

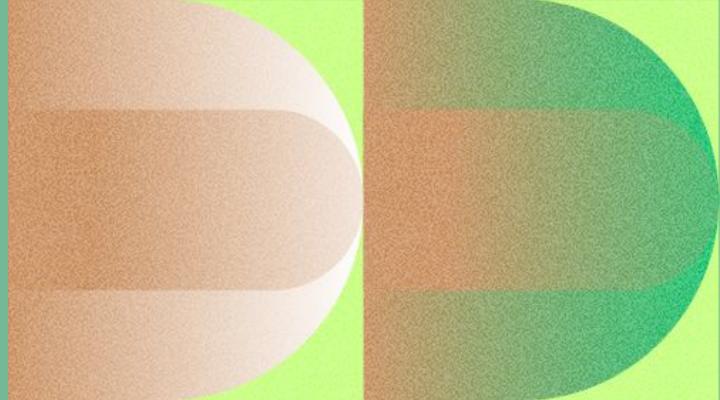


# AgroDrone Patrol



Larry & Leon

# Background Informations



40%      \$123B      1k ac

## *Global Crop Lost*

due to pests and diseases,  
mostly without sufficient  
monitor and timely actions

## *Economic Lost*

In Total due to crop lost  
with pests, diseases, and  
other natural factors

## *100 times faster*

Than traditional human  
inspection about 10 acres / day.

# Navigation Programs

The image shows a Scratch script for a robot to follow a spiral path. It uses nested loops and conditionals:

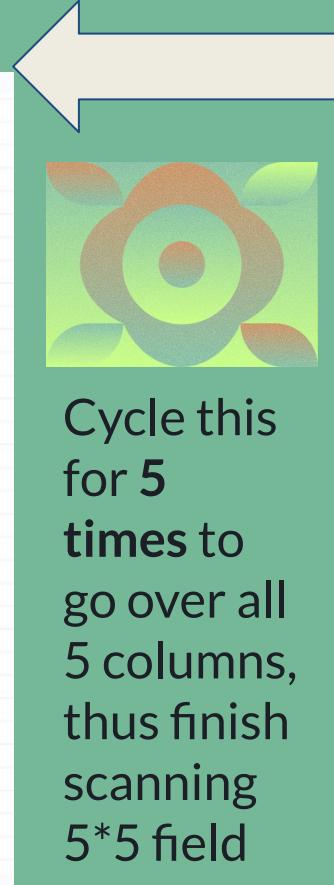
- A main loop repeats 5 times.
- Inside the main loop, there is a repeat loop that runs 2 times.
- Inside the repeat loop, there is a do loop that runs 4 times, performing the following sequence:
  - Move backward 1 second at 10% power
  - Turn left 1 second at 10% power
  - Turn right 1 second at 10% power
  - Move forward 2 seconds at 20% power
- After the inner do loop, the script moves left 1.5 seconds at 20% power to start the next column.

**Spiral**

Small movements in one unit while detecting the crops

Go forward to next unit

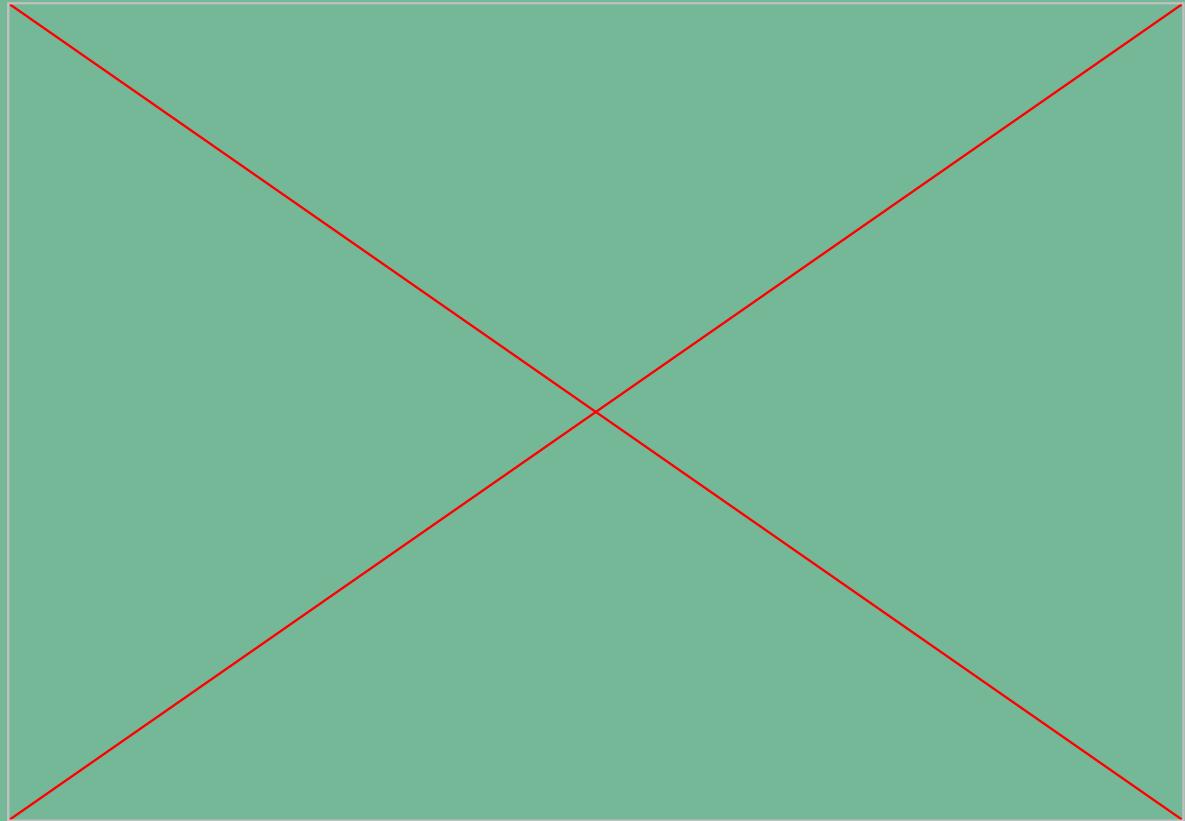
Go left after scanning over 1 column

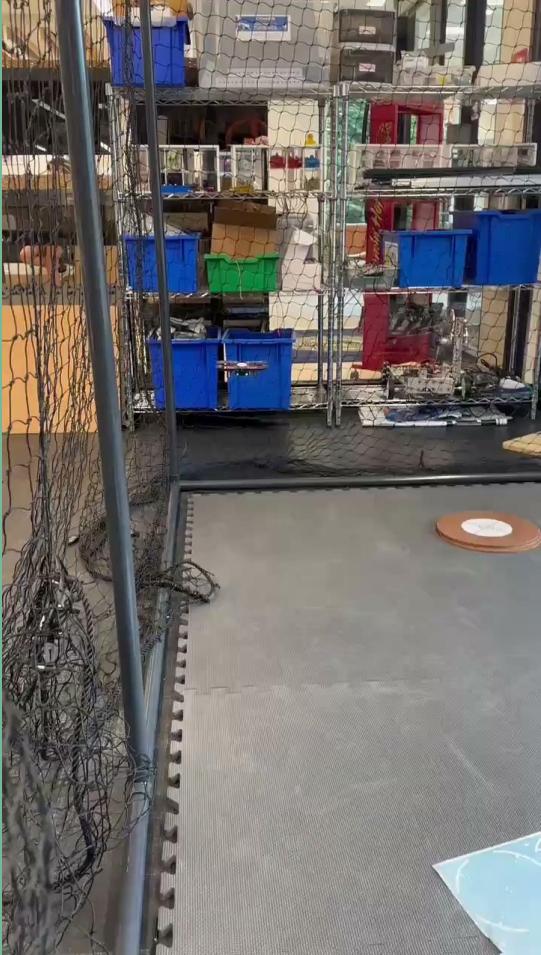


The image displays three Scratch scripts side-by-side:

- Spiral:** This script is identical to the one shown in the first section, creating a spiral path across the 5x5 field.
- Progressive:** This script creates a more progressive path, where the robot moves in a series of L-shaped steps, covering each square once. It uses a similar structure to the spiral script but with different movement sequences.
- linear:** This script creates a simple linear path, moving sequentially from the center square to the bottom-right corner of the field.

# Trial & Error



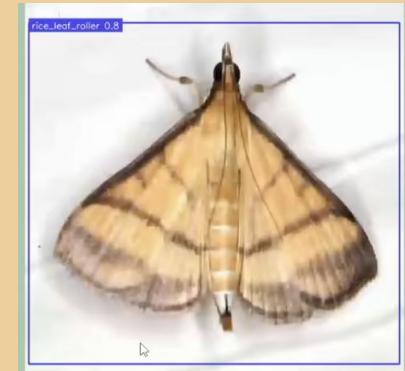
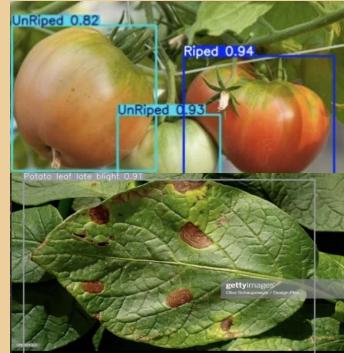


# Accomplishment & Challenge

What we've done so far:  
drones to go through 1 row of  
rubber pats with linear, progressive  
and spiral path planning.

Challenge:  
Various Errors and Hard to adjust  
during the process

# Detection Process



01.

Count



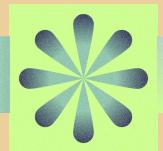
Decide Type

02.



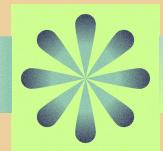
03.

Identify Pests



Monitor Status

04.

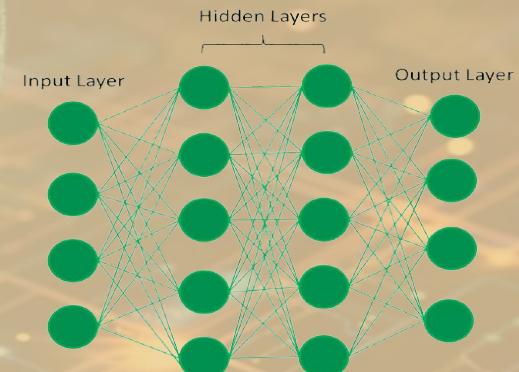


# AI Layer Example

Model ( $\frac{1}{3}$  step printing, 10fps =  $\frac{1}{3}$  30fps)

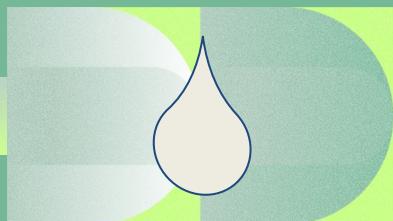
- Input Layer
  - Image (1024 x 512 pixels)
- Convolutional Layer 1
  - 32 Filters
  - Activation: ReLU
    - Function: Extracts low-level features (edges, textures)
- Max Pooling Layer 1
  - Pool Size: 2x2
    - Function: Reduces dimensionality, retains important features
- Convolutional Layer 2
  - 64 Filters
  - Activation: ReLU
    - Function: Extracts higher-level features
- Max Pooling Layer 2
  - Pool Size: 2x2
    - Function: Further reduces dimensionality

- Flatten Layer
  - Function: Converts 2D matrix to 1D vector for the dense layer
- Fully Connected Layer (Dense Layer 1)
  - 512 Nodes
  - Activation: ReLU
    - Function: Captures complex patterns
- Fully Connected Layer (Dense Layer 2)
  - 256 Nodes
  - Activation: ReLU
    - Function: Further learning of features
- Output Layer
  - Nodes: Number of Classes (e.g., Healthy, Diseased)
  - Activation: Softmax
    - Function: Provides probabilities for each class
- Loss Function
  - Categorical Cross Entropy
    - Function: Measures model performance during training



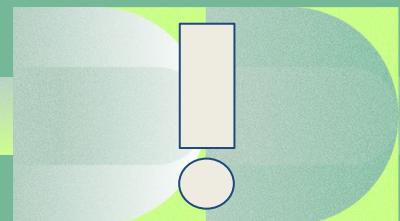
# Post Detection Actions

*Add*



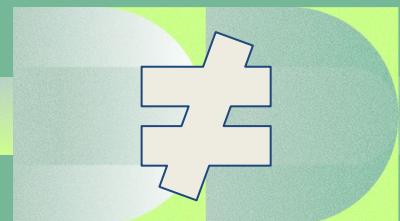
Water, fertilizer,  
pesticides, etc

*Report*



Issues to the  
backend (user)

*Remove*



Unwanted -  
Pests, Weeds, etc



Thank  
you!