A logo of a university

AI-generated content may be incorrect.

|  |  |
| --- | --- |
| Diagram of a computer vision diagram  AI-generated content may be incorrect.  AgribotSim Integration & Verification Log | Abstract  This project presents *AgribotSim* — a simulation-driven approach to intelligent crop monitoring using a Convolutional Neural Network (CNN) integrated with a MATLAB/Simulink-based robotic system. The robot navigates a grid-based environment, classifying tomato leaf health at each waypoint in real time using a trained MobileNetv2 model. Key implementation challenges, including class label misalignment between the CNN and Simulink, were systematically resolved using enumeration mappings and validation logic. The simulation prioritises realistic, sensor-driven decision-making, with visual debugging and validation mechanisms built in to confirm system reliability. Through modular design, structured data flow, and emphasis on scalability, AgribotSim lays the groundwork for future agricultural robotics — with potential for real-world deployment, multisensor integration, and expansion to other crop types. |

Final Submission Documentation for AGRIBOT\_FINAL\_SUB

## 1. Project Overview

This project demonstrates real-time AI-powered classification of tomato plant diseases within a Simulink-based robot simulation. A trained MobileNetV2 CNN model classifies leaf images into four categories: bacterial spot, fungal target spot, tomato mosaic virus, and healthy. The robot follows a waypoint path and simulates image analysis at each step, verifying predictions against expected labels.

## 2. CNN Development Summary

- Architecture: MobileNetV2, pre-trained and fine-tuned on a reduced subset of the PlantVillage dataset.  
- Input Size: 224×224 RGB.  
- Accuracy:  
 - Validation: 99.2%  
 - Test: 97.2% (batch result confirmed via external script).  
- Efficiency: Training time reduced from ~116 hours to ~3.5 using MATLAB Parallel Computing Toolbox.  
- Output: cleanTomatoNet.mat (DAGNetwork object).

## 3. Dataset & Preprocessing

Images were resized and renamed consistently using a MATLAB script. A labels.csv was generated to preserve class alignment post-renaming.  
Dataset structure:  
/resized\_images/  
 ├── Tomato\_\_\_Bacterial\_spot/  
 ├── Tomato\_\_\_Target\_Spot/  
 ├── Tomato\_\_\_Tomato\_mosaic\_virus/  
 └── Tomato\_\_\_healthy/

## 4. Image Sorting & Label Fix

Issue: MATLAB dir loads files alphabetically (e.g., 10.jpg before 2.jpg).  
Solution: Implemented numeric sort in the image loader using:  
sort(str2double(extractBetween(files, '', '.')))

## 5. CNN Verification (MATLAB Script)

A batch image test was run pre-integration:  
predictedLabels = classify(trainedNet, images);  
accuracy = mean(predictedLabels == categorical(labels)');  
Logged Accuracy: ~90.0% (realistic post-renaming). Output table included in report.

## 6. Simulink Integration & Architecture

The CNN model was integrated using the Predict block. Argmax logic converts softmax vector to class index.  
Simulink uses a selector-based structure to step through the image batch based on robot movement.  
A visual grid and logic panel aid debugging, with visualisation elements for class index display and validation.

## 7. Simulation Logic & Configuration

- Zig-Zag Navigation: Configured using waypoint\_generation.m which defines a 6×6 scan pattern.  
- Pause Logic: The robot pauses at each waypoint to simulate image capture.  
- Grid Field: Red markers were initially used for visualisation but removed to reflect real-world design.

## 8. Prediction Validation Logic

A validation subsystem compares the CNN prediction to the expected class at each waypoint.  
Categories: valid, invalid, systemOFF.  
These checks are only active when the robot is “on-task,” ensuring accurate runtime validation.

## 9. Debugging & Engineering Decisions

Original red spheres were removed from logic to reflect real-world conditions. The simulation evolved to rely on sensor-based classification only.  
This was a deliberate engineering decision to prioritise functionality over aesthetics.

## 10. Scalability Considerations

Future plans include:  
- EKF-based multi-sensor fusion  
- Webcam or live sensor data  
- Classifier for other crops  
- Improved simulation environments for scalability

## 11. Files & Reproducibility

All required files are available in: https://github.com/CarlaMED/AGRIBOT\_FINAL\_SUB  
Includes: cleanTomatoNet.mat, saveWaypointImages.m, waypoint\_generation.m, ClassLabel.m, Simulink models, config scripts, logs.

## 12. Toolboxes and setup:

• Deep Learning Toolbox

• Computer Vision Toolbox

• Simulink

• Image Processing Toolbox

• Parallel Computing Toolbox

• DPS Toolbox

• Simscape Electrical

• ImageMagick (used optionally to resize and rename images via command line)

## 13. References

MathWorks, 2023. *ImageDatastore*. [online] Available at: https://uk.mathworks.com/help/matlab/ref/imagedatastore.html [Accessed 16 June 2025].

MathWorks, 2023. *MobileNet-v2 Pretrained Network*. [online] Available at: https://uk.mathworks.com/help/deeplearning/ref/mobilenetv2.html [Accessed 16 June 2025].

MathWorks, 2023. *Simulink*. [software] The MathWorks, Inc. Available at: https://uk.mathworks.com/products/simulink.html [Accessed 16 June 2025].

MathWorks, 2023. *Enumerations in Simulink*. [online] Available at: https://uk.mathworks.com/help/simulink/ug/using-enumerated-types.html [Accessed 16 June 2025].

Hughes, D.P. and Salathe, M., 2015. An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint* arXiv:1511.08060.

**Simulink CNN Integration Summary**

• CNN\_Predictor subsystem includes Predict block, MATLAB Function (argmax), and display.

• Test harness isolates prediction logic.

• Label-to-index enum (ClassLabel.m) aligns outputs.

• A Selector block uses waypoint index to feed test images to the CNN in sync with robot motion.

**System Layout Diagram (Simplified)**

[testImage Selector] --> [Predict Block] --> [Argmax] --> [Label Display]

| ↘

[Image Index] [Color Logic (future)]