

# VR Space Shooter Game with Leap Motion

Xuandong Lei, Chen Zhu

{xuandong, chen0908} @stanford.edu

Stanford University

## 1 Introduction

We would like to build a First-Person shooter game using the HMD, IMU and Leap Motion (if we can borrow one). FPS games, like CS (Counter-Strike), are really popular around the world. We want to bring that experience into the VR world.

## 2 Goal

The game is similar to common First-Person shooting games. The game board would always be the player's view. The player could move/rotate his head around to see the world around him. The enemies will keep coming to the player from all different angles. The player could fire missiles to strike the enemy. The game will end if the player's HP goes to empty.

We would like to integrate orientation, hand tracking into a virtual game environment, creating an immersive gaming experience with the VR devices we have.

## 3 Method Overview

### 3.1 Methods

- Virtual Environment:

Our first step is building a virtual 3D environment with unity3D. We want to design and implement a complex and real 3D shooter game scene. The scene should be first-person view, where player can inspect the surrounding environment by purely rotating his/her head. To do this, we will use Unity and some existing unity assets.

- Orientation:

The next step is to integrate the 3D environment into a head mounted display. We want to simulate a more real shooter scene, where the player can control his/her view simply by rotating head instead of pressing keys on the keyboard. So we will need an IMU to detect the rotation angle of the player. We will use the same method as in hw5 to do the orientation tracking.

- Missile Firing:

The last thing is to make the game more interactive. In traditional shooter game, player control his/her guns and shoot by pressing keys on computer's keyboard. Here we would like to utilize Leap Motion (or trackpad) to track the position of the player's hand, so that he/she can control the direction of shooting by moving his hands, like what a real shooter will do.

We will use track pad to accomplish this functionality if we can't get a Leap Motion device.

### 3.2 Device Needed

- Unity  
Create the virtual environment.
- Head Mounted Display  
Display the game and environment.
- IMU  
Orientation tracking for changing the player view.
- Leap Motion (if provided)  
Control the place of the player's firing direction.

- TrackPad (if Leap Motion not available)

We will use trackpad to control firing if leap motion is not available.

## 4 Timeline

- **Week1:**

Build a basic game environment, integrate the IMU to control the orientation of player's head. Make sure we can correctly change the player's view.

- **Week2:**

Optimize the game environment, make it more interesting and complex. Integrate the leap motion or trackpad to control the firing system.

- **Demo Day:**

Make an interactive and interesting demo and have fun!

## 5 Optional Work

If time permits, we would also like to integrate the following functions into our game:

- More interactive game:

We would like to add more interactive stuff to our game. For example, we want to add different kind of guns to pick up by the player. Also to add health supply. But these are depended on the time. If we have time, we would make the game more interactive and interesting.

- Player's position tracking

Right now we a player can only control perspective by rotating head. We would like to integrate the player's position tracking into the game to make the playing experience more realistic. Here a kinect might be helpful to help locate the player, and with the location information of the player, we can let the player control his position in the 3D virtual scene by himself.

## 6 Reference

1. Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.
2. Marin, Giulio, Fabio Dominio, and Pietro Zanuttigh. "Hand gesture recognition with leap motion and kinect devices." Image Processing (ICIP), 2014 IEEE International Conference on. IEEE, 2014.