PROJECT SPEARHEAD

POST MORTEMS, FORMS, AND DATA

Table of Contents

Synopsis:

In the spring of 2013, after the Oculus Rift had successfully been kickstarted and had begun to ship to backers, Austin Booker, Alex Loughran, Frank Hamilton, and Tushar Arora pitched an idea to the faculty at the Entertainment Technology Center in CMU. They wanted to spend a semester experimenting with several small Oculus Rift prototype games.

At the time, most of the excitement surrounding the Oculus was focused on ports of existing games, like Team Fortress 2 and Doom 3 and on a few smaller indie titles being developed for the device; all of which were first person titles.

The basis of Project Spearhead was, at the very beginning, a desire to figure out what else the Oculus was good for, besides first person gameplay. Our main goal was to create games that defied the current model of Virtual Reality gaming, which we defined as a strict adherence to realistic immersion and first person POVs. The hope is that by creating more abstract games for the Oculus we can open up and increase its potential.

To accomplish this in only 16 weeks we began by testing with the device to learn what we could before starting on our prototypes. These tests were separated into 4 categories: Gesture Tests, Art Tests, Control Variation Tests, and Motion Tests. Each category had 3 or 4 tests that asked a specific question about the capabilities of the device and it's usability in certain situations. Over a 6 week period each test was designed, built, and tested using at least 10 users, except for the Motion Tests.

This document serves as the results and conclusions for every one of our tests. Additionally we've included short descriptions of the different tests and the questionnaire sheets that we utilized to gather our data. We encourage anyone who is so inclined to continue our research or refine our methods.

Project Spearhead
Post Mortem - Gesture Tests

Note: The testing done by the Spearhead team is by no means scientific. We aren't researchers; we're game designers attempting to learn as much about a new device as possible in a 6 week crunch period. As a result much of our data is incomplete and many of our conclusions rely on anecdotal evidence. Please feel free to disagree with our conclusions or better yet, continue our testing and refine our methods and results.

Overview

The Oculus Rift Developer Kit is able to track a user's head movement in three directions of tilting. The best descriptors of these motions are tilting forward and back, shaking left and right, and head-tilt-sideways. And while the Oculus cannot track positional motion data, that is motion in the X, Y, Z coordinates, (aka, if you step forward without tilting or shaking your head in anyway, the Oculus won't notice a chance) sensing tilt is should be enough to recognize certain movements of a user's head and create controls based on their gestures. Our 'Gesture Tests' are designed to answer whether it's possible to create working gestures with the Oculus's APIs and whether or not those gestures will be accurate enough and comfortable enough for its users.

The four tests we conducted were: “Gesture Speed Differentiation”, “Gesture Resolution”, “Gesture Fatigue”, and “Camera Transition and Gestures as Controllers”.

Test #1
Gesture Speed Differentiation

Hypothesis:

The Oculus Rift's ability to detect head movement is precise enough to detect variations in the user's gesture (shaking) velocity.

Description:

The Oculus will be in a locked camera position facing two lights that are shut off, one blue, one red. The red is activated by a slow head nod, the blue by a fast head nod. After a period of time getting used to the different speeds and which speed ignites which lights, the test will begin. A text prompt will appear asking the user to activate one of the lights. The user will attempt to do so. This will be repeated 10 times and a score out of 10 will be recorded.

Results:

User 1: N/A (The code broke)
User 2: 6/10
User 3: 10/10
User 4: 9/10
User 5: 8/10
User 6: 8/10
User 7: 8/10
User 8: 2/10
User 9: 10/10
User 10: 4/10
User 11: 8/10
User 12: 8/10

We chose to discount the result of User 1's trial; the error was the fault of our application and not the user. In that case, the average of the data is 7.36.

Test #2
Gesture Resolution

Hypothesis:

The data involved in tracking head motion on the Oculus is precise enough to allow the user to make miniscule adjustments with their head.

Description:

The user will be in control of the forward motion and the rotation of the shape in front of them. Using the Oculus they will have to rotate the shape to match the hole in the wall in front of them. Forward motion will be handled by 'W' and 'S'. They will have to complete this exercise with four different shapes, and this will be timed. They will also complete the exercise without the Oculus, using 'A' and 'D' to control rotation; this will be compared against the Oculus time.

Results:

	Test 2: Tilt Res (Oculus) in secs	Test 2: Tilt Res (Keyboard)
User 1	35	39.7
User 2	118	95
User 3	84	85
User 4	86	79
User 5	105	90
User 6	52	30
User 7	25	35
User 8	159	109
User 9	116	86
User 10	134	108
Average	91.4	75.67

Test #3
Gesture Fatigue

Hypothesis:

Using the Oculus as a gesture detecting input device will cause the user discomfort over time, especially for common actions, such as opening a door.

Description:

The user will be moved through a hallway at a set speed, they encounter a door that needs to be opened with a head gesture (left to right nod). This door is repeated 15 times and they will only be able to open the door when they stop at it. If they do not perform the correct gesture at the right time they will be stuck until they successfully open the door. The user will be timed through the entire course and the number of times they get stuck at a door (for performing a gesture erroneously) will be recorded.

Results:

	Total Time in secs	Times Stuck
User 1	120	3
User 2	50+	14
User 3	66	2
User 4	65	1
User 5	DNF	-
User 6	93	5
User 7	92	1
User 8	80	0
User 9	90	3
User 10	DNF	-
User 11	50.4	1
User 12	83	5
Averages (not counting DNFs)	82.16	3.5

Test #4
Camera Transition and Gestures as Controllers

Hypothesis:

It is possible and comfortable to use gesture inputs in tandem with the standard Oculus camera head-tracking, by creating instances when the Oculus transitions, in-game, from camera to gesture input device.

Description:

Users are allowed to walk around a small room freely and use the Oculus to look around. On one end of the room there is a door with a simple ‘puzzle lock’ on it. When users get close enough to the door, the game forces the camera to move around and lock onto the puzzle. Now, the Oculus no longer can be used to look around, but instead to move a block around within the puzzle to solve it and open the door. This will be a more subjective trial as success or failure will largely depend on the reported comfort of the users.

Results:

For this test we were unable to collect large amounts of objective data, but out of 12 subjects 11 were able to find their way out of the room. The one who was unable to open the door lock expected the unlocking mechanism to respond to head-tilt-sideways movements instead of twisting left and right. Beyond that we found that 7 of the 12 people were able to operate the lock without any prompting. And finally only 2 out of the 12 participants found the forced camera move to be uncomfortable or nauseating.

Gesture Tests Conclusion:

The major focus of these tests was to explore the possibility of using the Oculus as an input device. Could your primary display also work as your controller? The short answer is yes, the Oculus is perfectly capable of being implemented as a controller, especially when you only look at the usability of the data that comes from the device. When you take the view into account the answer gets more complicated.

In our first test we asked whether or not the Oculus provided the type of data that would allow us to track differentiations in velocity for the user. We learned that this is completely possible, 100%, but it can be difficult for the user to utilize this type of input. The first issue we noticed was that it took a few attempts to understand what speed of ‘head shaking’ was needed to activate each light. Fast and slow head movements mean different things to different people and many of our users had trouble staying consistent, at first. After they fell into a rhythm and began to get comfortable with the concept of head shaking as an input we invited them to start their 10 official attempts to activate the blue and red lights. The final result was an average of 75% accurate attempts. Before we move on we’d like to mention that during our prototyping phase we often use a head shake input to begin a game or advance through a start screen. It takes some experimentation to find the right speed that is both, slow enough to be easily reached by the average user and quick enough that users don’t accidentally skip through a menu option as a result of idle head movements; but in our experience this is a perfectly acceptable mechanic.

Our second asked the question of whether or not the Oculus was capable of head tracking accuracy on a minute level. The data we got showed that going through our test course using Oculus controls was, on average, 15 seconds slower than using traditional keyboard controls. We want to bring up that during our testing we had an issue where some of the obstacle required the users to turn their heads greater than 90 degrees (not an easy feat) and this arguably, lead to slower times. This issue has since been fixed in later versions of the test, but we have not done any formal data collection with the new version. What we’ve learned, regardless of whether or not Oculus control is ‘slower’ than keyboard control, and is that the Oculus provides an impressive amount of head tracking accuracy. Users are able to hold their position steady within a few pixels and match the angle of an in game object with their head orientation with ease, and more importantly without discomfort (though we must warn, games with head-tilt-side mechanics such as this test employees, are highly susceptible to simulator sickness).

The gesture fatigue test gave us perhaps, the most dichotic data in this category of tests. We asked if repeated physical motions, in the case of the test, shaking your head, to complete a task, would become uncomfortable overtime. The answer is, without a doubt, yes. Out of 12 users we had 2 complain of discomfort and elect not to complete the test. Other complained of nausea, headaches, and that the Oculus would shake around on their head and the eyepieces would smash into their noses. Despite this, the accuracy of those that did complete the test was remarkable. In our testing of those who completed the test, users got 'hung up' on a door only an average of 3.5 times out of 15 doors. If it's possible to operate that accurately under uncomfortable scenarios we feel that with a better motion (something not as violent as head-shaking) repeated gestures could serve as a unique mechanic, especially if used sparingly.

Finally our last test was really asking two questions: could users go from using the Oculus as a first person camera to using it as a gesture based controller, and would a forced camera move during the transition of camera to controller be uncomfortable. The former question was informed by the fact that 11 out of 12 of our users were able to leave the room. All of our users understood that when directed to the door lock the Oculus's role in the world had changed and was no longer a camera. Only one was unable to figure out that they needed to look up and down, left and right, to move the mechanism around. This is arguably a result of our poor directions and in-direct control; the subject did try several times to operate the mechanism through 'head-tilt-sideways' movements before giving up. In regards to our second question, whether or not a forced camera move was nauseating to the user. This is something that has been discussed at length in reference to the Oculus Rift's usability. It's often remarked that any movement of the in-game controller, not controlled by the user is a terrible idea. We find this not to be the case at all. In almost every case the user didn't realized that the camera was moving, only that they were now locked on the door lock object. Only 2 users took issue with the camera movement, leading us to believe with some careful animation and play testing you could move the camera a great deal without upsetting the user.

Project Spearhead
Post Mortem - Art Tests

Note: The testing done by the Spearhead team is by no means scientific. We aren't researchers; we're game designers attempting to learn as much about a new device as possible in a 6 week crunch period. As a result much of our data is incomplete and many of our conclusions rely on anecdotal evidence. Please feel free to disagree with our conclusions or better yet, continue our testing and refine our methods and results.

Overview

The Oculus Rift is unlike any display on the market today. Not only does it (at least in its current 'dev kit 1.0' iteration) suffer from a limited resolution, just 1280x800 total or 600x800 per eye; but it also has to contend with the fisheye effect forced on it by its lenses. Another issue that must be taken into consideration is that the 3D effect often makes it difficult for the user to calibrate the device just right, leaving them with a slightly blurry or out image of focus. Because of these limitations we decided to investigate exactly what was visually possible when designing for the Oculus. Each of these tests would be completely subjective and the data would be gathered directly from the users. Our first experiment was the Color Test which dealt with the peculiar situation where large amounts of certain colors used on the Oculus would cause discomfort in the user. Our second test focused on how small you could make text on the Oculus before it became unreadable; furthermore it explored different font types and whether they would impact readability. The next issue we tackled was 'text placement'; basically we tested which areas of the Oculus's screen would cause problems for text. And lastly we experimented with several styles of textures and whether or not they would affect perception in a 3D environment.

Test #1
Color Test

Hypothesis:

Certain colors at certain brightness levels will cause an Oculus user discomfort.

Description:

The user will be testing Gray, Blue, Teal, Green, Yellow, and Red variants all at 5 different levels of saturation and brightness. Each color will be placed on a plane in Unity in front of the camera. A paragraph of text will be applied to the plane and the test subject will be asked to read it. Different shades of gray will be used to ensure the text contrasts the base color properly. The user will report on whether the background color causes them discomfort and this will be recorded.

All of the different colors that were used during this test.

Results:

Color	**# of users who found it uncomfortable (of 12)**
Blue	
000033 (Darkest)	0
000066	1
0000aa	6
0000ff	2
969bff (Lightest)	2
Gray	
333333 (Darkest)	0
666666	1
aaaaaa	3
dddddd	2
ffffff (Lightest)	3
Green	
003300 (Darkest)	4
006600	6
00aa00	3
00ff00	5
79ff92 (Lightest)	4
Red	
330000 (Darkest)	3
660000	6
aa0000	8
ff0000	7
ff7f76 (Lightest)	4
Teal	
003333 (Darkest)	1
006666	3
00aaaa	4
00ffff	9
aaffff (Lightest)	3
Yellow	
333300 (Darkest)	1
666600	3
aaaa00	5
ffff00	7
fffe71 (Lightest)	4
	3.67 Average

The standard deviation of the data set was 2.28 so with rounding any color that received 6 or more reports of uncomfortable is considered to cause issues with Oculus users; any color that received less than 1.5 reports, so 1 or 0 reports is considered to be a great color to use with the device.

Test #2
Font Size and Type

Hypothesis:

Compared to a normal screen at the same resolution as the Oculus, font choices and sizes will be more limited.

Description:

The user will be shown a series of text walls, each with a different sized font on it. All the letters are spaced equally and are written in Droid Sans on a 1280x800 texture and placed on a plane in unity so that the entire texture is just visible to the Oculus Unity camera. The font starts at 100pt and shrinks to 10pt. The user will read off the letters to the best of their ability and the accuracy will be recorded. The second part of this test is a single texture, 1280x1280 with several fonts on it. Each font is written at 30pt size and the plane will be placed at the same distance as the Font Size plane. The user will attempt to read each sentence and report whether they are struggling to read the words, unable to make sense of the letters, or can read everything with ease. The Oculus camera will scroll vertically so that each sentence will be exactly in the center of the Oculus Camera when being read, this will eliminate any blur from the OR's fisheye lenses.

Results:

Font Size	**Average**	**Percent Missed**
100pt (Out of 15)	15.00	0.00%
75pt (Out of 16)	16.00	0.00%
50pt (Out of 24)	23.83	0.69%
30pt (Out of 21)	20.83	0.79%
20pt (Out of 29)	27.50	5.17%
15pt (Out of 14)	11.17	20.24%
12pt (Out of 18)	12.08	32.87%
10pt (Out of 12)	3.08	74.31%

Font Type	**# users who had trouble reading (out of 11)**
Droid Sans Mono	2/11
Times New Roman	1/11
Arial	0/11
DJ Gross	7/11
Alien Encounters	2/11

Papyrus	1/11
Old English Text	0/11
Dunkin Sans	0/11

Test #3
Text Placement

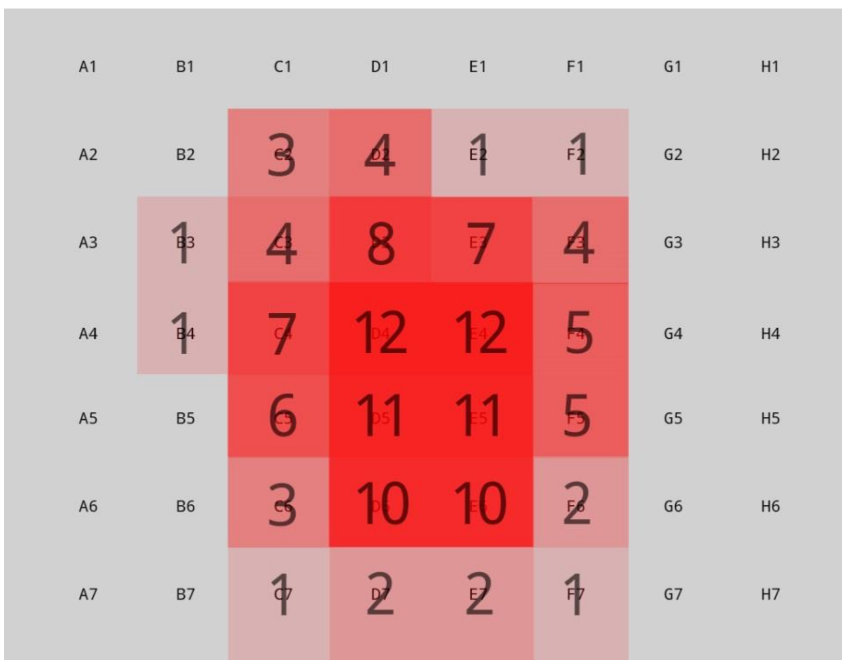
Hypothesis:

Tilting the plane on which text lies towards the Oculus's in-game camera makes it easier to read text nearer to the edge.

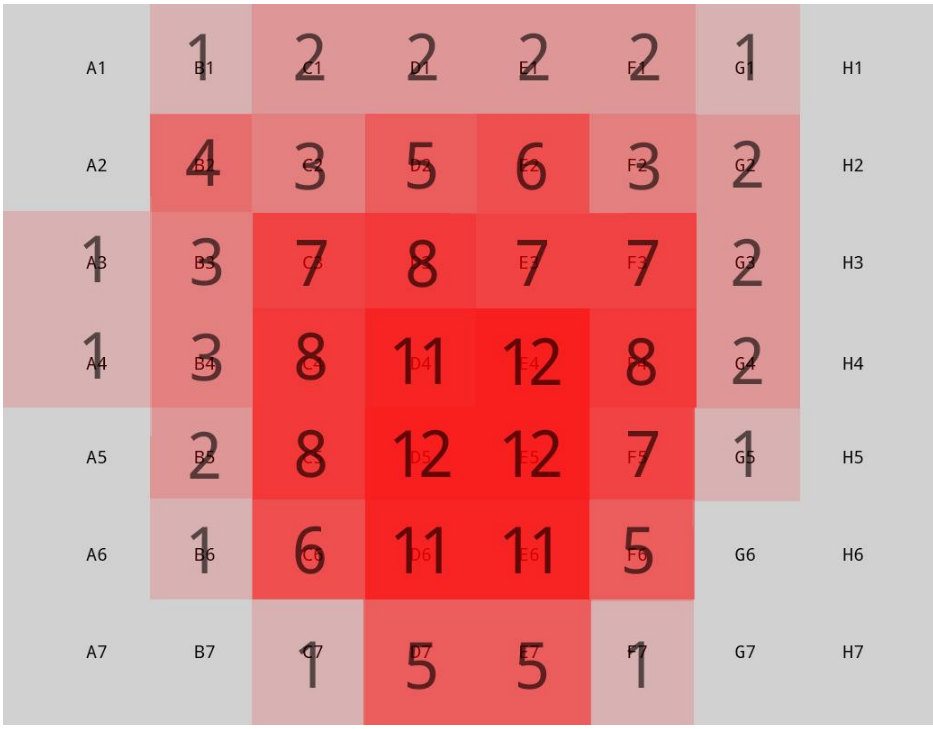
Description:

The user will first view a grid of 20pt text on a 1280x800 texture (again placed at the same distance of the last 2 tests) and report on the coordinates that they can read comfortably. The grid will then be tilted towards the bottom of the in-game camera. The user will be asked what coordinates they are able to read comfortably on the new grid.

Results:



Number of users (out of 12) that reported they could read each coordinate.



The results from the same test after the bottom of the plane was pivoted towards the camera (the #5 row was the pivot point and stays the exact same distance in both tests).

Test #4
Textures and Perception

Hypothesis:

Due to the Oculus's 3D nature some textures will cause issues with a user's perception of distance.

Description:

The user will be placed in a virtual room textured with a specific art style. 5 units in front of them will be a block 1x1x1 units. 100 units in front of them will be another block. On key press a third block will spawn in the middle (at a random distance between 10 and 90 units from the user) and the user will be asked to guess the distance the block is away from them. Then the next texture will be applied. They will not be made aware of how far away the middle blocks actually were until the end. The four styles of textures are Borderlands, Counter-Strike, Pixel Art, and Doodle Art.

Results:

Twelve users completed this test and these are the averages of the guesses, the actual distance of the spawned block, and the averages of differences.

	Borderlands: Guess 1	Borderlands: Actual 1	Difference	Borderlands: Guess 2	Borderlands: Actual 2	Difference
Averages	51.25	46.08	12.50	44.17	38.67	7.67

	Counter-Strike: Guess 1	Counter-Strike: Actual 1	Difference	Counter-Strike: Guess 2	Counter-Strike: Actual 2	Difference
Averages	40.50	34.42	9.08	44.17	35.50	10.50

	Pixel Art: Guess 1	Pixel Art: Actual 1	Difference	Pixel Art: Guess 2	Pixel Art: Actual 2	Difference
Averages	57.50	51.08	10.08	51.67	42.25	9.92

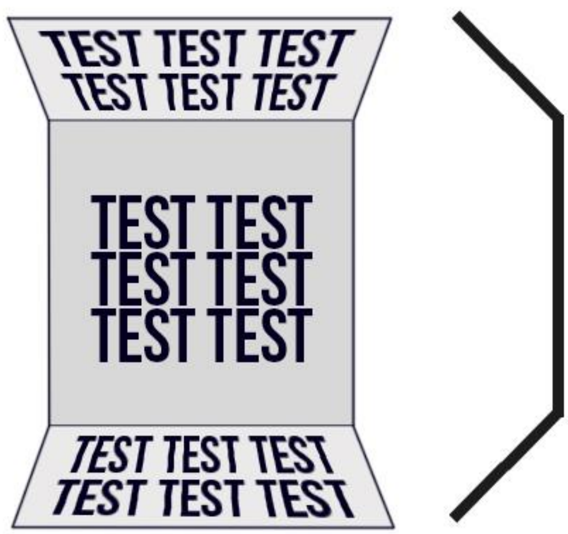
	Doodle: Guess 1	Doodle: Actual 1	Difference	Doodle: Guess 2	Doodle: Actual 2	Difference
Averages	55.17	44.25	11.75	71.83	62.33	13.00

Art Tests Conclusion:

The main point of these tests was to learn practical lessons about the nature of creating art for the Oculus Rift. The first of these tests focused on colors and whether or not certain colors would cause discomfort for users to view on the Oculus. We found that a few colors caused enough users discomfort to warrant caution when using them in excess. These colors are listed above in detail, but we will state generally that the problem lies in any color of extreme saturation. Light and dark colors typically did not cause any problem, especially towards the extreme ends of the brightness values. However, when a color is fully saturated it can cause problems. As for hues, red was the worst of all with three levels of brightness causing issues with at least 6 users. And while we actively avoid the 'problem colors' we've listed, we want to point out that our testing was using an entire wall of each color. Using red for a few game objects is not an issue, but we'd recommend staying away from red menu and start screens.

Font size and type was a test that is possibly only relevant while the Oculus is stuck at 1280x800 resolutions. That being said we still learned that small font is possible, but the cut off size is a steep curve. We started to have issues around 20pt font, but they were minimal. At 15pt font the users were missing every 5th letter. In regards to font types, no one font was considered to be unreadable, but more intricate fonts caused issues for the users. During our work with our prototypes we've learned that kerning plays an important part in whether or not a font is readable. If the letters are two close together the font will be unreadable on the Oculus. We've found that it's best to make the text look just a little bit too spread out on our regular monitors; this usually results in readable text on the Oculus.

More importantly than text size and type is the placement. Our third art test found that usable area on the Oculus for displaying text is very small. For absolute reliability you want to stay to a central area with a radius of about 12% of the device's total FOV. You could double that without too much of an issue, but text outside that second range is going to be completely unreadable. This means that the standard way of approaching UI (placing vital information around the edge of the screen) is no longer a viable way of designing UI. We have found that if you tilt the plane on which the UI lies in relation to the bend of the Oculus's fisheye lenses you can push the text nearer to the edge of the display while still maintaining readability.



A representation of how best to display textures on the Oculus; the texture planes near the edge of the display should be tilted towards the in-game camera.

One thing we didn't manage to test was creating curved text planes that exactly offset the fisheye lens curve. We hypothesize that this would increase readability on the edge of the display even further, but unfortunately we did not have time to test this.

Our final art test attempted to find a correlation between certain types of textures or art styles and a drop in the accuracy of a user's depth perception. Between the four styles we tested there was almost no change in accuracy. From what we can tell your depth perception is completely unaffected by texture on the Oculus Rift. However we did discover that textures that are rather intricate succumb quickly to aliasing on the Oculus's display. The low resolution and fisheye effect combine to cause issues with anything but simple types of art styles.

Project Spearhead
Post Mortem - Control Variation

Note: The testing done by the Spearhead team is by no means scientific. We aren't researchers; we're game designers attempting to learn as much about a new device as possible in a 6 week crunch period. As a result much of our data is incomplete and many of our conclusions rely on anecdotal evidence. Please feel free to disagree with our conclusions or better yet, continue our testing and refine our methods and results.

Overview

The most interesting aspect of the Oculus, in our opinion, is its use as a new form of controller. We believe that it's completely possible to create an entire game experience with the only needed peripheral being the Oculus Rift (no keyboard, no mouse, no gamepad) or to use the Oculus to augment and add additional mechanics to a game. In this same train of thought we wanted to explore the Oculus's usability in reference to several already established variations of control. Each of the tests in this category are taken directly from mechanics present in many current games. We want to see if they will translate to the Oculus. How adaptable is the Oculus? How much customization should game developers need to make available for the user to play with?

Test #1
Vertical Axis Inversion

Hypothesis:

Inverting the vertical axis will no longer be a viable option for video game controls with the Oculus.

Description:

The vertical axis will be inverted on the Oculus; when the user looks up, the in-game camera will look down and visa-versa. The horizontal axis will remain the same. The user will play through an obstacle course that requires them to look around to move in a direction. First, they will play through a course with the vertical axis set normally. Second, they will play through a similar course with the vertical axis inverted. The courses will be of equal length, and the times through the courses will be compared. Because of the difficulty of the end of the courses, the times recorded were taken at the very beginning of the final obstacle (the large barrel with two separate paths cut into it).

Results:

	Normal Controls				Inverted Controls			
	Time 2	Time 3	Average	Did they finish?	Time 2	Time 3	Average	Did they finish?
User 1	24.3	26.2	25.3	3 for 3	32.9	63.1	48.0	0 for 3
User 2	24.9	22.8	23.9	3 for 3	24.8	22.8	23.8	2 for 3

User 3	22.8	27.0	24.9	3 for 3	40.8	22.8	31.8	1 for 3
User 4	30.4	33.1	31.8	3 for 3	47.1	41.0	44.1	0 for 3
User 5	22.8	22.8	22.8	3 for 3	58.1	54.2	56.2	0 for 3
User 6	38.0	28.0	33.0	3 for 3	112.0	60.0	86.0	0 for 3
User 7	24.0	22.0	23.0	3 for 3	48.0	85.0	66.5	1 for 3
User 8	23.1	24.9	24.0	3 for 3	45.0	52.1	48.6	0 for 3
User 9	37.0	25.5	31.3	2 for 3	32.0	27.0	29.5	2 for 3
User 10	30.8	23.6	27.2	3 for 3	22.8	44.5	33.7	1 for 3
Average			26.7	29 for 30			46.8	7 for 30

Test #2
180 Degree Turn

Hypothesis:

Hot keying a 180 degree turn to the controller will continue to be a useful mechanic with the Oculus.

Description:

The user will use the space button to trigger a 180 degree turn. After the turn is complete they will need to use the Oculus to look at three shapes marked with either an A, B, or C. The time it takes them to find the shapes will be recorded and compared to a test done with a normal monitor and mouse input. The monitor version of the test will be conducted first. 5 trials of each version of the tests will be done.

Results:

	Monitor Trials			Oculus Trials		
	Average	Nausea?	Difficult?	Average	Nausea?	Difficult?
User 1	2.76	No	No	4.15	No	Yes, wanted to look with eyes, not turn head.
User 2	2.83	No	No	3.16	No	Mouse is Easier
User 3	2.84	No	No	3.26	Only when shaking head quickly	Don’t find natural, mouse easier
User 4	3.43	Little bit	Difficult to control Mouse	3.11	No	Camera move was fine
User 5	2.73	No	No	3.39	No	Different feel than mouse
User	4.52	No	Yes	3.38	No	Yes

6						
User 7	3.62	No	Yes	3.44	No	Yes
User 8	3.26	No	No	2.90	No	No
User 9	3.21	No	Mouse Sensitivity Annoying	3.26	No	Easier, felt it went quicker
User 10	2.83	No	No	3.35	No	More Convenient, better precision than mouse
User 11	3.08	No	No	3.39	No	No
	3.19			3.34		

The global average for the monitor trials was 3.19 seconds and the global average for the Oculus trials was 3.34 seconds.

Test #3
Orientation Sensitivity

Hypothesis:

Some users will find it useful to adjust the sensitivity of the Oculus in regards to its rotation tracking.

Description:

The user will need to use the Oculus to follow the path of a brightly colored ball. The ball will move rapidly around the user's character and pause for a period of time then begin moving again, this will continue for 15 secs. The amount of time the user spends looking at the ball will be recorded as a numbered score. This test will be done for a range of sensitivity settings and the results compared. Before each test the user will be allowed to play with the new setting and get used to it.

Results:

	Sensitivity .25	Sensitivity .5	Sensitivity 1	Sensitivity 2	Sensitivity 4
	Average	Average	Average	Average	Average
User 1	407.5	626.5	588	560	333.5
User 2	526	549.5	634.5	651	525.5
User 3	537	715.5	695	612	673
User 4	489	627	647.5	575.5	468.5
User 5	626.5	745	738	721	652

User 6	453	606.5	711	752.5	776
User 7	492	661.5	666	671	637
User 8	588	644.5	616	655	534
User 9	591	743	707	672	635
User 10	508.5	673.5	646.5	746.5	616.5

Green cells mark the sensitivity where the user performed the best on average.

Control Variation Tests Conclusion:

When you look at the Oculus Rift as a controller and not just as a new form of display the possibility for VR games increase exponentially. These three tests explored the possibilities and options that come with using the Oculus as a controller. The first test dealt with the possibility of inverting the Y-Axis on the Oculus so that whenever you tilt the device up, the camera in-game will look down and visa versa. We found that when dealing with inverted controls users took almost twice as long to get through ¾ of an obstacle course. The final part of the course was more difficult and as a result many of our users took 100s of seconds more to finish the course with inverted controls than with normal controls. Some users were unable to complete the entire course at all. The chief complaint was that the controls did not feel natural; the next biggest complaint was that inverting the controls meant that users had to look in the opposite direction from the one they were headed in. From our standpoint this made it clear that an inverted y-axis is not a viable option for the Oculus Rift. We did notice that as users played with the setting more and more they adapted to it, some members of Project Spearhead are able to run the course perfectly even with an inverted y-axis. This is not reflected in our data, but bears mentioning. Another thing to note is that we briefly tested inverting both the y and x axis's. While still not as usable as the normal controls the opinion of the members of Project Spearhead was that this set up was more usable than just inverting the y-axis. And while we do not recommend inverting the y-axis, inverting both could work in the right setting.

The 180 Degree test is reminiscent of a control that is common in console first person shooters; where a user can quickly turn their in game character around 180 degrees with a button press. Our test focused on the users' ability to find objects after completing the turn, simulating a scenario in which users might encounter during an FPS. We found that users were able to find the objects just 0.15 seconds faster with keyboard and mouse controls than they were with Oculus controls. We feel that this difference is negligible, especially considering that users were able to adjust the sensitivity of the mouse, but not of the Oculus (which was kept at a 1:1 ratio or sensitivity setting of 1 in terms of our 'Sensitivity Test'). Perhaps, more importantly though, this test proved that camera moves that aren't controlled by the user are not as uncomfortable or experience-breaking as previous thought. This goes against just about everything that has been said by Oculus VR about in-game camera movements not controlled by the user. And while we aren't saying they are necessarily wrong (one or two of our users take issue with such movements), we think there isn't a need to cut the movements out complete, especially since we found them very useful in directing the user's attention.

The final test in our Control Variation Tests asked a simple question, do you need to offer sensitivity options for the Oculus Rift? Our test explored 5 different settings (.25, .5, 1, 2, and 4). For reference, .25 is where if you move your real head 40 degrees the in-game camera will move 10 degrees. We had users complete the same test 3 times with each sensitivity and the setting they were most successful with was recorded as the sensitivity they preferred. The standard sensitivity, 1:1 or 1 was only preferred by one user. The two most popular settings were .5 and 2 at 4 users each. To us this means that users need to make their own choice in regards to the sensitivity of the Oculus. This has also been reflected in our prototyping. There are times when it is beneficial to have the Oculus respond with more or less sensitivity than 'normal'. One of the biggest reasons for purposefully adjusting the sensitivity of the Oculus is to reduce strain on the neck. If a game mechanic requires the user to continuously move their neck the sensitivity of the Oculus can be increased to give the user better control.

Project Spearhead
Post Mortem - Motion Tests

Note: The testing done by the Spearhead team is by no means scientific. We aren't researchers, we're game designers attempting to learn as much about a new device as possible in a 6 week crunch period. As a result much of our data is incomplete and many of our conclusions rely on anecdotal evidence. Please feel free to disagree with our conclusions or better yet, continue our testing and refine our methods and results.

Overview

The Oculus Rift presents a lot of opportunities for the Location Based Entertainment Industry, if it can be used in conjunction with outside motion. All of these tests are focused on the usability of the Oculus while the user is in motion. The first test involves the user attempting to navigate a digital space and a physical space simultaneously. We see this as something that would be useful to the LBE and AR industries. The second and third test both deal with a user attempting to utilize the gesture controls from an earlier test while under motion from an outside force, in our case, a car. The purpose of these tests is twofold. The first is to see whether or not it's possible to ever play Oculus games in a car; can it be done with the current technology and are users able to play a VR game in a moving vehicle. The second purpose is to investigate the level of discomfort that comes with being in a world that is moving in a way that contrasts how the user is moving in the real world.

Test #1
Walking in real life while viewing a virtual world

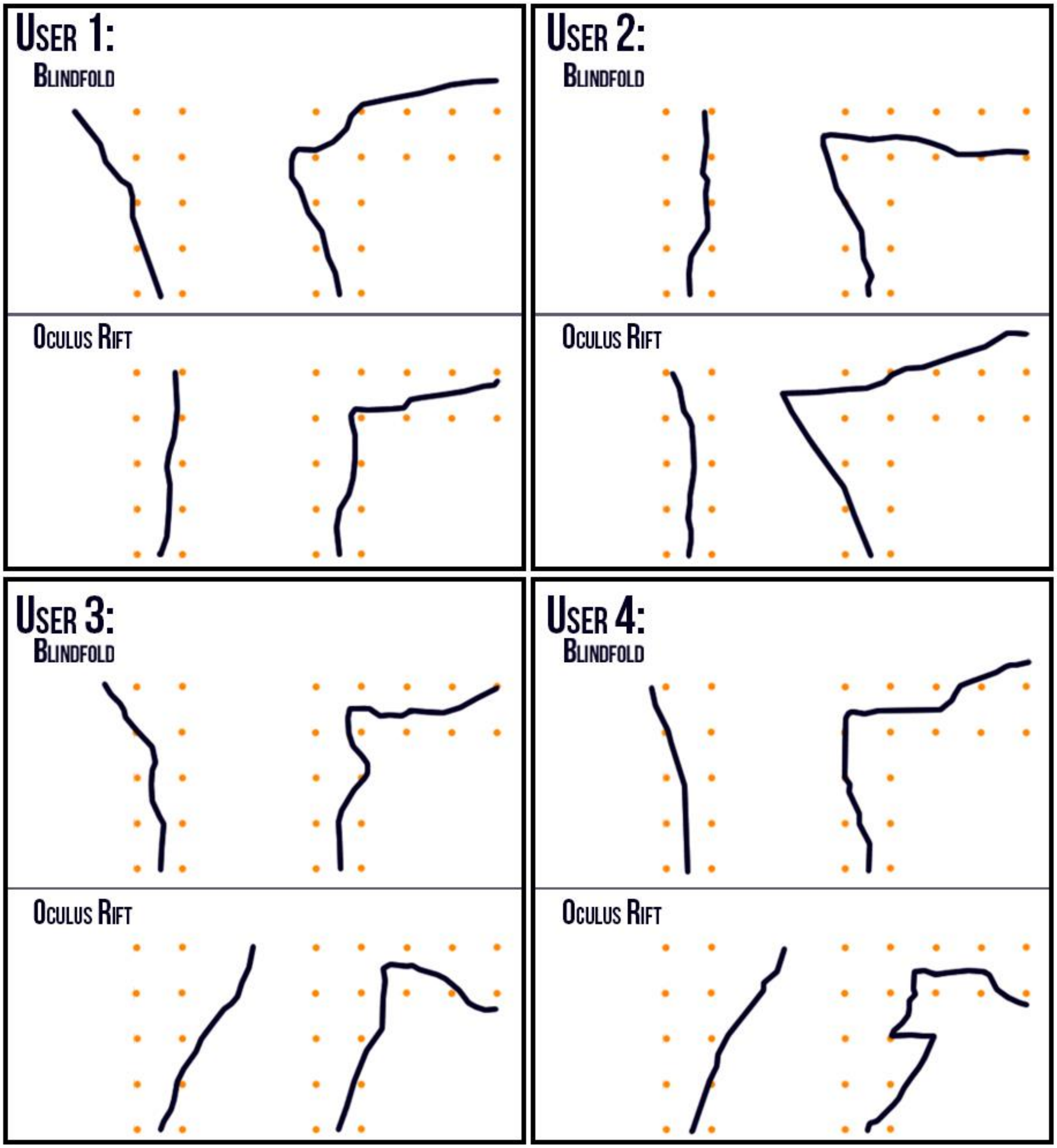
Hypothesis:

It is possible to follow a set path in real life while only being able to see a similar path in a virtual environment.

Description:

The test subject will first be blindfolded and asked to walk in a straight path of cones. The cones should be placed to create a straight hallway about 5 meters wide. The path they take through the cones will be recorded, stop them when they reach the final cones. Next, they will put on the Oculus Rift that is loaded with a level with a straight path of virtual cones similar to the real path they are walking. They will try to walk the real path and the virtual path simultaneously (the virtual path will be navigated with a control stick) and the number of deviations will be recorded. This will be done with 2 different paths, a straight line and a right 90 degree path. The right angle should also be a 'hallway' about 5 meters wide, and 40m on each side. The number of times they veered of each path during the different tests will be compared.

Results:


User 1:
Blindfold
Oculus Rift
User 2:
Blindfold
Oculus Rift
User 3:
Blindfold
Oculus Rift
User 4:
Blindfold
Oculus Rift

Test #2
Forward Motion

Hypothesis:

The Oculus Rift will still be playable in a vehicle as long as it moving in a straight line.

Description:

The test subject will wear the Oculus Rift while in a car or other vehicle. The vehicle will be driven in straight line with at least one stop sign in between the start and finish. During the trips the user will play the 'Rocket Game' from our 'Y-Axis inversion test'. This version of the game will be modified so that it is able to network with a separate laptop and Oculus Rift. The second Oculus will be set in a stationary, level, position facing forward; this will offset the motion of the vehicle and allow the user to utilize head gestures unabated. The distance of the trip should not be much longer than a minute. The user will record if they are able to finish the level and how they felt during play.

Results:

Each of the three users we tested were able to complete the course for the Rocket Game. The offset method we used worked fine and glitches were kept to a minimum and did not interfere with the gameplay.

Test #3
Chaotic Motion

Hypothesis:

The Oculus Rift will still be playable in a vehicle moving eradicating and making sharp urgent turns.

Description:

The test subject will wear the Oculus Rift while in a car or other vehicle. The vehicle will be driven in chaotic path. For our test, we used a slalom pattern with varying straight lengths and sudden turns. However, any course that quickly alternates straights and turns and direction will work. It should simulate chaotic city driving. During the trips the user will play the 'Rocket Game' from our 'Y-Axis inversion test'. This version of the game will be modified so that it is able to network with a separate laptop and Oculus Rift. The second Oculus will be set in a stationary, level, position facing forward; this will offset the motion of the vehicle and allow the user to utilize head gestures unabated. The distance of the trip should not be much longer than a minute. The user will record if they are able to finish the level and how they felt during play.

Results:

The three users we tested were able to complete the course for the Rocket Game on at least one of their attempts. Two users encountered glitches to the point where the game was rendered unplayable.

Motion Tests Conclusions:

Outside motion affects an Oculus user in strange ways that range from helpful to interesting to nauseating. From a general standpoint we found that, like most things with the Oculus Rift, moving while wearing it starts out being mildly to extremely uncomfortable, but after a few uses and a small adjustment period, the user grows accustomed to the sensation.

Our first test involved learning whether or not wearing the Oculus and having a virtual world to guide themselves through would assist the user in walking a straight line over them being blindfolded. To clarify, the user was hooked up to any device that measured motion tracking and had to move themselves in the virtual world using a controller. What we found was that the users were much better at completing the two courses while wearing a blindfold than the Oculus Rift. The Oculus created a false sense of confidence and the users ended up losing track of their position and ending up farther of course than they ever got while wearing the blindfold. The confidence also showed up in the speed at which they completed the course. The users would tread carefully, in contrast, while wearing the Oculus they would walk with a normal stride, at a normal pace. The final piece of information uncovered by this test was that if the user kept their in-game avatar still and focused on a point in the distance in the Oculus world they were able to walk in a straight line almost as quickly and accurately as if they were wearing nothing.

The second two tests were linked in that they dealt with playing an Oculus game, and using Oculus gestures while in a moving vehicle. The first test focused on vehicles moving in a straight line, as one might do on a highway or in an airplane. This test showed that users are able to play games accurately while moving in a straight line. Slowing at a stop sign did cause some discomfort during the first trial, but for subsequent attempts the stops were not a problem.

For the third test, the car was driven in an erratic fashion, turns and speed changes occurred abruptly and without warning or planning; each turn was also at least 90 degrees. This inflicted several angles of force and momentum on the user that were in no way constant with the virtual world they were viewing. In a similar fashion as the stop signs, the users were initially left feeling nauseous and uncomfortable. However, at no point did this feeling affect their ability to finish the course. What did occur was that the method in which we were offsetting the car’s movement started to break down. This was the reason why users were not able to complete the course every time.

The method in which we are offsetting the car’s movement has serious issues. The biggest of which is that if the car turns around even a single degree past 90 degrees from its original position the Oculus interprets this as though the entire apparatus has inverted. For the user this means that the controls suddenly invert in a similar fashion as our ‘Y-axis Inversion Test’. Another major issue is that the two Oculus’s are networked via Ad-hoc Wi-Fi and that has a substantial amount of lag and during rapid changes in direction this causes havoc with the game being played.

Space Bball Game

The Idea:

This was one of our earlier prototypes and began as an attempt to create a full game with only Oculus Rift gesture controls. From the beginning of our project we has a fascination with the idea of utilizing the Oculus in a way that would make it the only device you needed, acting as both the controller and the display. And so for this prototype we focused making a game where the only control was shaking your head around to manipulate an on screen object.

Gameplay:

The game starts with three different colored balls in a space basketball court. To start the game you shake your head left to right. Once the game the balls will float around the space until hit by the walls of the court, then they will move around at a constant speed. You control the court's movement with your head, as you move your head around, the court will move with it. By controlling the court you need to bounce the balls around and score baskets. However, only the ball which is the same color as the court will count as positive points. Other balls will take away points, and watch as the correct color changes every few seconds. Also, every few seconds one of several power ups will appear outside the court. Collect them to gain an advantage in the game. The game lasts 60 seconds, try to get as many points as you can.

Equipment Required:

- Oculus Rift Development Kit

Running the Prototype:

1. Connect the Oculus Rift as specified in normal operation to a laptop
2. Take some time to adjust the lenses on the Oculus Rift so that the image on screen is as clear as possible
3. Most likely you will have to adjust the angle and location of your webcams to get the best effect

Controls:

Oculus Rift Controls:
Shake head to start or restart the game.
Move head around to control the court.

Conclusions:

The project was plagued by a number of issues from the beginning. The chief of which was the lack of a decent physics engine. We found that with Unity, the engine we were working with, we had issues with mesh colliders and the rate at which Unity physics updates its collision detection. To put it plainly, we had to put a cap on the speed at which the basketballs could achieve or else they would move too fast for Unity to track them and fly out of the court. This meant that it was harder to get the basketballs bouncing around, and the user needed to really throw their head around, which in turn caused fatigue. This is why we gave the balls a constant velocity, otherwise it was too much to ask of the users. On a side note, something to consider regarding the Oculus Rift as a platform is the strain caused by playtestings. There were times during our development where we had to take time out to rest our heads or even lay to stop the discomfort. It's something that is going to need to be addressed when developing VR games, either by shortening the hours of QA testers or perhaps by alternating their focus between VR and non-VR games.

The other major issue we ran into with Space Bball was that people did not identify with the court and wanted to control the basketballs. However, we feel this is more a matter of improperly branding our game and not indicative of an issue with purely Oculus games or gesture controls.

What went well with this prototype was that we were able to manipulate the gesture controls in a way that, despite all the problems, gave us a fun, playable game. This speaks to the flexibility of gesture based controls with the Oculus. With some work you can make the Oculus do just about anything you want.

Third Person Rig

The Idea:

One of the most popular camera angles in video games is the third person camera. Games like The Legend of Zelda, Gears of War and Uncharted use some variation of the camera. The main concept of the third person rig was to see what life would be like if you saw in third person.

Gameplay:

The gameplay of the third person rig was simply experiencing life from a new perspective.

Equipment Required:

- Laptop
- Oculus Rift Development Kit
- 2 Web Cameras
- Third Person Camera Rig

Running the Prototype:

1. Connect the Oculus Rift as specified in normal operation to a laptop
2. Replace the typical AC adapter with: http://www.amazon.com/gp/product/B003MQO96U/
3. Connect 2 identical webcams vertically separated by 2.5 inches
4. Execute the application
5. If the cameras are flip-flopped press “a” to swap them
6. Take some time to adjust the lenses on the Oculus Rift so that the image on screen is as clear as possible
7. Most likely you will have to adjust the angle and location of your webcams to get the best effect

Controls:
Keyboard Controls:
A Key: Swaps Cameras

Conclusions:

The third person rig was an extremely unique experience. Every time a user put on the rig there was an immediate “woah”. Then usually they would complain about the bald spot on the top their head or comment on how good they looked. Moving and interacting while using the rig was difficult, because your point of perspective is 3 feet behind you.

Headless Horseman

The Idea:

Headless Horseman is a game inspired by the Headless Horseman character of “The Legend of Sleepy Hollow”. The headless horseman is often depicted carrying his head, a pumpkin, in his hand. We were struck by the idea of what it would be like to move your head around and get new perspectives on the world and felt that the Oculus Rift was a great way to explore this concept.

Gameplay:

You play as a ‘headless horseman’ a creature who holds his head in his right hand which is represented by the PS Move controller. You use your ability to raise and lower your head to look around the world and over obstacles. This helps you solve puzzles and navigate through the dungeon. The second power you have is the ability to aim and throw your head around the environment. This is useful as it allows you to see over tall walls and activate triggers that your body can’t get to.

You must use both of your powers to tackle each of the puzzles in your path. You ultimate goal is to pass through the dungeon and escape.

Equipment Required:

- PC
- Oculus Rift Development Kit
- Playstation 3 (PS3)
- Navigation Controller for PS3
- Move Controller for PS3
- MoveMe, an app for PS3
- PS3 Eye

Running the Prototype:

1. Connect the Oculus Rift as specified in normal operation
    1. Make sure the Oculus is mirroring your main display by checking in the screen resolution settings of your PC
2. Make sure both the PS3 and the PC have internet access
3. Turn on both the PC and PS3
4. Connect the PS3 Eye to the PS3 and make sure it is facing a clear area of floor space
5. Start up the MoveMe application on the PS3
    1. In the upper left corner in the MoveMe application is an IP address and a port. These will needed to connect the PC to the PS3
    2. Turn on your Navigation and Move Controller

   3. The Move Controller will need calibrated at the beginning of each play session. This is accomplished by pointing the Move controller at the PS3 Eye and pressing the move button.
6. Make sure the Oculus Rift can extend into your cleared play area and that it is on
7. Start the executable
8. Take some time to adjust the lenses on the Oculus Rift so that the image on screen is as clear as possible
9. On your PC monitor you should see an IP address and a Port. Adjust these values to match the values seen in the MoveMe application and click connect

Controls:

Move Controller Controls:
Square Button: Calibrate Move controller location.
Position Tracking: The placement of the move controller in 3D space affects the view the player has in the world.
Trigger Button: Throw/ Retrieve head

Navigation Controller Controls:
X (Cross) Button: Next page in the introduction
Circle Button: Open door when close
L1 or L2 Button: Show predicted head landing location if thrown.
- Changes operation of Analog Stick from movement to adjust throw vector.

Analog Stick: Movement/ Adjust throw vector (depends on L1 or L2 Button activation)

Conclusions:

The reason we decided to make ‘Headless Horseman’ was our desire to push Oculus Rift users to experience something unlike any other game in existence. We wanted to find out if users would be able to play a game that involves, literally holding your head in your hand, and throwing your head around the level. Those two functions rapidly alter the perspective and position of the in-game camera, which were we concerned would cause users far too much discomfort to ever be enjoyable.

Our concerns, as it turns out, were valid. We found that some users were unable to play our game, especially the first time they played our game and those that were unfamiliar with the Oculus. We also had to alter our ‘throwing’ mechanic so that the head would not bounce. Bouncing the camera off of objects after each throw made the game unplayable to pretty much everyone including the development team. Once the throwing mechanic had been toned down we found more players who were able to enjoy our game and the ability to toss the in-game camera across the level. This is what we were hoping for. And it made it clear that some Oculus users can tolerate a great deal before becoming uncomfortable. Unfortunately, that’s not a statement that is true for all users. In truth, it’s a sliding scale, some users can tolerate a great deal and a game like ‘Headless Horseman’ is perfect for them. For others, simple demos are too much.

Rift Swarm

The Idea:

First Person Shooters, Action RPGs, and Racing games are a natural fit for the Oculus Rift, however is it possible to get other genres to work with the device. We at Project Spearhead have often read people joke about getting top down games or RTS games working on the device, thinking that it wouldn't offer anything.

For our Rift Swarm prototype we are attempting to create a top down tower defense game that not only works on the Oculus, but does something different in the genre.

Gameplay:

You are perched on top of a tower in the center of the bottom of a bowl. In the center of your FOV is a continuously firing laser. The bowl is divided into three sections, blue, yellow, and pink. Enemies will begin in the blue section and 'attack' in waves. In total there are 11 waves. To defeat enemies you can either aim the laser directly at them or you can aim at the color blocks (turrets) dotted around the dome. This will charge the turrets (you will see a circle spinning around the turret) when this is full the turret is completely charged. At any level of charge a turret will begin firing at nearby enemies. The two alternatively colored turrets near the center will fire at a high rate of damage and at a greater distance (though not at anything close by).

The object of the game is to defend your central tower and stop the enemies from reach you. Once 18 enemies reach you, the game is over.

Equipment Required:

- PC
- Oculus Rift Development Kit
- Recommended
    - Backpack
    - USB to 5v Power Adapter
    - Laptop

Running the Prototype:

1. Connect the Oculus Rift as specified in normal operation
    1. Make sure the Oculus is mirroring your main display by checking in the screen resolution settings of your PC
2. Start the executable
3. Take some time to adjust the lenses on the Oculus Rift so that the image on screen is as clear as possible

Optional:

1. To get the best experience, run the game on a laptop, place laptop in a backpack while connected the Oculus.
2. Use the USB to Power Adapter cord to power the Oculus and place backpack on the back of an office chair. This will help you spin around easier.

Controls:

Oculus Controls:
Look around to aim the laser.

Conclusions:

Our goal for this prototype was very straight forward, bring the tower defense genre to the Oculus Rift. The biggest worry was that the Oculus would not have enough resolution to create enough detail in the enemies and turrets to make a dynamic tower defense titles. Another worry was that if we made the enemies as small as we wanted to, the Oculus wouldn't be precise enough to hit them with the laser, or users wouldn't be able to hold their heads still long enough. None of this was the case.

Not only that, but because you are sitting in the middle of a 360 degree world your brains physical memory kicks in and helps you remember what is going on in the game. Instead of wondering, 'Ok where on the map am I being attacked… oh yeah, they are in the blue section'; a user remembers, physically that they are being attacked on their right, or that just behind them the need to keep two turrets charged. The dome, which was used to ensure no matter how from the center an object was it would be seen as 'top-down', had a added benefit of creating a sort of poor-man's holodeck. You can't move around in it, but we managed to create a living world with this prototype.

The Rift of Wall Street

The Idea:

This was the clean up prototype; the final project idea we came up and the one that ended up being a combination of every [major] idea we had left on the drawing board. We wanted to try working with positional tracking on the Oculus Rift (not of the person, but the device itself) since Oculus VR talks about wanting to include that technology into the final version. We also wanted to work with using a second screen that holds different information than the the Oculus display; our hope was that this would create tension in the user as they would have to juggle time between both screens. And finally we wanted to use the Oculus as an in game object; something the user would have to manipulate with their hands while looking at the second screen. All of these ideas combined to become 'The Rift of Wall Street'.

Gameplay:

You are in a square surrounded by building, this is a our 'Wall Street'. When the game begins coins and gavels will start flying into the square from the right and left. You must collect as many coins as you can while avoiding the gavels. If you hit a gavel you will lose all the coins in your wallet. Your wallet contains all the coins on your person; it is represented by 15 circles on the Oculus's screen. You can only have 15 coins at a time.

Once you have 15 coins or as many as you feel comfortable carrying you need to deposit them at the RiftDAQ building, the large black building in the front of the square. To deposit the coins you need to step forward to the edge of the square, take off the Oculus, hold it over top the second monitor, and shake. You should hear a coin dropping sound. Coins deposited into the bank will not be lost when you hit a gavel.

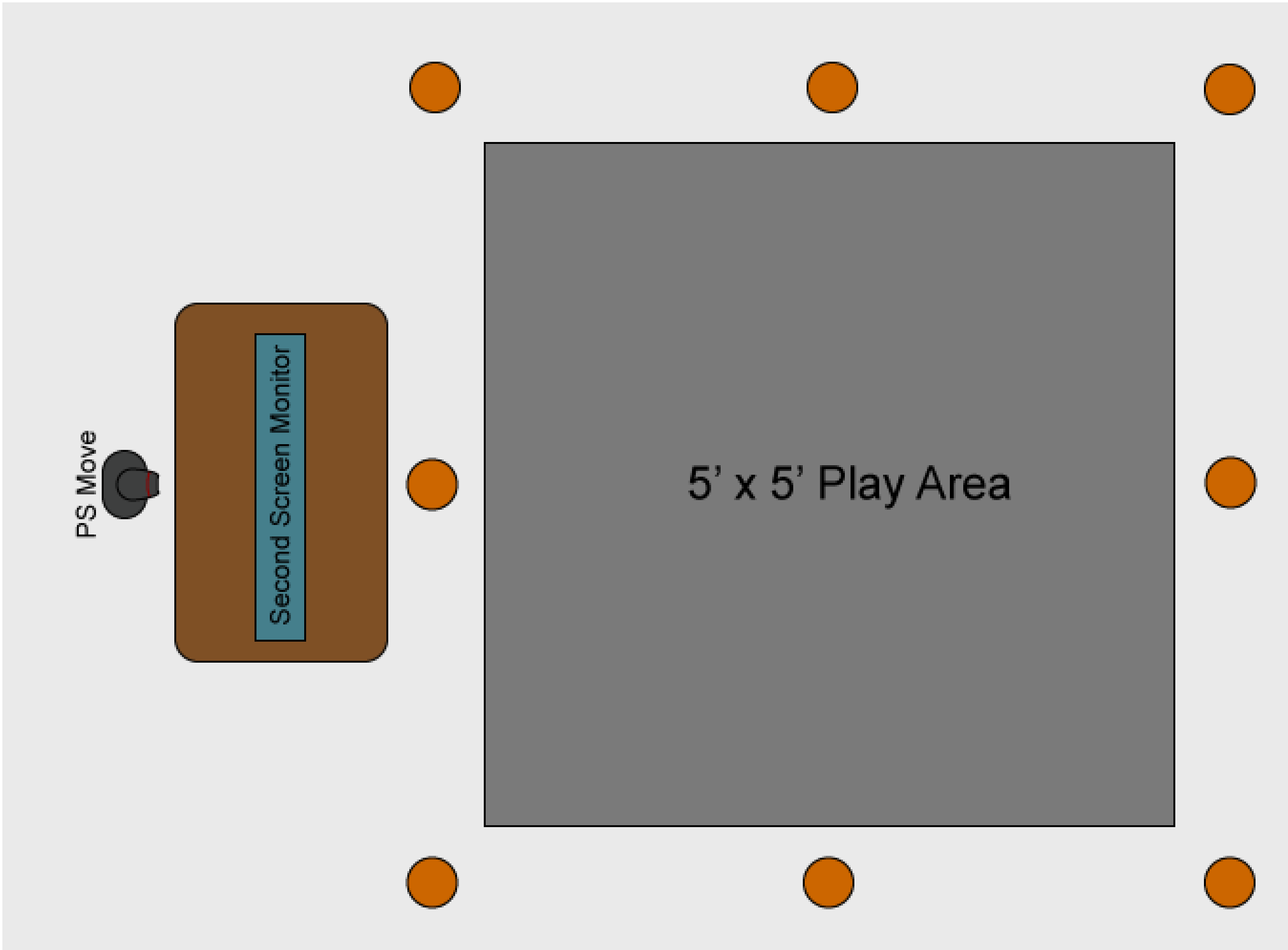
The second screen will update every time you deposit coins into the bank. This screen keeps track of the different values of each coin. The 4 different coins will become less value the more you collect them (in relation to the other coins). This means that if you focus on collecting a specific color, it will diminish in value, while the other coins increase in value. It's important to remember that you only have 3 minutes until the market closes again and the game ends, so use your time wisely.

Equipment Required:

- PC
- Oculus Rift Development Kit
- PS Eye
- 2 PS Moves
- PS3

Running the Prototype:

1. Connect the Oculus Rift as specified in normal operation
    1. This game requires two screens, one being the Oculus and one being your monitor
    2. Make sure both screens are set to 1280 by 800 resolution
    3. Make sure your monitor is your main display
    4. Make sure your monitor is to the right of the Oculus in the display settings
2. Setting up the PS Eye and Play area
    1. Place the PS Eye facing a clear section of the room
    2. Raise the PS Eye to chest level and tilt it slightly towards the ground
    3. The front of your play area will be about 60 inches away from the PS eye
    4. The back of the play area will be an additional 50 inches behind the front
    5. The sides of the play area will be about 40 inches to the left and the right of the center of the front
    6. It is a good Idea to mark off this area so that you can make sure the area remains clear
3. Connect the PS3 Eye to the PS3
4. Attach one PS Move to the front of the Oculus.
5. Turn on the PS Move
6. Attach the second PS Move to the back of the Oculus.
7. Turn on the second PS Move
8. Launch the MoveMe application on the PS3
9. Calibrate the front then the back move
10. Launch the DoubleMonitorVersion.bat (The Triple Monitor version can be used if a third display is attached to the left of the oculus in the display settings at 1280x 800. It simply shows the view of the player)
11. Change the IP address and port to match that of the MoveMe Application
12. Press the connect button
13. Put on the Ocuylus while standing in the play area
14. Take some time to adjust the lenses on the Oculus Rift so that the image on screen is as clear as possible
15. Ask a friend to press the up and down arrow keys to adjust the green cube until it is about eye level
16. Press right arrow to confirm this height
17. Finally press space to start the game



The layout of the play area; the PS Move should be placed at just below eye level on a stand. The second screen monitor should be on a short coffee table.

Controls:

Keyboard Controls:
Space Bar: Starts and stops the game
Up Arrow: Raise calibration cube
Down Arrow: Lower calibration cube
Right Arrow: Confirm the Calibration height

Oculus Rift Controls:
Move around the play area to collect coins and dodge gavels.
Shake controller while over the 'RiftDAQ' building to deposit coins.

Conclusions:

The first portion of this prototype went over very well. Adding position tracking to the Oculus, even just to the device itself works well and adds a distinct degree of immersion. More importantly users were able to move around the world comfortably. There was an initial feeling of imbalance, but once they realized they were able to walk around normally they felt at home in the Oculus world.

Our second idea was to have a secondary screen that would hold information that wasn't available on the Oculus screen. This worked in so far as it had the effect of encouraging users to occasionally remove the Oculus in order to read the extra info. Some of our users, about two thirds ,were playing as we had planned and were carefully juggling their time between the Oculus screen and the secondary screen. The other third ignored the extra info altogether and just collected coins. We believe this is for two reasons. One the information we presented on the second screen was not paramount to the success of the user. And secondly, some users felt that taking off the device broke their immersion and they preferred to stay in the Oculus world. For the latter group, we think that the L.B.E. (location based entertainment) industry might have the fix. If you had a room themed to look almost exactly like the virtual world (or maybe on it a past version of the other) it would hold the immersion and possibly add the experience.

The final idea we were working with was the idea of the Oculus as an object. This was a mixed bag. Many users, especially the ones who 'protested' the second screen mechanic were content to hold their head of the 'RiftDAQ' building and shake their around until the coins were deposited. Our biggest issue with this mechanic was that Oculus needed to have two PS Moves taped to it in order to gain positional tracking. This made the device unwieldy and uncomfortable to remove and put on quickly.

We could see each of these mechanics being successfully implemented into a game. Though we do think that special care needs to be taken so that each mechanic is meaningful and there is actual motivation for the user to utilize the mechanic.

Prototype #6

The Idea:

Prototype was our attempt at expanding upon and gamifying two of our projects we used in our testing; the Gesture Resolution Test and the Y-Axis Inversion test. We also wanted to incorporate a separate mechanic that required arm movements to control an in-game object. This was an experiment to see whether a user could handle controlling one mechanic using head gestures and controlling a separate mechanic via body movements.

Gameplay:

You are controlling a cursor made up of several cubes. The size and shape of the cursor change depending on how far the Razer Hydra controllers are held apart from each other. You will need to change that distance in order the collect ‘coin blocks’ throughout game to increase your score. You will also need to resize and use the Oculus’s head tracking to navigate around the red obstacles.

Your goal is to collect as many ‘coin blocks’ as possible while avoiding obstacles to achieve a high score. The game is procedurally generated so there is no end as long as you avoid contact with the obstacles.

Equipment Required:

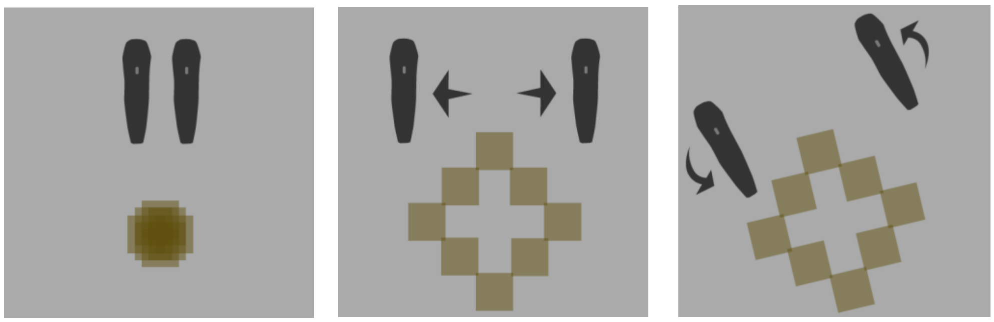
- PC
- Oculus Rift Development Kit
- Razer Hydra Controller

Running the Prototype:

1. Connect the Oculus Rift as specified in normal operation
    1. Make sure the Oculus is mirroring your main display by checking in the screen resolution settings of your PC
2. Clear an area on your desk in front of your keyboard.
3. Place the Razor on desk make sure base is orientated correctly with the connecting wires facing away from the user.
4. Connect the PS3 Eye to the PS3 and make sure it is facing a clear area of floor space
5. Start the executable
6. Take some time to adjust the lenses on the Oculus Rift so that the image on screen is as clear as possible

Controls:

Razer Controller Controls:
Keep controllers close to make game cursor smaller.
Spread controllers to grow the object.
Turn controllers (like a steering wheel) to turn in game cursor.



Oculus Rift Controls:
Look via tilting the device to steer the direction of the game cursor.

Conclusions:

Our goal was to create a prototype using two of our test projects and to explore the idea of combining head gestures with hand movements. What we found was that users tended to blur together their head and hand movements. As they turned their hands they would accentuate the gesture by tilting their head and sometimes full bodies simultaneously. This is a natural reaction (people do it even when play keyboard/mouse games) and isn't a problem as long as you don't have two separate functions controlled by similar gesture (ie: if tilting your head does function A and tilting your head does function B). This is something we purposefully avoid in Prototype, and so tilting your head doesn't affect the steering of the game cursor. By taking care in how we institute the motion gestures we managed to create a unique (if silly looking) way of controlling a game.

Gesture Test Data

General User Data:

	Age	Sex	VG Hours/Week	Years Played	Athlete?	OR Exp?	VR Exp?	Motion/Sea Sick?
User 1	25	m	0-5	0	Table Tennis	No	No	No
User 2	33	M	0-5	0	No	No	No	No
User 3	23	F	0-5	5	No	2 weeks	No	No
User 4	25	M	16-20	13	No	0	No	Motion
User 5	24	M	0-5	12	No	Once	Yes (Older HMD)	No
User 6	24	F	0-5	3	4 Years	5 Hrs	No	Yes
User 7	24	M	0-5	16	No	1-2 Hours	No	No
User 8	23	M	6 - 10	19	Longboard, golf, basketball	No	No	No
User 9	23	M	16-20	19	Tennis	Yes, 1 Hr	Yes, Older HMD	No
User 10	22	M	0-5	0	No	No	Yes	No
User 11	24	M	0-5	12	No	1 hour	No	No
User 12	29	M	0-5	15	No	2-3 hours	Not Really	Car sick (sometimes)

	Test 1 Speed Diff (Out of 10)	Test 1 Comments?	Test 2: Tilt Res (Oculus) in secs	Test 2: Tilt Res (Keyboard)	Test 2: Comments
User 1	0 (test failed)	N/A	35	39.7	Oculus better w/o big changes
User 2	6	Didn't get the concept	118	95	Perferred Keyboard
User 3	10	N/A	84	85	N/A
User 4	9	N/A	86	79	Less Fun with OR felt weird with keyboard
User 5	8		105	90	Had tendency to move head even when using keyboard controls
User 6	8	Red was super hard	52	30	N/A
User 7	8		25	35	Oculus controls were comfortable
User 8	2		159	109	N/A
User 9	10	Unclear of distance needed to move	116	86	
User 10	4		DNF	Unk	Nauseas
User 11	8		Faster	Unk	
User 12	8		134	108	N/a
	7.36		91.4	75.67	

Test 3: Fatigue (Total Time in secs)	Test 3: Fatigue (Times Stuck)	Test 3: Comments	Test 4: Get Out?	Test 4: Camera Move?	Test 4: Understand what to do next?
120	3		Yes	Felt Natural	Yes
50+	14	DNF	Yes	Didn't Notice	No, didn't understand gesture controls
66	2		No	Didn't feel too weird	Wanted to rotate puzzle
65	1		Yes	Wanted Control	Nope, needed instructions
DNF	-	Quit, felt dizzy	Yes	Felt Nauseas, not happy when control was taken	Yes
93	5	Dizzy Exhausted	Yes	fine compared to previous test	Yes mostly
92	1	Exhausted, a little dizzy	Yes	N/a	No, wanted to rotate
80	0	Felt sick, had to take a break	Yes	Made home nauseas, but drew attention to keyboard	Had to be prompted
90	3	No pain or discomfort	Yes	Noticed, didn't feel wrong	Yes, quickly
DNF	-		Yes	As if I am moving in there	Yes
50.4	1	Very Tiring, dizzy	Yes	Didn't notice	Took time to figure out
83	5	Oculus hits your eyes when you shake it back and forth	Yes	Didn't notice	Understood
82.16	3.50		11/12/2013		

Art Test Data

General User Data

	User 1	User 2	User 3	User 4	User 5	User 6	User 7	User 8	User 9	User 10	User 11	User 12
Age	25	25	29	24	23	23	25	23	21	22	-	-
Sex	M	M	M	F	M	M	F	M	M	M	-	-
VG Hours/Week	0-5	0-5	0-5	0-5	6-10	6-10	0-5	6-10	10-15	0-5	-	-
Years Played	5 years	20	23	-	6	5	0	~15	16	17	-	-
Athlete?	No	No	No	-	No	No	No	No	No	No	-	-
OR Exp?	Tried Once	1-2hr	1 hr	-	5-6 hours total	No	Tried Once	4 hrs	1hr	almost none	-	-
VR Exp?	No	No	Yes	-	No	No	No	Yes	No	No	-	-
Motion/Sea Sick?	No	No	No	-	No	Sea Sickness	No	No	Car Sick	Motion Sickness	-	-
Particpated in Other Tests	Gesture	Gesture	No	-	No	No	No	No	No	No	-	-

Color Test Data

Color	User 1	User 2	User 3	User 4	User 5	User 6	User 7	User 8	User 9	User 10	User 11	User 12	Total Ucomf	Difference	
Blue															
000033 (Darkest)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	0	3.67	13.44
000066	Yes	Yes	Yes	yes	Yes	Yes	Yes	Yes	Yes	Yes	x	Yes	1	2.67	7.11
0000aa	X	X	Yes	yes	yes	Yes	Yes	x	Yes	x	x	x	6	2.33	5.44
0000ff	Yes	Yes	X	yes	yes	Yes	Yes	x	Yes	Yes	Yes	Yes	2	1.67	2.78
969bff (Lightest)	Yes	Yes	Yes	X	yes	Yes	Yes	x	Yes	Yes	Yes	Yes	2	1.67	2.78
Gray															
333333 (Darkest)	Yes	Yes	Yes	Yes	yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	0	3.67	13.44
666666	Yes	Yes	Yes	yes	yes	Yes	Yes	x	Yes	Yes	Yes	Yes	1	2.67	7.11
aaaaaa	Yes	Yes	Yes	X	x	Yes	Yes	x	Yes	Yes	Yes	Yes	3	0.67	0.44
dddddd	Yes	Yes	Yes	X	yes	Yes	Yes	Yes	Yes	x	Yes	Yes	2	1.67	2.78
ffffff (Lightest)	Yes	Yes	Yes	X	yes	Yes	x	x	Yes	Yes	Yes	Yes	3	0.67	0.44
Green															
003300 (Darkest)	Yes	X	X	X	yes	Yes	x	Yes	Yes	Yes	Yes	Yes	4	0.33	0.11
006600	X	Yes	X	Yes	x	Yes	Yes	x	Yes	Yes	x	x	6	2.33	5.44
00aa00	X	Yes	X	X	yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	3	0.67	0.44
00ff00	Yes	X	Yes	X	x	Yes	Yes	Yes	x	x	Yes	Yes	5	1.33	1.78

79ff92 (Lightest)	Yes	X	Yes	Yes	yes	Yes	Yes	Yes	x	x	x	Yes	4	0.33	0.11
														0	
Red														0	
330000 (Darkest)	Yes	X	X	Yes	yes	Yes	Yes	x	Yes	Yes	Yes	Yes	3	0.67	0.44
660000	X	X	X	Yes	yes	Yes	Yes	x	Yes	Yes	x	X	6	2.33	5.44
aa0000	X	Yes	X	X	x	x (A little too bright)	Yes	Yes	Yes	x	x	x	8	4.33	18.78
ff0000	X	X	Yes	Yes	x	Yes	Yes	x	Yes	x	x	x	7	3.33	11.11
ff7f76 (Lightest)	Yes	X	Yes	Yes	yes	Yes	Yes	Yes	x	x	x	Yes	4	0.33	0.11
														0	
Teal														0	
003333 (Darkest)	Yes	Yes	Yes	Yes	yes	Yes	x	Yes	Yes	Yes	Yes	Yes	1	2.67	7.11
006666	Yes	Yes	Yes	X	yes	Yes	Yes	Yes	Yes	x	Yes	x	3	0.67	0.44
00aaaa	Yes	Yes	Yes	x	yes	Yes	Yes	Yes	x	x	x	Yes	4	0.33	0.11
00ffff	Yes	X	X	x	x	Yes	x	x	x	x	x	Yes	9	5.33	28.44
aaffff (Lightest)	Yes	Yes	Yes	x	x	Yes	Yes	Yes	Yes	Yes	x	Yes	3	0.67	0.44
														0	
Yellow														0	
333300 (Darkest)	Yes	X	Yes	Yes	yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	1	2.67	7.11
666600	X	Yes	Yes	Yes	yes	Yes	x	Yes	Yes	Yes	Yes	x	3	0.67	0.44
aaaa00	Yes	Yes	X	X	yes	Yes	x	Yes	Yes	x	x	Yes	5	1.33	1.78
ffff00	X	Yes	X	X	x	Yes	Yes	x	x	Yes	x	Yes	7	3.33	11.11
fffe71 (Lightest)	Yes	Yes	X	Yes	yes	Yes	x	Yes	x	Yes	x	Yes	4	0.33	0.11
													3.67		5.22
Number Of Uncomfortable	8	10	11	14	8	1	7	12	7	11	14	7			Standard deviation = 2.28
													9.17		6 or above = Bad
															1.5 or below = great

Font Size Data

	User 1	User 2	User 3	User 4	User 5	User 6	User 7	User 8	User 9	User 10	User 11	User 12	Averages	Average % Missed
Font Size														
100pt (Out of 15)	15	15	15	15	15	15	15	15	15	15	15	15	15.00	0.00%
75pt (Out of 16)	16	16	16	16	16	16	16	16	16	16	16	16	16.00	0.00%
50pt (Out of 24)	23	24	24	24	24	24	24	24	24	24	23	24	23.83	0.69%
30pt (Out of 21)	21	21	21	21	21	21	21	21	21	20	20	21	20.83	0.79%
20pt (Out of 29)	29	16	29	29	29	28	29	29	29	28	26	29	27.50	5.17%
15pt (Out of 14)	10	0	12	14	13	12	14	11	9	11	14	14	11.17	20.24%
12pt (Out of 18)	13	0	13	17	18	4	16	14	7	11	18	14	12.08	32.87%
10pt (Out of 12)	0	0	0	9	8	0	8	0	0	1	11	0	3.08	74.31%

Font Type Data

Font Type	User 1	User 2	User 3	User 4	User 5	User 6	User 7	User 8	User 9	User 10	User 11	User 12	Averages
Droid Sans Mono	Yes	Yes	Yes (bad)	Yes	Yes	Yes	Yes	Yes	Yes ('m' unclear)	Yes	Yes	-	2
Times New Roman	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes (harder)	Yes	Yes	-	1
Arial	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	-	0
DJ Gross	Yes	Yes	Yes (hard to read)	Yes	Yes/No (hard to read)	Yes	Yes (Difficult to read)	Yes (some trouble)	Yes (thick)	No (Missed one word)	Yes (Hard but can read)	-	7
Alien Encounters	Yes	Yes	Yes (hard to read)	Yes	Yes	Yes	Yes	Yes	Yes (pretty hard)	Yes	Yes	-	2
Papyrus	Yes	Yes	Yes (hard to read)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	-	1
Old English Text	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	-	0
Dunkin Sans	Yes	Yes	Yes	Yes (Best)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	-	0

Perception Test Data

	Borderlands: Guess 1	Borderlands: Actual 1	Difference	Borderlands: Guess 2	Borderlands: Actual 2	Difference
User 1	90	63	37	15	16	1
User 2	50	35	15	70	72	2
User 3	40	43	3	85	65	20
User 4	70	75	5	40	27	13
User 5	30	43	13	25	28	3
User 6	30	22	8	20	17	3
User 7	40	27	13	30	17	13
User 8	40	58	18	35	42	7
User 9	55	50	5	15	13	2
User 10	60	49	11	75	70	5
User 11	60	50	10	50	43	7
User 12	50	38	12	70	54	16
	51.25	46.08	12.50	44.17	38.67	7.67

	Counter-Strike: Guess 1	Counter-Strike: Actual 1	Difference	Counter-Strike: Guess 2	Counter-Strike: Actual 2	Difference
User 1	30	19	11	50	33	17
User 2	30	21	9	55	63	8
User 3	20	23	3	12	13	1
User 4	70	42	28	60	27	33
User 5	40	37	3	75	66	9
User 6	50	42	8	45	38	7
User 7	30	18	12	60	46	14
User 8	50	52	2	40	36	4
User 9	45	52	7	25	27	2
User 10	11	17	6	38	28	10
User 11	70	60	10	30	22	8
User 12	40	30	10	40	27	13
	40.50	34.42	9.08	44.17	35.50	10.50

	Pixel Art: Guess 1	Pixel Art: Actual 1	Difference	Pixel Art: Guess 2	Pixel Art: Actual 2	Difference
User 1	30	29	1	50	39	11
User 2	30	14	16	50	47	3
User 3	70	65	5	80	72	8
User 4	80	64	16	40	25	15
User 5	70	75	5	40	30	10
User 6	90	76	14	50	34	16
User 7	50	22	28	75	51	24
User 8	55	64	9	35	38	3
User 9	60	66	6	30	24	6
User 10	25	27	2	50	46	4
User 11	60	52	8	30	25	5
User 12	70	59	11	90	76	14
	57.50	51.08	10.08	51.67	42.25	9.92

	Doodle: Guess 1	Doodle: Actual 1	Differen ce	Doodle: Guess 2	Doodle: Actual 2	Difference
User 1	15	16	1	100	77	33
User 2	40	25	15	50	52	2
User 3	70	68	2	60	64	4
User 4	80	55	25	100	77	23
User 5	20	20	0	60	65	5
User 6	40	26	14	80	73	7
User 7	77	42	35	95	72	23
User 8	20	24	4	40	45	5
User 9	70	69	1	80	75	5
User 10	80	66	14	67	45	22
User 11	60	42	18	50	38	12
User 12	90	78	12	80	65	15
	55.17	44.25	11.75	71.83	62.33	13.00

Control Variation Tests Data

General User Data

	User 1	User 2	User 3	User 4	User 5	User 6	User 7	User 8	User 9	User 10	User 11
Age	25	23	28	24	38	23	24	23	25	23	25
Sex	M	M	M	F	M	F	F	F	M	M	M
VG Hours/Week	0-5	16-20	0-5	6 - 10	20+	6 - 10	6 - 10	0	6 - 10	6 - 10	10 - 15
Years Played	0	20	26	15	33	15	0	0	15	15	16
Athlete?	No	No	No	No	No	No	No	No, Dancer	No	No	Yes, Athlete
OR Exp?	Yes	Yes	Yes	Yes	No	Yes	Yes	No	Yes	Yes	Yes
VR Exp?	Yes	Yes, HMD	Yes	No	Yes, Cave	Yes	No	No	No	No	Yes
Motion/Sea Sick?	No	No	No	No	No	No	No	No	Yes, Motion Sick	Yes, Sea Sick	No
Particpated in Other Tests	All	Gesture	No	No	No	No	No	No	Gesture Tests	No	No

180-Degree Data

180 Degree Turn	Monitor Trials	In Seconds						
	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Average	Nausea?	Difficult?
User 1	2.00	3.85	2.31	2.54	3.10	2.76	No	No
User 2	3.06	2.88	2.84	2.63	2.72	2.83	No	No
User 3	3.61	2.70	2.70	2.76	2.43	2.84	No	No
User 4	3.83	4.25	2.81	3.24	3.03	3.43	Little bit	Difficult to control Mouse
User 5	2.85	2.35	2.82	2.71	2.93	2.73	No	No
User 6	6.94	4.00	4.07	3.65	3.92	4.52	No	Yes
User 7	4.23	3.71	3.93	2.70	3.51	3.62	No	Yes
User 8	3.55	3.51	2.68	2.88	3.70	3.26	No	No
User 9	3.57	3.33	3.43	2.80	2.90	3.21	No	Mouse Sensistivity Annoying
User 10	3.16	2.88	2.94	2.89	2.30	2.83	No	No
User 11	4.17	2.53	2.76	3.78	2.18	3.08	No	No
						3.19		

180 Degree Turn	Oculus Trials							
	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Average	Nausea?	Difficult?
User 1	3.64	4.07	3.62	5.53	3.90	4.15	No	Yes, wanted to look with eyes, not turn head.
User 2	3.33	3.57	3.30	2.53	3.06	3.16	No	Mouse is Easier
User 3	3.70	3.65	2.57	3.35	3.05	3.26	Only when shaking head quickly	Dont find natural, mouse easier
User 4	3.28	2.91	2.44	3.67	3.24	3.11	No	Camera move was fine
User 5	3.55	3.56	3.42	3.87	2.54	3.39	No	Different feel than mouse
User 6	4.26	3.70	3.02	3.46	2.46	3.38	No	Yes
User 7	3.41	3.95	3.31	3.10	3.45	3.44	No	Yes
User 8	2.65	3.08	3.03	2.51	3.21	2.90	No	No
User 9	3.55	3.48	3.22	3.03	3.00	3.26	No	Easier, felt it went quicker
User 10	3.56	2.61	3.06	3.61	3.93	3.35	No	More Convient, better precision than mouse
User 11	4.75	3.58	2.60	2.76	3.24	3.39	No	No
						3.34		

Sensitivity Data

	Sensitivity: .25			Sensitivity: .5			Sensitivity: 1		
	Score 2	Score 3	Average	Score 2	Score 3	Average	Score 2	Score 3	Average
User 1	367	448	407.5	593	660	626.5	621	555	588
User 2	562	490	526	556	543	549.5	583	686	634.5
User 3	541	533	537	724	707	715.5	699	691	695
User 4	482	496	489	605	649	627	643	652	647.5
User 5	630	623	626.5	749	741	745	695	781	738
User 6	439	467	453	610	603	606.5	729	693	711
User 7	533	451	492	692	631	661.5	614	718	666
User 8	556	620	588	621	668	644.5	682	550	616
User 9	567	615	591	750	736	743	769	645	707
User 10	433	584	508.5	629	718	673.5	633	660	646.5

	Sensitivity: 2			Sensitivity: 4		
	Score 2	Score 3	Average	Score 2	Score 3	Average
User 1	573	547	560	291	376	333.5
User 2	666	636	651	570	481	525.5
User 3	636	588	612	687	659	673
User 4	627	524	575.5	484	453	468.5
User 5	743	699	721	667	637	652
User 6	743	762	752.5	793	759	776
User 7	674	668	671	599	675	637
User 8	607	703	655	509	559	534
User 9	667	677	672	625	645	635
User 10	702	791	746.5	696	537	616.5

Y-axis Inversion Data

	Normal Controls				Inverted Controls			
	Time 2 (Until last obstacle)	Time 3	Average	Did they finish?	Time 2 (Until last obstacle)	Time 3	Average	Did they finish?
User 1	24.3	26.2	25.3	3 for 3	32.9	63.1	48.0	0 for 3
User 2	24.9	22.8	23.9	3 for 3	24.8	22.8	23.8	2 for 3
User 3	22.8	27.0	24.9	3 for 3	40.8	22.8	31.8	1 for 3
User 4	30.4	33.1	31.8	3 for 3	47.1	41.0	44.1	0 for 3
User 5	22.8	22.8	22.8	3 for 3	58.1	54.2	56.2	0 for 3
User 6	38.0	28.0	33.0	3 for 3	112.0	60.0	86.0	0 for 3
User 7	24.0	22.0	23.0	3 for 3	48.0	85.0	66.5	1 for 3
User 8	23.1	24.9	24.0	3 for 3	45.0	52.1	48.6	0 for 3
User 9	37.0	25.5	31.3	2 for 3	32.0	27.0	29.5	2 for 3
User 10	30.8	23.6	27.2	3 for 3	22.8	44.5	33.7	1 for 3
			26.7	29 for 30			46.8	7 for 30

Project Spearhead

Gesture Test Form

Subject Information

Name:

Age:

Sex:

How many hours a week do you play video games?

0	0-5	6-10	10-15	16-20	20+

How many years have you been playing video games regular (if applicable)?

Are you currently an athlete, dancer, or gymnast? (Please add what sport, style, or event you perform and number years active)?

Do you have any experience with the Oculus Rift? How many hours in total and in the last week have used the device?

Do you have any experience with any other form of Virtual Reality?

Are you prone to motion sickness or sea sickness?

Gesture Test Results

Gesture Speed Differentiation

Score: ____/10

Gesture Resolution when Twisting Your Head

Time through the course using Oculus Gestures:

Time through the course using Keyboard Controls:

Gesture Fatigue

Did they complete the course?:

Total Time:

Number of times stuck on a door:

Transition from Camera POV to Gesture Input POV

Did they get out of the red room?

How did they feel when the camera moved on its own?

Did they understand what to do next (without prompting)?

Project Spearhead

Art Test Form

Subject Information

Name:

Age:

Sex:

How many hours a week do you play video games?

0 0-5 6-10 10-15 16-20 20+

How many years have you been playing video games regular (if applicable)?

Are you currently an athlete, dancer, or gymnast? (Please add what sport, style, or event you perform and number years active)?

Do you have any experience with the Oculus Rift? How many hours in total and in the last week have used the device?

Do you have any experience with any other form of Virtual Reality?

Are you prone to motion sickness or sea sickness?

Have you participated in any of our other tests? Which ones?

Test #1: Color

Starting with the first color (dark blue), start reading the text on the plane in your head. After around 5 seconds you will be asked if the color is comfortable to look at. If 'yes' you'll move on, if 'no' you'll be asked to read the text out loud for as long as you can stand to.

Color:	Comfortable?	Total Time Read?
Blue – Darkest	Yes No	______________
Blue – Dark	Yes No	______________
Blue – Middle	Yes No	______________
Blue – Light	Yes No	______________
Blue – Lightest	Yes No	______________
Gray – Darkest	Yes No	______________
Gray – Dark	Yes No	______________
Gray – Middle	Yes No	______________
Gray – Light	Yes No	______________
Gray – Lightest	Yes No	______________
Green – Darkest	Yes No	______________
Green – Dark	Yes No	______________
Green – Middle	Yes No	______________
Green – Light	Yes No	______________
Green – Lightest	Yes No	______________
Red – Darkest	Yes No	______________
Red – Dark	Yes No	______________

Red – Middle	Yes	No	_______________
Red – Light	Yes	No	_______________
Red – Lightest	Yes	No	_______________
Teal – Darkest	Yes	No	_______________
Teal – Dark	Yes	No	_______________
Teal – Middle	Yes	No	_______________
Teal – Light	Yes	No	_______________
Teal – Lightest	Yes	No	_______________
Yellow – Darkest	Yes	No	_______________
Yellow – Dark	Yes	No	_______________
Yellow – Middle	Yes	No	_______________
Yellow – Light	Yes	No	_______________
Yellow – Lightest	Yes	No	_______________

Test #2: Font Test

The first part of this test explores font sizes on the Oculus. A texture will appear and the user will be asked to read through the letters on screen.

100pt	I K L W V S D R M N Q A Z P	Correct _____ / 15
75pt	U N G B H Y S J L A O P M B J Y	Correct _____ / 16
50pt	M H Y S W C Z J G L O I P F I M C D F V S Y B I	Correct _____ / 24
30pt	Q Y H F V B I G H K P O M S E F L X U I A	Correct _____ / 21
20pt	A B Y U D V T E Z O N R Y V G F W N J L D K U P X Y E F C V	Correct ______/ 29
15pt	N H U W M G S Q O I U N L P	Correct _____ / 14
12pt	J G Y U M S W C G T O P L Q E X A N	Correct ______/ 18
10pt	I M N O Y G V B S Y J Q	Correct _____ /12

The second part of the test will determine if users can comfortably read different font types comfortably.

Font	Read Accurately?	Comments?
Droid Sans Mono	Yes / No	
Times New Roman	Yes / No	
Arial	Yes / No	
DJ Gross	Yes / No	
Alien Encounters	Yes / No	
Papyrus	Yes / No	
Old English Text	Yes / No	
Dunkin Sans	Yes / No	

Test #3: Text Placement

Normal Plane:

Circle the coordinates the user is able to see clearly

A1	B1	C1	D1	E1	F1	G1	H1
A2	B2	C2	D2	E2	F2	G2	H2
A3	B3	C3	D3	E3	F3	G3	H3
A4	B4	C4	D4	E4	F4	G4	H4
A5	B5	C5	D5	E5	F5	G5	H5
A6	B6	C6	D6	E6	F6	G6	H6
A7	B7	C7	D7	E7	F7	G7	H7

Tilted Plane:

Circle the coordinates the user is able to see clearly

A1	B1	C1	D1	E1	F1	G1	H1
A2	B2	C2	D2	E2	F2	G2	H2
A3	B3	C3	D3	E3	F3	G3	H3
A4	B4	C4	D4	E4	F4	G4	H4
A5	B5	C5	D5	E5	F5	G5	H5
A6	B6	C6	D6	E6	F6	G6	H6
A7	B7	C7	D7	E7	F7	G7	H7

Test #4: Perception with Textures

Borderlands:

Trial 1:	Guess:	Actual:
Trial 2:	Guess:	Actual:

Counter-Strike:

Trial 1:	Guess:	Actual:
Trial 2:	Guess:	Actual:

Pixel Art:

Trial 1:	Guess:	Actual:
Trial 2:	Guess:	Actual:

Doodle:

Trial 1:	Guess:	Actual:
Trial 2:	Guess:	Actual:

Project Spearhead

Control Variations Test Form

Subject Information

Name:

Age:

Sex:

How many hours a week do you play video games?

0 | 0-5 | 6-10 | 10-15 | 16-20 | 20+

How many years have you been playing video games regular (if applicable)?

Are you currently an athlete, dancer, or gymnast? (Please add what sport, style, or event you perform and number years active)?

Do you have any experience with the Oculus Rift? How many hours in total and in the last week have used the device?

Do you have any experience with any other form of Virtual Reality?

Are you prone to motion sickness or sea sickness?

Have you participated in any of our other tests? Which ones?

Test #1: 180 Degree Turn

The user will perform this test first with a flat monitor and then with an Oculus. They will start in a room facing a blank wall. On their command the tester will hit the space bar causing the in-game camera to spin, facing the other side of the room. The instant the camera stops spinning on its own, the user will need to use the mouse or the Oculus to 'look at' (hover over with the cross hair) 3 different items in order marked A, B, and C. The time it takes to find these objects will be recorded.

Monitor Trials

Trial 1:

Trial 2:

Trial 3:

Trial 4:

Trial 5:

Is this version nauseating?

Do they find it difficult to locate the objects each time?

Oculus Trials

Trial 1:

Trial 2:

Trial 3:

Trial 4:

Trial 5:

Is this version nauseating?

Do they find it difficult to locate the objects each time?

Test #2: Sensitivity Variation

The user will be shown a green target that they must follow with a red dot fixed in the center of their screen. The longer they can keep the red dot on top of the green target the higher their score will be. This test will be repeated 3 times for each sensitivity setting. The first score will not be counted.

Sensitivity Setting

.25 Score 1 _____ Score 2 _____ Score 3 _____
Average (2&3)_____

.5 Score 1 _____ Score 2 _____ Score 3 _____
Average (2&3)_____

1 Score 1 _____ Score 2 _____ Score 3 _____
Average (2&3)_____

2 Score 1 _____ Score 2 _____ Score 3 _____
Average (2&3)_____

4 Score 1 _____ Score 2 _____ Score 3 _____
Average (2&3)_____

Test #3: Invert Y-Axis

The user will pilot through a tunnel of obstacles using the Oculus Rift's tilt sensors. For one of the tunnels the controls will be left normal. For the second tunnel the vertical axis will be inverted. The times will be compared to see if inverting the vertical axis with the Oculus is a usable option. Like the previous test only the 2nd and 3rd times will count.

Normal Controls

Time 1 _____ Time 2 _____ Time 3 _____ Average (2&3)_____

Inverted Controls

Time 1 _____ Time 2 _____ Time 3 _____ Average (2&3)_____

Project Spearhead

Motion Test Form - Walking

Test #1: Walking in real life while viewing a virtual world

Straight Line Course

Draw the path the user takes.

Blindfold:

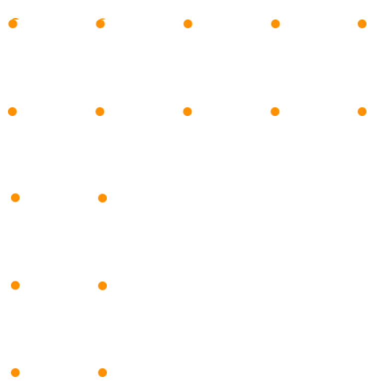
Oculus Rift:

Which course was easier to navigate?

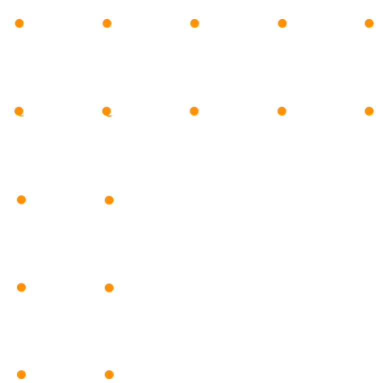
Right Angle Course

Draw the path the user takes.

Blindfold:



Oculus Rift:



Which course was easier to navigate?