**A close-up of a logo

AI-generated content may be incorrect.**

**Team Name:** GIFTS

**Team Members:** R Hemesh, M B Chaithanya, R Y Samhitha, N D Priyanka

**Multimodal Crop Disease and Insect Detection System**

An AI-powered diagnostic tool for farmers using image and symptom data

**Git Repo:** [**Link**](https://github.com/Hemesh11/Insect-and-disease-detection-in-a-Leaf)

**Table of Contents**

1. CVAT (offline) setup procedure
2. Annotation techniques used
3. Annotated dataset
4. Architecture of Multimodal
   1. Description
   2. Pictorial Representation
5. Implementation codes of the multimodal
6. Procedure to run the entire application
7. Output screenshots

**CVAT (offline) setup procedure**

1. Install Git
2. Install Docker

A screenshot of a computer program

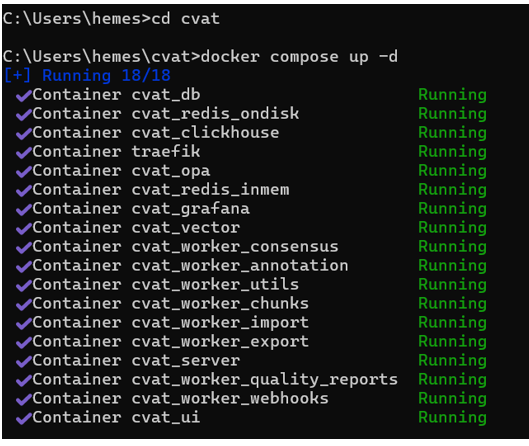
AI-generated content may be incorrect.

1. Clone the github repo of cvat

A screenshot of a computer

AI-generated content may be incorrect.

1. Run “cd cvat” to start cvat docker container



1. When you run cvat server for first time you should create superuser account and signup

A black screen with white text

AI-generated content may be incorrect.

1. Go to “https://localhost:8080”
2. Enter your credentials and sign in to use cvat in offline mode:
3. Create new task, upload dataset and annotate images (bounding box, polygon)

A close up of a caterpillar

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

1. Save and export in yolo 1.1 (bounding box) or yolo – seg (polygon) format
2. Close cvat, and run “docker compose down”

**Annotation techniques used**

**1. Insect Detection Dataset**

**Annotation Type**: Bounding Boxes

**Tool Used**: CVAT (Rectangle Tool)

**Export Format**: YOLO v1.1 (Object Detection format)

**Purpose**: To localize visible insects in images for YOLO detection training.

**2. Disease Detection Dataset**

**Annotation Type**: Polygon Segmentation

**Tool Used**: CVAT (Polygon Tool)

**Export Format**: YOLO Seg (Segmentation format)

**Purpose**: To accurately highlight disease-affected regions on crop leaves for segmentation training.

**Data Augmentation for both the datasets:** using the Albumentations library.Applied Transformations:Horizontal flip, Rotation (±25°), Brightness/contrast adjustment, Hue/saturation shift, Random scaling, Gaussian or motion blur, Gaussian noise

All bounding boxes were automatically adjusted to stay valid post-transformation using YOLO-compatible settings.

**Annotated dataset (after augmentation):**

**Diseases dataset:** [Link for Diseases Dataset](https://drive.google.com/drive/folders/1HgMg9GTp47qo2NzA6-xJzoVuLYULWgv2?usp=sharing)

**Insects dataset:** [Link for Insects Dataset](https://drive.google.com/drive/folders/1o3vhxlDZnlK8xHFM6qaMsRYItHtf9bma?usp=sharing)

**Architecture of Multimodal**

**System Architecture Description**

The Multimodal Crop Disease and Insect Detection System is built around three major machine learning components, each playing a distinct role in the diagnostic process. These components work in sequence to analyze visual and textual inputs, then combine their insights to deliver a highly accurate prediction.

**YOLOv8s – Looks at the Image**

Detects visible pests or disease spots in crop photos.

YOLOv8s (detection) → finds insects using bounding boxes.  
YOLOv8s (segmentation) → identifies leaf diseases by highlighting affected areas.

**Input:** Crop image  
**Output:** Confidence scores

**TabNet – Listens to the Symptoms**

Predicts disease or insect based on farmer-reported symptoms.

Uses 30 yes/no questions   
 Works even if some answers are missing  
 Explains which symptoms influenced the result

**Input**: 30 binary features  
**Output**: Probability

**Fusion Model – Combines the Best of Both**

Merges YOLO and TabNet outputs to make the final decision.

MLP (Neural Network) with two hidden layers (16 → 8 units), or  
 Logistic Regression for lightweight devices  
 YOLO scores are weighted 2x to prioritize visual evidence

**Input:** fusion\_input = concatenate([yolo\_scores \* 2.0, tabnet\_probs])  
 **Output**: Final prediction + confidence (e.g., 0.91)

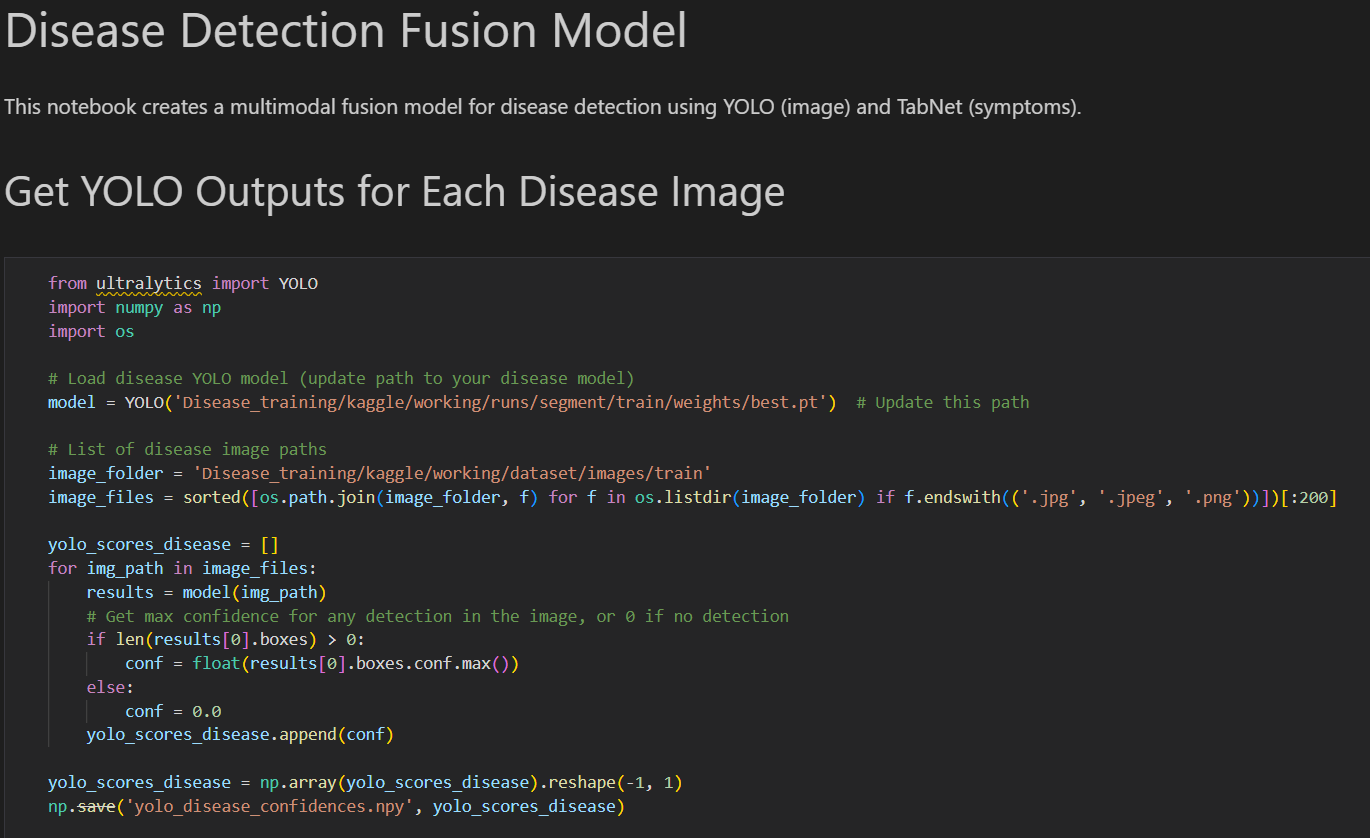
**Pictorial Representation**

A diagram of a flowchart

AI-generated content may be incorrect.

**Implementation codes of the multimodal**

**Multimodal Disease Detection**

****

**A screenshot of a computer program

AI-generated content may be incorrect.**

**A screen shot of a computer code

AI-generated content may be incorrect.**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**A screen shot of a computer program

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Multimodal Insect Detection**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**A screen shot of a computer program

AI-generated content may be incorrect.**

**A screen shot of a computer code

AI-generated content may be incorrect.**

**A screen shot of a computer program

AI-generated content may be incorrect.**

**A screen shot of a computer program

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Procedure to run the entire application**

**Installation**

Prerequisites

* Python 3.8+
* CUDA-capable GPU (recommended)

**Setup**

git clone <repository-url>

cd AGRITHON

pip install -r requirements.txt

Or manually:

pip install ultralytics pytorch-tabnet scikit-learn pandas numpy opencv-python joblib albumentations matplotlib seaborn

**Usage Guide**

**Step 1:** Train Models

YOLOv8 Training

python insect\_dataset\_split/train\_yolo.py

python train\_disease\_yolo.py # Edit paths in script

TabNet Training

python tabnet\_Train.py

**Step 2:** Train Fusion Model

* Insect: fusion\_insect.ipynb
* Disease: fusion\_disease.ipynb

**Step 3:** Inference (Prediction)

from fusion\_models import predict\_insect, predict\_disease

Example: symptoms = [1, 0, 1, 0, ...] # 30 binary values

image\_path = "path/to/image.jpg"

prediction, confidence = predict\_insect(image\_path, symptoms)

**Output screenshots**

**Insect Detection**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Disease Detection**

**A screenshot of a computer

AI-generated content may be incorrect.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | Task | Accuracy | Precision | Recall | F1 |
| YOLO (Insect) | Detection | 92%+ | - | - | - |
| YOLO (Disease) | Segmentation | 90%+ | - | - | - |
| TabNet (Insect) | Classification | 76% | 0.75 | 0.78 | 0.76 |
| TabNet (Disease) | Classification | 90% | 0.89 | 0.91 | 0.90 |
| MLP Fusion (Insect) | Classification | 85%+ | 0.84 | 0.86 | 0.85 |
| MLP Fusion (Disease) | Classification | 92%+ | 0.91 | 0.93 | 0.92 |