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 and aiming of a target as a state machine governed by the combination of RGBD camera data and other sensor inputs to correctly rotate the turret and incline the NERF gun to aim at the target. The goal will be to accurately detect a target and fire accurately for maximum effect.

II. Project Requirements

The general requirements that can be measured to determine the success of our turret are as follows:

- The turret platform supports full 360° rotation.
- The turret platform supports an incline angle between 0° and 30°.
- The turret is able to perform facial recognition.
- The turret can accurately hit a stationary target (human face) within the range of 5 meters.
- In the case of communication error, turret support successful tracking target to the last known location.

III. System Components

A diagram illustrating the system components for this project is shown in Figure 1.

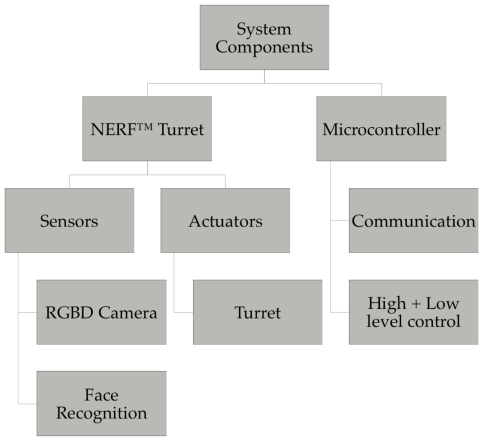

Fig. 1. System components hierarchy

In general, the system is divided into two overarching components: (1) the NERF Turret and (2) the Microcontroller. Each component can be subdivided further as shown in the diagram. In general, the NERF Turret component is responsible for providing the actuators that control the rotating platform - a turntable made of bearings, motors, and a winch -, an electronically triggered NERF gun mounted on the platform, and the RGBD camera sensor that performs the face recognition. On the other hand, the microcontroller component is responsible for providing the controller api calls that appropriately handle the high and low level control of the system. The microcontroller is connected to the actuators via wired connections, and to the sensors using a WiFi connection via the Adafruit CC3000 chip.

The four person group is divided into two subgroups that each handle one of the two overarching components. That is, one group handled the difficulties of acquiring a NERF gun, constructing a turret, and setting up sensors, while another handled the control system and the communication.

IV. Updates

The updates that we have for the various system components are described below.

A. NERF Turret Actuators

Fig. 2. The CS-18 that has been cut open to allow electrical input.

The NERF gun and the motors on the turret are the actuators in our system. We selected the CS-18, which was chosen since it was not only relatively lightweight, but also entirely electronically controlled which allowed us to swap out the switches of the firing mechanism. Figure 2 below shows the result of disassembling the gun to allow electronic firing.

Currently, we have disassembled the NERF gun and reverse-engineered it to the point where we can electronically trigger it to fire a NERF dart. In addition, we have obtained two relays to replace the trigger switches in the gun in order to connect it to the microcontroller. Relays are used since they are more easily controlled via a GPIO pin on the microcontroller.

For the turret we have begun drafting what we will be using to rotate the turret and how we will mount the gun on it.

B. RGBD Camera and 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. The RGB information is important in order to perform face recognition, and the depth information will be useful to allow the NERF gun to properly target the face once it is detected.

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 2 meter USB3 extension cable.

On the software side, we have C++ code that reads from the camera and uses OpenCV to detect faces. Figure 3 shows a demo of this behavior.

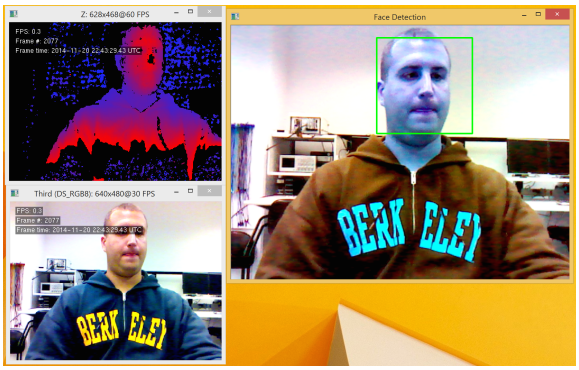

Fig. 3. Going counter-clockwise from the top right - Depth Map, RGB Image, and Face Detection

C. Controller

```
#include "mbed.h"
#include "TSISensor.h"

DigitalOut flywheel(PTB8);
DigitalOut trigger (PTE20);

TSISensor tsi;

float eps = 0.001;

int main() {
  flywheel = 1;
  while(1) {
    if (tsi.readPercentage() < eps)
      continue;
    trigger = 1;
    wait(0.4);
    trigger = 0;
  }
}
```

Listing 1. Basic demo code controlling the NERF gun.

We have decided to use the MBED microcontroller to handle the control algorithm. Currently, we are able to control firing of the NERF gun with the MBED, via wired connection between the microcontroller and the relays. In the code shown in Listing 1, we have set the microcontroller to trigger the NERF gun when the capacitive touch sensor is touched. Figure 4 shows the chip connected to the relays via the proper GPIO pins.

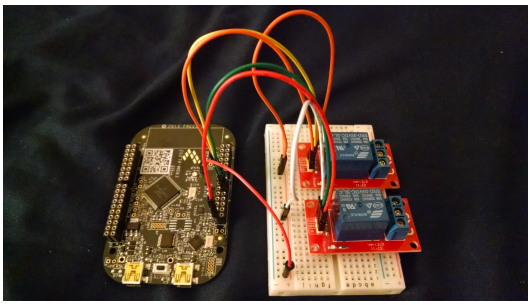

Fig. 4. An image of the MBED chip connected to the two relays. These would be connected to the NERF gun to electronically fire the gun.

D. Communication

We have selected WiFi for our connection scheme mainly since it is easy and serves its purpose well for our requirements. We are tracking a static target, and, therefore, are perfectly fine with some latency with our communication protocol. However, we have had several problems using the Adafruit CC3000 chip with the MBED processor to connect to a network, and have yet been able to maintain a consistent connection.

V. Future Plans

This is what we expect to accomplish moving forwards:

1) Build the turret platform.
2) Implement the targeting algorithm.
3) Set up server for communication between sensor and controller, via WiFi connection.
4) Integrate all the components.

A. Turret

The plan for the turret has changed, and 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 microcontroller, which will be used to rotate the gun to line up in one direction. The winch will be mounted on the turntable and connected to the NERF gun, which can be be turned by a motor until the target is in position.

B. Communication

We are still working on creating a simple server connection so that the microcontroller can recieve data from the face recognition software and, therefore, move the turret based off of a face's position in space.

References

[1] Bjoern Hartmann and Ben Zhang, "CS294-84 / ME C290U: Interactive Device Design," http://husk.eecs.berkeley.edu/.

[2] B. Zhang, "CC3000 Demo," http://developer.mbed.org/users/nebgnahz/code/CC3000_demo/.