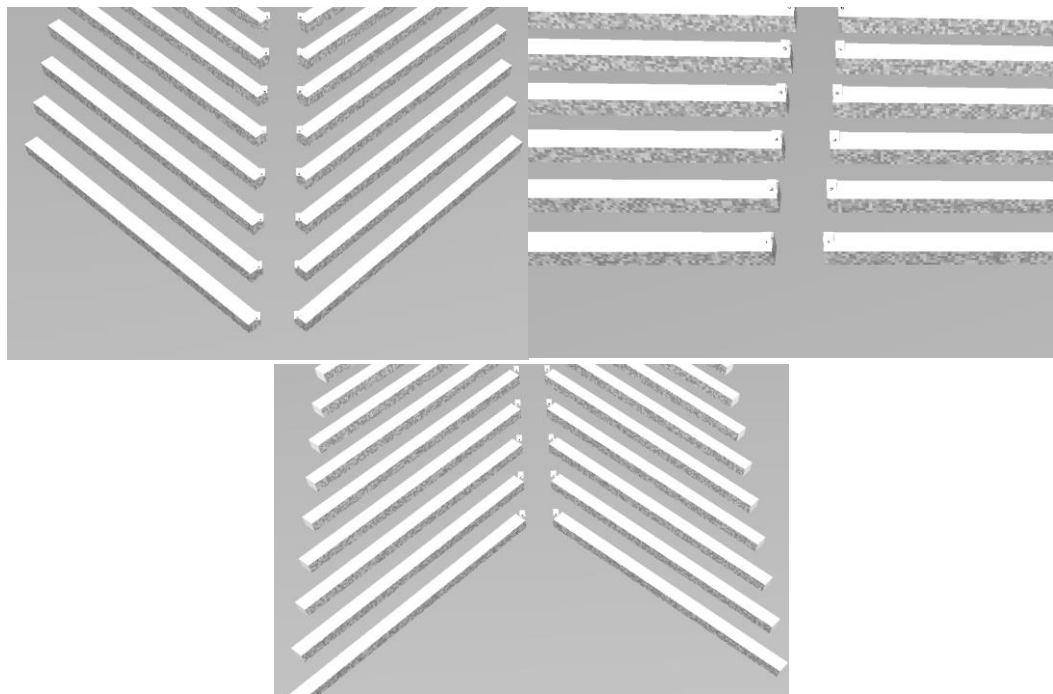


Figure 10. ( $\theta = 45^\circ$ ,  $90^\circ$ , and  $135^\circ$ ) Overviews.

I needed to specify the starting point of the first rack in both rows of racks, but the placement of racks 2 through 20 was determined parametrically, based on the location of the previously added rack and a constant offset value. Figure 11 line 108 shows that row 2 is offset by 325.6”. The speed of the system is set in the configuration file for each layout - the only control a participant in this experiment had was when to begin moving through the environment. Also included in the system was a set of 21 images: one for the target (a red dot on a grey

background), and a set of images to display the numbers 1 through 20 for the numbered signs above each aisle.

```
85  OpenGLObject
86      PlaneObject 2700 9000
87      Rotate 90.0 1.0 0.0 0.0
88      Translate -1350.0 78.0 -1600.0
89      Color 0.4 0.4 0.4
90      #TextureFrontID 0
91  OpenGLObjectEnd
92
93  # row 1 left
94  OpenGLObject
95      BoxObject 1260 78 58.8
96      Rotate 60 0.0 1.0 0.0
97      Translate 60.0 0 0
98      #Scale 0.01
99      #Color 0.7 0.7 0.0
100     MarkerRaster 150 6
101     MarkerSideRaster 6 7
102  OpenGLObjectEnd
103
104  # row 2 left
105  OpenGLObject
106      BoxObject 1260 78 58.8
107      Rotate 60.0 0.0 1.0 0.0
108      Offset 0 0 325.6
109      #Scale 0.01
110      #Color 0.7 0.7 0.0
111      MarkerRaster 150 6
112      MarkerSideRaster 6 7
113  OpenGLObjectEnd
```

Figure 11. Excerpt from a ( $\theta = 60^\circ$ ) with floor.

The side of the aisle to the right of the participant was where the initial BoxObjects were created and the left side was populated BoxObjects initialized on the right and rotated into position. Figures 11 and 12 show the right side is set to rotate at  $120^\circ$ , and the left side rotated at  $60^\circ$ , this is so both racks are angled at  $60^\circ$  from the central aisle. As a result of this, the BoxObject's BackFace is the side which actually faces the participant, with the FrontFace facing away from them. There were initial issues caused by this confusion when placing targets, which

delayed progress and required the additional step of pulling up the rendered layout for visual confirmation that the targets were properly oriented

```
393 # row 7 right
394 OpenGLObject
395     BoxObject 1260 78 58.8
396     Rotate 120.0 0.0 1.0 0.0
397     Offset 0 0 325.6
398     #Scale 0.01
399     #Color 0.7 0.7 0.0
400     MarkerRaster 150 6
401     MarkerSideRaster 6 7
402     TextureBackID 0
403     MarkerBackPosition 1 4
404 OpenGLObjectEnd
405
406 # row 8 right
407 OpenGLObject
408     BoxObject 1260 78 58.8
409     Rotate 120.0 0.0 1.0 0.0
410     Offset 0 0 325.6
411     #Scale 0.01
412     #Color 0.7 0.7 0.0
413     MarkerRaster 150 6
414     MarkerSideRaster 6 7
415     TextureBackID 0
416     MarkerBackPosition 50 4
417 OpenGLObjectEnd
```

Figure 12. Excerpt from a ( $60 = \theta$ ) right side rack with targets.

## 2 Phase One - Flat Racks

### 2.1 Questions

For Q1, we had created a model to estimate the exposure and intensity at which a shopper would view any part of a given rack within the layout, but we needed to verify the accuracy of this model. To test the model, we decided to focus specifically on the orientation of the racks for the initial experiment, settling on ( $\theta = 30^\circ, 45^\circ, 90^\circ, 135^\circ, \text{ and } 150^\circ$ ). Each layout would have

targets distributed through-out in spots of both we type, i.e. that believe should and should not be seen. Figure 13 below demonstrates the FOV being applied to a curved rack. The various  $\theta$  values would be used to test the orientation portion of Q2, but curvature and heights would be saved for later experiments.

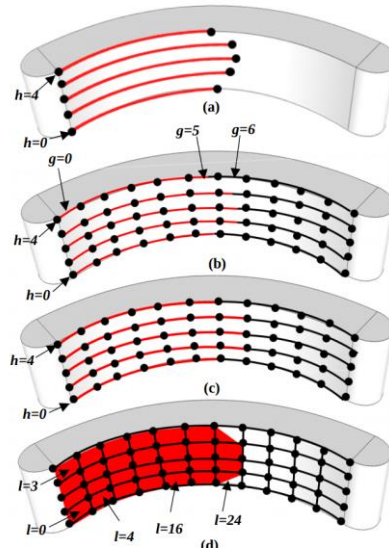


Figure 13. Exposure/intensity at step y [1].

To answer Q3, we first needed to determine how we would quantify this data. We determined the best way was to keep the participant restricted to moving down one aisle, only giving the participant control over when the movement through the environment begins. This way, the participant can focus solely on the task of finding targets, which keeps the data analysis simpler (as the participant's control over speed would add another component for analysis). To analyze this data, we would use the view direction to determine how far they turned their head from center, as well as how quickly they looked across the center aisle to look at the other side. This data is collected from a participant and graphed as seen in Figure 6. The data is then analyzed to determine how a participant's scanning pattern evolves throughout the experiment. After this, the data is compared to that of other participants with the same orientations, as well as



the scanning patterns of other orientations.

To quantify Q4, we decided all layouts would consist of 20 rows of racks, which would allow us to derive the model of exposure and intensity from the space requirements. We could quantify this both before and after the experiment with the model, then verify if our hypothesis on which orientation would perform the best holds true.

## **2.2 Experiment Setup**

With the orientations chosen, we next needed to decide on how the orientations would be tested. Factors to consider included the amount of time allowed for the experiment, the safety of the user, how to gather significant data from each orientation, and how the user would learn the controls. First, it was decided the experiment should not be longer than an hour, as continuous exposure to the 3D stereoscopic screens is when side effects typically appear. These side effects include headaches, dizziness, nausea, and eye strain.

To check on a participant's health, we carried out stability tests and health surveys (one set of each) to test the participant for possible side effects of extended exposure to the 3D stereoscopic displays. The stability test would include balancing on their non-dominant foot with their eyes closed while putting the dominant foot behind the other, crossing their arms, and close their eyes. We would determine how long they could remain in this position (up to 30 seconds). After the experiment, the stability tests would be run again to determine if the participant's performance declined from their initial ability. Additionally, we had the participant take a survey which listed the various symptoms of the exposure to 3D, which was answered on a scale (ranked from 1 to 10 or not applicable) of how severely they had experienced the given symptoms. Once again, this entailed a test both before and after the experiment, to verify the health of the participant.

Options on how to run the experiment ranged from having a participant experience every orientation to focusing on only one layout type. If a participant went through every layout, any bias of order in experiment would need to be accounted for, as participants would be expected to become more adept with the use of the system as they experience it. Another concern was that ( $\theta = 30^\circ, 45^\circ, 135^\circ$ , and  $150^\circ$ ) would all be new to participants, which introduces a risk of participants missing targets on a single run through a new layout (while adapting to the change). This could lead to these new angles and earlier-experienced layouts to underperform, resulting in skewed data.

From this, we decided we wanted to focus on only two layouts with a participant. A ( $\theta = 90^\circ$ ) and ( $\theta = 30^\circ, 45^\circ, 135^\circ$ , or  $150^\circ$ ) would be what a participant would experience. This meant that every participant would experience ( $\theta = 90^\circ$ ) allowing us to have each participant focus more on the new orientation. This was chosen because ( $\theta = 90^\circ$ ) is the standard orientation used in grocery stores, and therefore the layout most consumers would be familiar with. We could use ( $\theta = 90^\circ$ ) as a baseline to compare how each participant prepared in their given  $\theta$  opposed to ( $\theta = 90^\circ$ ). This could further quantify Q4 through their performance, as well as by giving them surveys asking them how they liked the new orientation compared it. Ultimately, we decided the ratio would be four ( $\theta = 90^\circ$ ), one of which would be training, and six of ( $\theta = 30^\circ, 45^\circ, 135^\circ$ , or  $150^\circ$ ), this gives us four test groups.

The next consideration in what order the layouts would appear was whether we should have the three non-training ( $\theta = 90^\circ$ ) spread out across the experiment or all at the beginning. An example experiment order could be: training, one ( $\theta = 90^\circ$ ), two ( $\theta = 45^\circ$ ), one ( $\theta = 90^\circ$ ), two ( $\theta = 45^\circ$ ), one ( $\theta = 90^\circ$ ), and two ( $\theta = 45^\circ$ ). This would have the advantage of keeping the ( $\theta = 90^\circ$ ) fresh in the participants mind when taking the surveys asking to compare the two layouts.

The downside would be that the participant would not have time to fully adapt to the ( $\theta = 45^\circ$ ), as they would be constantly reminded of the layout which they are already familiar with.

The alternative would be to run training, three ( $\theta = 90^\circ$ ) and six ( $\theta = 45^\circ$ ). This would have the benefit of having the participant learn the system with the ( $\theta = 90^\circ$ ) layouts, and all participants would experience the ( $\theta = 90^\circ$ ) at the same time without this experience being interrupted by another layout. With this, the ( $\theta = 90^\circ$ ) data from each participant could more easily be compared to one another, as they would all have the same level of experience within the system at that point. The time spent in each layout would not be equal to one another, as ( $\theta = 45^\circ$ ) would take less time than ( $\theta = 150^\circ$ ). Because all layouts contain 20 rows, the amount of space each layout occupied would not be the same. If we used the spread out ( $\theta = 90^\circ$ ) approach, ( $\theta = 45^\circ$ ) participants would have less time in the VE when they reached the final ( $\theta = 90^\circ$ ) than the ( $\theta = 150^\circ$ ) participants.

To confirm if the participant truly saw a target, we had two steps for confirmation. The first step entailed the participant pressing the “left” or “right” trigger when a target has been seen on the left or right side of the main aisle respectively. This method was expanded upon during the second phase of the experiment. The operator would have a list of all targets which appear in the given layout, and the participant would be required to verbally announce the rack number and side the target was found on. With this verification step, we were able to determine true positives, false positives, and determine which targets were missed.

The last item to consider is the movement. As previously stated, we determined that the user should be restricted to the center aisle, but how fast they should move was another concern. The speed in which the participant is moved through the environment had several factors that needed to be considered, with the first and foremost being the user’s ability to see the “targets.”

The other major factor was the goal of keeping each experiment to no more than an hour in length. This hour would not only include the experiment itself, but also the explanation of the experiment to the participant and getting consent forms signed, surveys on experience within the VE, an eye test, and training with the system. Finally, we had to include enough space for the participants to keep their head still for five seconds without them missing the target. This meant that the participant should start far enough from the entrance of the aisle, especially for rack placements with obtuse angles (i.e.  $\theta = 135^\circ$  and  $150^\circ$ ). Figures 14 below show the drastic difference between the “entrance of the aisle” in the acute angle as opposed to the obtuse angle.

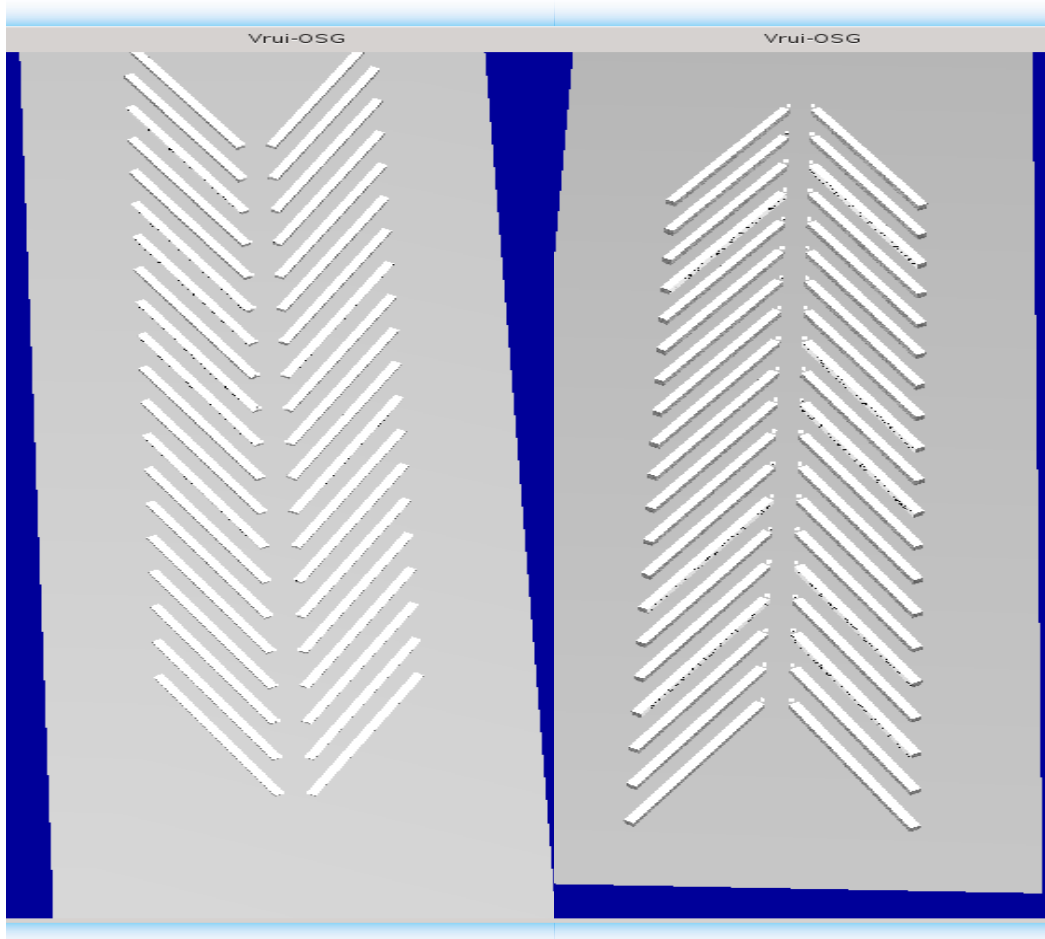


Figure 14. ( $\theta = 30^\circ$ ) Overview and ( $\theta = 150^\circ$ ) Overview.

## **2.3 My Contribution**

In this phase, I worked with the configuration files to create the layout that matched the parameters required to analyze the model of intensity and exposure. This would entail placing the targets where they supposed to be, determining the correction orientation and the distance between racks, and matching the length, height, and depth of all racks for each layout. I edited the graphical user interface (GUI) to have groups of radio buttons, each with an “execute” button under each group to render the selected layout. Each group would be one of the four patterns. The GUI also included a text box where the user could name the output file generated from the layout.

It was my job to allow the team to experience various speeds and starting locations to match these requirements. For ease of reading data, all participants would start 200” away from the entrance of the central aisle and would be moved at 3.33 fps. Overall, this increased the distance they would need to cover in each layout. These additional tasks left little time for running the participant through all nine layouts, which meant that participants would be moved through the environment at faster-than-typical speeds.

## **3 Phase Two - Curved Racks**

### **3.1 Questions**

To answer Q1 and Q3 we follow the same procedures as in phase one. The biggest change from the previous phase is the inclusion of curvature, which allows us to further investigate Q2. The height remains the same as in phase one, and ( $\theta = 30^\circ$  and  $150^\circ$ ) were excluded due to ( $\theta = 90^\circ$ ,  $45^\circ$ , and  $135^\circ$ ) performing better based on previous results. The layout orientations chosen were ( $\theta = 90^\circ$ ,  $45^\circ$ , and  $135^\circ$ ), with the addition curvature of ( $\alpha = 0^\circ$ ,  $30^\circ$ , and  $90^\circ$ ). Due to the increased space required for the curvature of racks, the number of rows was

reduced from 20 to 10, and the racks were about half of the length used in phase one. Figure 15 shows how the racks would look if they were not shortened and Figure 16 shows example of racks shortened.

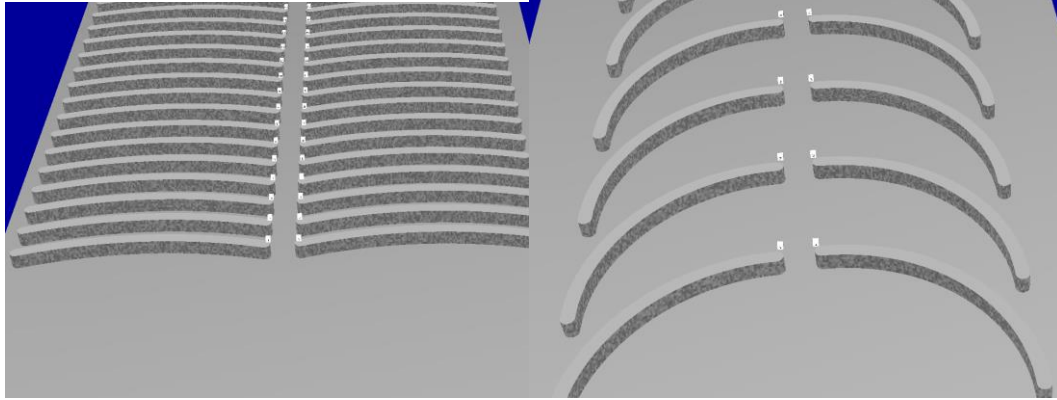


Figure 15. Before ( $\theta = 90^\circ$ ,  $\alpha = 30^\circ$ ) and Before ( $\theta = 135^\circ$ ,  $\alpha = 90^\circ$ ).

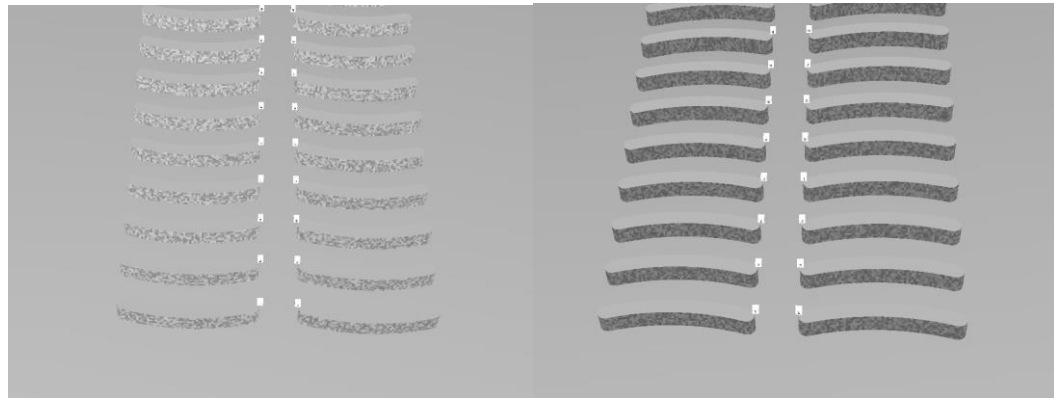


Figure 16. After ( $\theta = 90^\circ$ ,  $\alpha = 30^\circ$ ) mirror and after ( $\theta = 90^\circ$ ,  $\alpha = 30^\circ$ ).

### 3.2 Experiment Setup (Phase 2)

In this phase, all the same considerations as in phase one hold true. Factors which needed consideration still include the time of the experiment, the safety of the user, how to gather significant data from each orientation, and how the user would learn the controls. The additional concern when determining the experiment in this phase was how the curvature affects the data needed to be collected to get a full picture of a given layout.

To get full data on a layout with the new curvature, we were required to create two versions of each layout: the first with the original orientation, and the other a mirrored version of the previous layout. Figure 14 shows how ( $\theta = 30^\circ$  and  $\theta = 150^\circ$ ) are essentially reflections of each other. The curvature of ( $\alpha = 30^\circ$  and  $90^\circ$ ) results in a lack of equivalent layouts within the rest of the data set. Figures 16 and 17 show the examples of these mirrored layouts. To increase the realism of the environment, we included a verbal prompt for the participant to know they are going back down the same aisle and contains all the same target locations, with the aisle numbers now counting down from ten to one.

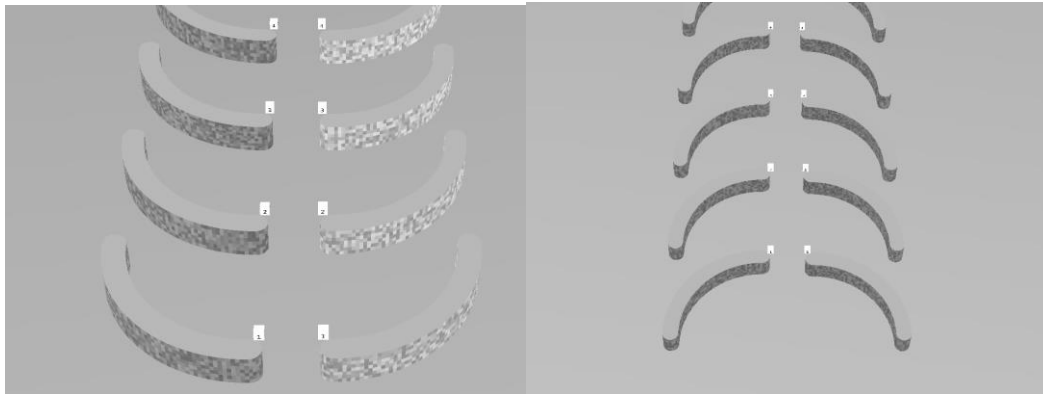


Figure 17. After ( $\theta = 135^\circ$ ,  $\alpha = 90^\circ$ ) mirror and after ( $\theta = 135^\circ$ ,  $\alpha = 90^\circ$ ).

This time, there are nine different types of layouts, meaning a participant would need to experience 18 iterations of a ( $\theta = 90^\circ$ ,  $45^\circ$ , and  $135^\circ$ ) each with a version with ( $\alpha = 0^\circ$ ,  $30^\circ$ , and  $90^\circ$ ). To run this experiment, we could have each participant experience one  $\theta$  with every  $\alpha$ , one  $\alpha$  with every  $\theta$ , or every ( $\theta$ ,  $\alpha$ ) combination. While the need for the mirrored runs would increase the time needed for any of our three options, the choice of  $\alpha$  as the separator would cause times to vary drastically between participants. ( $\alpha = 90^\circ$ ) layouts took up significantly more space than ( $\alpha = 0^\circ$  and  $30^\circ$ ) - see Figure 17 and 18 as an example of these differences in space. Along with the racks taking up more space, Table 1 lists the distances required between each rack. As can be seen, ( $\alpha = 90^\circ$ ) has more distance to cover and would require more time per run through,

which means we could not fit in as many run throughs in the hour time limit as opposed to ( $\alpha = 0^\circ$  and  $30^\circ$ ). The same issue is true with the use of  $\theta$  as a basis, except ( $\theta = 90^\circ$ ) would take significantly less time compared to the other two.

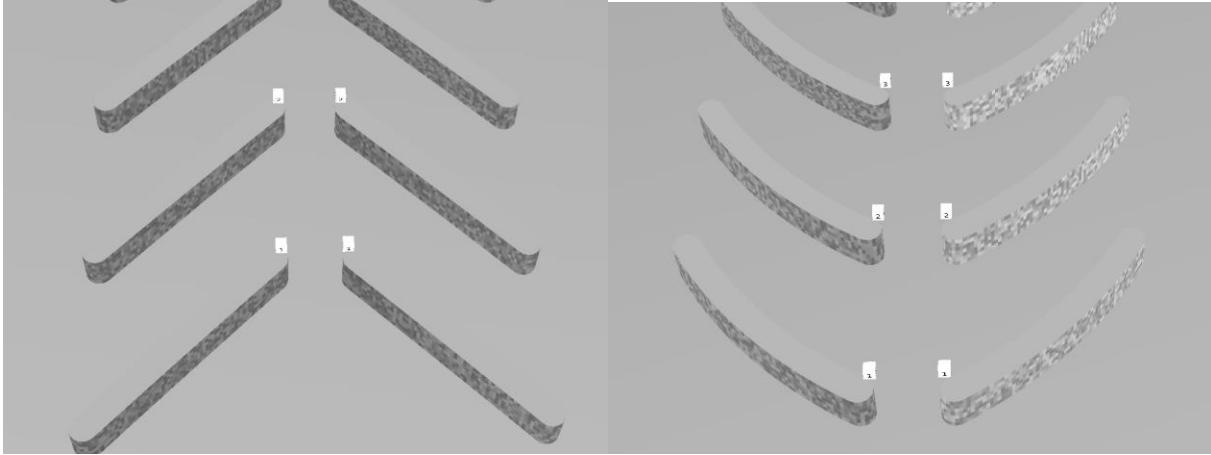


Figure 18. ( $\theta = 135^\circ$ ,  $\alpha = 0^\circ$  and  $30^\circ$ )

	$\theta = 45^\circ$	$\theta = 90^\circ$	$\theta = 135^\circ$
$\alpha = 0^\circ$	220.68"	156.0"	220.68"
$\alpha = 45^\circ$	262.56"	160.68"	262.56"
$\alpha = 90^\circ$	343.44"	191.52"	343.44"

Table 1. Distances between racks for every  $\theta$  and  $\alpha$ .

Based on the timing issue, it made the most sense this time around for a participant to experience every  $\theta$  and  $\alpha$  combination. Because we went in this direction, we then needed to consider the participants' experience within the system when determining how the layouts should be ordered in the experiment. Due to the concern of participants performing worse on earlier layouts in the experiment or being biased towards earlier racks, we set up nine patterns. Each combination would appear in every position, first to last, in one of the nine patterns, with the initial racks always starting with the racks curving away from the participant (Figure 15 being the example of that curvature).



### **3.3 My Contribution**

Due to the curvature of some racks, the numbers floating above them were positioned such that they could be cut off by the time the participant notices a target. Along with the user constantly switching between aisle orders of “one to ten” and “ten to one,” the participants of our pre-test cases would often become confused, needing to resort to mental math to determine which rack they saw the target on. With the participant constantly moving, they could also easily become confused on whether the numbers represent aisles or racks or perform the necessary mental math incorrectly. These factors could result in incorrectly identified targets (false positives) or missed targets (false negatives). Participants could also miss a target while in the process of announcing where they had seen the previous one. To ease this issue, I added the number of the rack on each end cap. The end caps would not be cut off the top of the screen like the numbers floating above, meaning that in the worst case the user would only need to subtract one to know what rack a target is on.

I felt that the previous training was a bit lacking, mainly due to a trend of participants not realizing targets could appear on the back of racks. To address this, I split the training into two parts. The first would now place the participant in the in the middle of a layout around rack six. They would not be able to move but could still look around to see the targets within the area. The second part followed the example of other training used during phase one, with the participant experiencing the type of movement and controls used in the experiment.

#### **3.3.1 CurBoxObject**

Originally, the plan was to reuse the pre-existing rack models with only the addition of curved end caps. I first worked on designing a circle object as a basis for these end caps, but it was too difficult to line these objects up with the existing racks in the environment itself.

The main challenge in this task was that the collaborator's model was designed with a smooth surface. In order, for targets to be implemented easily, it had to be adjusted such that vertices were spaced out much further apart.

When placing targets on the layouts on the previous phase of the experiment, I had issues remembering that the left side of the environment had the FrontFace and BackFace facing the opposite direction. This resulted in difficulty implementing the targets. Also considering how the curvature is implemented on the end caps, I chose to make the right side able to accept targets in the same manner that the left can. The end caps were also affected in that I needed to fully define the both left and right sides of the CurBoxObject. The BoxObject had only defined SideFace, which represented the left side, while the right, top and bottom faces were all just filled in with one big square. I decided to call the right side the FarSideFace in order for it to accept "target" information. I also retroactively added the FarSideFace to the BoxObject at this time. As a result of this, the left side was no longer only a rotated rendering of the objects on the right side, with the FrontFace now facing the participant. Some consideration was required to ensure that the end caps each had correct information, although this was much easier to keep track of as the object much smaller and not meant for use with targets.

The top and bottom faces could not be filled in the same way as in BoxObject. My options were either to recalculate all the vertices the faces would need to need to connect to, or to save all vertices in an array which is iterated through in order to render geometry. Recalculating the vertices required less memory and kept each face separated from the others and would also reduce the size of the parser file. Saving the vertices while they are calculated allows for the code to be more concise and to save time when rendering, as each object would not have to run two more rounds of the same calculations.

### 3.3.2 Unique issues with CurBoxObject

The main challenge in working with the curved racks was with ( $\alpha = 30^\circ$ ), the cut off for another square was just slightly off for ( $\theta = 45^\circ$  and  $135^\circ$ ,  $\alpha = 30^\circ$ ). This resulted in a column not being generated and a loss of alignment with the end cap. The layout with ( $\theta = 90^\circ$ ,  $\alpha = 30^\circ$ ), however, did not have this issue. To resolve this, I increased the amount each of columns to be generated, hard coding a check for the specific case with ( $\theta = 90^\circ$ ,  $\alpha = 30^\circ$ ), programming it to display one less column than before. This left a very slight overhanging edge, although this would be difficult to find even if someone is aware of its existence.

With the curvature of the racks and all necessary rotations, the starting point of a rack no longer corresponded to the exact coordinates that were programmed in. In order to resolve this issue, I created a PlaneObject with the a hard-coded distance to correct this discrepancy in the coordinates for the racks. The racks added after the first row would then be positioned using the predetermined offset values.

### 3.3.3 Playback Overhead View

The next major deliverable for this project was to design a program for viewing previous run throughs of the grocery store. To quantify the data, this would include a representation for the participant's head and FOV, which would briefly change colors when the participant had pressed a button. This would serve two purposes, the first being that it could be used to "replay" a participant's trial within the system. Additionally, a recording of a trial could be used to demonstrate what the DIVE is capable of without the need to be present in the lab. These replays would provide an additional means of confirmation on whether a participant had seen a target or not. Furthermore, the replay can be applied to any layout, and could reasonably tell us if the participant would have seen a target (had it existed in the layout they went through).

To implement this, I first needed to add the capability to read an output file containing all head movement data from the system. The system already had a method to create the name of an output file, which was connected to other parts such as the GUI. I opted to add a third input, which would require layoutConfigFile, dummyFile, and outputFile. This did not require further changes to existing code. As this feature would not be used as often, it is also advantageous in that the dummy files could be deleted easily.

After this, keyboard controls needed to be added to the system. The playback function used the “<” key to reset the playback position to 0, “>” to advance by one frame, and “?” to play the whole file.

To create a playback video, I would start with an existing layout and replace the target image with a fully red image, darkening the floor and background to make the racks and participant stand out visually. The representation of the participant’s head was set to bright blue, and a cone representing their field of vision was transparent grey. This cone would flash blue at times, indicating when the participant had pressed a button. The flashing would make confirming the participant's view of target when they would press the left or right button easier.

Each output file included data on which layout it was connected to, so I looked through the data and found images which would be of particular interest to watch. Primarily, these samples were images in which the participant was active and reasonably successful in the simulation, rather than simply not trying or not noticing the targets in the environment. After experimenting with the different screen recording options available, I determined which would be the quickest and easiest one to use.

## **4 Phase Three - Fully Populated Racks**

### **4.1 Questions (Phase 3)**

The aim was to give the user more control of their movement, as well as to use actual images of different products to populate all the racks. There were many of potential directions for the experiment that we were interested in, and we ultimately decided on taking a small sample size of participants in order to further refine what to emphasize in the project in the future. The different factors that would be considered were how realistic can the VE grocery store be made, the implementation of orientations of ( $\theta = 45^\circ$  and  $90^\circ$ ,  $\alpha = 0^\circ$ ) with no curvature on the end caps either and introducing racks both below ( $h = 42''$ ) and above eye height ( $h = 84''$ ).

With this goal in mind, Q1 would not be the main focus of this phase but could follow the same principle. Q2 would be addressed in a similar manner to the previous phases, just with added height and no curvature. For Q3, the data could now include the shopper's movement patterns (traffic), rather than only the scanning pattern of the participant. Q4 could be refined with how much information a participant could read with the exposure in mind. Overall, Q4 would not be a main focus of this phase either. A store the size of the previous phases would require thousands of unique images, where we would need to make sure each image is of good quality and is properly priced and formatted. This would not be feasible for the goal of a quick and small experiment, such as testing out potential future directions for the project. As a result, the VE grocery store would be much smaller than previous phases, because all images would need to be collected and processed.

The main focus of this phase compared to the previous two phases was to lay the groundwork to answer Q5, Q6, and Q7. In this phase, we aimed to answer a simplified Q5 -

“Can we promote participants to impulse buy a product?” For Q6, we would observe the performance of ( $\theta = 45^\circ$  and  $90^\circ$ ) of both heights and examine if one layout had more impulse purchases than any other layout. Finally, data used when answering Q3 can also be applied to Q7, with the additional consideration of how well a participant performed within each layout.

#### **4.2 Experiment Setup (Phase 3)**

With actual product images, we were able to begin to experiment with regard to impulse purchases. To encourage purchases, we had to give the participant an idea of how much money they had to spend. One approach was to give participants a specific amount of money and a shopping list. To keep the trial simple, we settled with telling the participant that they would have enough money to get two more items which were not on the list. This encouraged impulse purchases further, without the need to make the instructions for the participant more complex.

Targets were redefined within this set to no longer as specific points to look for, but rather items from the participant’s shopping list. Two or three items would be very specific, and the rest were given as general categories. Examples of each level of specificity might be "Mountain Dew," "ice cream," and "up to two additional items of your choice." To further immerse the participant, we provided scenarios to create a theme for the shopping list, such as last-minute shopping for a "birthday party," "barbecue," or "shopping for dorm move-in."

The training consisted of three parts. First, the participant was taught how to use the controls. Next, the participant would run through one of four possible layouts ( $\theta = 45^\circ$  or  $90^\circ$ ), with the racks either above or below eye height. The last part would then feature one of two layouts ( $\theta = 45^\circ$  or  $90^\circ$ ) which had not been used in the previous part, with both being above eye height.

The new movement system started with expanding upon the conveyor belt system of the

previous phases by giving participants controller over the speed. This would remain similar to the theoretical model, allowing analysis of Q1 to be fairly simple. Q3 could be expanded upon by seeing how a participant moves up and down an aisle, although this would cause other issues that needed to be addressed. The first of these issues is that of image visibility. This movement system would require us to make all images readable to the participant from the main aisle. This would reduce the realism of the product images and might require us to stick with the “target” based approach with an added distraction of other products. Due to this, a second issue which results from a “target” based search is that the impulse purchases desired would be difficult to encourage with a strict system restraining them from shopping as normal.

The movement system was therefore expanded to a free roaming model, which allows the participant to move around the VE on the 2D plane. This model has the benefit of allowing us to examine the traffic pattern a participant would follow for Q3, and not just how many times they move up and down the main aisle searching for the missing item. The downside is that the movement system no longer matches the theoretical model, meaning additional computations would be required to analyze the data. At the same time, this new movement system could be used to expand on the theoretical model to include new positions other than the main aisle.

The new free roaming movement system necessitated a different setup for the training. There would only be one set of training, but it would take up about as much time as one of the two trial runs. First, we designed an environment which included three ( $\theta = 90^\circ$ ) with two in the front and third in the back, one ( $\theta = 45^\circ$ ) and one ( $\theta = 180^\circ$ ) both within the middle of the layout. Two of these were below eye height, and three were above eye height. Six “targets” were spread throughout the layout, with a “target” now being a unique product image with a price. The rest of the racks were greyed out, unlike in the full trial where products would be in every slot.

## **4.3 My contribution**

### **4.3.1 Increasing image capabilities**

The way the system handled images on racks (as described previously) only allowed for one image per face in one square. At first, I considered expanding the array of images, but was concerned that doing so would not only load a very large number of image files into the configuration file, but also necessitate the assignment of coordinates to each image. This would significantly slow down the process of making changes to each layout, due to both the large file length and the conversion of each image into array indices, which requires the user to first find the correct image number in the list, and then find it again in the image names.

In order to avoid this problem, I decided to keep the current implementation by configuring the parser to populate the racks from a starting coordinate. This would still require only a couple image numbers (four per BoxObject), which is significantly less cumbersome than specifying one for each square. The next issue to address was that the system could not handle the workload of updating every picture on every rack while the user moved around.

While investigating this issue, the number of images correlated with the severity of the stuttering and freezing in the display. In order to mitigate this, the images were first stitched together into columns (an example can be seen in Figure 19 below), which were then stitched into one image for the face of a rack. The end caps were left to be populated by 4 columns which were not stitched together. This was to allow the columns to be usable in the curved rack implementations of future experiments. There were still noticeable performance issues with this setup, however. To fully resolve this issue, the face images were shrunk by 30%. This not only improved performance significantly, but also resulted in images which were still clear and readable by participants.





Figure 19. Example of half a column of a shelf.

#### 4.3.2 New Control System

While the images were being developed, I worked on improving the movement capabilities of the simulation. At the same time, I wanted to make sure the original movement options were available as well. Initially, I had implemented a control for the participant to increment and decrement the speed, which could even allow them to move backwards through the environment. The main challenge for this was determining what buttons on the controller were not already reserved by the system or already used in the previous experiment. With this in

mind, there were no intuitive buttons to use for the speed manipulation, which made this movement type less ideal.

My second approach was to use the joystick, allowing for the completely free movement of the participant. One of the reasons to push for this is because it aligned much better with the new experiment goal of gathering information on Q5, Q6, and Q7. Now we could focus more on how the user moved through the system and not just scanning patterns of the participant. We could further immerse the participant in our scenario when they are not restrained to one aisle, and there would not need to be a concern of readability of products further away from the center.

I got access to the system code with similar functionality after making tweaks, such as preventing the Y value from increasing as the participant moved and correcting what the system considered “positive” and “negative” movement to be consistent with the system already in place. With this option, the participant had much more freedom: they would only move while pressing a button, rather than being required to press less-intuitive buttons multiple times to adjust their speed incrementally. Additionally, the direction considered to be “forward” would be based on the direction the participant is facing in the system. With this implemented, the participant could even go into the side aisles, rather than being restricted to the main one.

The last hurdle for this adjustment was that the three-wall system would cut off racks which were located behind the participant, resulting in a less immersive simulation overall. In order to correct this, I attempted to add the capability to rotate the system around the participant. With the left and right buttons already in use, I opted to use the left and right inputs of the joystick as the trigger for rotation. One issue that came up in doing this was that the rotation I had based the system on did not have the right axis of rotation for the environment. With the shorter racks, the error was not caught immediately due to the fact that a participant would not

stray far from the axis. This resulted in the appearance that the issue where the participant would spiral out from the given point, moving the participant further or closer to the correct axis of rotation, making the problem vary widely in scale. Due to the manifestation of the problem was not immediately connected to the axis issue, but the issue was resolved.

### 4.3.3 Creating images

To create images for the system, other team members collected images of cereal boxes for use as a test set. For this set, no prices were added, and each image would fill up the entire box given. It was decided that this was visually too overwhelming for the participant, and a different format had to be used for the images. Figure 20 displays the difference of the initial cereal images, (displayed on the end cap of the second rack on the right) and the rest of the images given the additional space between products.

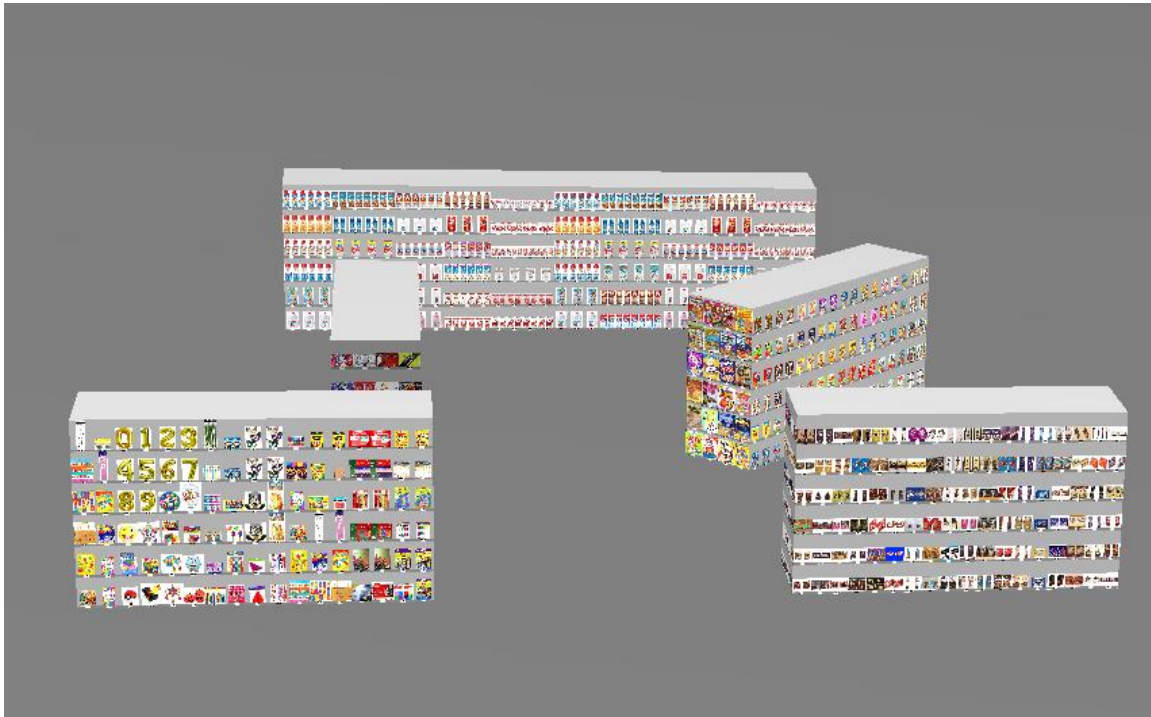


Figure 20. Sample Training layout side view.

I chose to use GIMP to create the template, not only because it was free and I had prior experience working with it, but because it could convert the images to the format (.ppm) required for the system to read the images. I created the basic grey background to mimic the general color that would be produced for a square in the usual program, and a darker border on the bottom with a blank white rectangle in the center. I created layers with different prices, which could easily be hidden or edited if a price was not already included. This way, the spacing and font for all prices would be consistent for every image. I included many guidelines that the product image can be easily resized to. Figure 21 shows the template with the guidelines on. The amount of space each takes up depended on the product - some would be stacked up or multiple included side by side to more accurately represent how the given product would appear inside an actual store, examples of both can be seen in Figure 22.

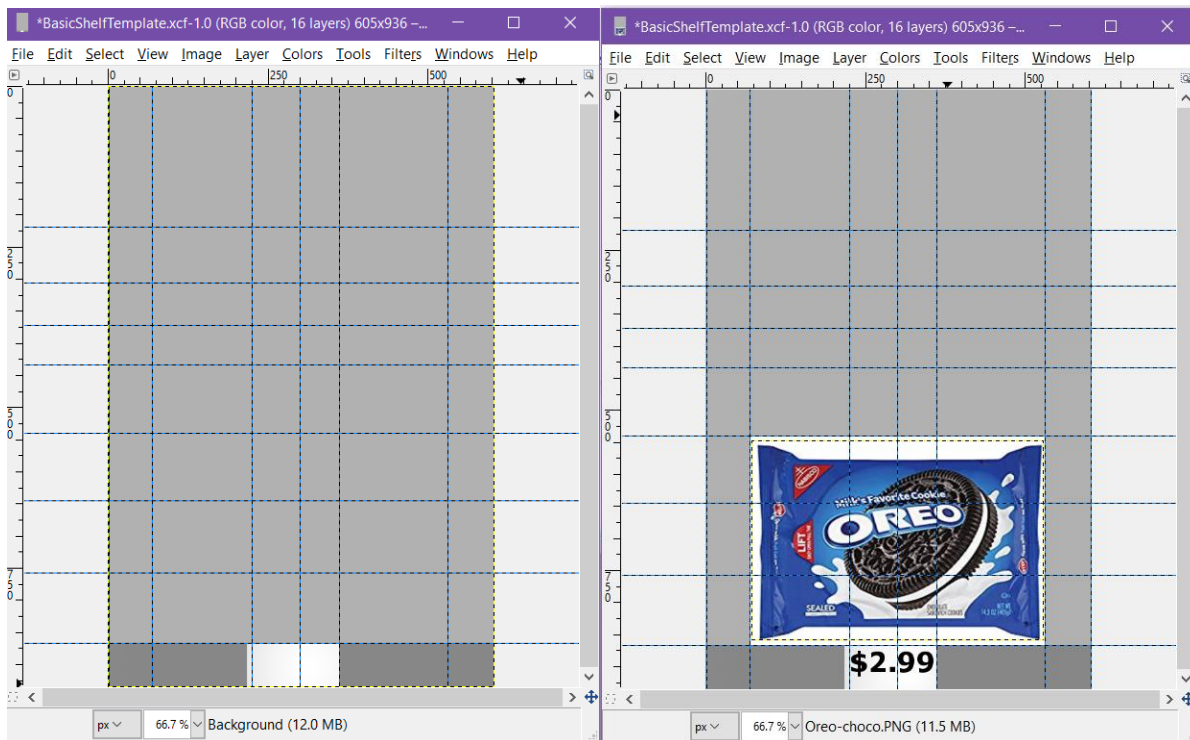


Figure 21. Basic shelf template with and without a product image.



Figure 22. Example of a slot shown with different stacking options.

## 5 Additional Research

### 5.1 Why Head Mounted Displays?

With the re-creation of a realistic virtual store underway, I decided to investigate the possibility of expanding the experiment VE to utilize an HMD. One main reason for this would be the less restrictive cost of HMDs compared to a DIVE. The DIVE costs about \$120,000 [6] to make. The two potential HMDs I considered were the Oculus Rift First Devkit and the Vive. A professional HMD with all necessary equipment (i.e. Vive PRO) could cost up to \$1,399 [9]. This does not include the cost of the devices needed to run the VE but is a significant decrease in price when compared to a DIVE.

The HMDs have even fewer space requirements compared to the DIVE and would be more easily transported to different locations for testing. If grocery stores wanted to test out a new store layout in a specific region, they could move the system with little hassle to that area,

allowing for the use of local residents as participants. Besides portability, HMDs have the advantage of having access to the 360° environment without out the need to program the controller to allow the participant to turn.

## **5.2 The Vive**

The Vive featured an adjustable headset, FOV of 110°, capability to utilize space up to 15' by 15', and the “Chaperone” system which allows user to see boundaries while within the system [9]. Additional features of the Vive include controllers designed to work well within the virtual environment and wiring layouts designed to minimize the risk of the user tripping while in the virtual environment, as seen in Figure 23. Specifically, the sensors used with the system are wireless, and the wires of the headset are threaded to be positioned behind the user. The downsides include the increased weight of the HMD (as a result of the built-in sensors) and controllers based around teleport transportation than a consistent motion (which can cause motion sickness for some users). Furthermore, the Vive would require additional code to allow the system to read and write data in a similar way to the previous design, as well as to integrate the controller which is packaged with the Vive. Finally, the Vive requires Vrui 4.6, which was incompatible with the current set up of the DIVE (which used version 4.2). As phase three of the experiment was still in progress, it was not possible to update the system to version 4.6.



Figure 23. Vive HMD [9].

### 5.3 The Oculus Rift First Dev Kit

The Oculus Rift first devkit features FOV 90° horizontal, 110° diagonal, a 7" 1280x800 screen, and adjustable head straps. Unlike the Vive, the Oculus does have internal tracking which is compatible with the DIVE's tracking system. This tracking allows for data output without additional design, but has a cord positioned in a manner that increases the risk of tripping for the user. This cord can be seen in Figure 24. The same controller can also be used, although it was not designed specifically for the HMD as the Vive's controllers were, making it more difficult to use with the HMD on.





Figure 24. Oculus Rift First Devkit HMD.

The last major downside of the Oculus Rift is that the First Devkit is no longer on the market. As a result, systems utilizing it which are to be marketed for commercial use would require the additional work of being ported to another viable HMD. As the Vive was not a viable option (due to the incompatibility with Vrui 4.2) at that point in time, I was forced to settle on the Oculus for creating a proof-of-concept for the HMD implementation. Unfortunately, a major issue emerged which prevented the free movement system from working: rather than moving on the intended plane, the system would leave the user fixed to one location, instead rotating them around that point. This led to another issue where the only possible solution seemed to require changing the current design of the system itself, which prevented the design of this proof-of-concept from moving further ahead.



## **5.4 Future of HMDs with the Experiment**

With either HMD, adjustments were needed to the code which had risk of clashing with the current experiment, as either Vrui needed to be updated or controls needed to be adjusted to work with to work with the HMD. Other issues would also need to be resolved for an HMD to be viable within this experiment, including motion sickness risks and collecting participant data when exploring the VE with an HMD. The motion sickness would be a major concern with the type of movement required to gather all the data needed for the experiment. The Vive's effects on motion sickness were tested with both walking and running, with all participants being affected (whether to a minor or severe extent) [10].

Our experiment requires the “walking” style of movement, which was tested, and this concern would need to be addressed if an HMD was used for the experiment. While our experiment could benefit from the additional immersion, the motion sickness and additional work required to get an HMD working with the system does not seem to be worthwhile for our team at this time. With us already having the DIVE fully operational, a HMD would not bring anything significantly different to the table, that we do not already have with the DIVE. An HMD may have a place in the experiment in the future, but improvements with reducing motion sickness or a different focus which would allow for a teleportation-based system would likely be required.

## **6 Conclusion and Future Research**

In conclusion, we found that the DIVE works very well with the experiment, with the benefits greatly outweighing the drawbacks. The VE can be further improved, however, by reducing rendering time of fully populated racks and reducing lag in larger environments. In terms of the research questions posed by our team, Q1 and Q4 were thoroughly answered in

phases one and two. Q2 requires further research into height's effect on exposure to be more conclusively answered. The scanning pattern aspect of Q3 has been heavily researched in the previous phases, but more information on shopper traffic is required to answer this question. In phase three, we determined that it was possible to have participants make impulse purchases, indicating that Q5 can be quantified in future phases of the experiment. This would likely require a larger virtual environment compared to that used in phase three. Q6 and Q7 can also be expanded upon in future phases, most likely with larger virtual environments as well.

## **6.1 Summary of Contribution**

My role started with familiarizing myself with the system and designing rack layouts with targets in the designated locations. The framework already existed for flat racks, but curved racks needed to be designed from scratch. While a model had been designed for curved racks, I needed to design the necessary code to properly render these in the environment, as well as to allow placement of targets on curved racks. I also implemented the capability of watching a previous participant's experience within a given layout by recreating it with the data collected from their run through the VE. Overall, my design process in the work I completed in this experiment was to allow for both backwards compatibility and for a level of simplicity such that a new researcher could more quickly learn how the system works. This significantly expanded our abilities to interpret the data we had been collecting. I also made key design decisions regarding the implementation of faces with more than one target, control configurations for the participant interface, and the design of a visual "product image" template. I provided advice regarding what the system could and could not do, giving input on what I had considered to be the best way to collect the data we needed.

## 6.2 Phase Four and Beyond

This experiment has a multitude of directions to expand in, such as those which emphasize significant aspects of phase three. These aspects include the heights of racks, cutting out variable heights and focusing solely on impulse purchases, or the introduction of fully populated curved racks. Curved racks will have a challenge of running smoothly when fully populated, therefore it might be better if a version of the CurBoxObject was made which could have curved end caps where the main faces would be ( $\alpha = 0^\circ$ ), but where the FrontFace and BackFace would be based on those of a BoxObject. This would allow for the mitigation of this performance issue, by designing the main faces for one solid image, instead of the column-based approach.

Another option is to run an experiment similar to those in phases one and two, with the free roaming system enabled. This could be used to expand upon the theoretical model and analyze exposure from within racks (not just from the main aisle). The way in which participants go about searching for targets when given full control of their movement would also be a point of analysis for these phases. Putting a greater emphasis on the realism of the environment is another possibility. In its current design, the store is similar in style to the Tesco Virtual Store, but it would be possible to increase the environment's realism by implementing interactive models, similar to those in the experiments on produce shopping and VR-aided rehabilitation. Introducing obstacles is also a possibility, which might be implemented as include a BoxObjects with images of shoppers which can be placed in the middle of an aisle. We could also work towards implementing walls, a store entrance, and a checkout area.

Additionally, fully populated racks allow for recreating experiments from phase one, except with multiple targets on the same face. This feature was not implemented phase one was

carried out and could add another layer to that experiment. Lag would be less of an issue in this case, as only rack faces with targets would require an image.

Furthermore, the video playback feature can be expanded to fully integrate with the new movement type introduced in phase three. The main issue for this currently is that the rotation is applied to the environment, rather than the participant. In order to correctly implement this, the rotation will first need to be read, which would then be reversed and applied to the head's cone.

## References

1. Guthrie, B. and Parikh, P. J. (2017). Analyzing a shopper's visual experience in a retail store in 3D. In IISE Annual Conference. Proceedings (pp. 746-751), Pittsburgh, PA.
2. C. E. Levy, D. M. Miller, C. A. Akande, B. Lok, M. Marsiske, and S. Halan, "V-Mart, a Virtual Reality Grocery Store," *American Journal of Physical Medicine & Rehabilitation*, vol. 98, no. 3, pp. 191–198, 2019.
3. C. Lombart, E. Millan, J.-M. Normand, A. Verhulst, B. Labbé-Pinlon, and G. Moreau, "Consumer perceptions and purchase behavior toward imperfect fruits and vegetables in an immersive virtual reality grocery store," *Journal of Retailing and Consumer Services*, vol. 48, pp. 28–40, 2019.
4. J. Strother, "Shopping by phone at South Korea's virtual grocery," *BBC News*, 21-Oct-2011. [Online]. Available: <https://www.bbc.com/news/business-15341910>. [Accessed: 25-May-2019].
5. P. Desmet, R. Bordenave and J. Traynor, "Differences in purchasing behaviour between physical and virtual laboratory stores", *Recherche et Applications en Marketing (English Edition)*, vol. 28, no. 2, pp. 70-85, 2013. Available: 10.1177/2051570713487473.
6. T. Wischgoll, M. Glines, T. Whitlock, B. Guthrie, C. Mowrey, P. Parikh, and J. Flach, "Display Infrastructure for Virtual Environments", *Electronic Imaging*, vol. 2018, no. 1, pp. 060406-1-060406-11, 2018. Available: 10.2352/j.imagingsci.technol.2017.61.6.060406.
7. C. Cruz-Neira, D. Sandin, and T. DeFanti, "Surround-screen projection-based virtual reality: the design and implementation of the cave", in *20th annual conference on Computer graphics and interactive techniques*, 1993, pp. 135–142.
8. B. Guthrie, P. Parikh, T. Whitlock, M. Glines, T. Wischgoll, C. Mowrey, J. Flach, and S. Watamaniuk, "Comparing and Enhancing the Analytical Model for Exposure of a Retail Facility Layout with Human Performance.", in *IISE Annual Conference*, 2018.
9. "VIVE Virtual Reality System", Vive.com, 2018. [Online]. Available: <https://www.vive.com/us/product/vive-virtual-reality-system/>. [Accessed: 11- Dec- 2018].
10. Suarez, A. (2018). *How and why our experiments with virtual reality motion made us ill*. [online] VentureBeat. Available at: <https://venturebeat.com/2018/02/27/how-and-why-our-experiments-with-virtual-reality-motion-made-us-ill/> [Accessed 11 Dec. 2018].