Nerf Firing Milestone Report

Craig Hiller, Kevin Wu, Leo Kam, Christopher Hsu

I. Introduction

THIS project will create an automated NERF turret that can detect a target (a human face), aim, and fire at the target. The project will model the detection of a target and aiming as a state machine governed by the combination of RGBD camera data and other sensor inputs to correctly aim the turret. The goal will be to accurately detect a target and aim/fire accurately for maximum effect.

II. Project Requirements

Based off concerns in the presentation, list out explicit minimum requirements.

III. System Components

Description of the various components (see next sections) + figure showing how they connect.

IV. NERF Gun Turret

The NERF Gun and Turret are the actuators in our system. So far, we have been able to replace the two trigger switches in the NERF gun with relays connected to our micro controller. We use relays since they are easily controlled by a micro controller. There NERF Gun we have is a NERF CS-18, this was chosen since it is relatively small and light but most importantly, it is controlled entirely electronically which allowed us to swap out the switches of the gun. The plan to build the turret has changed, we are currently planning on making a turntable out of some bearings (ordered), a motor, and a winch. The motor will turn the table and be controlled by the micro controller, this is for rotating the blaster too line up in one in one direction. The winch will be mounted on the turntable and connected to the NERF gun. The winch will be turned by a motor until the target is in position.

V. Face Recognition

We have decided to use an Intel RealSense 3D Camera to handle face recognition. This camera was chosen since it has a small form factor and provides RGBD information at 30 frames per second. A limitation of this camera is that it must be connected to a Windows 8 computer via USB3, but it only came with a short cable. To get around this limitation, we ordered a 2m USB3 extension cable. On the software side, we have C++ code that reads from the camera and uses OpenCV to detect faces. We are still working on creating a connection from this computer to the micro controller so that we can move the turret based off of a face's position in space. Below shows (counter clock-wise from bottom left) input put image, face detected image, and depth image.

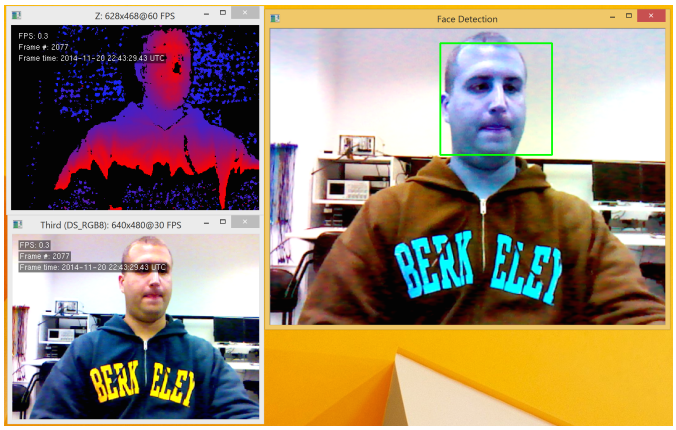

Fig. 1. Depth Map, RGB Image, and Face Detection

VI. Controller

Updates on the controller for the turret + figure showing state diagram

VII. Communication

Updates on various types of communication methods between camera + controller.

VIII. Future Plans

Future stuff