PiEye

EyesOnThePi’s

Kevin Masse, Aaron Wise, Noah Mintz-Roberts

Ultimately designed with the blind community in mind, we designed a raspberry pi with dynamic haptic feedback, a device which allows the wearer to effectively navigate complex environments using the cues of gloves as they learn to understand it. This version is ultimately a prototype for our actual entry in the CS fair; we plan on improving the sensor array, compressing the hardware of the glove, as well as custom make a weatherproof case. The PiEye is a mobile raspberry pi application, using sensors to allow the blind have the ability to perceive some aspects of sight. The base design is incredible versatile, allowing for personal sensor (or other necessities) expansion and modification. The mobile design also allows it to be worn anywhere and automating the software (startup, updates, etc.) to help make it easier to be booted with limited technical knowledge or ability.

Definitions, Acronyms, and Abbreviations:

## Acronyms

IR - Infrared

RPi - Raspberry pi

V - Volt

MM - Millimeter

NM - Nanometer

USB - Universal Serial Bus

FOV - Field of View

ASL - American Sign Language

## Sensors/Components

Infrared Distance Sensor

Ultrasonic Distance Sensor

UBEC - Universal battery elimination circuit

ADC - Analog digital converter

## 

## Definitions

Infrared - a range of the electromagnetic spectrum above red from 800nm - 1mm

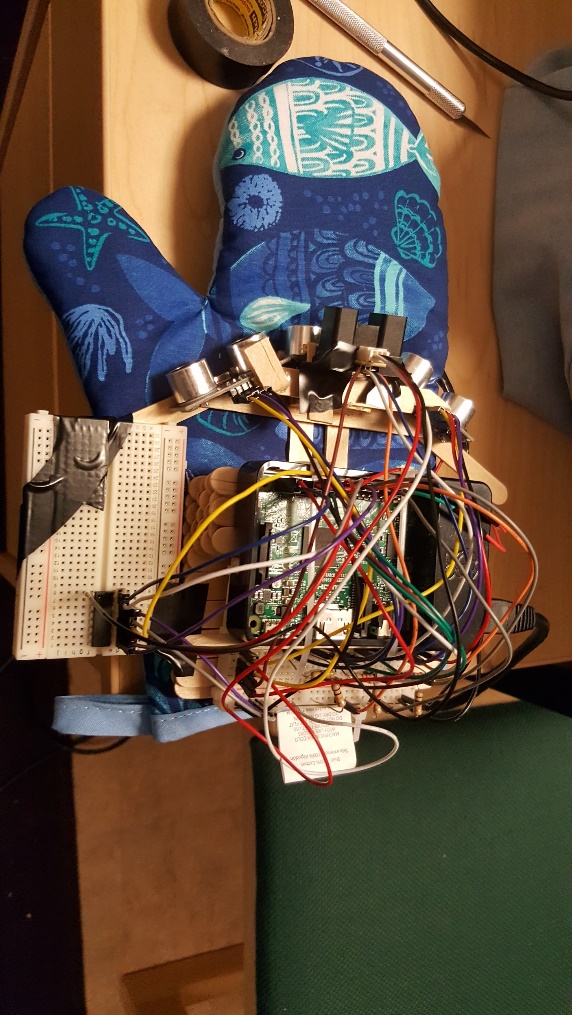
Ultrasonic - sound waves of a frequency above human hearing capabilities

Solder - joining less fusible metals, typically with a low-melting alloy

Haptic - relating to the sense of touch

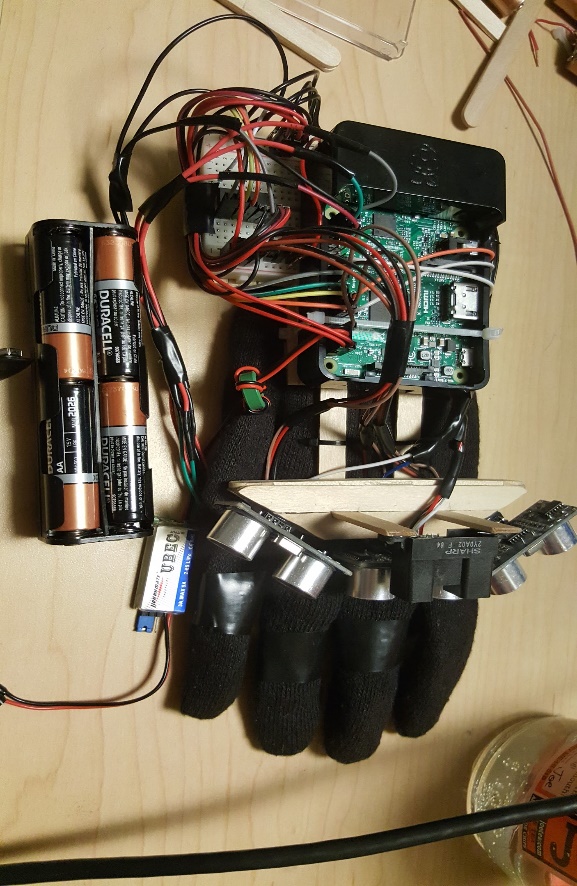
Laser - a device which generates beams of coherent monochromatic light by stimulating atoms/molecules to emit photons

Field of View - the extend of the observable world that can be seen in a single instance

 Servo - a powered mechanism producing motion or forces at a higher level of energy than the input level, e.g., in the brakes and steering of large motor vehicles, especially where feedback is employed to make the control automatic.

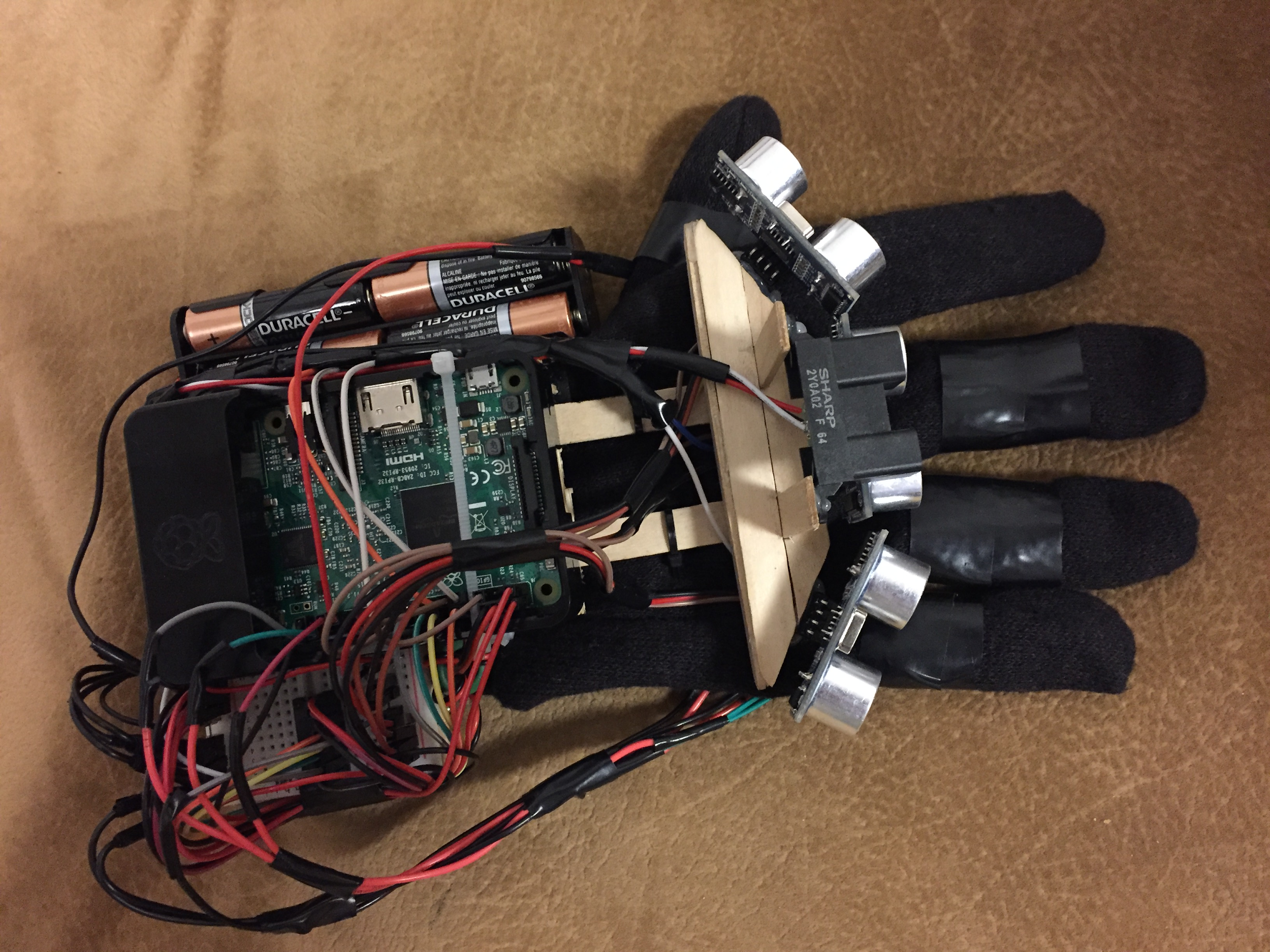
[First Prototype Picture]

* USDS array
* Oven mit
* Sprawling hardware
* Pre-IR capability
* Only wired power



[Final Prototype Photos]

* Smaller fingered glove
* Compact wiring & hardware
* IR distance sensor
* AA Battery pack with UBEC
* Runnable from startup



Our prototype has been a constantly evolving project. Originally being very dependent on the breadboard, we were ultimately able to reduce this dependency almost completely with the use of an ADC (Analog to Digital Converter). The ADC adds 8 channels of 10-bit analog input to your microcontroller. This allows us to read the output from our IR sensor. Using Software SPI means it only uses 4 pins on the PI as well. Other features of our final prototype include an optimized array of 3 ultrasonic sensors, as well as using an UBEC to allow us to properly integrate a AA pack into the raspberry pi at a safe input voltage. Our output model for the haptic feedback sensors is implemented using an exponential scale, to more effectively differentiate between distances.

**Hardware:**

Connecting Ultrasonic Distance Sensors: Most pins on the Ultrasonic distance are easily plugged directly into the PI, power, or ground, however due to the voltage limit of 3.3v input on the Gpio pins a simple circuit had to be made to step down the voltage on the echo pin of the ultrasonic distance sensors.

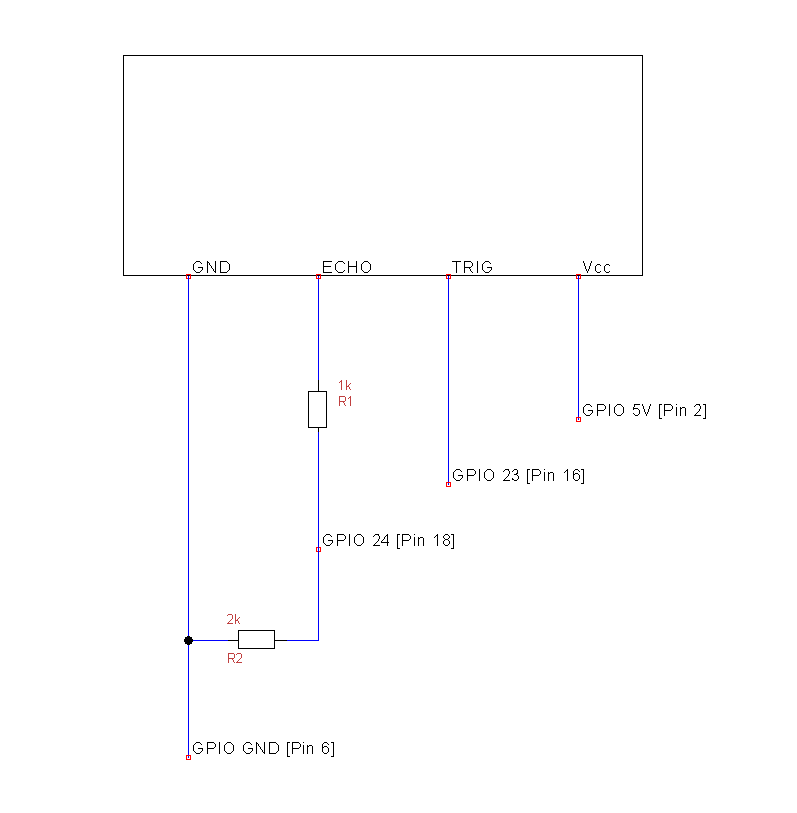


Diagram of the circuits used to connect the ultrasonic distance sensors

After the circuits where complete and tested to minimize breadboard usage, we have soldered resistors directly in line.

Connecting the IR sensor was more difficult because our IR sensor outputs in analog but he PI can only take digital. To solve this issue we used an analog to digital converter.



The MCP3008 ADC we used to translate the analog IR into digit output for the Rpi

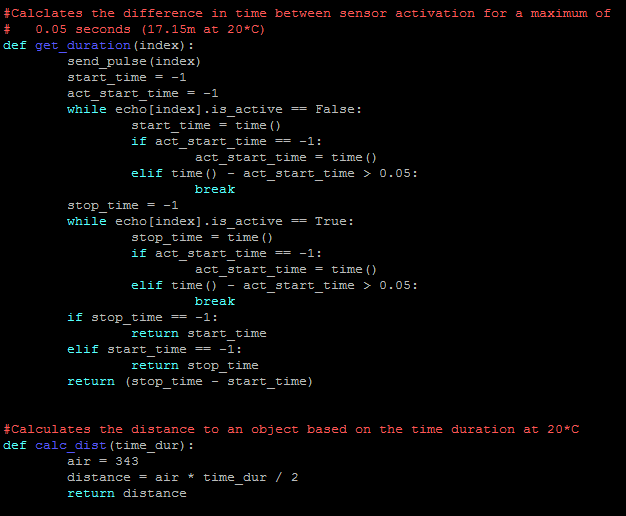
(Source: http://www.raspberrypi-spy.co.uk/2013/10/analogue-sensors-on-the-raspberry-pi-using-an-mcp3008/)

The ADC was connected to the PI using Software SPI through 4 of the GPIO pins. This is easier to set up and more flexible than hardware SPI.

After all the sensors were connected four vibration motors where soldered to wires with pins and connected to ground and a GPIO pin on the PI. Using pulse width modulation the PI can control the motors in correspondence with the sensors by sending signals out of the connected GPIO pins.

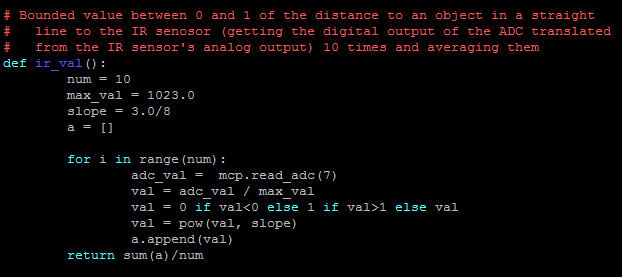
Another hardware challenge was to power the PI to a mobile power source such as a battery pack However a PI has very specific power requirements with just batteries it is difficult to supply correct voltage while providing enough amps for the PI and the sensors. To solve this we used a Universal Battery Elimination Circuit. The UBEC allowed us to use larger battery pack to provide enough amps as well as stepping down the voltage to a steady 5v.

Significant portions of code:



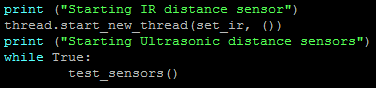
**Time and Distance calculation of ultrasonic distance sensors:**

These two functions are responsible for calculating the difference in time that the sensor is active and using that to determine distance. The function only waits a maximum of 0.05 seconds until breaking to avoid an infinite loop in which the signal is never received. The distance calculation function uses the speed of sound in air at 20°C (343 m/s) and divides by 2 to account for the distance to and from the object.



**Reading the digit input values from the ADC:**

A function which runs through a loop a number of times and reads the value from the ADC’s 8th index (0-indexed) and divides that by the maximum value of 1023 to scale it between 0 and 1 for the vibration motor. The value is then scaled by being raised to the ⅜ and then averaged to determine the approximate distance to an object. The value raised to a power so that it would still be bounded between 0 and 1 but the curve of the equation could be adjusted to facilitate a better feel and range of the glove.



**Different threads to concurrently run the two (USDS and IR) clocks:**

The data of the IR sensor is being taken in another thread which allows it to run faster than the ultrasonic distance sensors which require a time delay to function. This results in the IR sensor being able to be much more accurate and faster than the ultrasonic distance sensors can be.



**Program running on startup using crontab:**

We edited the crontab with “sudo crontab -e” in order to schedule the cron to run the program on the startup of the raspberry: this allows us to have the program be run and for the glove be “on” immediately once it receives power and has started up.

**Where are we going from here:**

We are going to submit a revised version of this project in the fall to the CS fair, we have many ideas to generally improve the design, as well as interesting sensors we could implement to further expand the device usage range. There are many things that a sensor like this could effectively communicate to a disabled user, albeit color, gps location, temperature etc. the possibilities of the disability assistive range of this tech are limitless.

### 

Our initial estimate for time was ~36 hours in our initial project proposal. Ultimately, this was a realistic goal as a true final prototype won’t be finished until the CS Fair in the Fall. We spent countless hours realistically working on the project so far, and are continuing to do so.

In regards to our proposed budget, the inclusion of the Model B 3 pi kit bumps it up near 250% what we imagine being able to reduce the cost to using a cheaper (yet just as reliable) microcontroller alternative. After factoring out the pi 3, the final costs of the design is roughly 60$, depending on the amount of bulk materials used (i.e. wire, solder, filament etc.), these types of materials the total cost of a spool doesn’t accurately reflect the miniscule amount that you can potentially use in an efficient build.

Final Reflections:

We’re really excited with how it’s turned out so far, this class has been an incredibly unique opportunity to get creative and make a project that both reflects our technical ability and our dedication to the UVM community.

There’s been many challenges along the way;

* Crunching numbers and coding without the IR sensor for benchmarking analog voltage to distance ratios has been challenging. There doesn’t really exist a market standard for the conversions, so we will have to do it out by haptic response testing with the vibration motors.
* Creating circuits with appropriate resistance using just 330 and 10k ohm resistors. The vibration motors, the ultrasonic sensors and the bread board are now set up in an experimental circuit. (Just had to buy 4.7k resistors)
* Realized due to the output type of the IR sensor being analog, we need an ADC (Analog digital converter) circuit for our breadboard or we won’t be able to use the data received by the laser directly.
* We are using roughly 8 AA batteries to reach a suitable amperage, then using a UBEC circuit to dial the voltage back down to 5V, while it gives expanded mobile use, the battery pack is pretty hefty.
* Lead Engineer to the ER after a laceration during a fabrication part of the project
* Zeroing the ultrasonics to work properly in array, especially with cross sensor interference
* Prototyping the frame using limited material resources, we will hopefully be 3d printing a better static mount for the fall.
* Realizing the ADC was going to require us to rely on the breadboard, we had originally assumed we could completely remove it because of our soldering work.

We will be continuing to document our process, and hope to have inspired others with our project.