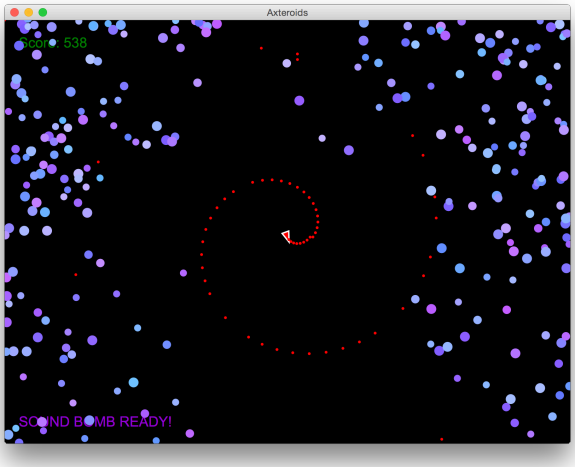
CS5041 P1: Axteroids

Student 120008200
University of St Andrews

INTRODUCTION

This implementation, inspired by the classic video game 'Asteroids' takes inspiration from the 'Bullet hell' or 'maniac shooters' genre of video games. Gameplay is simple: shooting rocks increases player score, hitting a rock ends the game.

Figure 1: Shooting whilst turning



GAMEPLAY

Rocks (in various shades of blue/purple) spawn outside the visible bounds, with random initial velocity. Missiles are fired on press of the spacebar.

Figure 2: A sound bomb destroys all visible rocks

Movement is controlled by the mini-joystick on the controller. Ship acceleration (positive and negative) is applied according to the exact y-axis value of the stick allowing for precise control of ship movement. Likewise, turning is achieved by directing the stick to the left or right. This high level of precision allows the player to accelerate or turn at exactly the rate required.

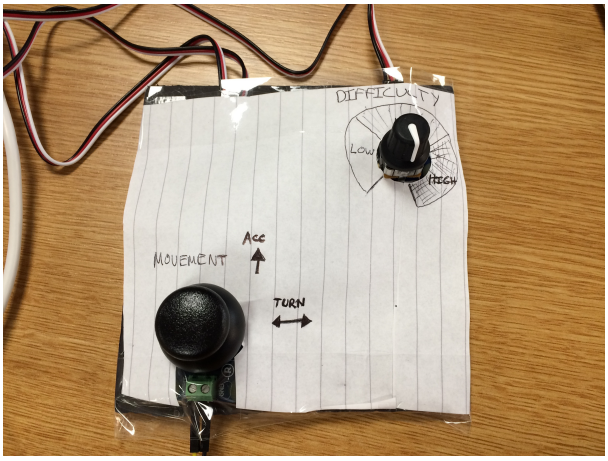
An interesting feature added to the game is the use of a 'sound bomb'. The sound bomb is a bomb which takes some time to recharge and is detonated by a loud sound e.g. shouting. On explosion, the sound bomb destroys all visible rocks, helping the player in times of peril! An indicator can be seen at the bottom left of figure 1 ("Sound bomb ready!") when the bomb is charged and ready to be used.

CONTROLLER

The controller is made up of two parts (figures 3&4). A paper overlay helps the player to easily pick up the controls, and understand how position of the sensors is used in the game.

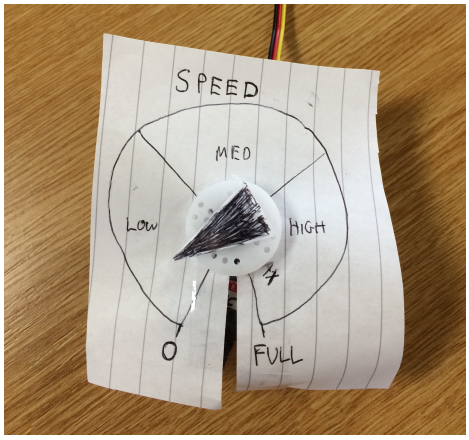
The rotation sensor controls rock spawn rate. Higher spawn rate increases difficulty, but allows the player to score points more quickly. The mini-joystick can be pressed down which causes the ship to stop in its current position, acting as an 'emergency stop' button. The sound sensor is used to fire the 'Sound bomb' as explained previously. The final input sensor used is the light sensor. The value of the light sensor affects the colour of the game background, making it darker in darker light conditions and vice versa.

Figure 3: Controller



The second part of the controller is the speedometer as shown in figure 4:

Figure 4: Speedometer



The speedometer is created using the servo phidget device, with a paper underlay, indicating the actual speed of the ship.

HIT TESTING

Collision detection has been fully implemented, enabling the shooting and losing components of the game. For hit testing purposes, the ship, missiles and rocks are treated like perfect circles. Hit testing of the ship, missiles and rocks is simply checking that the distance between any two positions is greater than the sum of their radiuses. Otherwise, a hit is detected, and the appropriate elements are destroyed.

CONCLUSION

The use of phidget sensors makes for exciting interactive gameplay. Thinking of ways to use different sensors to add new dimensions to the game is challenging but fun. In particular, I think the 'Sound Bomb' feature used in this implementation is a great way of making sound integral to the game. Video game players are renowned for getting frustrated or making sounds reflecting their feeling of the state of gameplay. I believe that integrating these urges in to the game itself, both encourages an action that players enjoy, and makes them more involved.