COATL Millimeter wave RADAR

Test Plan

Team 4

Portland State University

Maseeh College of Engineering & Computer Science

Department of Electrical & Computer Engineering

Industry Sponsor:

Joshua Mendez, joshmen@pdx.edu

Faculty Advisor:

Mark Martin, marmart2@pdx.edu

Engineering Team:

Wallace McKenzie, mwallace@pdx.edu

Chris Kane-Pardy, chrisk@pdx.edu

Henry Sanders, hensan@pdx.edu

Kamal Smith, kamal@pdx.edu

May 8, 2025

ECE 412/413

Senior Project Development I/II

1 Top Down Test Plan v1.0

Date of Test: 4/25/2025
Testers: Kamal Smith, Henry Sanders, Wallace McKenzie, Chris Kane-Pardy

1.1 Purpose

The purpose of this test plan is to test the accuracy of the machine learning model used in the COATL mmWave RADAR project. This model is currently being developed to differentiate between 3 varieties of green coffee beans of various moisture contents based on I/Q data obtained through the XM125 radar. Current testing involves using a sample data set acquired through our data collection program. This test plan involves using that sample data set, but successive testing will involve live prediction of at least 3 green coffee varieties, once our machine learning model is trained. This test plan will include suggestions for optimizing the machine learning model as well as a description of how to make additional datasets for it.

Note: Typical machine learning models are considered *good* if they are above 70% accurate. Machine learning models that are within the range of 80-90% are considered ideal. This test plan is designed to achieve accuracy within the range of 80-95%.

1.2 Equipment Needed

- System Equipment
  - A computer with Python (version 3.8, 3.9, 3.10, 3.11, or 3.12)
  - The Acconeer Exploration Tool
    - Installation: Within your terminal, type

      ```
      python -m pip install --upgrade acconeer-exptool[app]
      ```
  - PyTorch
    - Installation: Within your terminal, type

```
pip3 install torch torchvision torchaudio --index-url \
https://download.pytorch.org/whl/cu118/
```

  - Scikit-learn
    * Installation: Within your terminal, type

```
python -m venv sklearn-env
sklearn-env\Scripts\activate  # activate
pip install -U scikit-learn
```

  - Download "Full_Array_Average_Scan_Fixed.py" from GitHub repo: https://github.com/chriskp-pdx/COATL-RADAR/blob/main/pyFiles/Beanis%20Scan%20Data%20Collection/Consistent%20Amplitude/Full_Array_Average_Scan_Fixed.py
  - Download "Chris_MLTests.py" from Github repo: https://github.com/chriskp-pdx/COATL-RADAR/blob/main/pyFiles/Machine%20Learning/Chris_MLTests.py
  - Download "MedBeanis_Testing v2 - BEANSV.csv" from GitHub repo: https://github.com/chriskp-pdx/COATL-RADAR/blob/main/pyFiles/Datasets/MedBeanis_Testing%20V2%20-%20BEANSV.csv

- Test Equipment
  - EVK125 Module Evaluation Kit with XM125 board and A121 Radar Sensor
  - LH112 Lens Kit and Holder
  - Aluminum tube
  - 3 varieties of unroasted coffee beans
  - 3-d printed Christmas tree stand
  - 8 M2.5 x 10mm screws
  - Corner reflector
  - Copper tape

- USB to USB-C cable
- Metric scale

1.3 Pre-Test Setup

1.3.1 Hardware Setup

1. Insert the Hyperbolic lens into the holder.
2. Insert the aluminum tube into the 3D-printed Christmas tree stand and tighten the 8 screws on the sides such that the aluminum tube is level and positioned centered over the center of the Hyperbolic lens.
3. Plug the USB cable into a USB port on your computer and the USB-c end into the connector on the XM125.

1.3.2 Software Setup

1. Open Device Manager on your computer and make sure the COM port is being recognized (it should be the "enhanced" COM port).
2. Open "Full_Array_Average_Scan_Fixed.py" in your IDE or terminal of choice. You may have to ensure you have both the acconeer and PyTorch packages installed within the same environment as your IDE.
   Running Full_Array_Average_Scan_Fixed.py will not work otherwise.
3. Change line 27 (the line with the variable definition "client") to reflect the correct COM port that matches your local "enhanced" port determined in step 1 above.
4. Open "Chris_MLTests.py" in your IDE or terminal of choice.
5. Change line 13 (the line with the variable definition "TrainingData") to reflect the correct data path where you have "MedBeanis_Testing v2 - BEANSV.csv" downloaded.

1.4 Top-down Test Steps for Training the Machine Learning Model with a Sample Dataset

1. Run "Chris_MLTests.py". It will take roughly 2 minutes to finish training the model. Take note of the printed accuracy. If the accuracy is below 80%, proceed with the following steps. If the accuracy is above 80%, proceed to subsection 1.5.

2. Experiment with modifying the number of epochs. The epochs variable is on line 67 and is currently set at 25000. Increasing or decreasing this value will have mixed results and may or may not yield improved accuracy.

   Note: This value is considered high for typical machine learning models; however, the dataset for this experiment is large, requiring a large number of epochs to yield a higher accuracy.

3. Experiment with modifying the learning rate. The learning rate is on line 64 within the parameter "lr" within the "optimizer" variable. Increasing or decreasing this value will have mixed results and may or may not yield improved accuracy.

1.5 Top-down Test Steps for Data Collection

1. Choose a bean variety. Using the scale, measure out 3g of beans.

2. Carefully insert the beans into the aluminum tube.

3. Using an IDE of your choice, open and run "Full_Array_Average_Scan_Fixed.py". It will export the amplitude values to a CSV file within the same directory as "Full_Array_Average_Scan_Fixed.py".

4. For testing this new data set, open "Chris_MLTests.py" and change line 13 (the line with the variable definition "TrainingData") to reflect the correct data path where you have your new CSV dataset downloaded.

5. Return to subsection 1.4.

1.6 Top-down Test Plan Conclusions / Discussion

The goal of these tests is to create datasets for the machine learning model that yield accuracy results within the range of 80-95%. Currently, we are working on a program that allows for live input data to be compared against our sample dataset. Achieving consistent accuracy of prediction of 3 varieties of green coffee beans is the requirement we are ultimately striving to meet. Meeting this goal will require a combination of modifying the machine learning model and the data collection program, as well as much, much more testing.

The current model accuracy heavily depends on the amount and diversity of data available. Collecting additional datasets under slightly varied environmental conditions (e.g. slight variations in bean orientation within the tube) can help improve model generalization and reduce overfitting.

Moving forward, we are prioritizing increasing the amount and diversity of collected training data and continuing to refine the machine learning model. Implementing live prediction capabilities, along with broader validation strategies, will be critical steps toward achieving the ultimate requirement: reliable differentiation between three varieties of green coffee beans based on moisture content.