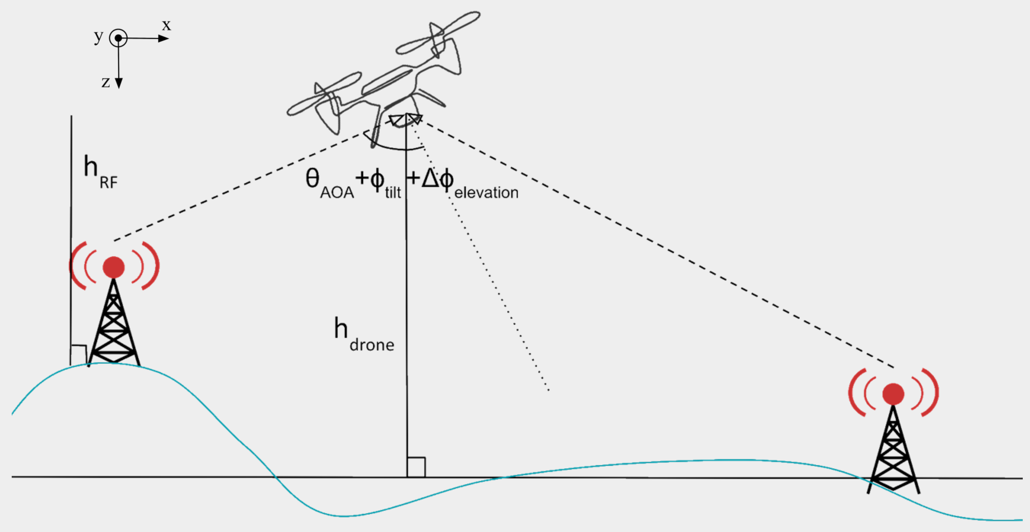
Silvus Technologies Clinic

Liam Chalk (TL-SP), Joe Zales (TL-FA), Arya Goutam, Zeneve Jacotin (SP), Alexa Wright (SP), Alec Candidato (FA), James Clinton (FA)
Advisor: Professor Josh Brake
Liaisons: Eric Contee II (HMC '19), Matthew Hayes, Ivan Aguilar

Problem Statement

The Silvus Technologies clinic project aims to use machine learning with geolocation-based signal processing to locate the position of a transmitting RF source. The final deliverable is a **Python neural network**, trained on simulated data, **to geolocate the RF source** in environments with varied terrain with an accuracy of 10 meters. The neural network is validated using emulated data collected on a Silvus SC4400 Radio.



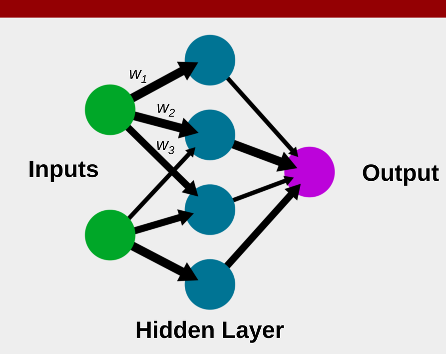
Solution Design

The neural network architecture has three sets of inputs: one **phase difference** for each pair of antennas, the sensor's **height**, and a slice of **terrain** data. The inputs pass through hidden layers consisting of nodes that combine these inputs. These nodes have different **weights**, which are tuned through an iterative process of optimization to maximize performance on a set of **simulated training data**. Finally, the neural network's performance is evaluated based on performance on simulated test data.

Neural Network

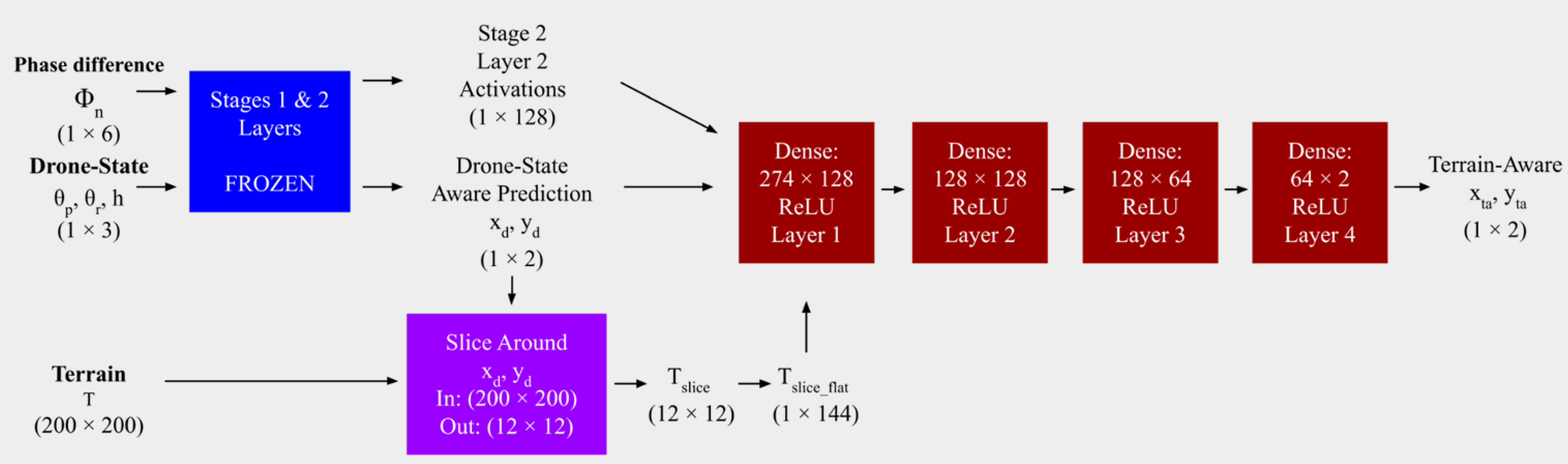
Initial Iterations and Motivation

A **neural network** was chosen because they can **adapt** to changes in signal characteristics, making them well-suited for tasks like RF source localization that require ongoing analysis and adjustment. When the neural network trains, it adjusts the **weights**, $\mathbf{w_n}$, to make more accurate output predictions over time.



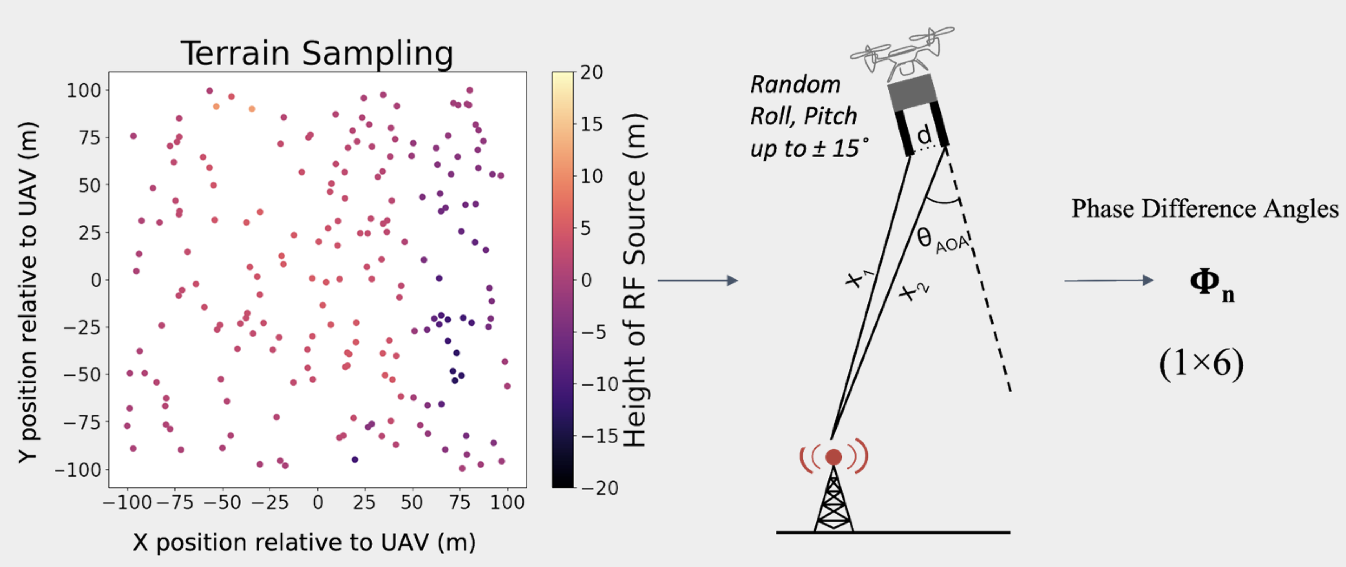
Simplified Model Iteration
Stage 1: *Level terrain, No tilt*
Stage 2: *Level terrain, With tilt*
Unified: *Variable terrain, With tilt*

Final Unified Model Architecture



This unified model uses two frozen models as a feature extractor, slices out a key section of terrain based on a terrain-blind prediction, and then updates the prediction with four dense layers. This is an example of transfer learning and is the architecture of the team's final deliverable.

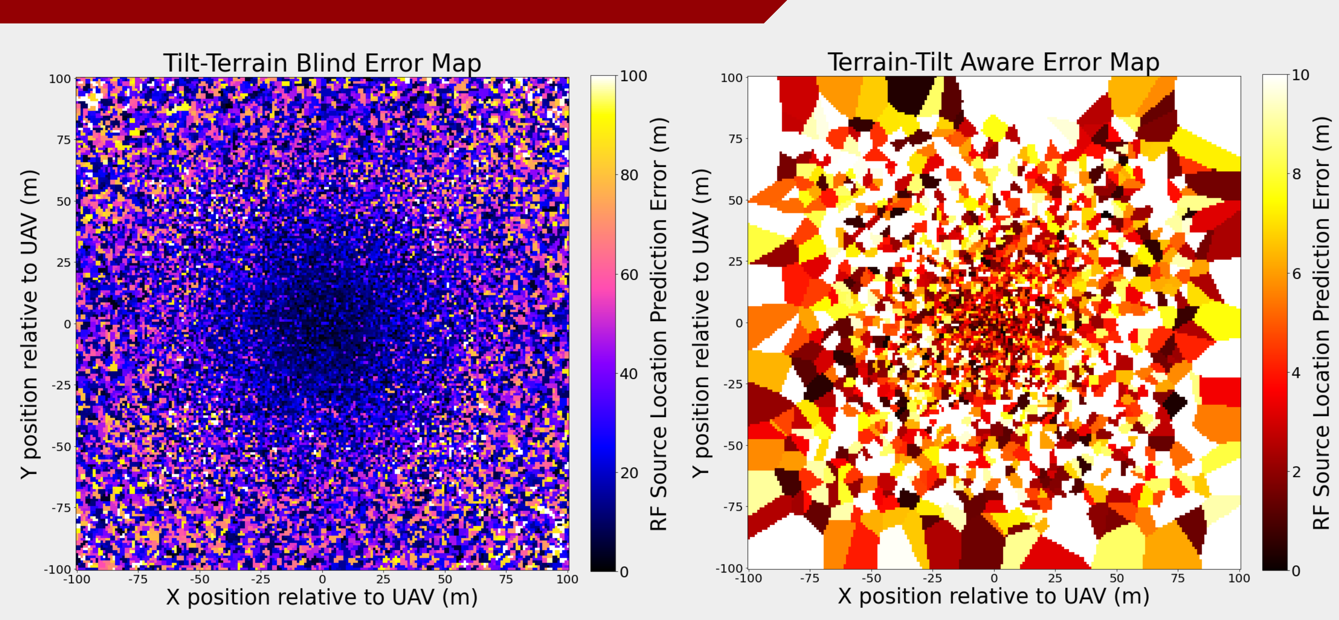
Training Data Generation



The neural networks were trained on simulated data created from random RF source points sampled on a 200 m by 200 m map. These points were assigned a **height** relative to the origin (±20 m) creating a terrain. For each RF source, the receiver was randomly assigned a **pitch, roll and height**. The difference in the **phase** of the signal at each antenna pair was calculated using geometric and signals engineering principles.

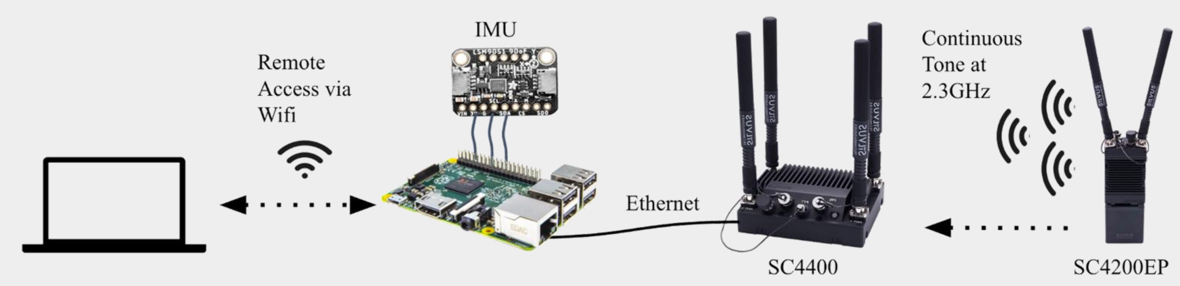
Unified Model Results

Visualization of errors from Stage 2 and Unified Stage 3 models. The left plot illustrates a model that has been trained on data that hasn't incorporated terrain. In the above network architecture this is the Stage 1 Model. The right plot is the result of the unified model which was trained on real terrain data from United States Geological Survey (USGS). Without the ability to use terrain or tilt data, the Stage 1 Model has a mean error of **58.1 m**, while the unified model has an error of only **8.7 m**.



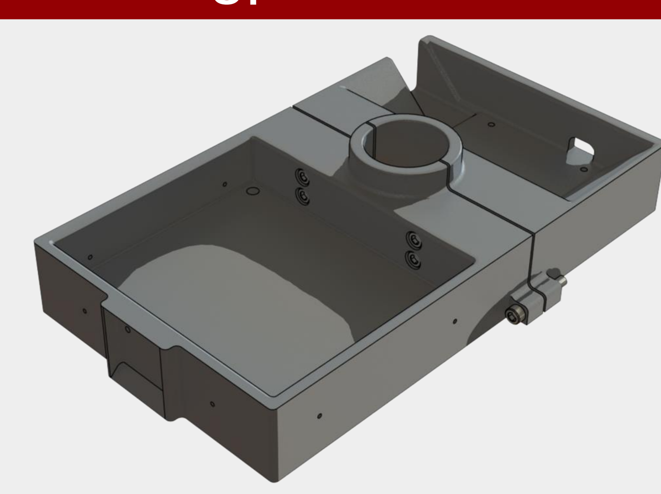
Field Testing

Field Connections



The team designed and constructed a test setup in order to validate the model. A Silvus SC4200 transmitted a constant tone at **2.3GHz**, which was received by the Silvus SC4400. A Raspberry Pi connected to the receiver by ethernet collected data from both the SC4400 and an IMU. The Pi was connected to a **5GHz wifi signal** allowing a computer to gain remote control.

Flagpole Mount



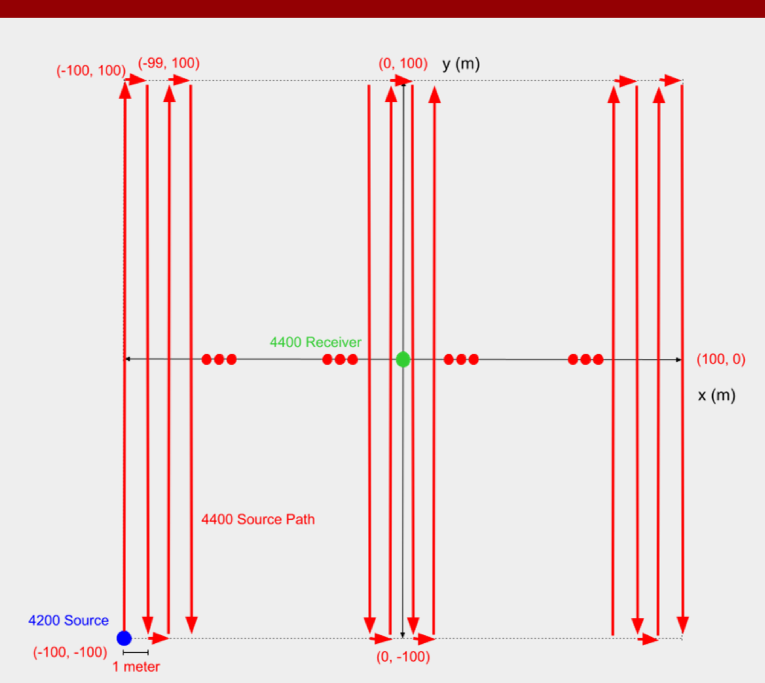
The project's ultimate goal is to attach a Silvus radio to a drone. However, due to the lack of repeatability with a drone, the team opted to test using a 30ft telescoping **flagpole**. The radio must attach to the pole upside down and the side. The team created multiple prototypes before creating the **3D-printed mount** above. On the left, the Silvus SC4400E is secured using ten mounting points. The **IMU** is attached to the front of this side, allowing for accurate radio tilt and sway measurements. The **Raspberry Pi** is mounted on the right, with cutouts for ethernet and power. These two sides clamp together on the pole using six 10-24 bolts.

Field Test Setup

Field tests took place behind Linde Residence Hall, allowing access to power and wifi. A Silvus SC4400E radio was used to record the RF footprint of a known transmitting source and use the proposed Neural Network to determine the source's location. However, the team was not able to find a solution that takes into account the random phase from each antenna channel that is a result of its respective RF chain.

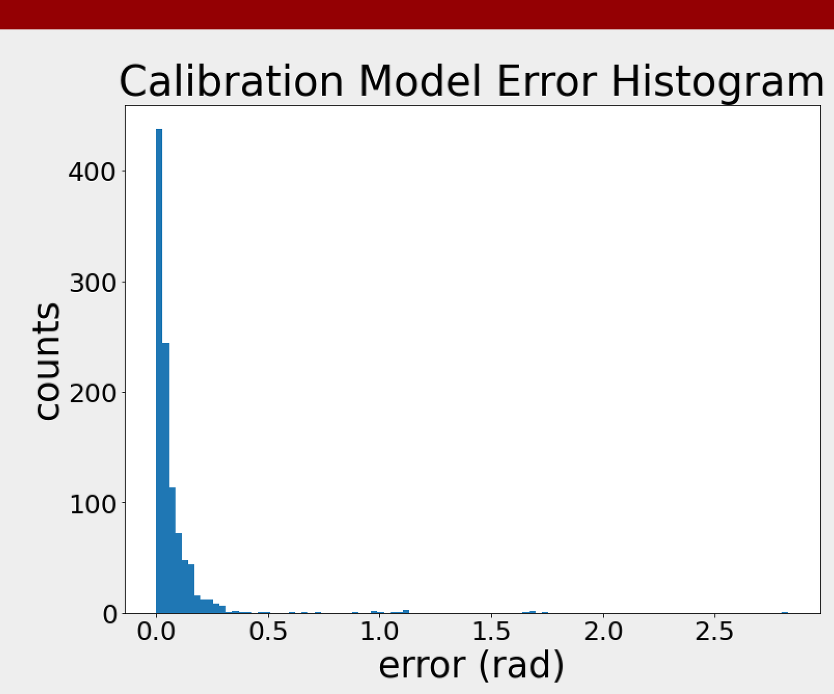
Network Emulator Validation

Network Emulator Validation



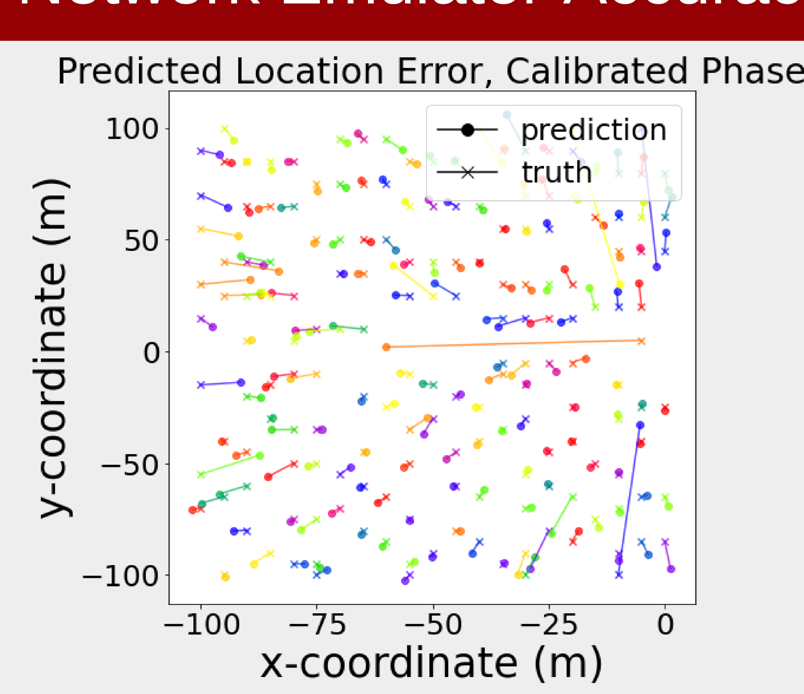
The network emulator data validated the neural network by comparing the **actual** locations to the **model** generated locations. The simulation above shows a moving source and a flat receiver.

Calibration Neural Network



The calibration neural network was trained to **correct** the correlation angles for an offset caused by the hardware. The mean error of the calibration model was **0.074 radians**.

Network Emulator Accuracy



The error map shows the **accuracy** of the network emulator neural network at locating the position of the source. The mean error of the unified model on channel emulator data was **5.4 m**.