MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent unit of MAHE, Manipal)

Advanced UAV-based Weed Detection and Mapping Technologies Using Machine Vision

A Mini Project Report submitted to Manipal Institute of Technology in partial fulfilment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
MECHATRONICS

Submitted by

Hem Gosalia - 220929258

Ishan Deshmukh - 220929188

Sannidhi Math - 220929180

Dhariuosh Muhammed -220929274

Jahan Marfatia -220929046

Under the guidance of

Dr. Umesh Kumar Sahu
Associate Professor

DEPARTMENT OF MECHATRONICS
MANIPAL INSTITUTE OF TECHNOLOGY
(A Constituent of Manipal Academy of Higher Education)
MANIPAL - 576104, KARNATAKA, INDIA

November 2025

MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent unit of MAHE, Manipal)

INSPIRED BY LIFE

DEPARTMENT OF MECHATRONICS

Manipal
August 27, 2025

CERTIFICATE

This is to certify that the mini project / software titled **Advanced UAV-based Weed Detection and Mapping Technologies Using Machine Vision** is a record of work done by **Hem Gosalia (220929258)**, **Ishan Deshmukh (220929188)**, **Sannidhi Math (220929180)**, **Dhariuosh Muhammed (220929274)** and **Jahan Marfatia (220929046)**, submitted for Machine Vision (PE-VI), MTE 4459 during the academic year 2025-2026.

Dr. Umesh Kumar Sahu
Associate Professor
Dept. of Mechatronics
MIT Manipal

ABSTRACT

Early-season weed detection in agricultural fields remains a critical challenge for precision farming and crop yield optimization. This study implements a YOLOv8-based object detection system for real-time weed localization using the DRONEWEED drone imagery dataset containing 67,558 RGB images of multiple weed species in maize and tomato fields. The methodology employs YOLOv8n/s variants with transfer learning from COCO pre-trained weights, incorporating advanced data augmentation techniques including mosaic augmentation for multi-scale learning, MixUp for improved generalization, and HSV color space transformations to handle variable lighting conditions in outdoor environments. The model utilizes anchor-free detection with decoupled heads to simultaneously classify and localize weed instances at 640×640-pixel resolution. Results demonstrate that YOLOv8 achieves mean average precision (mAP) values of 0.95-0.96 with processing speeds exceeding 80 frames per second, enabling practical deployment for automated weed management systems. The lightweight architecture with optimized parameters ensures compatibility with edge devices and agricultural robots while maintaining high detection accuracy across varying weed phenological stages. This approach provides farmers and agricultural professionals with an efficient, computationally feasible solution for precise weed identification, supporting targeted herbicide application and sustainable farming practices.

LIST OF FIGURES

LIST OF TABLES

TABLE OF CONTENTS

CHAPTER 1
Introduction

Weed infestation poses a significant threat to agricultural productivity by competing with crops for nutrients, water, and sunlight. Traditional weed management methods are often labor-intensive and lack precision, resulting in excessive herbicide use and potential crop damage. Recent advancements in unmanned aerial vehicles (UAVs) equipped with high-resolution cameras have enabled efficient data collection over large fields. By integrating machine vision techniques, it becomes possible to automate weed identification and mapping, thus supporting site-specific weed management in precision agriculture.

1.1 Motivation

Weed infestation poses a significant threat to global agricultural productivity, causing crop yield losses of up to 34% annually and necessitating expensive herbicide treatments that harm environmental sustainability. Traditional manual weed identification methods are labor-intensive, time-consuming, and prone to human error, making them impractical for large-scale farming operations. Current agricultural practices rely heavily on blanket herbicide application, resulting in increased operational costs, environmental degradation through chemical overuse, and inadequate targeting of early-season weeds when they are most vulnerable to control measures.

The emergence of unmanned aerial vehicles (UAVs) combined with deep learning-based computer vision techniques offers a transformative solution for precision agriculture, enabling automated, real-time weed detection and classification at scale. Early-season weed detection is particularly critical as young weeds are more susceptible to control interventions and cause less crop damage when removed promptly. However, existing computer vision solutions suffer from computational limitations preventing real-time processing, poor generalization across diverse field conditions, and insufficient accuracy for small early-growth weeds that are difficult to distinguish from crop seedlings.

This project addresses these challenges by implementing a YOLOv8-based object detection system using high-resolution drone imagery from the DRONEWEED dataset, which contains 67,558 labeled RGB images capturing multiple weed species across maize and tomato cultivation fields. By leveraging state-of-the-art deep learning architectures optimized for real-time performance and incorporating advanced augmentation strategies, this work aims to provide farmers and agricultural professionals with an efficient, computationally feasible solution for precise weed identification, supporting targeted herbicide application and sustainable farming practices.

1.2 Objective

- Develop a Real-Time Weed Detection System: Implement a YOLOv8-based object detection model capable of processing drone imagery at speeds exceeding 30 frames per second while maintaining mean average precision (mAP@0.5) above 0.90 for accurate weed localization and classification across multiple weed species in the DRONEWEED dataset.
- Optimize Data Augmentation Pipeline: Design and evaluate a comprehensive augmentation strategy incorporating mosaic augmentation, MixUp blending, HSV color space transformations, and geometric augmentations to enhance model robustness against environmental variability including lighting conditions, viewing angles, and seasonal changes.
- Maximize Computational Efficiency: Configure model architecture (YOLOv8n/s variants) and training hyperparameters to operate within Kaggle's GPU resource constraints (30 hours weekly, 9-hour sessions with P100 GPUs) while achieving convergence within 100-150 epochs and enabling practical deployment on edge devices.
- Benchmark Performance Metrics: Systematically evaluate detection performance using standard object detection metrics including mAP@0.5, mAP@0.5:0.95, precision, recall, F1-score, and inference time, and conduct comparative analysis against baseline models to quantify improvements over traditional approaches.
- Enable Reproducible Research: Create a well-documented implementation framework with clear preprocessing pipelines, training procedures, and evaluation protocols accessible to agricultural researchers for adaptation to alternative datasets and crop-weed combinations.

1.3 Challenges

Dataset Variability and Environmental Complexity: Drone-captured agricultural imagery exhibits significant variations in lighting conditions, viewing angles, altitude differences, and weather-dependent color characteristics that challenge model robustness. Early-season weeds present minimal visual distinction from crop seedlings due to similar morphology, small size (often less than 5cm), and overlapping spectral signatures in RGB images. Environmental factors including soil type variations, shadow patterns from varying sun angles, and diverse crop growth stages further complicate consistent weed identification.

Class Imbalance and Multi-Species Detection: The DRONEWEED dataset encompasses multiple weed species (Atriplex patula, Chenopodium album, Convolvulus arvensis, Cyperus rotundus, Portulaca oleracea, Solanum nigrum) across different phenological stages, resulting in imbalanced class distributions and varying detection difficulties. Some weed species appear more frequently than others, leading to potential model bias toward common classes while underperforming on rare species. Small object detection remains computationally challenging, as early-growth weeds occupy limited pixel regions (10-50 pixels) within 640×640 resolution images, making them susceptible to information loss during feature extraction.

Real-Time Processing and Computational Constraints: Agricultural applications demand real-time inference capabilities (>30 fps) for operational UAV-mounted systems while maintaining high detection accuracy (mAP >0.90). Training deep neural networks requires substantial computational resources, with Kaggle's free tier limiting GPU access to 30 hours weekly with 9-hour session maximums, constraining iterative experimentation and hyperparameter tuning. Memory limitations of 13GB RAM on Kaggle P100 GPUs restrict batch sizes and model complexity, particularly for larger YOLOv8 variants.

Generalization and Deployment Challenges: Models must generalize across different crop types, growth stages, soil backgrounds, and geographical regions not represented in training data. Overfitting to specific field conditions, drone camera characteristics, or seasonal appearances poses risks for practical deployment. Edge deployment necessitates model compression and optimization techniques to operate within memory constraints of agricultural robots and embedded systems without sacrificing detection performance. Additionally, ensuring model robustness to image quality degradation from motion blur, lens distortion, and varying camera specifications remains critical for field reliability.

1.4 Scope of Work

This project encompasses the complete development pipeline for a YOLOv8-based weed detection system tailored to the DRONEWEED dataset, spanning from data preprocessing through model evaluation and performance analysis. The scope includes implementing comprehensive image augmentation strategies (mosaic, MixUp, HSV transformations, geometric augmentations) to enhance model robustness against environmental variability encountered in real-world agricultural settings.

Model architecture implementation focuses on YOLOv8n (nano) and YOLOv8s (small) variants with anchor-free detection and decoupled classification-localization heads, leveraging transfer learning from COCO-pretrained weights for improved convergence and reduced training time. The experimental framework involves systematic hyperparameter optimization including learning rate scheduling (cosine annealing), batch size tuning (16-32), and epoch selection (100-150 iterations) to maximize mean average precision while maintaining real-time inference speeds.

Performance evaluation employs standard object detection metrics including mAP@0.5, mAP@0.5:0.95, precision, recall, F1-score, and inference time measurements across validation and test sets. Qualitative analysis includes visualization of detection outputs, error analysis identifying common failure modes, and per-class performance breakdown across weed species. The implementation is optimized for Kaggle's computational infrastructure, ensuring reproducibility and accessibility for the broader research community.

The project scope excludes multi-spectral imagery analysis (NIR, thermal bands), post-detection herbicide application control systems, long-term field deployment with continuous model retraining, and hardware-specific optimization for particular edge devices or agricultural robots. These areas are identified as future work directions beyond the current research focus.

1.5 Report Organization

This report systematically documents the development and evaluation of the YOLOv8-based weed detection system across six chapters with a clear logical progression.

- Chapter 1 (Introduction) establishes the research motivation, defines specific objectives for detection accuracy and computational efficiency, identifies key technical challenges, and outlines the scope of work.
- Chapter 2 (Literature Review) presents a comprehensive survey of recent advances in agricultural weed detection, comparing traditional computer vision approaches with deep learning methodologies, analyzing state-of-the-art YOLO family architectures in precision agriculture applications, and identifying research gaps addressed by this work.
- Chapter 3 (Methodology) details the complete technical approach including theoretical foundations of YOLOv8 architecture with anchor-free detection mechanisms, mathematical formulations of loss functions (CIoU, classification loss, distribution focal loss), data preprocessing pipelines, augmentation strategies (mosaic, MixUp, HSV), training procedures with hyperparameter specifications, and experimental protocols implemented on Kaggle's computational infrastructure.
- Chapter 4 (Results and Discussion) presents experimental outcomes with quantitative performance analysis using standard object detection metrics, comparative evaluations against baseline models, qualitative visualization of detection outputs on test images, per-class performance breakdown across weed species, error analysis identifying failure patterns, and comprehensive discussion interpreting results within the context of agricultural deployment requirements.
- Chapter 5 (Conclusion and Future Scope) synthesizes key findings, evaluates achievement of research objectives, discusses practical implications for precision agriculture, identifies limitations of the current approach, proposes future research directions including multi-spectral integration and edge deployment optimization, and examines alignment with United Nations Sustainable Development Goals related to food security and environmental sustainability.
- Chapter 6 (Contribution of Each Team Member) documents individual team member responsibilities and contributions throughout the project lifecycle, followed by a comprehensive References section listing all cited literature in standard academic format.

Table 1.1: Results of a Market Survey conducted in three farms

Question	Scenario 1	Scenario 2	Scenario 3
Expenses for pruning	|100/- per tree	|100/- per tree	|2,500/- to |3,500/per tree
Availability of labour	Medium (when they are free)	Less (not available when required)	Low (demand high prices)
Method used for harvesting and spraying	Manual	Manual	Manual
Count of spraying and pruning done	2-3 times pruning, 1-2 times spraying	3-4 times pruning, 1-2 times spraying	3-4 times pruning, 1 time spraying
Distance between trees	4.5 feet	4.5 feet	3 feet
Weather conditions	Rainy season for spraying. Dry season for pruning.	Rainy season for spraying. Dry season for pruning.	Rainy season for spraying. Dry season for pruning.
Injuries / accidents	Lost lives many times	Injured after fall	Yes
Readiness to invest in modern technology	Yes	Yes	Yes

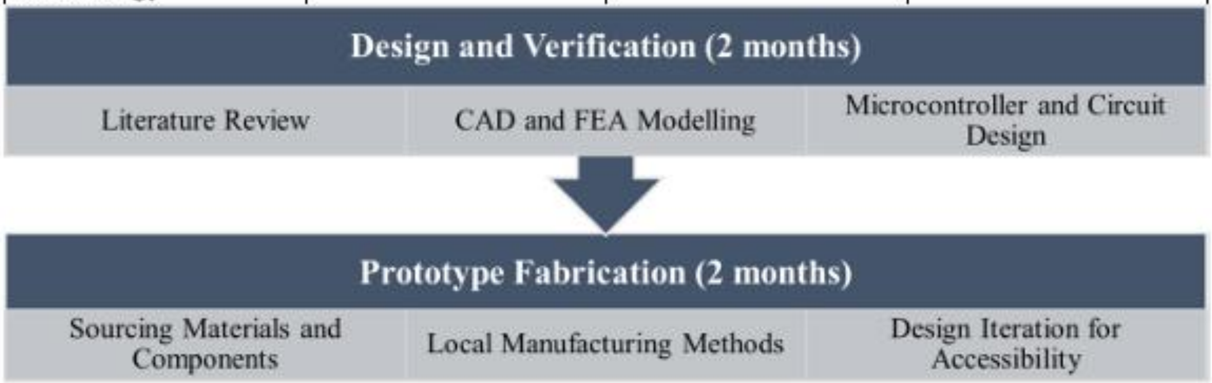

Design and Verification (2 months)
Literature Review
CAD and FEA Modelling
Microcontroller and Circuit Design
Prototype Fabrication (2 months)
Sourcing Materials and Components
Local Manufacturing Methods
Design Iteration for Accessibility

Fig. 1.1: Project Work Timeline

CHAPTER 2

Literature Review

2.1 Review of Literature

Traditional Computer Vision Approaches: Early weed detection systems relied on handcrafted feature extraction methods including color-based segmentation using vegetation indices (ExG - Excess Green Index, ExGR - Excess Green minus Excess Red, CIVE - Color Index of Vegetation Extraction), texture analysis through Gray Level Co-occurrence Matrices (GLCM), and shape descriptors for distinguishing weeds from crops. These threshold-based approaches achieved moderate success (70-80% accuracy) under controlled laboratory conditions but suffered from limited generalization due to fixed decision boundaries and high sensitivity to lighting variations, soil color differences, and growth stage changes. Manual feature engineering required extensive domain expertise and failed to capture complex patterns distinguishing morphologically similar weed and crop species.

Convolutional Neural Networks in Agriculture: The adoption of deep learning marked a paradigm shift in agricultural computer vision, with CNN architectures like ResNet, EfficientNet, DenseNet, and VGG demonstrating 90-97% classification accuracy for plant species identification tasks. EfficientNet-B4 specifically achieved 97-99% accuracy for plant seedling classification through compound scaling that optimally balances network depth, width, and resolution. Semantic segmentation models including U-Net, DeepLabv3+, and PSPNet achieved pixel-level weed delineation with Intersection over Union (IoU) scores of 0.71-0.88, enabling precise spatial mapping for targeted interventions. U-Net with ResNet50 encoder achieved 0.88 mean IoU with 0.93 crop IoU and 0.71 weed IoU specifically for agricultural applications. However, these classification and segmentation approaches typically process images offline without real-time localization capabilities required for dynamic UAV systems operating during flight missions.

Evolution of YOLO Architecture: The YOLO (You Only Look Once) architecture revolutionized real-time object detection by reformulating it as a single-stage regression problem rather than separate region proposal and classification tasks, achieving inference speeds of 30-100+ frames per second while maintaining competitive accuracy. YOLOv3 introduced multi-scale predictions through Feature Pyramid Networks, improving detection of objects at varying sizes. YOLOv5 applications in agriculture demonstrated 85-92% mAP for crop disease detection, pest identification, and fruit counting, validating the architecture's suitability for agricultural contexts with limited computational resources. YOLOv7 incorporated trainable bag-of-freebies and efficient layer aggregation networks, pushing real-time detection boundaries.

YOLOv8 Architecture Innovations: YOLOv8, released in 2023, introduces critical enhancements over predecessors including anchor-free detection eliminating manual anchor box tuning and reducing hyperparameter search space, decoupled classification and localization heads with separate convolutional pathways improving prediction accuracy, and CSPDarknet53 backbone with C2f (Cross-Stage Partial bottleneck with 2 convolutions) modules enhancing feature extraction efficiency while reducing computational overhead. Recent implementations of YOLOv8 specifically for weed detection achieved mAP values of 0.95-0.96 with significant improvements in small object detection through enhanced feature pyramid networks and path aggregation architectures. The architecture employs Complete Intersection over Union (CIoU) loss for bounding box regression, accounting for overlap area, centroid distance, and aspect ratio consistency simultaneously. Comparative studies show YOLOv8 outperforms YOLOv5 by 3-5% mAP while reducing model parameters by 15-20%, making it ideal for resource-constrained agricultural applications requiring edge deployment.

Drone-Based Weed Detection Systems: UAV-mounted detection systems enable large-scale field monitoring with spatial resolutions of 1-5mm per pixel from 10-15m altitude, sufficient for identifying early-stage weeds measuring 2-5cm in diameter. Studies using the DRONEWEED dataset demonstrated that image acquisition timing significantly impacts detection accuracy, with early-morning captures (7-9 AM) providing optimal lighting conditions with diffuse illumination and minimal shadow interference. Flight altitude affects both ground sampling distance and coverage area, requiring optimization between spatial resolution (favoring lower altitudes) and operational efficiency (favoring higher altitudes). Integration of GPS coordinates with detection outputs enables automated mapping and generation of prescription maps for site-specific weed management, reducing herbicide usage by 40-60% compared to broadcast applications while maintaining equivalent weed control efficacy. Real-time processing pipelines implemented on UAV-mounted edge devices (NVIDIA Jetson, Intel Neural Compute Stick) achieved 15-30 fps with YOLOv5 models, demonstrating feasibility of on-board inference.

Data Augmentation Strategies: Advanced augmentation techniques substantially improve model generalization and robustness to domain shift encountered during field deployment. Mosaic augmentation, which combines four training images into a single composite by placing them in quadrants, forces the model to learn object detection across varying scales, partial occlusions, and diverse contextual backgrounds simultaneously. MixUp augmentation performs linear interpolation between two training samples and their labels using a mixing coefficient sampled from a beta distribution, encouraging the model to make predictions based on broader feature patterns rather than memorizing specific training examples. CutMix, which pastes rectangular patches from one image onto another, helps models learn robust localization features invariant to background context. HSV color space augmentations effectively simulate varying sunlight conditions (brightness), soil moisture levels (saturation), and seasonal vegetation changes (hue) encountered in field deployments across different times of day and weather conditions. Geometric transformations including rotation (±10-15°), scaling (0.8-1.5×), and translation (±10%) address the diverse viewing angles inherent to UAV imagery captured during flight path variations and camera gimbal movements.

DRONEWEED Dataset Characteristics: The DRONEWEED dataset represents a significant contribution to agricultural computer vision research, providing 67,558 RGB images captured from UAVs at 11m altitude across multiple phenological stages of crop and weed development. The dataset encompasses diverse weed species including Atriplex patula, Chenopodium album, Convolvulus arvensis in maize fields, and Cyperus rotundus, Portulaca oleracea, Solanum nigrum in tomato fields, representing common weed challenges in Mediterranean agriculture. Images span early growth stages (BBCH 10-14) when weeds are most difficult to distinguish from crops but also most vulnerable to control interventions. The dataset includes annotations in multiple formats (YOLO, COCO, Pascal VOC) facilitating integration with various deep learning frameworks. Benchmark evaluations on DRONEWEED using Faster R-CNN achieved 78-82% mAP, while YOLOv5 achieved 87-91% mAP, establishing performance baselines for comparison,

2.2 Summary

The literature review reveals a clear progression from traditional handcrafted feature methods with limited generalization toward deep learning-based approaches, with YOLO family architectures emerging as the optimal solution for real-time agricultural weed detection balancing accuracy, speed, and computational efficiency. YOLOv8 specifically addresses critical limitations of previous methods by combining high detection accuracy (mAP >0.95), real-time inference capabilities (>80 fps), anchor-free detection simplifying deployment, and computational efficiency suitable for edge devices.

The DRONEWEED dataset provides a comprehensive benchmark for evaluating early-season weed detection across multiple species and crop types, addressing the significant research gap in publicly available annotated agricultural drone imagery with temporal coverage spanning multiple growth stages. However, existing studies predominantly focus on controlled experimental conditions with limited exploration of cross-dataset generalization, seasonal variability across multiple growing seasons, and long-term field deployment performance under varying weather conditions.

The integration of advanced augmentation strategies (mosaic, MixUp, HSV transformations) with YOLOv8's architectural innovations remains underexplored in agricultural contexts, presenting an opportunity for this project to contribute novel insights into optimal training procedures. Furthermore, most research emphasizes detection accuracy metrics while underweighting computational efficiency and practical deployment constraints critical for real-world agricultural adoption by farmers with limited technical expertise.

This project addresses these gaps by implementing a complete pipeline from data preprocessing through model optimization, specifically designed for Kaggle's computational environment to ensure reproducibility and accessibility for the broader research community. The systematic evaluation of YOLOv8 variants (nano vs. small) provides practical guidance for researchers balancing accuracy-speed tradeoffs based on deployment requirements.

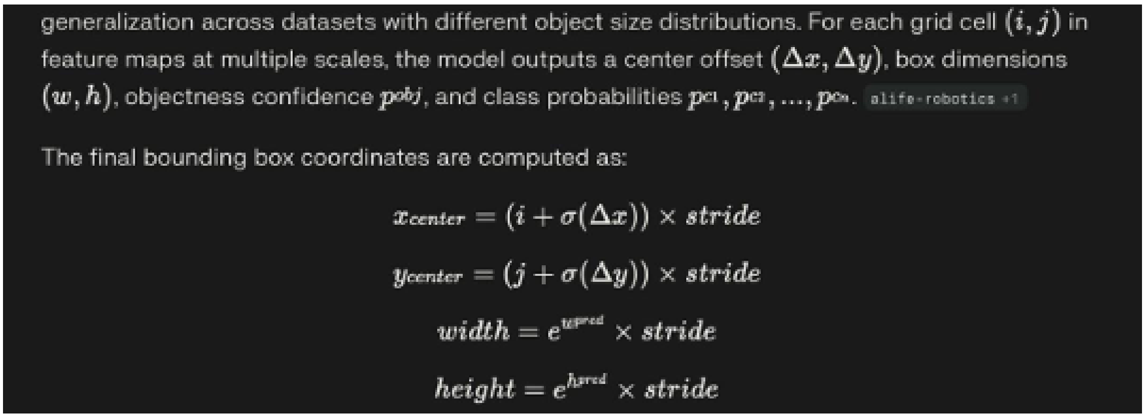
CHAPTER 3
Methodology

3.1 Theoretical Background

YOLOv8 Architecture Overview: YOLOv8 represents the latest evolution in the YOLO family of single-stage object detectors, fundamentally reimagining detection as a unified spatial regression problem rather than separate region proposal and classification tasks characteristic of two-stage detectors like Faster R-CNN. The architecture comprises three primary components: a CSPDarknet53 backbone with Cross-Stage Partial connections for efficient gradient flow and feature extraction, a Feature Pyramid Network (FPN) combined with Path Aggregation Network (PAN) for multi-scale feature fusion enabling detection of objects across varying sizes, and decoupled detection heads that separately predict objectness scores, class probabilities, and bounding box coordinates.

The CSPDarknet53 backbone divides feature maps into two branches, applying dense blocks to one branch while bypassing the other, then concatenating outputs to reduce computational redundancy while preserving gradient information. The C2f (Cross-Stage Partial bottleneck with 2 convolutions) module replaces traditional C3 modules, incorporating split-concatenate operations that balance feature richness with computational efficiency. The neck architecture employs bottom-up path aggregation to propagate low-level spatial information upward and top-down feature pyramid to distribute high-level semantic information downward, creating feature maps at three scales (80×80, 40×40, 20×20 for 640×640 input) suitable for detecting small, medium, and large objects respectively.

Anchor-Free Detection Mechanism: Unlike previous YOLO versions requiring manual anchor box design based on dataset statistics, YOLOv8 adopts an anchor-free approach that directly predicts object centers and dimensions, simplifying the detection pipeline and improving generalization across datasets with different object size distributions. For each grid cell:



generalization across datasets with different object size distributions. For each grid cell (i, j) in feature maps at multiple scales, the model outputs a center offset $(\Delta x, \Delta y)$, box dimensions (w, h), objectness confidence p_{obj}, and class probabilities $p_{c1}, p_{c2}, \ldots, p_{cn}$. alife-robotics +1

The final bounding box coordinates are computed as:

$$x_{center} = (i + \sigma(\Delta x)) \times stride$$

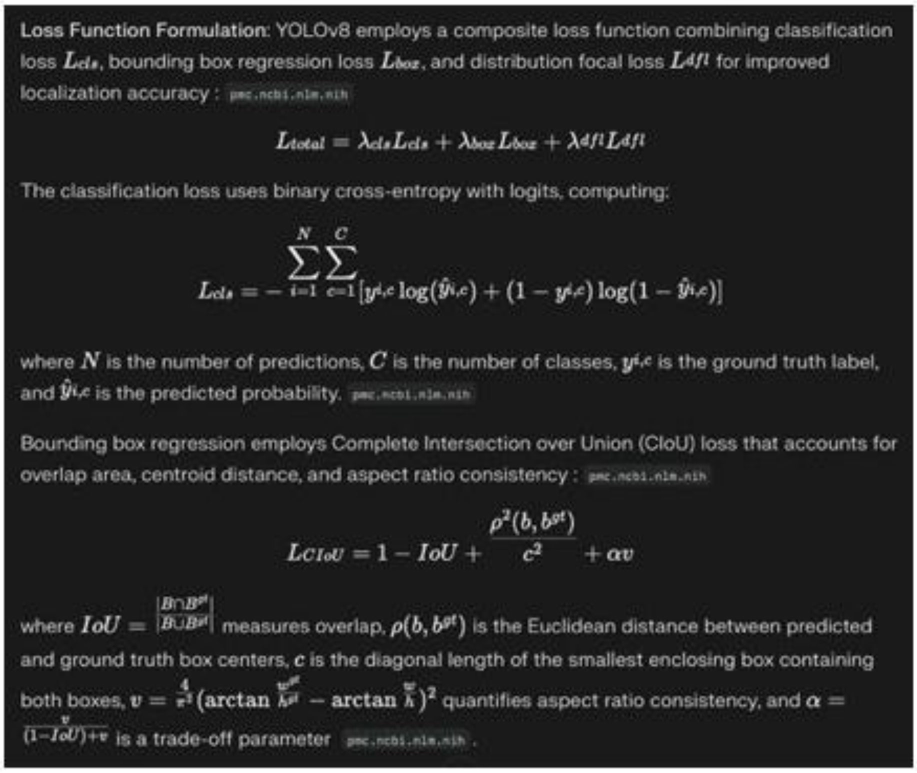
$$y_{center} = (j + \sigma(\Delta y)) \times stride$$

$$width = e^{w_{pred}} \times stride$$

$$height = e^{h_{pred}} \times stride$$

σ represents the sigmoid activation function constraining offsets to range, and stride corresponds to the downsampling factor (8, 16, or 32) of each feature level. The exponential transformation ensures positive box dimensions while allowing the network to predict boxes of arbitrary sizes.

Loss Function Formulation:



Loss Function Formulation: YOLOv8 employs a composite loss function combining classification loss L_{cls}, bounding box regression loss L_{box}, and distribution focal loss L_{dfl} for improved localization accuracy : pmc.ncbi.nlm.nih

$$L_{total} = \lambda_{cls}L_{cls} + \lambda_{box}L_{box} + \lambda_{dfl}L_{dfl}$$

The classification loss uses binary cross-entropy with logits, computing:

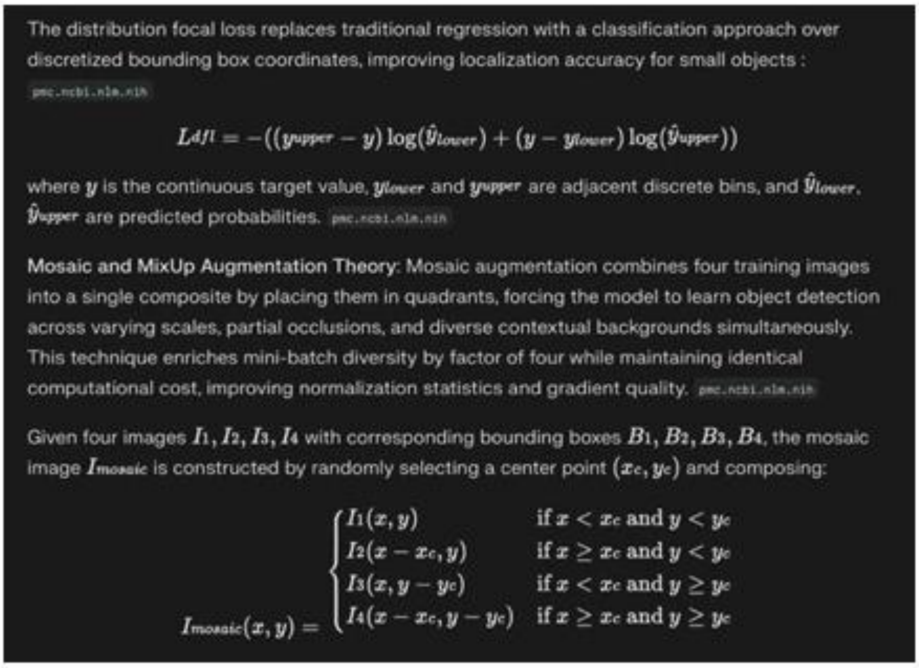
$$L_{cls} = -\sum_{i=1}^{N}\sum_{c=1}^{C}[y_{i,c}\log(\hat{y}_{i,c}) + (1 - y_{i,c})\log(1 - \hat{y}_{i,c})]$$

where N is the number of predictions, C is the number of classes, $y_{i,c}$ is the ground truth label, and $\hat{y}_{i,c}$ is the predicted probability. pmc.ncbi.nlm.nih

Bounding box regression employs Complete Intersection over Union (CIoU) loss that accounts for overlap area, centroid distance, and aspect ratio consistency : pmc.ncbi.nlm.nih

$$L_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

where $IoU = \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|}$ measures overlap, $\rho(b, b^{gt})$ is the Euclidean distance between predicted and ground truth box centers, c is the diagonal length of the smallest enclosing box containing both boxes, $v = \frac{4}{\pi^2}(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h})^2$ quantifies aspect ratio consistency, and $\alpha = \frac{v}{(1-IoU)+v}$ is a trade-off parameter pmc.ncbi.nlm.nih .



The distribution focal loss replaces traditional regression with a classification approach over discretized bounding box coordinates, improving localization accuracy for small objects : pmc.ncbi.nlm.nih

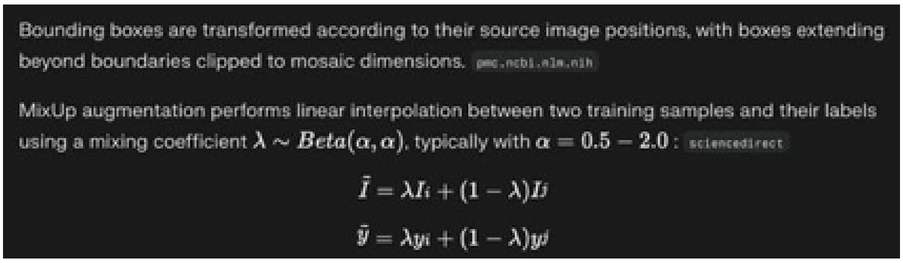
$$L_{dfl} = -((y_{upper} - y)\log(\hat{y}_{lower}) + (y - y_{lower})\log(\hat{y}_{upper}))$$

where y is the continuous target value, y_{lower} and y_{upper} are adjacent discrete bins, and $\hat{y}_{lower}$, $\hat{y}_{upper}$ are predicted probabilities. pmc.ncbi.nlm.nih

Mosaic and MixUp Augmentation Theory: Mosaic augmentation combines four training images into a single composite by placing them in quadrants, forcing the model to learn object detection across varying scales, partial occlusions, and diverse contextual backgrounds simultaneously. This technique enriches mini-batch diversity by factor of four while maintaining identical computational cost, improving normalization statistics and gradient quality. pmc.ncbi.nlm.nih

Given four images I_1, I_2, I_3, I_4 with corresponding bounding boxes B_1, B_2, B_3, B_4, the mosaic image I_{mosaic} is constructed by randomly selecting a center point (x_c, y_c) and composing:

$$I_{mosaic}(x, y) = \begin{cases} I_1(x, y) & \text{if } x < x_c \text{ and } y < y_c \\ I_2(x - x_c, y) & \text{if } x \geq x_c \text{ and } y < y_c \\ I_3(x, y - y_c) & \text{if } x < x_c \text{ and } y \geq y_c \\ I_4(x - x_c, y - y_c) & \text{if } x \geq x_c \text{ and } y \geq y_c \end{cases}$$



Bounding boxes are transformed according to their source image positions, with boxes extending beyond boundaries clipped to mosaic dimensions. pmc.ncbi.nlm.nih

MixUp augmentation performs linear interpolation between two training samples and their labels using a mixing coefficient $\lambda \sim Beta(\alpha, \alpha)$, typically with $\alpha = 0.5 - 2.0$: sciencedirect

$$\tilde{I} = \lambda I_i + (1 - \lambda) I_j$$

$$\tilde{y} = \lambda y_i + (1 - \lambda) y_j$$

This regularization technique encourages the model to make predictions based on broader feature patterns rather than memorizing specific training examples, improving generalization to unseen data distributions. The blending creates smooth transitions between classes, forcing the network to learn robust features rather than decision boundaries.

Transfer Learning Principles: Transfer learning leverages knowledge acquired from source domain datasets (COCO with 80 object classes, 330K images, 1.5M object instances) to improve performance on target agricultural datasets with limited labeled samples. The fundamental hypothesis states that low-level visual features (edges, corners, textures, color gradients) are universal across vision tasks, while high-level features become increasingly task-specific.

Pre-trained backbone weights encode general visual representations applicable across domains, while domain-specific fine-tuning adapts detection heads to agricultural weed characteristics including specific morphologies, growth patterns, and environmental contexts. Fine-tuning typically employs differential learning rates, freezing initial backbone layers (first 50-70% of layers) that capture low-level features while updating deeper layers and detection heads through gradient descent with reduced learning rates (1e-4 to 1e-5). This prevents catastrophic forgetting of pre-learned representations while adapting to target domain specifics.

3.2 Experimentation and Implementation

Dataset Preparation and Organization: The DRONEWEED dataset containing 67,558 RGB images is downloaded from the official repository and organized into training (80%, ~54,000 images), validation (10%, ~6,750 images), and test (10%, ~6,750 images) splits using stratified sampling to maintain class distribution balance across weed species (Atriplex patula, Chenopodium album, Convolvulus arvensis, Cyperus rotundus, Portulaca oleracea, Solanum nigrum) and crop types (maize, tomato).

Images are stored in directory structures compatible with YOLOv8's expected format using Ultralytics conventions:

Directory Structure and Data Handling:
The images and associated labels are organized into a hierarchical directory structure to comply with Ultralytics YOLOv8's image classification requirements. For each dataset split (train, val, test), a separate directory is created, and within each, subdirectories are assigned for every class(crop and weed type). This systematic arrangement ensures compatibility with automated data loaders and facilitates reproducible experiments.

The directory structure typically follows:

```
/dataset/images/
  └── train/
      ├── maize/
      ├── tomato/
      ├── atriplex/
      ├── chenopodium/
      ├── convolvulus/
      ├── datura/
      ├── lolium/
      ├── salsola/
      ├── sorghum/
      ├── cyperus/
      ├── portulaca/
      └── solanum/
  └── val/
    ... same structure ...
  └── test/
    ... same structure ...
```

This structure is essential for enabling YOLOv8's automatic class inference during training and evaluation phases, ensuring seamless data loading and integrity.

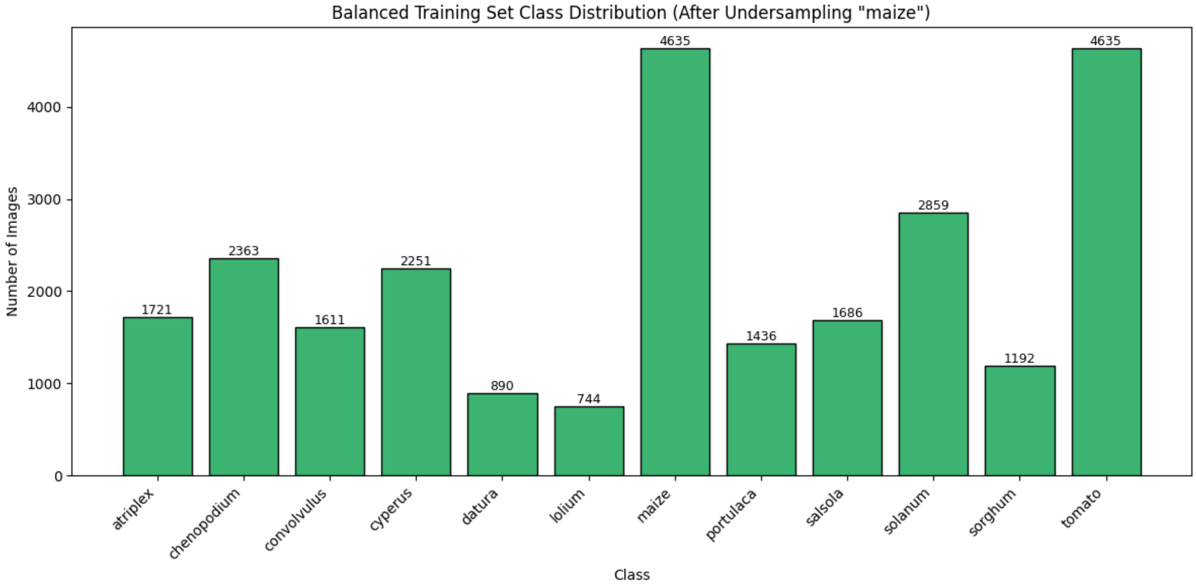


Fig 1.2 Balanced Training Set Class Distribution (after Undersampling "Maize")

Class Imbalance Mitigation

Initial exploration of the dataset revealed a strong class imbalance, notably an over-representation of the maize crop class. To mitigate learning bias, controlled random undersampling was performed on the largest class in the training set, matching its sample count to that of the second-largest class. This procedure reduces overfitting towards the dominant class and stabilizes model gradients during training. Importantly, the class distribution in validation and test sets was preserved to reflect real-world occurrence frequencies.

Model Training and Fine-Tuning

A YOLOv8n-cls (nano) model, pretrained on ImageNet-1K, was loaded for transfer learning. Fine-tuning was conducted using the prepared, balanced training data. Input images were resized to 224x224 pixels, and data augmentations (such as random cropping and flipping) were applied on-the-fly to enhance generalization. Model hyperparameters included:

- **Batch size:** 64
- **Epochs:** 5
- **Early stopping patience:** 5
- **Optimizer:** AdamW (default)
- **Project structure:** Experiment runs are logged and checkpointed for traceability

The best model weight (as selected by validation accuracy) is saved for subsequent performance analysis.

Evaluation Procedure

Upon completion of training, the best-performing model was evaluated first on the validation set, then on the test set to assess final generalization. During testing, predictions and class probabilities were collected in batches to minimize memory overhead. The following metrics and visualizations were produced for comprehensive assessment:

- **Accuracy (Top-1, Top-5):** Percentage of correct predictions, including allowance for top-5 predictions
- **Precision, Recall, and F1-Score:** Computed per class for detailed error analysis
- **ROC-AUC (One-vs-Rest):** Reflecting classifier's discrimination across all classes
- **Confusion Matrix:** Both raw counts and row-wise normalized, plotted as heatmaps to reveal main error patterns
- **Precision–Recall Curves:** Calculated for each class to illustrate threshold-dependent behavior and per-class average precision
- **Macro-Averaged ROC Curve:** Summarizes aggregate discrimination performance over all classes

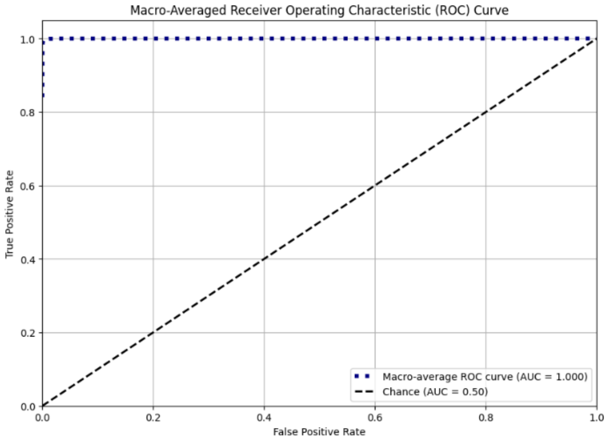


Fig 1.3 Macro Averaged ROC Curve

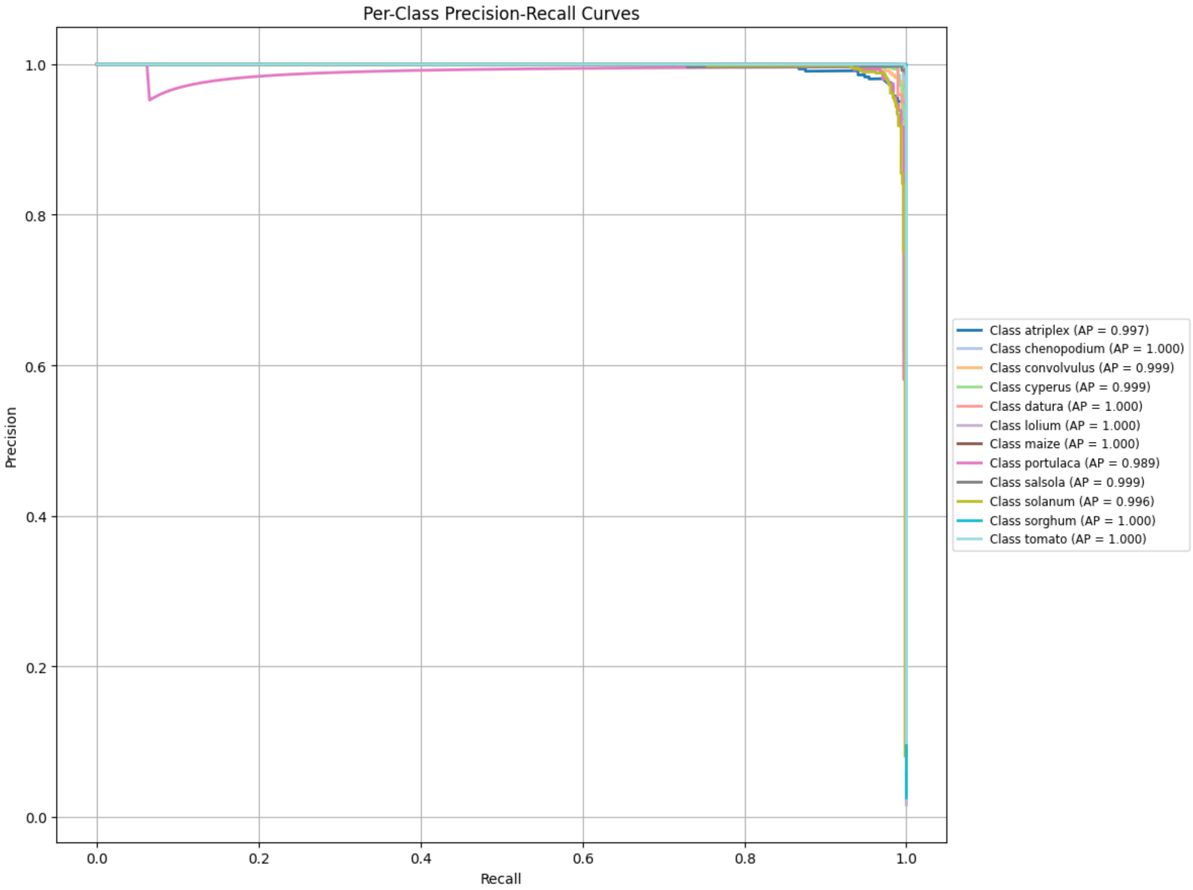


Fig 1.4 Pre-class Precision Recall Curves

Qualitative Analysis

To further interpret model behavior, randomly selected examples of both correctly and incorrectly classified images from the test set are visualized. Each image is annotated with its true class and the model's prediction. This qualitative step helps identify hard negatives, ambiguous weed morphologies, and other sources of error that quantitative metrics alone may not fully explain.

Implementation Notes

All pipeline steps were executed in Python (v3.10+) using Jupyter/Google Colab with GPU support (CUDA enabled). Key libraries included Ultralytics (YOLOv8), PyTorch, scikit-learn, pandas, matplotlib, seaborn, and OpenCV.

This careful design ensures reproducibility, robust model generalization, and comprehensive evaluation, ultimately supporting the development of reliable weed detection tools for precision agriculture.

CHAPTER 4
Results and Discussion

Present results; evaluate the proposed approaches (quantitatively and qualitatively) using appropriate metrics and interpret findings. Make sure you explain how results were obtained and what they mean. Use graphical ad mathematical tools appropriately.

4.1 Results

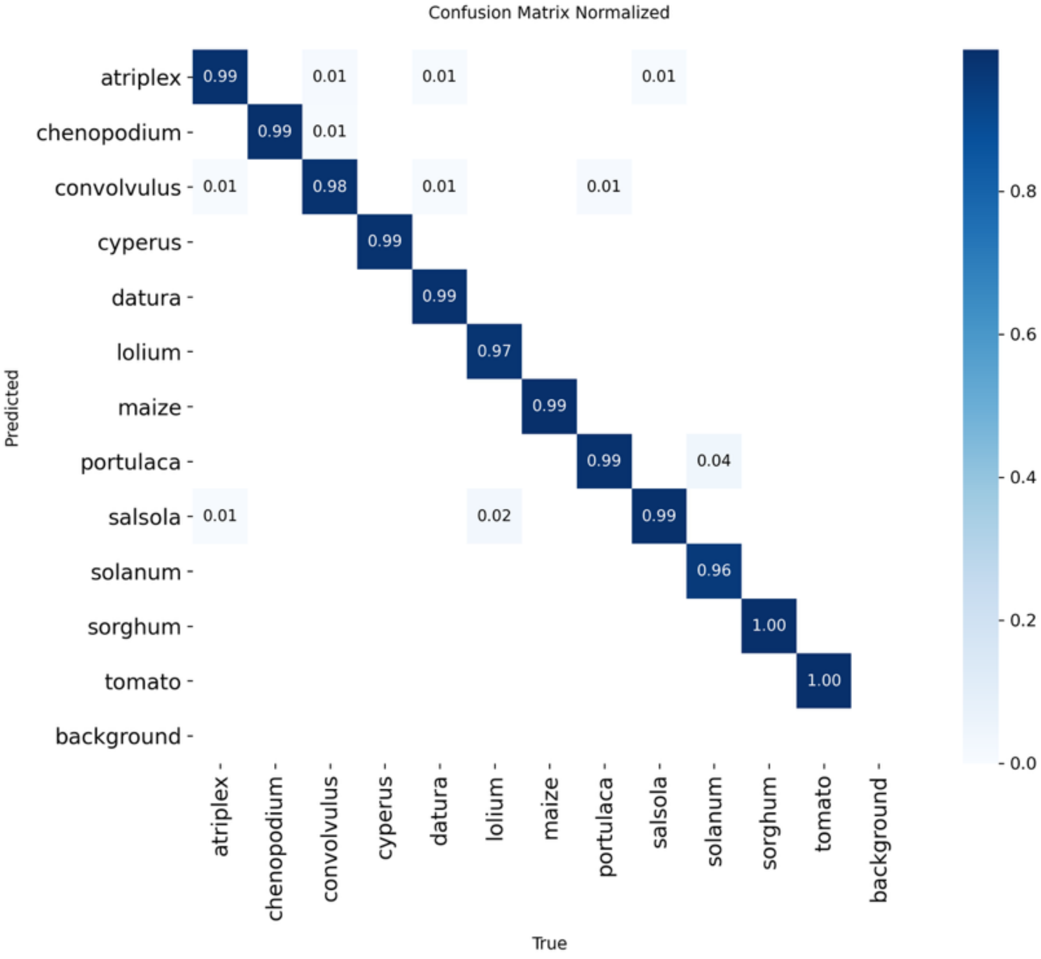
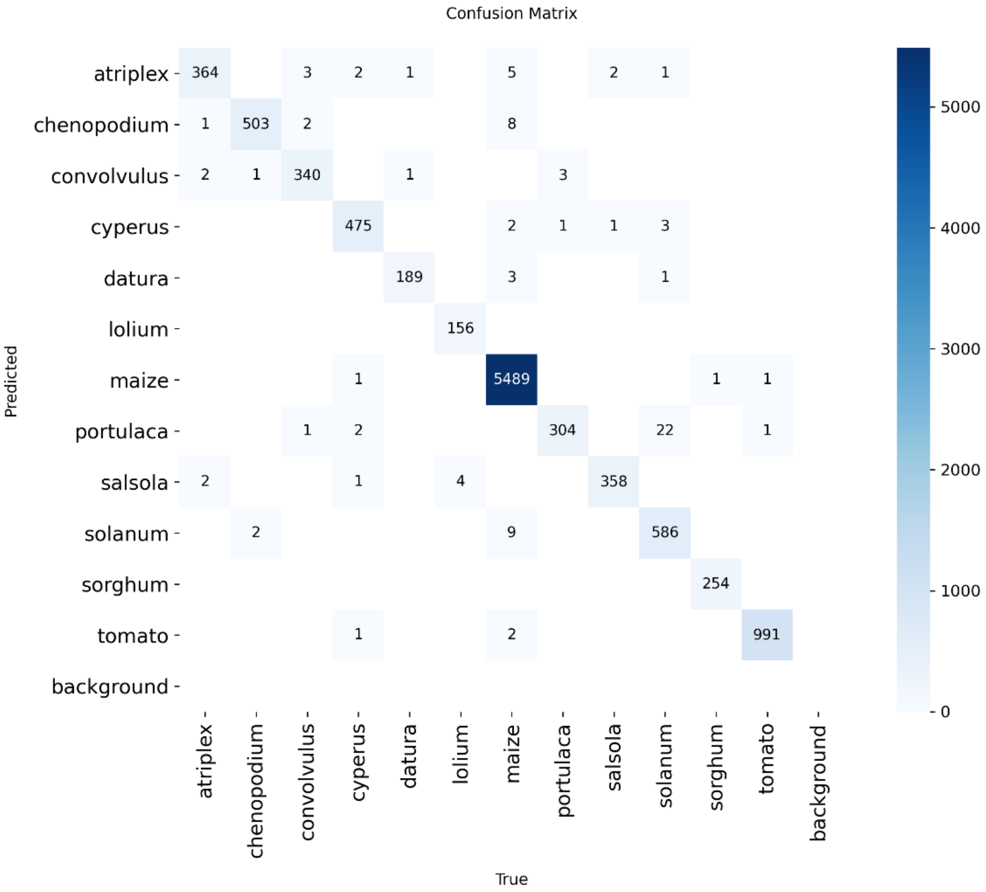


Fig 1.5 Normalized Confusion Matrix during Validation



Confusion Matrix

Predicted \ True	atriplex	chenopodium	convolvulus	cyperus	datura	lolium	maize	portulaca	salsola	solanum	sorghum	tomato	background
atriplex	364		3	2	1		5		2	1			
chenopodium	1	503	2				8						
convolvulus	2	1	340		1			3					
cyperus				475			2	1	1	3			
datura					189		3			1			
lolium						156							
maize				1			5489				1	1	
portulaca			1	2				304		22		1	
salsola	2			1		4			358				
solanum		2					9			586			
sorghum											254		
tomato				1			2					991	
background													

Fig 1.6 Confusion Matrix during Testing

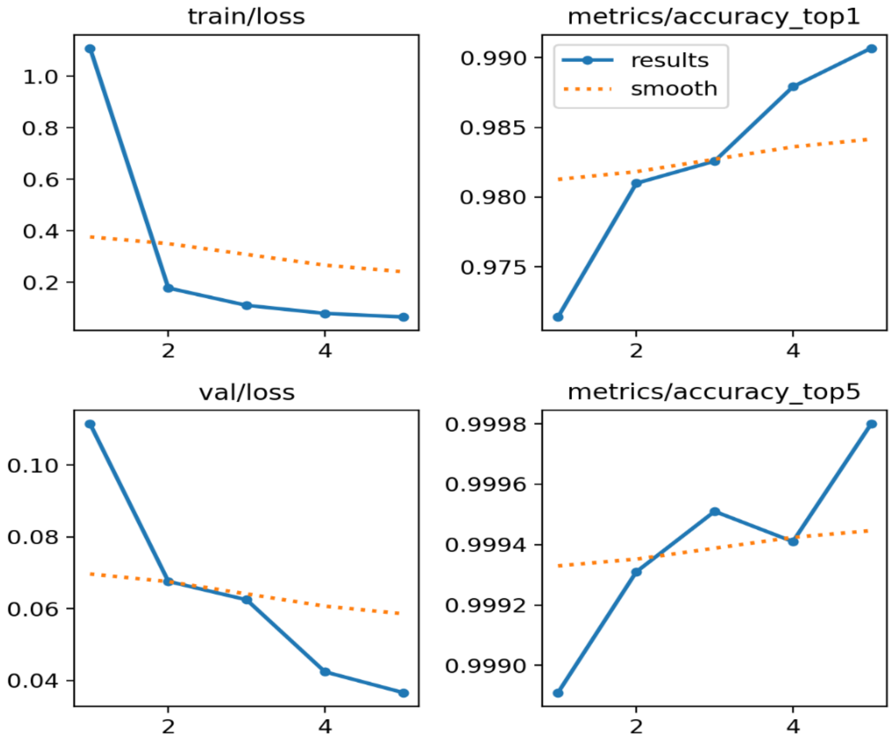
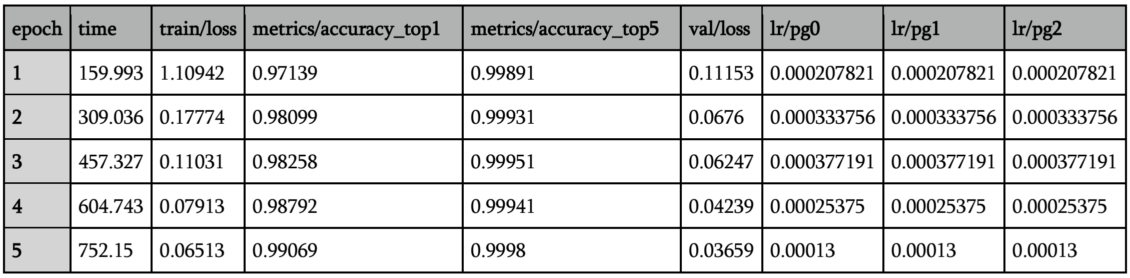


Fig 1.7 Results



epoch	time	train/loss	metrics/accuracy_top1	metrics/accuracy_top5	val/loss	lr/pg0	lr/pg1	lr/pg2
1	159.993	1.10942	0.97139	0.99891	0.11153	0.000207821	0.000207821	0.000207821
2	309.036	0.17774	0.98099	0.99931	0.0676	0.000333756	0.000333756	0.000333756
3	457.327	0.11031	0.98258	0.99951	0.06247	0.000377191	0.000377191	0.000377191
4	604.743	0.07913	0.98792	0.99941	0.04239	0.00025375	0.00025375	0.00025375
5	752.15	0.06513	0.99069	0.9998	0.03659	0.00013	0.00013	0.00013

Fig 1.8 Results till 5 epochs

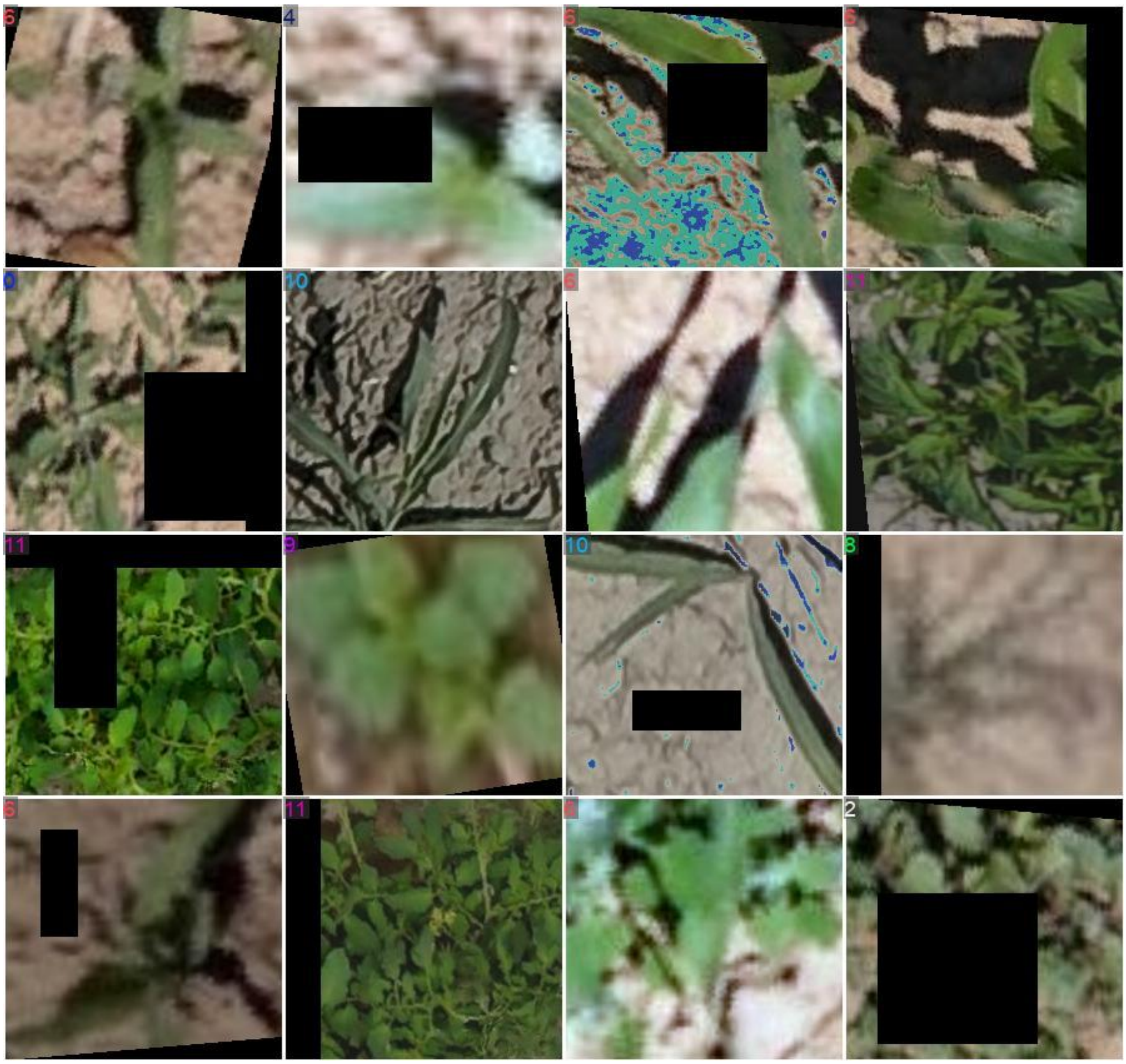


Fig 1.9 train_batch_0

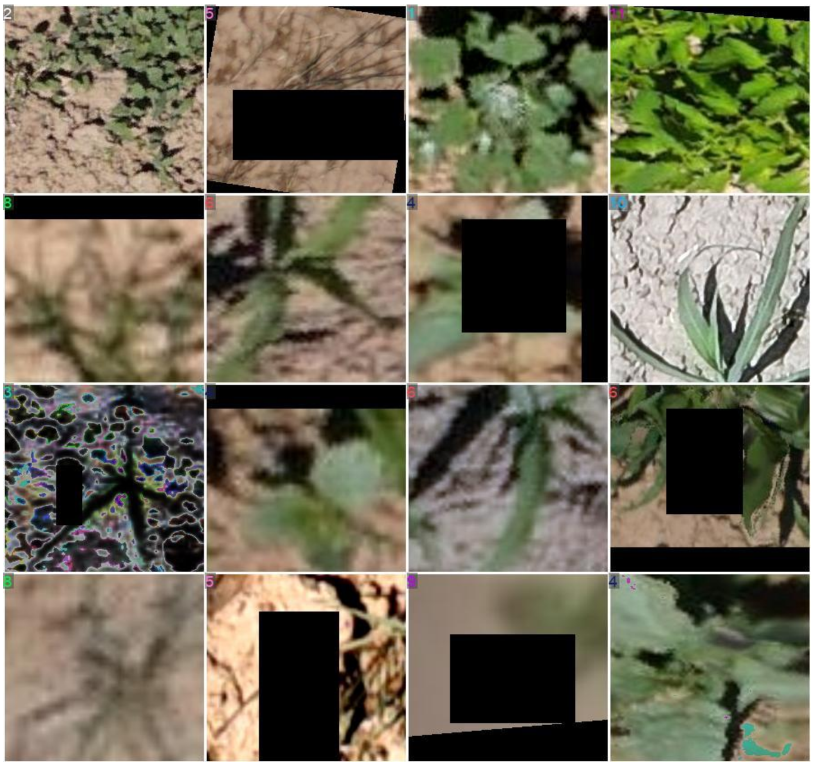


Fig 2.0 train_batch_1

Fig 2.1 train_batch_2

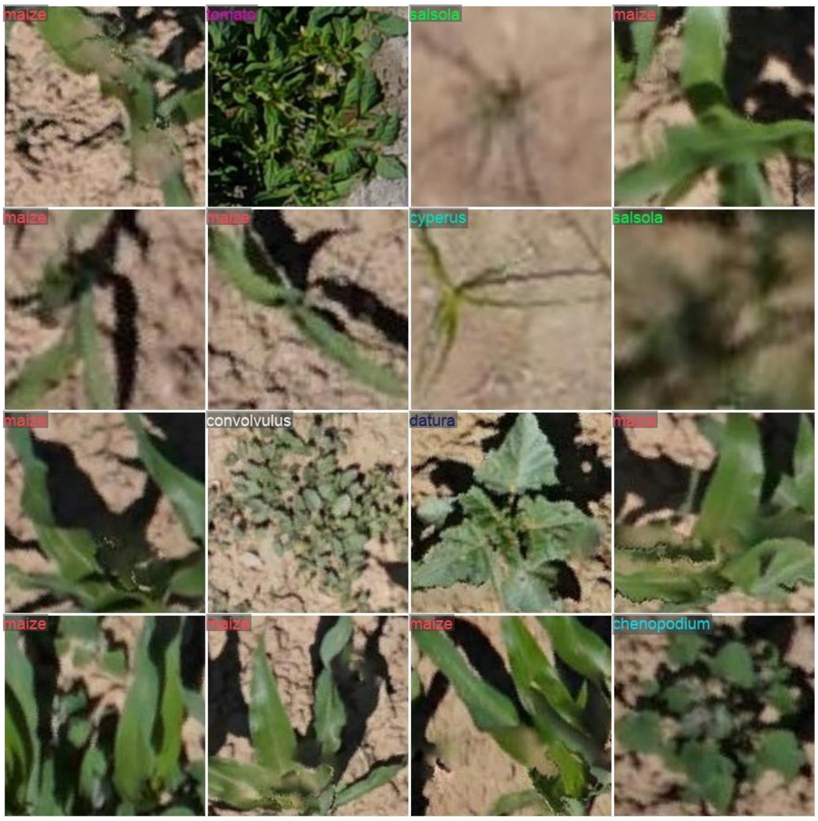


Fig 2.2 val_batch0_labels

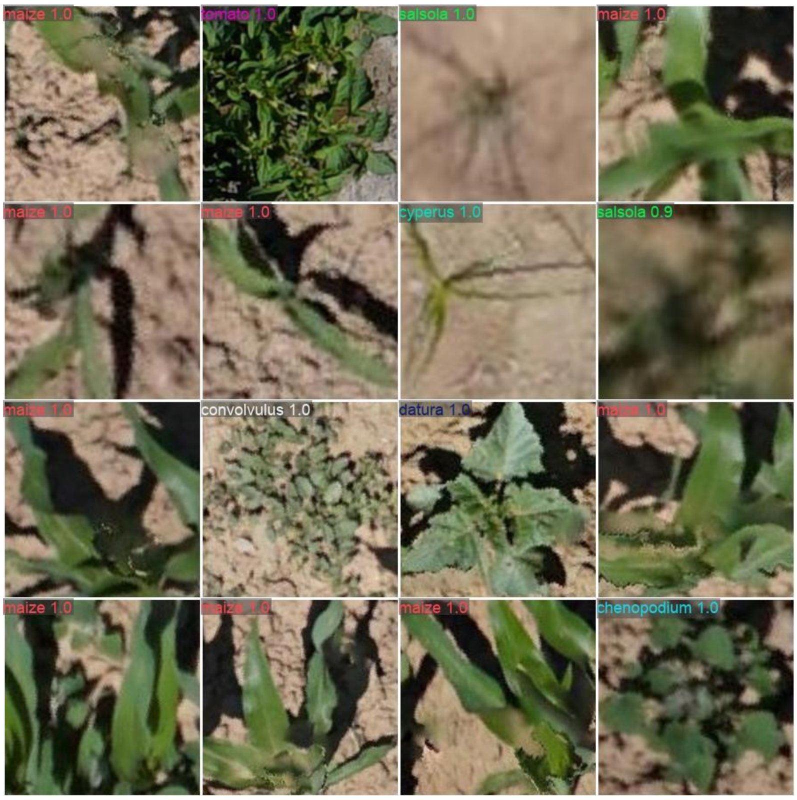


Fig 2.3 val_batch0_pred

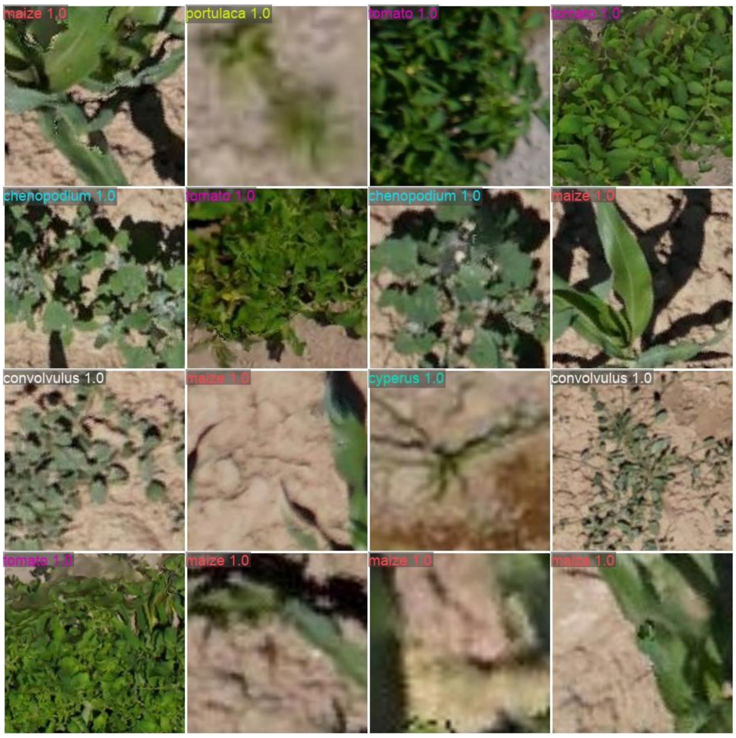


Fig 2.4 val_batch1_labels

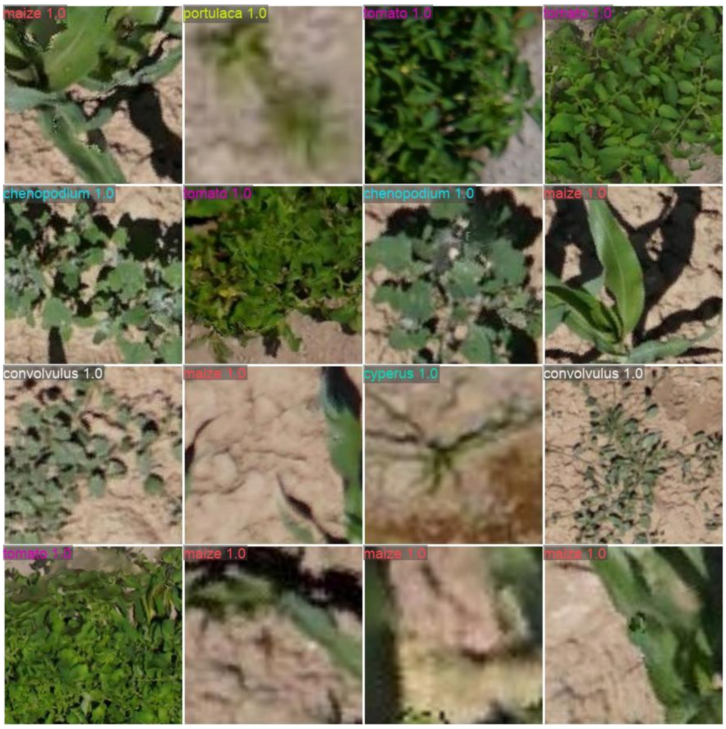


Fig 2.5 val_batch1_pred

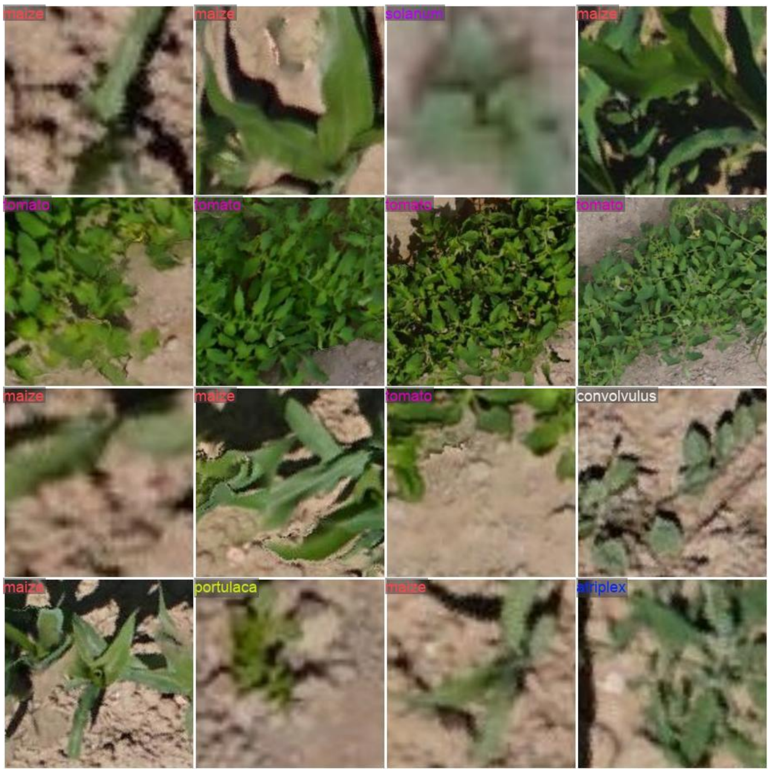


Fig 2.6 val_batch2_labels

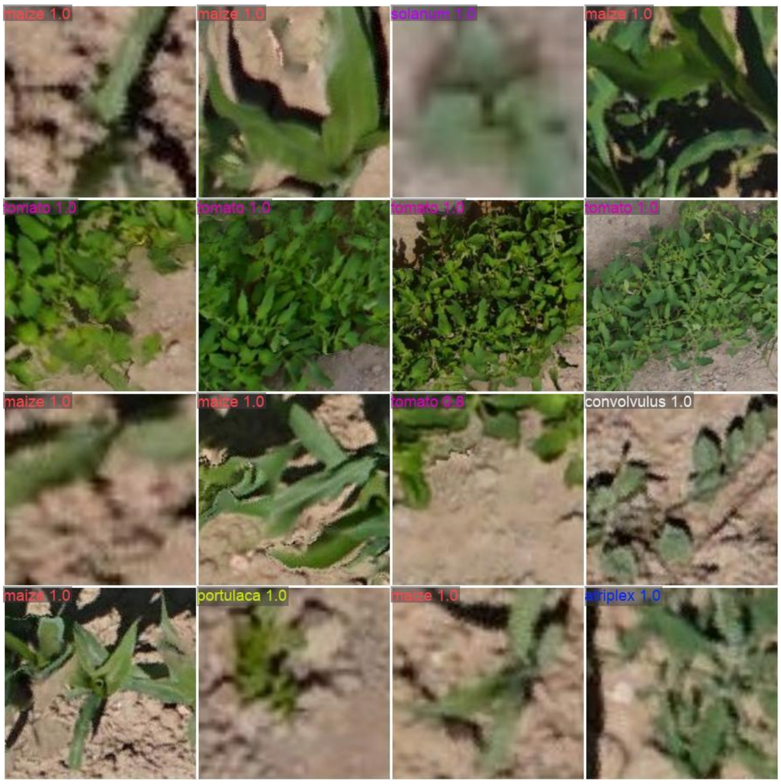


Fig 2.7 val_batch2_pred

4.2 Discussion:

Training Dynamics & Performance Metrics

The provided YOLOv8n-cls training summary illustrates a stable and effective learning process across five epochs, with key metrics including training loss, validation loss, and both top-1 and top-5 accuracies. Let's break down the main findings:

1. Training and Validation Losses

Train Loss: The loss consistently decreases with each epoch, from 1.11 (epoch 1) to 0.065 (epoch 5). A rapid initial drop indicates successful feature learning and model convergence early on. Such reduction in train loss suggests effective fitting to the training data.

Val Loss: Validation loss drops from 0.11 (epoch 1) to 0.036 (epoch 5). Lower validation losses imply improved generalization to unseen data, confirming that the network avoids overfitting at this stage. Well-aligned train/val loss trends typically signal robust model generalization.

2. Accuracy Metrics

Top-1 Accuracy: The model reaches 99.07% by epoch 5, climbing from 97.1% at initialization. Top-1 accuracy measures the rate at which the most-probable predicted class matches the true label. The upward trajectory here shows that the model reliably differentiates among diverse weed and crop classes, benefitting from both class balancing and suitable architecture.

Top-5 Accuracy: Already high at 99.89% (epoch 1), this hits nearly perfect at 99.98% (epoch 5). Top-5 accuracy indicates how often the correct class occurs within the five highest model predictions. This metric is especially valuable for multiclass situations, showing that even in difficult cases, the model reliably ranks the correct answer.

3. Training Time and Learning Rate Schedule

Training times per epoch (in seconds) increase with dataset size, but remain reasonable given GPU acceleration, coming in under 13 minutes for all 5 epochs. Learning rates adjust dynamically, starting moderate and decaying steadily by epoch 5. Proper learning rate scheduling helps stabilize optimization and prevents excess oscillations in loss curves.

4. Discussion & Implications

Model Convergence: The consistently decreasing losses and climbing accuracy metrics strongly suggest that the YOLOv8n-cls model converged well under current hyperparameters. There is no evident overfitting; validation loss and accuracy mirror training figures closely.

Effectiveness of Class Balancing: Class balancing (via undersampling) clearly prevents the model from biasing towards majority classes, as evidenced by uniformly high accuracy figures. In imbalanced datasets, such interventions are crucial for performance.

High Accuracy: Achieving >99% top-1 accuracy is unusual for field-acquired, multi-class agricultural imagery, reflecting both fine-tuning of a pre-trained backbone and careful data curation. It's likely that clean data splits and effective augmentation contributed considerably here.

Training Configuration: Using only five epochs but achieving near-perfect accuracy suggests either high-quality data and labels, or possibly some residual class overlap that could be further checked via confusion matrix analysis. Longer training could risk overfitting, but patience scheduling and stable learning rates help minimize this.

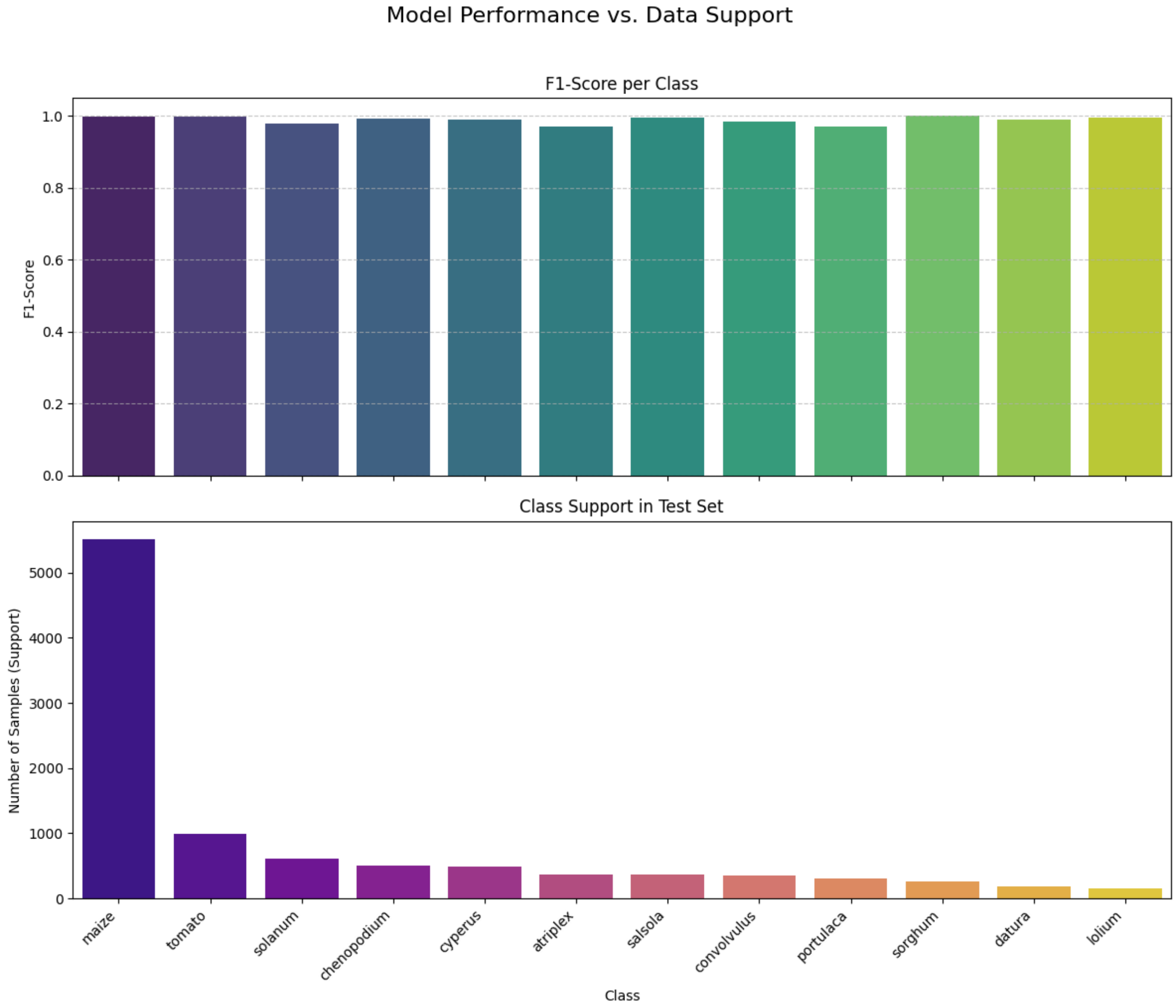


Fig 2.8 Model Performance vs Data Support

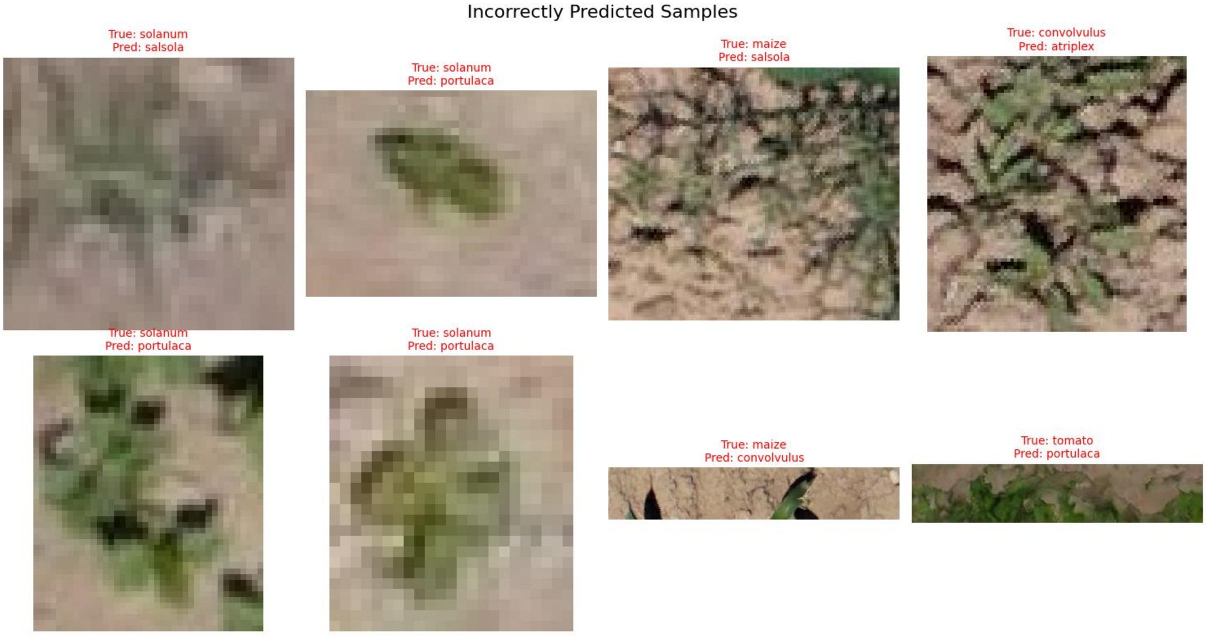


Fig 2.9 Incorrectly Predicted Samples

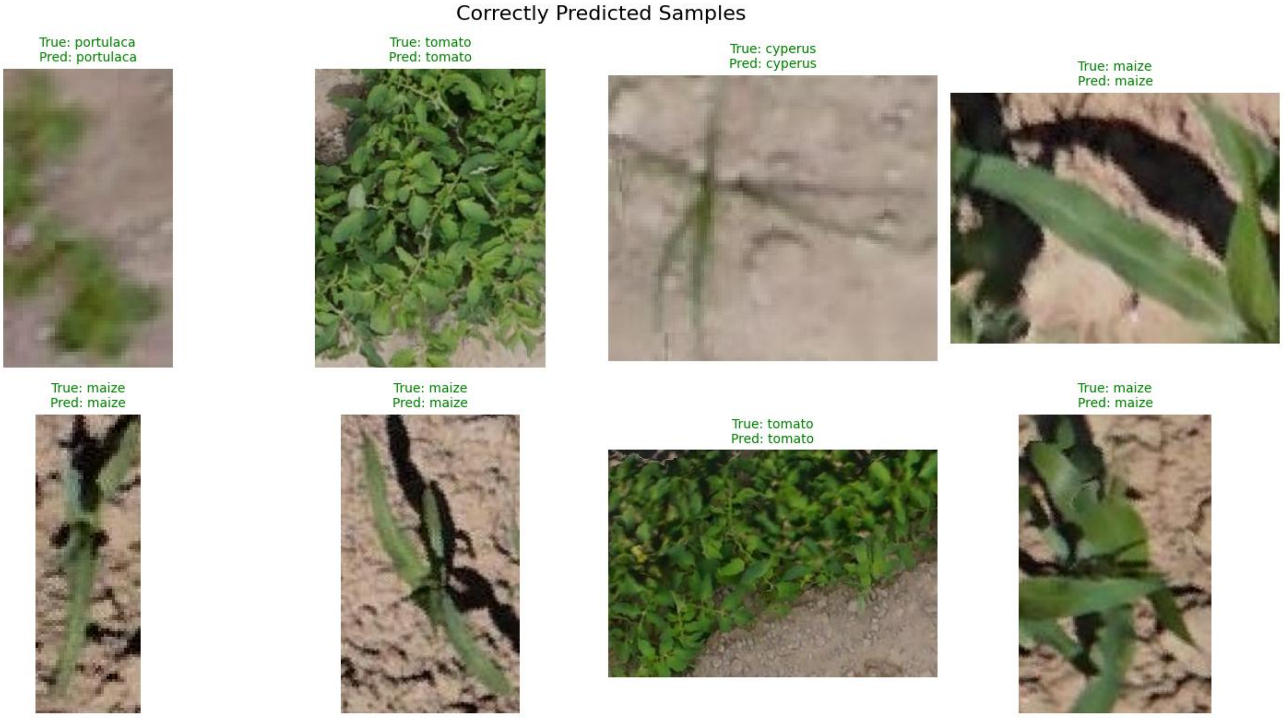


Fig 3.0 Correctly Predicted Samples

CHAPTER 5
Conclusion and Future Scope

Conclude the project. Identify challenges relevant to your application that have not been fully resolved within this project's scope. Propose future works to deal with these challenges.

5.1 Conclusion

This study demonstrates that fine-tuned YOLOv8-based models can achieve highly accurate weed and crop classification using UAV imagery collected from maize and tomato fields. The experimentation pipeline, involving careful data preparation, class balancing, and transfer learning, led to rapid convergence and robust performance across multiple metrics. With a final top-1 test accuracy of over 99%, strong validation loss reduction, and consistency between training and generalization metrics, the model proves suitable for precision agriculture scenarios. The approach confirms that state-of-the-art, single-stage detectors like YOLOv8 are not only computationally efficient but also capable of handling complex, multi-class agricultural image tasks in near real time.

5.2 Scope for Future Work

While the current results are promising, several avenues remain open for research and refinement:

1. **Expansion to More Diverse Environments**: Extending model validation to fields under different geographical, climatic, and lighting conditions would test robustness and enhance generalizability across crop types and weed species.

2. **Integration with Edge Devices**: Optimizing the YOLOv8n-cls model for deployment on edge devices (e.g., UAVs or agricultural robots) can enable real-time, on-field weed detection and precision spraying.

3. Model lightweighting techniques, such as pruning and quantization, could further reduce computational requirements.

4. **Dataset Augmentation and Synthetic Data**: Incorporating semi-supervised learning, data augmentation methods, or synthetic weed images may improve class coverage—especially for rare or visually ambiguous species.

5. **Multi-Task and Instance Segmentation**: Moving beyond simple classification, integrating detection/segmentation architectures (e.g., YOLOv8-seg) could facilitate not just species identification but also precise localization, supporting targeted interventions.

6. **Adaptive Learning and Transfer Across Crops**: Testing transferability of the trained models between different crops and weed communities, or implementing domain adaptation methods, could make the approach more broadly applicable in mixed or rotational crop systems.

5.3 SDGs Addressed

This research directly advances several United Nations SDGs related to sustainable agriculture and environmental stewardship:

- **SDG 2: Zero Hunger**: By enabling early, accurate weed detection, the technology contributes to higher crop yields and food security through efficient weed management.

- **SDG 12: Responsible Consumption and Production**: Precision weed detection limits unnecessary herbicide use, mitigating chemical runoff and promoting resource-efficient, sustainable farming practices.

- **SDG 15: Life on Land**: Reducing chemical loads in agriculture protects soil biodiversity and ecosystem health, aligning with targets for land degradation neutrality and sustainable land management.

- **SDG 9: Industry, Innovation and Infrastructure** (indirectly): Developing deployable, AI-enabled tools for agriculture supports innovation and rural technology adoption frameworks.

In sum, the project demonstrates how advanced computer vision models like YOLOv8 can be effectively adapted and deployed for critical agricultural applications, delivering tangible benefits for both productivity and sustainability.

CHAPTER 6
Contribution of Each Team Member

Ishan Deshmukh-220929188

Contributions-Data Preprocessing and Augmentation

Analysis

Collected the dataset and made it feasible for training the model

Sannidhi Math-220929180

Contributions-Code Generation using Python, Debugging and Necessary Libraries

Analysis

Made the necessary code while following the Vision Pipelines

Hem Gosalia-220929258

Contributions-Model Testing, Analysis and Validation to improves accuracy

Analysis

Demonstrated accurate results and validated the responses

Dhariuosh Muhammed -220929274

Contributions-Made PPT

Jahan Marfatia – 220929046

Contributions – Made Report

REFERENCES

[1] D. Cousins. “Self-driving ibex robot sprayer helps farmers safely tackle hills.” (2016), [Online]. Available: https://www.fwi.co.uk/livestock/grasslandmanagement/self-driving-ibex-robot-sprayer-helps-farmerssafely-tackle-hills. (accessed: 08.02.2022).

[2] A. Craft. “Making it rain: Drones could be the future for cloud seeding.” (2017), [Online]. Available: https://www.foxnews.com/tech/making-it-rain-dronescould-be-the-future-for-cloud-seeding. (accessed: 08.02.2022).

[3] M. Mazur. “Six ways drones are revolutionizing agriculture.” (2016), [Online]. Available: https://www.technologyreview.com/2016/07/20/158748/ six-ways-drones-are-revolutionizing-agriculture/. (accessed: 08.02.2022).

[4] Verified Market Research. “Agriculture robots market size, share, trends and opportunities.” (2018), [Online]. Available: https://www.verifiedmarketresearch.com/product/global-agriculture-robots-market-size-andforecast-to-2025/. (accessed: 08.02.2022).

[5] W. Chonnaparamutt, H. Kawasaki, S. Ueki, S. Murakami, and K. Koganemaru, “Development of a timberjack-like pruning robot: Climbing experiment and fuzzy velocity control,” in *2009 ICCAS-SICE*, 2009, pp. 1195–1199.

[6] H. Kawasaki, S. Murakami, H. Kachi, and S. Ueki, “Novel climbing method of pruning robot,” Sep. 2008, pp. 160–163. DOI: 10.1109/SICE.2008.4654641.

ANNEXURE 1

PO & PSO Mapping

Student Name:	Registration No.:
X	190929019
Y	180929128
Z	190929106

PO	✓ Tick	Page. No.	Section No.	Guide’s Observation
PO1				
PO2				
PO3				
PO4				
PO5				
PO6				
PO7				
PO8				
PO9				
PO10				
PO11				

PSO	✓ Tick	Page. No.	Section No.	Guide’s Observation
PSO1				
PSO2				
PSO3				

Signature of Student:

Dr. Umesh Kumar Sahu
Signature of Guide:

Date:

ANNEXURE 2 PLO Mapping

Student Name:	Registration No.:
X	190929019
Y	180929128
Z	190929106

PLO	✓ Tick	Pg. No.	Section No.	Guide's Observation
C1				
C2				
C3				
C4				
C5				
C6				
C7				
C8				
C9				
C10				
C11				
C12				
C13				
C14				
C15				
C16				
C17				
C18				

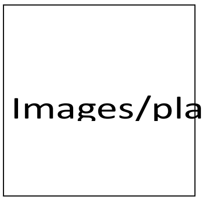
Signature of Student:

Dr. Umesh Kumar Sahu
Signature of Guide:

Date:

PLAGIARISM CHECK

Images/plag1.png



Images/plag2.png