Electrical, Computer & Energy Engineering
UNIVERSITY OF COLORADO **BOULDER**

((EIT Vision))

Wavegen Warriors: Dante Ausonio, Peter Braza, Ahmed Khan, Gustavo Sanchez, Graham Snyder, Gavin Unrue
With thanks to our sponsors: Andy Goldstein and Eric Bogatin

How it Works:

1. Place the test object within the polar grid of the platform. Remain clear of the platform when ready to scan
2. Press the **scan** button, and wait a few seconds to view the heat map
3. Press the **scan livestream** button to see the live imaging results on the screen updating every 1-2 seconds
4. Move the object around to see the heat map change
5. Press the **stop** button when finished

Technology Overview:

1. Test object - used to distort electric field when placed in between RF transceivers
2. Six LoRa Modules - transmit 915 MHz signals and measure the Received Signal Strength Intensity (RSSI)
3. Arduino Mega to process power (dBm) distortion data into vector of 30 points
4. Python UI and image results with heatmap using a Radial Basis Function (RBF) Interpolation library

ARDUINO adafruit

Electrical Impedance Tomography (EIT) that determines an object's location using Radio Frequency (RF)

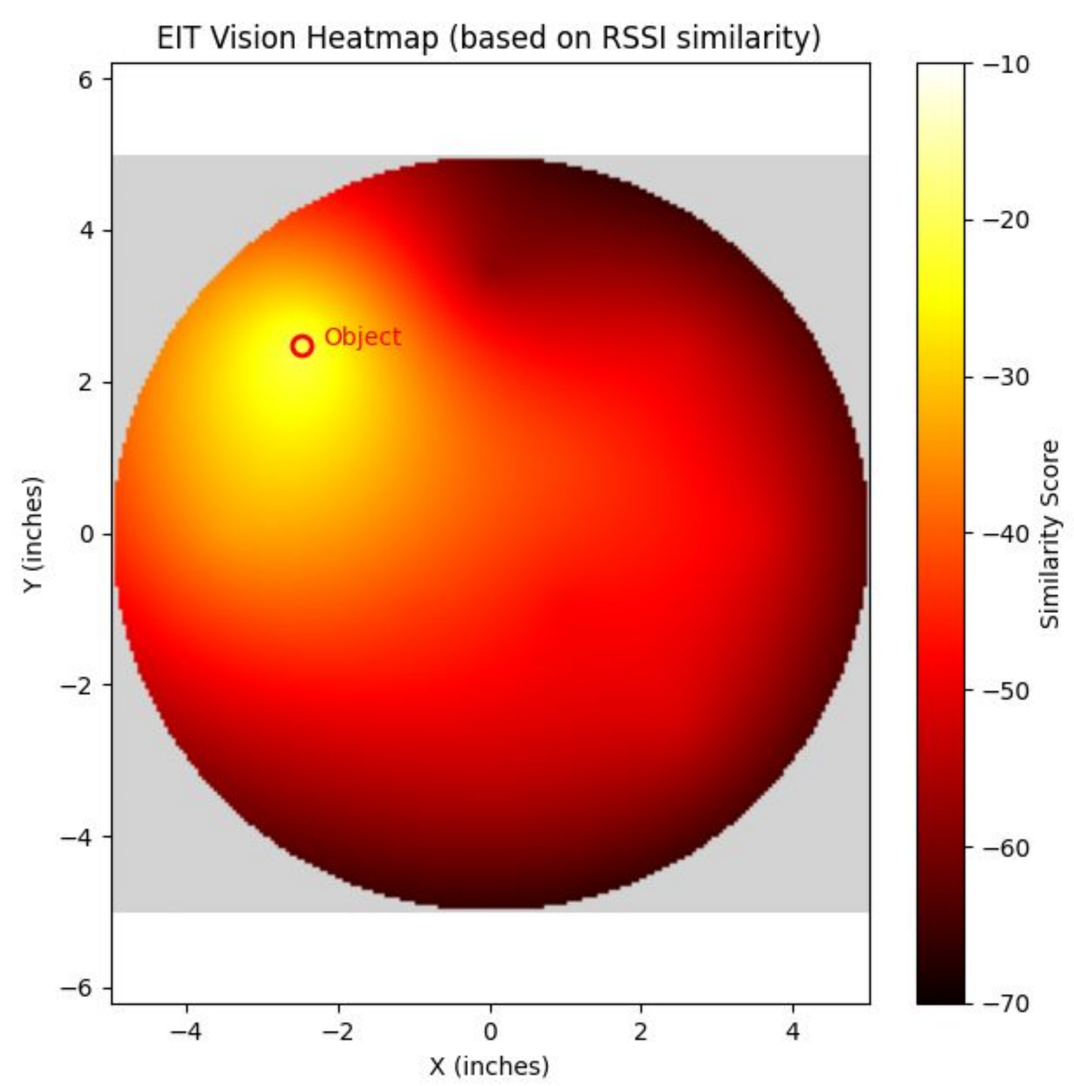
Test Object

Sensing Platform →

Purpose:

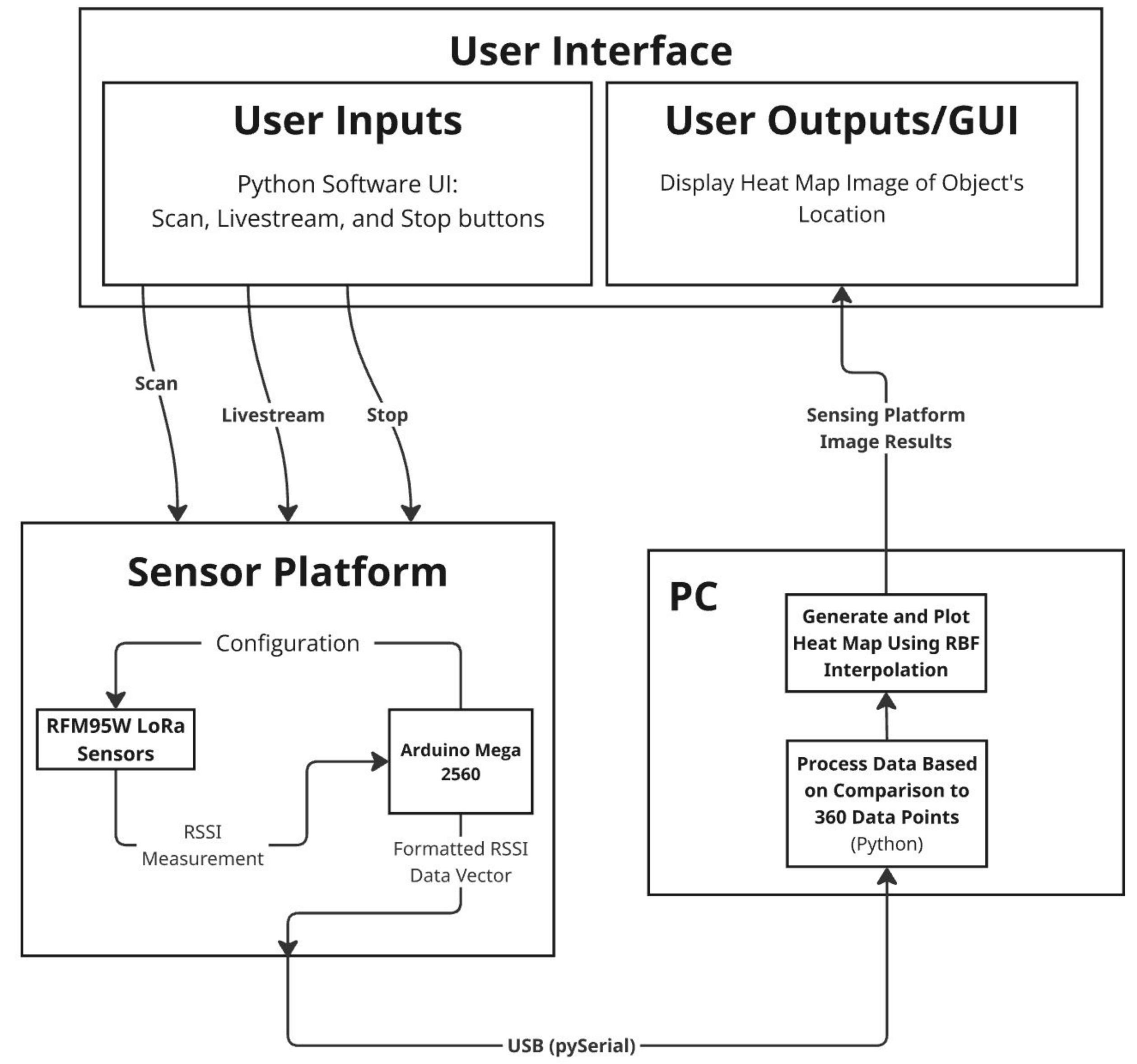
- Test vehicle to validate RF as a viable method for EIT
- Proves the feasibility of an underlying technology that can be used in various applications
- Potential applications of this technology: land mine detection, medical imaging, ag & soil monitoring, structural integrity testing

Output Results:



Python UI outputs the object's location using a similarity heatmap. Output updates every 1-2 seconds until turned off.

System Overview:



Development Journey:

LoRa Signal Strength:
- Original Plan ~ LoRa Modules and 1.3GHz antennas
- Problem ~ AGC issues due to close proximity and low overall signal strength
- Solution: Switched to properly frequency matched whip antennas

Imaging:
- Plan: EIDORS or pyEIT Library
- Problem: Imaging was inaccurate due to DC assumptions. Inapplicable to RF
- Solution: Radial Basis Function (RBF) Interpolation library

User Interface:
- Plan: Run pyEIT processing on firebase cloud function, and output to a Swift UI iOS app user interface
- End Result: Tkinter UI with one frame every 1-2 seconds