VR First-Person Space Shooter Game in Unity

Sarah Radzihovsky
Stanford University
450 Serra Mall,
Stanford, CA 94305
sradzi13@stanford.edu

Tien-Ning Hsu
Stanford University
450 Serra Mall,
Stanford, CA 94305
tiening@stanford.edu

Abstract

For our project, we created a first person VR shooter game. Our goal was to create as enjoyable an experience as possible both in terms of comfort and immersion by designing an environment that captivates the audience. To accomplish this, we experimented with the depth of field, image corrective technique, sound designs, world designs, movement designs, different ways to increase the game's difficulty, and much more.

1. Introduction

There are many shooter games in VR already, however many of these games do not provide an attractive, comfortable, and immersive experience for gamers. In examining why, several key factors can contribute to a game not being attractive to a user: (1) Eye fatigue and motion sickness (2) Not immersive (3) Not challenging enough.

We had previous game building experience from CS248 and chose to layer on top of a game Sarah had made from her final project in CS248 to make an engaging virtual reality game. We experimented with some tools and concepts we learned in class such as incorporating the VRduino into our game and adding depth of field to the space scene the user plays in.

2. Related Work

2.1. World Design

With the progression of virtual reality technology, the exploration of developing video games in virtual reality had became a large area of focus. In fact, there are already numerous first person shooter games built out in virtual reality. Many of these games, such as Arizona Sunshine and Hover Junkers, the player is tethered to the ground and can only explore the world by moving forward, left, and right. For our world we chose to keep our VR game in an infinite space abyss when transferring the game to VR because the endless nature of the environment more attractive in VR. Because the user has no ability to actually move around the world more than rotate their first-person spacecraft and direction they are flying into, we were able to make the world look quite good and believable from the vantage points the player is limited to, at very little expense.

Among first person shooter, PC games, Titanfall drew a large audience because its commitment to world design, allowing its players to travel across walls, over chasm, and high into the air. Doom similarly was designed around an extremely beautiful and immersive world that users enjoyed exploring in game play. On the other hand, games like World War II Combat: Iwo Jima (PC) flopped because the world design of the game was ugly and the difficulty of the game was increased too high such that the best way to gain tactical information is to dive into the low-res action, get killed, and start all over again. Similarly, when playing Terrawars: New York Invasion, players had difficulty believing the world and immersing themselves because the virtual environment was bare and dull. Also attempted to include a story and game-play for the player to have more agency but these were also not developed out well enough to make for smooth motions and seamless play. *NRA Varmint Hunter* was found unsuccessful because players complained that the targets were the same size as the bullets.

2.2. Sound Design

A large part of engagement in video games and VR experiences is the sound. A study concluded that players who experience accurate spatial sound in their environment generally feel far more immersed [1]. This same paper showed the use of spatial-sound techniques in a virtual environment contributes to "higher levels of presence experienced, and does so with a medium to large effect" [1]. They measured this by asking participants to enter a virtual scene of a forest where some participants experienced it with sound and some without sound. The questionnaires filled out after indicated that the users felt more immersed with spatial sound present.

2.3. Motion Sickness

An important limitation that often disrupts the user's experience when using VR is the feeling of motion sickness while wearing the HMD. There has been a lot of research and work done on causes motion sickness as well as how it can be prevented. According to Hettinger et al., to reduce VR induced motion sickness, the first step is to identify the cause of the sensory mismatch in VR. In their paper, they claim that there are two primary factors that contribute to vestibular and non-vestibular discontent. The first is the mismatch between perceived motion toward the direction you're facing and the player being physically stationary. A study by Columbia supports this point stating that "High-quality tracking systems can minimize the mismatch between a user's visual perception of the virtual environment (VE) and the response of their vestibular system, diminishing VR sickness for moving users. However, this does not help users who do not or cannot move physically the way they move virtually" [2]. The second is the delay between head rotation (which redirects the direction in the world the player faces) and the movement of the point of view in the virtual world. Both cases are present in our game and posed interesting challenges for us to make design choices to mitigates these factors [3].

Some studies showed that having good resolution and a large field of view are important contributors to combating motion sickness in VR. After experimenting with the resolution of our game we concluded that increasing the resolution didn't have a huge effect on the user, however did cause latency issues at times so we deemed resolution improvements more disruptive than helpful. Testing out the original field of view that comes with the game Unity exports, we also determined that the given field of view was sufficient for the effect we were trying to achieve.

3. The Game

3.1 Game design

In the game, the first person shooter flies through space in their spacecraft, trying to dodge obstacles (by moving or shooting them) and shooting green capsules, which give the player more ammunition. Hitting a red capsule or running into an obstacle penalizes the player by taking away ammunition. The farther the player travels through space, the larger their final score when they run out of ammunition.

We aimed to create an enjoyable VR space shooting experience both in terms of comfort and immersion by designing an environment that captivates the audience through a beautiful intergalactic environment, a diverse set of obstacles, increasing levels of difficulties so players stay challenged, sound effects, and intuitive movement control for the user. We will discuss each aspect in more detail in the implementation section.

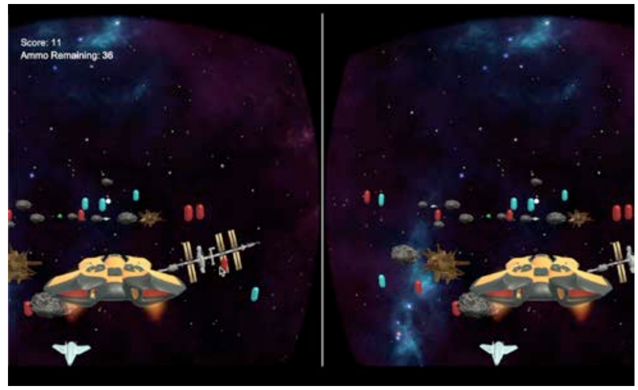

Figure 1. The VR game scene on the LCD panel.

For the VR hardware support, we used the ViewMaster housing a LCD panel. To make most use of the equipment handed out in class, we decided not to use iPhone but used LCD panel, with VRduino attached to the ViewMaster to record the user orientation. Additionally, we provided the user wireless mouse as the shoot trigger as well as wireless headphone to serve the best immersive environment.

Figure 2. VR Game setup

3.2 Implementation

3.2.1 Movement

When redesigning the game for VR, we first evaluated the options of player movement. After some experimentation, we decided that limiting the player's movement to only changing flying direction was most effective. Allowing the player to move more than this increased the difficulty of the game in an unpredictable way and created disorientation that could induce motion sickness.

We also decided that the most comfortable and engaging way to steadily increase the difficulty of the game in VR was to linearly increase the velocity of the player's spacecraft through the obstacles the further the player moves into the game.

To control the players, motion, we attached a Vrduino to our HMD and used the read in data to control the player's orientation (i.e. the camera and spacecraft). Thus, the spaceship flies in whichever direction the user turns towards and the other game objects spawn in that direction in front of the spacecraft.

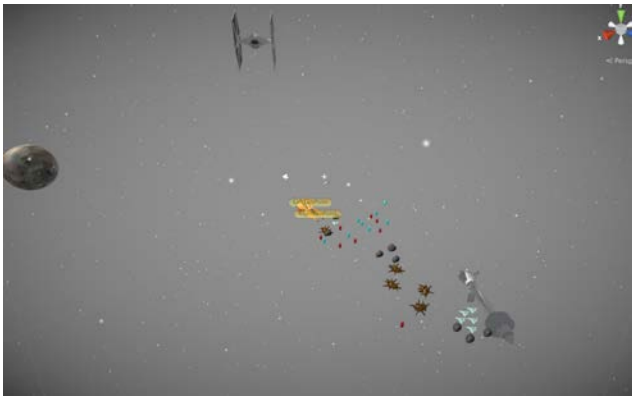

Figure 3. Birdview of the game. As VRduino sending orientation information, the main camera and the spaceship rotates accordingly.

3.2.2 Sound Design

Our VR game's graphics not of high enough quality to rely on always seeing the laser shots, so we found that sound effects are necessary to cue the player in as to when they are successfully targeting an object. We included two forms of audio to help our players feel more connected to the game. The first form of audio are the reactional sounds that are played in response the (1) the user shooting and (2) an explosion sound when the player's shot connects with an object. Because our game involves many obstacles, we found that the reactional sounds helped to orient players amid the commotion. The second form of audio is the background sound track that gradually gets more intense as the player progresses further on their voyage and combines with the speed-up effect to suck the user in.

3.2.3 Environment

The game's empty space environment is predominantly a combination of a nebula skybox with a infinite starfield. The skybox was carried over from the initial PC version of the game and generated by following an online tutorial and allows us to simulate and infinite space as the user travels through the virtual world. The infinite starfield was created using Unity's Particle system and was exaggerated for the VR version of the game. Even without exaggeration, infinite starfield is much more noticeable in the virtual environment and adds specs of bright light that zoom past the viewer in a manner that is subtle enough to avoid motion sickness but to make the world more believable. Additionally, we believe the space environment is very beneficial for VR game in the sense that even if the initialization of VRduino is not perfect, it's hard for user to feel dizzy since there is no exact "correct" orientation in the space.

Obstacles, capsules, and orbiting objects make up the matter that spawn in front of the player as they fly through the game. Each model was either made in Maya or premade and downloaded before being textured in Unity by solid colors or images.

Layering our game with the post-processing component in Unity allowed us to further enhance our game and make it more realistic using several concepts discussed in class. For instance, we experimented with depth of field and found that a focus distance of 15 and f-stop of 1.3 because it allowed our player to focus on their near targets. Objects too close or too far, as a result, were blurred, becoming clearer as they approached the focus distance. We activated the "screen space reflection" feature in the post-processing profile to illuminate the capsules that were textured with reflective materials and make them stand out amongst the other obstacles given their heightened importance to the game. We also added antialiasing to the profile to smooth the edges of our models as they move through the game and ambient occlusion to darken areas where two edges meet in models and intensify self-shadowing. Lastly, we added a motion blur feature that only is noticeable when the spaceship's velocity reaches a high enough value and the obstacles begin moving past more quickly.

Figure 4. The depth of field effect on the game.

4 Conclusion

Our demo game provides a fairly fun and comfortable 1st person shooter VR game. Although shooter games in VR face several limitations, motion sickness and latency being the largest worry, in general, the immersiveness and ability to move around the game add to the attractiveness and entertainment value of VR games.

References

[1] Sandra Poeschl, Konstantin Wall, and Nicola Doering. "Integration of spatial sound in immersive virtual environments an experimental study on effects of spatial sound on presence". In: Virtual Reality (VR), 2013 IEEE. IEEE. 2013, pp. 129–130.

[2] Fernandes, Ajoy S., and Steven K. Feiner. "Combating VR sickness through subtle dynamic field-of-view modification." 3D User Interfaces (3DUI), 2016 IEEE Symposium on. IEEE, 2016.

[3] Hettinger, L. J., Riccio, G. E. (1992). Visually Induced Motion Sickness in Virtual Environments. Presence: Teleoperators and Virtual Environments, 1(3), 306-310. doi:10.1162/pres.1992.1.3.306