CV Shooter

Yunkun (Ricky) Lu

# Description:

A top-down space shooter game controlled by red color detection with opencv, hence the name CV Shooter.

# Competitive Analysis:

There are a lot of shooter games, especially top-down shooters. My project will be a simple top-down shooter design, with the player’s ship at the bottom and enemies spawning from the top, with bosses as well as normal small enemies.

However, my shooter will be controlled by opencv, a python module that allows the detection of objects with the webcam. I plan on using a red object to make the spaceship move around, instead of the traditional keyboard/mouse movement. This also means the game will be harder to control, hence the margin of error should be larger in this game.

The game should also have a smart level system where the difficulty of the game adjusts to the player’s performance.

# Structural Plan:

I plan on having a startup screen that allows the user to go into the tutorial or the main game, and an end game screen that allows the user to go back to the startup screen or try again. These main functions should be in the same file.

I will have a file separately for all the classes that I’ll be writing.

The most interesting class is probably the Level class, which contains information needed for the spawner to spawn Enemies accordingly.

I have a folder named “Assets” in which I will put all the art assets. The assets folder is also organized into “Enemy”, “Player”, “Bullet”, etc…

# Algorithmic Plan:

I will first use opencv module to create a controller by detecting color in a specific range and identifying the center of the color as a point. I will then project two lines on the camera to represent areas the red color needs to be in in order to move the spaceship to other directions.

I will organize my objects in pygame groups, which is a useful tool for drawing everything at once.

The adaptive difficulty will be achieved by measuring the “performance”, which is the number of hits the player took in the past 10 seconds. Then the enemy’s speed, bullet’s speed will decrease, and the spawn rate will also decrease.

I plan on having a span function that takes in a class object called level, and spawn each level according to the class properties.

# Timeline Plan:

TP1: base game working with opencv control, playable with health/scoring.

TP2: Title screen working, game has levels with a boss at the end after all other small levels. Enemies should be “Smart” and adjust difficulty based on the performance of the player.

TP3: Final project, with everything working and try to optimize code so it runs faster.

# Version Control Plan:

I will be using GitHub and back up my code once per day.

# Module List:

Pygame, Opencv

# TP2 Update:

I added a small “Team formation” type enemy.

The enemy moves as a group, two of them shoot straight at the player, one other shoots towards the player’s right, the last one shoots towards the player’s left. This makes it harder to dodge.

Difficulty of the game is reflected by the interval between enemy fire and the repair tool drop rate/heal amount. The better the player performance, the more frequently the enemies will shoot. The worse the player performance, the more frequently the repair tools will drop, and it will heal for more health.

# TP3 Update:

Added a bomb for the player that clears the screen

Added sound effects for shooting and explosion

Added the challenge enemy smart AI that dodges player bullets, shoots predicting splitting bullets, and shoots a homing bullet that chases the player.

Added an endless “infinite” mode.