Flying Under the Radar

Team A6: Linsey Szabo, Ayesha Gupta, and Angie Bu
18-500 Capstone Design, Spring 2023
Electrical and Computer Engineering Department
Carnegie Mellon University

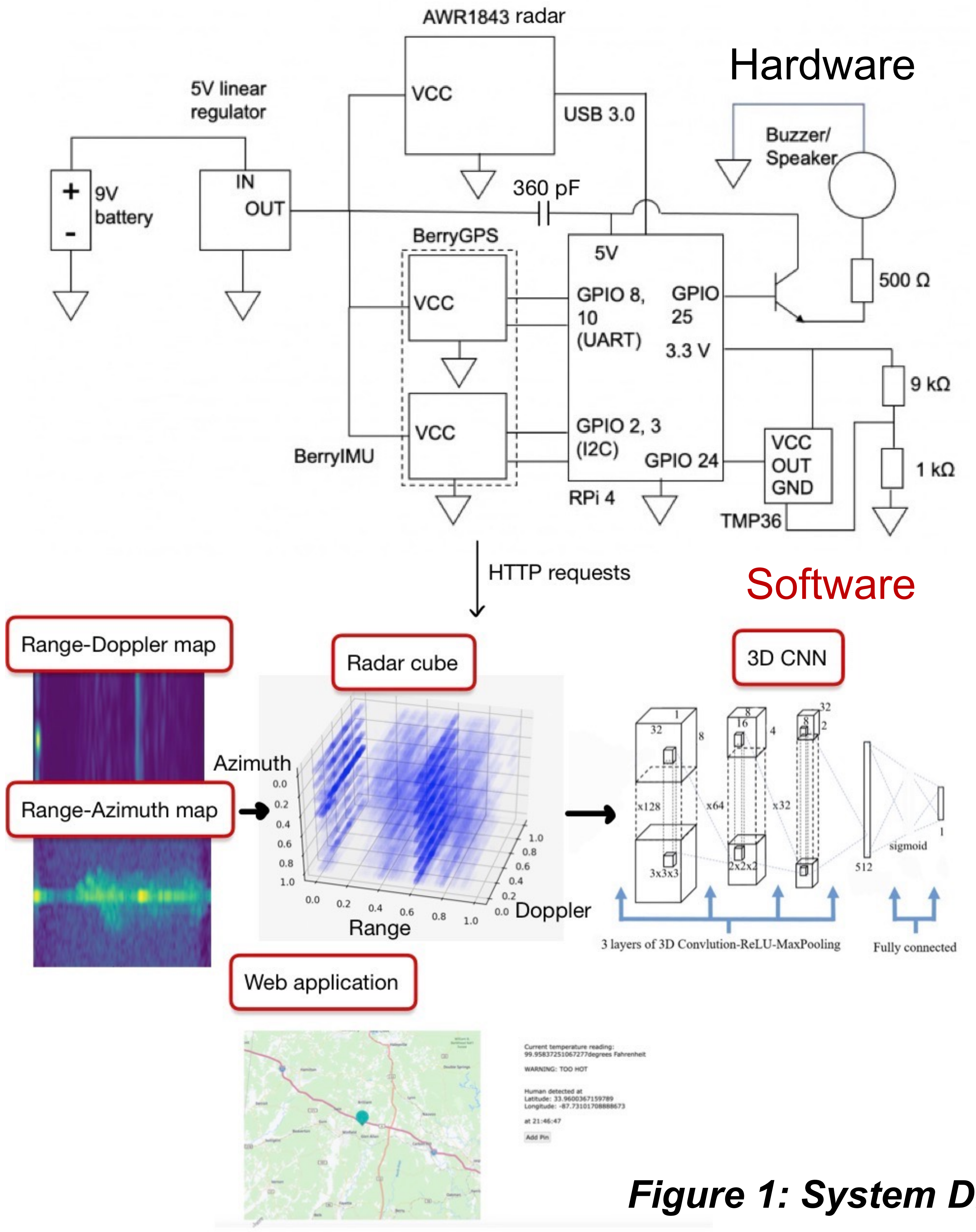
Product Pitch

Search and rescue (SAR) wilderness missions place first responders in harrowing terrain and expose them to extreme weather conditions like fire and smoke for prolonged periods of time. A current market solution uses drones with an HD camera and thermal camera, which are not only expensive but also fail in fire conditions due to heat interference.

We propose a mmWave radar drone attachment device that automates human detection through a 3D convolutional neural network (CNN) machine learning architecture. To interact with our device, we also created a web application to save the location of detected victims to further increase the efficiency of the SAR mission. In this way, we lower the cost barrier to SAR drone technology and help keep first responders safe.

System Architecture

Our data is collected through the AWR1843 mmWave radar and processed into a radar cube on the Raspberry Pi (RPi). It is then sent via an HTTP request to the web application, where the 3D CNN performs inference on it. Finally, the web application displays whether a human was detected, the location of the detection on the map if applicable, and temperature to warn if the device is in too high of heat.



Current temperature reading: 99.95837251067277degrees Fahrenheit

WARNING: TOO HOT

Human detected at
Latitude: 33.9600367159789
Longitude: -87.73101708888673

at 21:46:47

Add Pin

Figure 1: System Diagram

Conclusions & Additional Information

We were able to achieve many goals we set for ourselves, demonstrating that it is possible to improve human detection in SAR missions with a more affordable product. Our next steps would be to attach our device to a drone to test in realistic scenarios and conditions. We'd also like to test on very poor network signals to simulate said conditions. We learned a lot about dealing with finicky hardware and testing along the way to improve integration, which is always difficult.

System Description

Our system consists of three main subsystems: hardware, machine learning, and frontend. Our hardware has four components with the main one being our mmWave radar that captures images. The second component is a temperature sensor to capture the ambient temperature. The third component is a GPS and inertial measurement unit (IMU) sensor to capture positional information. All these components connect to an RPi, which sends the data to our frontend via HTTP requests. Lastly, we have a speaker, which plays a message to instruct victims to wave their arms, so that the radar can detect them better. Our frontend consists of a Django web application, which displays all the sensor information and uses the HERE Maps API to display the coverage area. Our web application also runs our last subsystem–the machine learning model. Using TensorFlow, it runs a 3D CNN architecture that accurately detects when a human is present and moving.

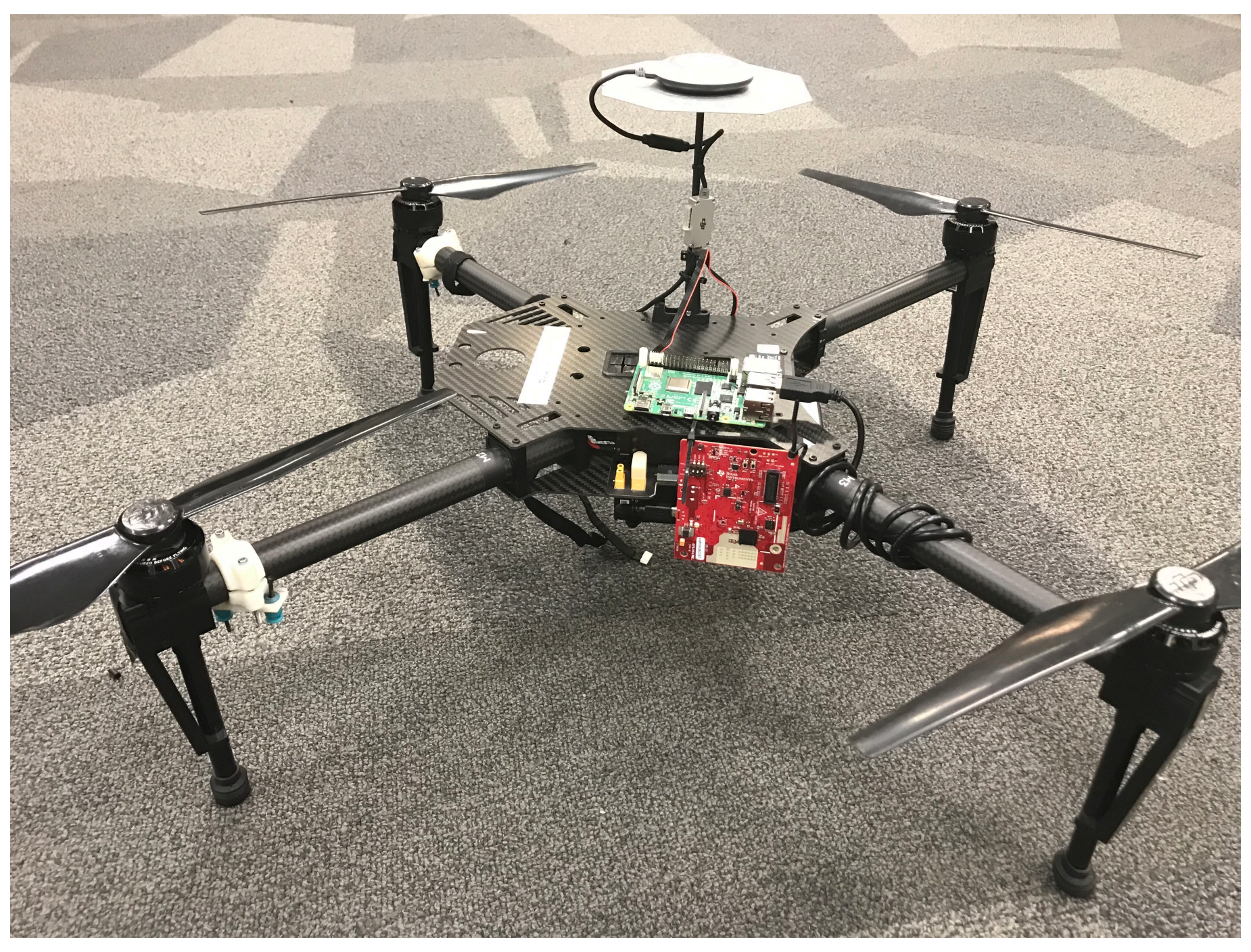


Figure 2: System attached to drone (no chassis)

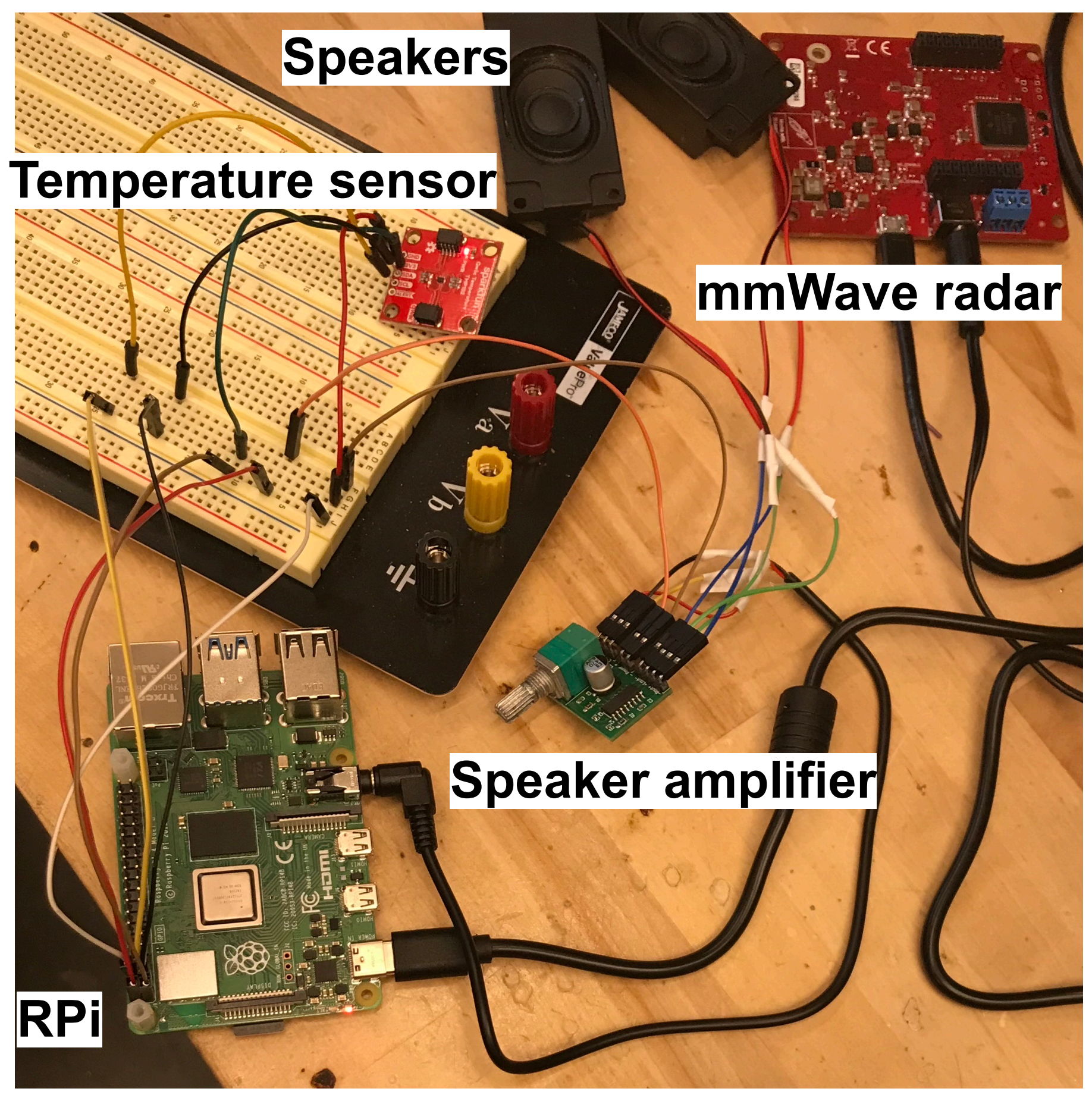


Figure 3: Hardware subsystem

System Evaluation

a. Hardware

Component	Goal	Result
Radar	**Capture high resolution range-Doppler and range-Azimuth data** at 5 Hz to determine a human presence from 5 m	***Success.*** The success of other subsystems attest to the radar image quality
Speaker	Play the message "Wave your arms if you are able in order to help us detect you better" at a volume such that it can be **heard within the 5 m range**	***Success.*** Playing the audio message from the speaker at 5 m is heard over din of TechSpark
Temperature Sensor	Alert the web application when ambient temperature exceeds **100 degrees Celsius**	***Success.*** Web application displays warning when temperature exceeds our set threshold of 84 degrees Fahrenheit and will generalize to 100 degrees Celsius
GPS/IMU Sensor	Locate the victim **within 0.5 m**	***Fail.*** Achieved a standard deviation of 1.5 m

b. Software

Machine Learning Architecture	**F1 score of 0.7** on unseen data	***Success.*** Achieved F1 score of .99 on unseen data from various scenes
Web Application	Accomplish all functions with **100 ms latency**	***Success.*** All actions, including map loading and sensor data display, occur with 100 ms latency

c. System

mmWave Radar Drone Attachment with Web Application	**1 s** for good WiFi, **3 s** for bad WiFi	***Success.*** 39.34166 ms in good WiFi; ***In Progress.*** Unable to test in bad WiFi