Industrial Internship Report on

” Crop and Weed Detection Using YOLOv8”

Prepared by

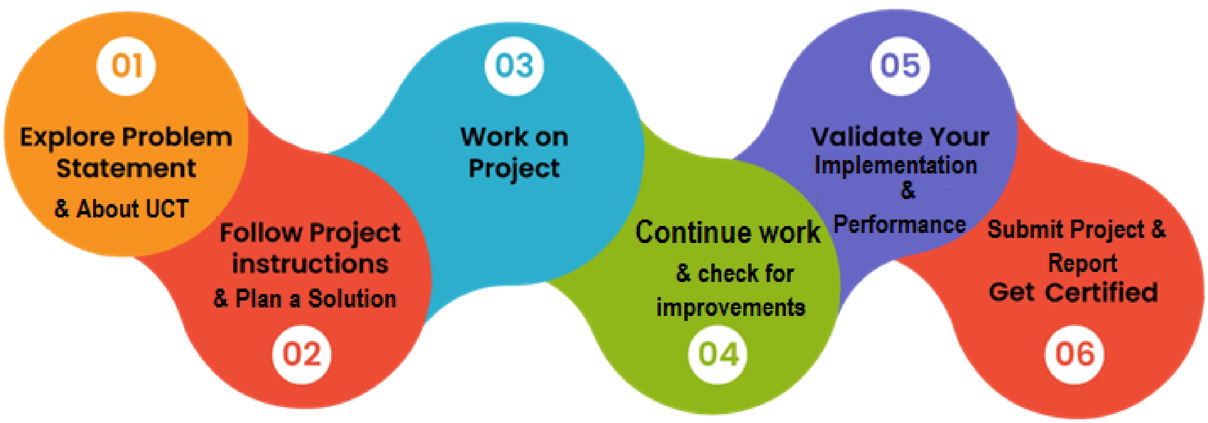
[DineshKumar S]

Executive Summary
This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT). This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 6 weeks’ time. My project was uses the Ultralytics YOLOv8 framework to build an object detector that distinguishes crops from weeds. We collected a custom dataset of agricultural images, annotated each image in the YOLO format, and trained a YOLOv8n (nano) model on this data using Python. We then implemented inference scripts for images, videos, and live webcam input, as well as a simple Tkinter GUI to run detection interactively. The trained model outputs detected bounding boxes labeled “crop” or “weed,” and we save the predictions (including box coordinates and confidences) to CSV files for further analysis. This documentation details the objectives, tools, dataset structure, training process, and inference code, with clear explanations and examples. This internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solution for that. It was an overall great experience to have this internship.

TABLE OF CONTENTS

1 Preface.

This report presents the outcome of my six-week internship program, during which I gained valuable exposure to both theoretical learning and practical application. Over the course of the internship, I followed a systematic approach starting with understanding the problem statement, exploring possible solutions, implementing the project, validating its performance, and finally documenting the outcomes. Each week contributed progressively to strengthening my technical knowledge and problem-solving ability.



Internships play a crucial role in career development by bridging the gap between academic learning and industrial practices. They provide an opportunity to apply classroom concepts to real-world challenges, enhance professional skills, and cultivate confidence in tackling practical projects. This internship has given me hands-on experience in using modern AI and computer vision tools, which are highly relevant for my future career in electronics, communication, and data-driven technologies.

The project undertaken during this internship focused on **"Crop and Weed Detection using YOLOv8"**. The objective was to design and implement an intelligent vision-based system capable of distinguishing crops from weeds in agricultural images. By leveraging a custom dataset, Python, PyTorch, and the Ultralytics YOLOv8 framework, the system was trained to perform object detection and classification with the aim of supporting precision agriculture practices.

I am grateful to **UCT (Unicoverage Technologies)** for offering me this opportunity. Their structured internship program enabled me to explore problem statements, follow step-by-step project guidance, and progressively work towards a functional solution. The planned modules—ranging from understanding the problem, developing and validating the solution, and finally presenting the results—helped in gaining a holistic experience of project development.

My Learnings and Overall Experience

Through this internship, I not only strengthened my knowledge of Python, AI frameworks, and computer vision but also learned the importance of systematic dataset preparation, model training, and validation. I developed confidence in debugging, optimizing performance, and handling real-time challenges. More importantly, I realized the significance of patience, continuous improvement, and teamwork in project success. This experience has given me a clearer vision of my career path and inspired me to explore further research in AI-driven agriculture and IoT solutions.

On the **technical front**, I learned how to:

- Prepare and organize datasets in the YOLO format, ensuring proper annotation and configuration for model training.
- Train a deep learning model (YOLOv8) using Python and PyTorch, while understanding the importance of hyperparameters like learning rate, epochs, batch size, and image dimensions.
- Use tools such as **OpenCV** for handling image and video inputs, and **Tkinter** for creating simple but effective GUIs.
- Run inference on images, videos, and live webcam streams, and export the results into CSV format for further analysis.

Acknowledgments

I extend my heartfelt gratitude to **[Insert Mentor/Guide's Name]**, who guided me throughout the internship with valuable suggestions and encouragement. I also thank the **UCT training team** for providing structured learning modules, my **college faculty** for their continuous support, and my **friends and peers** who indirectly motivated me to achieve my project goals. Without their guidance and encouragement, this achievement would not have been possible.

Message to Juniors and Peers

To my juniors and peers, I would like to emphasize that internships are golden opportunities to gain real-world experience—make the most of them. Do not hesitate to explore new technologies, ask questions, and learn from mistakes. Be consistent in your efforts, stay curious, and always connect your academic knowledge with practical applications. The skills and confidence you gain here will become stepping stones for your future career.

2 Introduction

2.1 About UniConverge Technologies Pvt Ltd

A company established in 2013 and working in Digital Transformation domain and providing Industrial solutions with prime focus on sustainability and RoI.

For developing its products and solutions it is leveraging various **Cutting Edge Technologies e.g. Internet of Things (IoT), Cyber Security, Cloud computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), Java Full Stack, Python, Front end** etc.

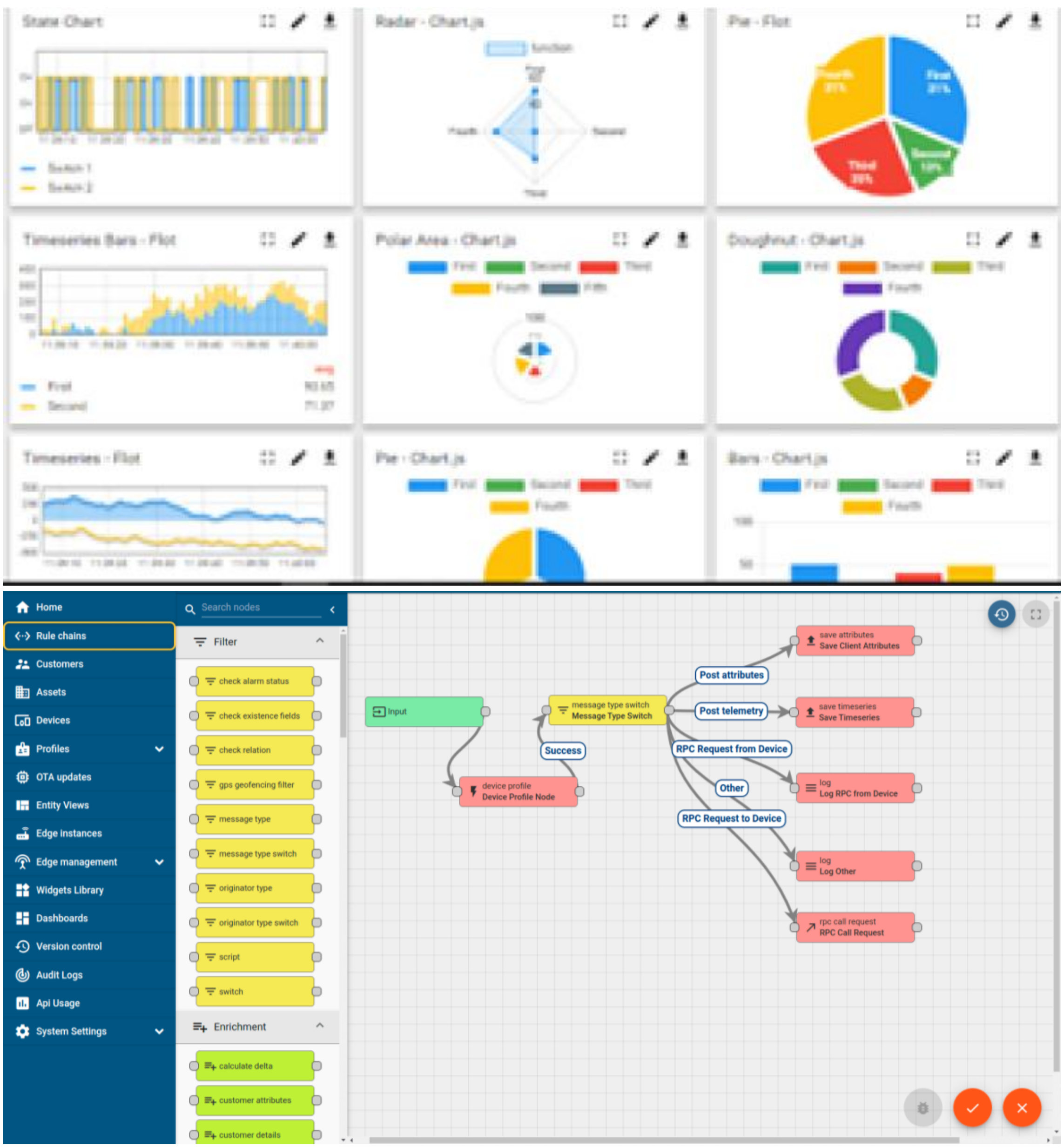
i. UCT IoT Platform (uct Insight)

UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA
- It supports both cloud and on-premises deployments.

It has features to

- Build Your own dashboard
- Analytics and Reporting
- Alert and Notification
- Integration with third party application(Power BI, SAP, ERP)
- Rule Engine



ii. Smart Factory Platform (FACTORY WATCH)

Factory watch is a platform for smart factory needs.

It provides Users/ Factory

- with a scalable solution for their Production and asset monitoring
- OEE and predictive maintenance solution scaling up to digital twin for your assets.
- to unleased the true potential of the data that their machines are generating and helps to identify the KPIs and also improve them.
- A modular architecture that allows users to choose the service that they what to start and then can scale to more complex solutions as per their demands.

Its unique SaaS model helps users to save time, cost and money.

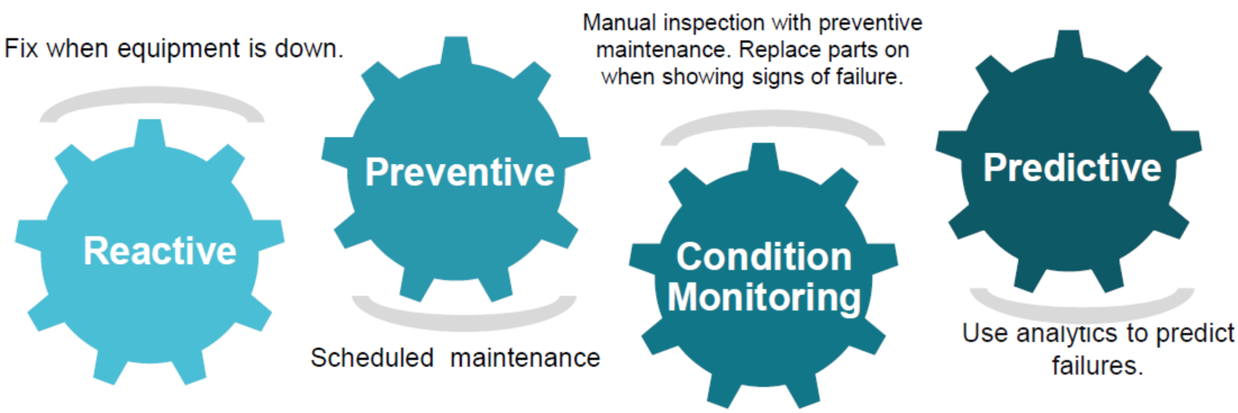
Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i

iii. LoRaWAN based Solution

UCT is one of the early adopters of LoRAWAN teschnology and providing solution in Agritech, Smart cities, Industrial Monitoring, Smart Street Light, Smart Water/ Gas/ Electricity metering solutions etc.

iv. Predictive Maintenance

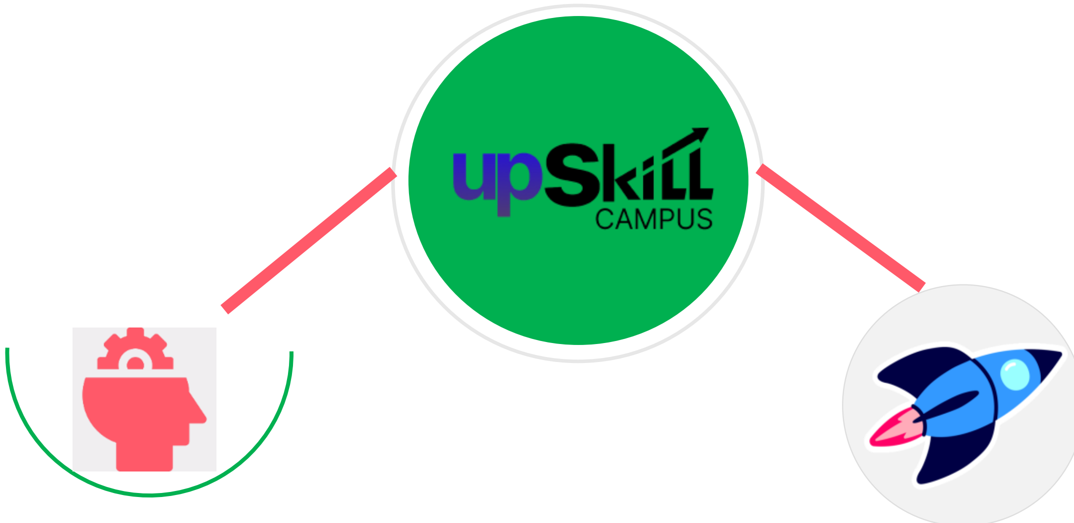
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.



2.2 About upskill Campus (USC)

upskill Campus along with The IoT Academy and in association with Uniconverge technologies has facilitated the smooth execution of the complete internship process.

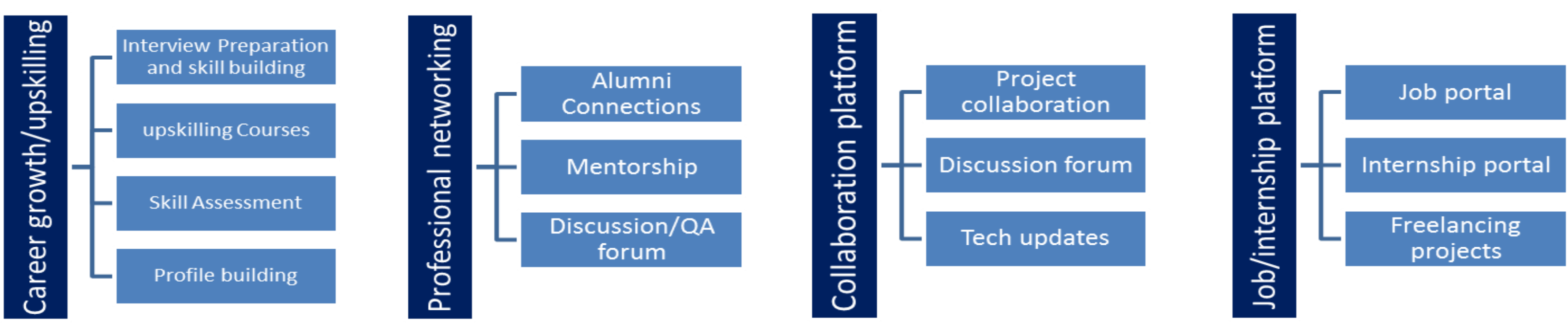
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



Seeing need of upskilling in self paced manner along-with additional support services e.g. Internship, projects, interaction with Industry experts, Career growth Services

upSkill Campus aiming to upskill 1 million learners in next 5 year

https://www.upskillcampus.com/



2.3 The IoT Academy

The IoT academy is EdTech Division of UCT that is running long executive certification programs in collaboration with EICT Academy, IITK, IITR and IITG in multiple domains.

2.4 Objectives of this Internship program

The objective for this internship program was to

☛ get practical experience of working in the industry.

☛ to solve real world problems.

☛ to have improved job prospects.

☛ to have Improved understanding of our field and its applications.

☛ to have Personal growth like better communication and problem solving.

2.5 Reference

• Ultralytics. *YOLOv8 Documentation – Custom Training and Inference.* Available at: https://docs.ultralytics.com (Accessed: 2025).

• Redmon, J., & Farhadi, A. (2018). *YOLOv3: An Incremental Improvement.* arXiv preprint arXiv:1804.02767. Available at: https://arxiv.org/abs/1804.02767.

• Sa, I., Popović, M., Khanna, R., Chen, Z., Lottes, P., Liebisch, F., Nieto, J., & Siegwart, R. (2018). *WeedMap: A large-scale semantic weed mapping framework using aerial multispectral imaging and deep neural networks.* Remote Sensing, 10(9), 1423.

2.6 Glossary

Terms	Acronym/definition
AI	Artificial Intelligence – Simulation of human intelligence in machines that can think, learn, and make decisions.
CNN	Convolutional Neural Network – A deep learning model widely used for image recognition and classification.
Dataset	A collection of images or data used for training, validation, and testing of machine learning models.
Epoch	One complete pass of the entire dataset through the training process of a model.
OpenCV	Open Source Computer Vision Library – A library of programming functions for real-time computer vision applications.

3 Problem Statement

Weed is an unwanted thing in agriculture. Weed use the nutrients, water, land and many more things that might have gone to crops. Which results in less production of the required crop. The farmer often uses pesticides to remove weed which is also effective but some pesticides may stick with crop and may causes problems for humans.

Weeds are **unwanted plants** that grow alongside crops in agricultural fields. Although they may seem harmless, weeds compete directly with crops for essential resources such as **nutrients, water, sunlight, and land space**. This competition reduces the growth and productivity of the main crop, leading to **lower yields** and economic loss for farmers.

Traditionally, farmers rely on **manual weeding** or the use of **chemical pesticides and herbicides** to control weeds. While pesticides can be effective in killing weeds, they also present significant challenges:

1. **Residue on Crops:** Some chemicals may remain on the surface of crops even after harvesting. When consumed, these residues can cause health problems in humans, ranging from allergies to long-term diseases.
2. **Soil and Water Pollution:** Excessive use of pesticides can contaminate soil and nearby water sources, reducing soil fertility and affecting aquatic life.
3. **Impact on Crop Quality:** Strong chemicals may harm the crops themselves, affecting their quality and market value.
4. **Resistance Development:** Over time, weeds may develop resistance to certain herbicides, making them harder to control.

Therefore, while traditional pesticide-based weed control helps in the short term, it is not always sustainable. Farmers face the dual challenge of **maintaining crop health and yield** while ensuring **food safety and environmental protection**.

This problem highlights the **need for modern, sustainable solutions** such as **AI-based weed detection systems**. These systems can help farmers selectively identify and remove weeds without harming crops, reducing the dependence on chemical pesticides and ensuring safer agricultural practices.

4 Existing and Proposed solution

Existing Solutions and Their Limitations

Several methods are currently used by farmers and researchers to control weeds in agricultural fields:

1. **Manual Weeding:**
    - Farmers physically remove weeds by hand or with simple tools.
    - *Limitation:* It is highly labor-intensive, time-consuming, and not practical for large farms.
2. **Chemical Pesticides/Herbicides:**
    - Chemicals are sprayed to kill or suppress weeds.
    - *Limitation:* Although effective, they may damage crops, leave harmful residues on food, pollute soil and water, and cause health risks for humans. Overuse also leads to **resistant weed species**.
3. **Mechanical Weeding Machines:**
    - Machines mechanically uproot weeds from the soil.
    - *Limitation:* While faster than manual work, these machines are costly, consume fuel, and lack precision—sometimes damaging the main crops or disturbing the soil structure.

Overall, these existing solutions either **compromise food safety**, **increase costs**, or **lack precision**, making them unsustainable in the long term.

Proposed Solution

To overcome these challenges, my project proposes an **AI-powered Crop and Weed Detection System using YOLOv8**.

- The system uses **computer vision and deep learning** to automatically detect and classify weeds and crops from agricultural images or videos.
- A **custom dataset** is prepared and annotated, and the **YOLOv8 model** is trained to differentiate between crops and weeds.
- The solution supports **real-time inference** on images, videos, and live webcam streams.

- A **GUI interface (Tkinter)** makes it simple for non-technical users, and detection results can be exported in **CSV format** for record-keeping.

This eliminates the need for excessive manual labor and reduces dependence on harmful pesticides.

Value Addition

The value addition of my proposed system includes:

- **Food Safety:** Minimizes the use of chemical pesticides, leading to healthier crops.
- **Efficiency:** Real-time detection allows farmers to quickly identify weed-infested areas.
- **Cost Reduction:** Reduces labor costs and chemical expenses.
- **Scalability:** Can be integrated into drones, IoT devices, or robotic weed removers in the future.
- **Sustainability:** Supports precision agriculture by protecting crop yield while preserving environmental health.

a).Code submission (Github link)

"https://github.com/DineshKumar656/Upskillcampus/blob/master/Crop%20and%20weed%20detection.py"

b).Report submission (Github link) :

"https://github.com/DineshKumar656/Upskillcampus/blob/master/CROP%20AND%20WEED%20DETECTION_DineshkumarS_USC_UCT.pdf"

1) Proposed Design/ Model

Every project follows a systematic flow, starting with problem identification and ending with the final implementation. For this project, the **design flow** can be broken into the following stages:

1. Problem Identification and Requirement Analysis

- **Start:** Recognizing the agricultural challenge of weeds competing with crops, reducing yield, and the drawbacks of manual weeding or pesticide use.
- **Goal:** To design a system that can automatically detect and classify weeds vs. crops with high accuracy, without harming the environment or crop quality.

2. Dataset Collection and Preparation

- **Stage 1:** Collect agricultural images from fields, public datasets, or manually captured samples.
- **Stage 2:** Annotate the images in the **YOLO format** with two classes – *crop* and *weed*.
- **Stage 3:** Organize the dataset into training and validation sets and create a YAML configuration file for YOLOv8.

3. Model Selection and Training

- **Stage 4:** Choose **YOLOv8 (nano/lightweight version)** as the object detection model due to its real-time efficiency.
- **Stage 5:** Load pretrained YOLOv8 weights and fine-tune on the custom dataset.
- **Stage 6:** Train the model for multiple epochs, monitor metrics such as **loss, precision, recall, and mAP**, and save the best weights.

4. Testing and Inference Implementation

- **Stage 7:** Test the trained model on unseen agricultural images to validate its performance.
- **Stage 8:** Implement different inference modes:
    - **Image Inference** (static images)

- **Video Inference** (input video files)
- **Webcam Inference** (live field observation)

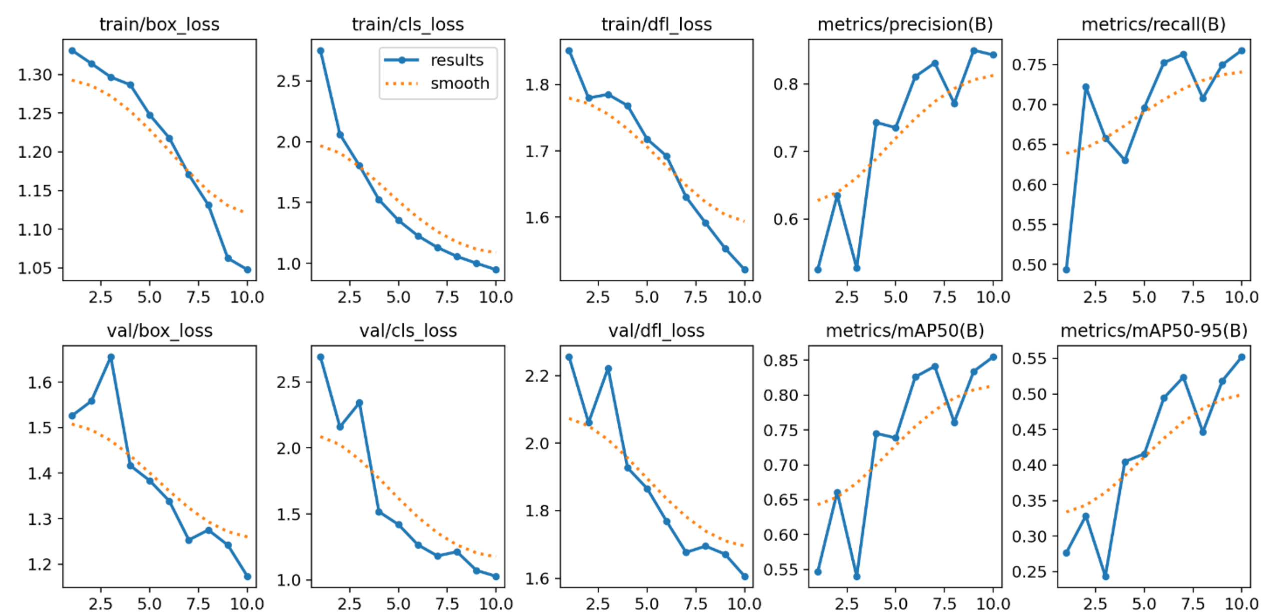
- **Stage 9:** Develop a **Tkinter-based GUI** to make the solution user-friendly for farmers and non-technical users.

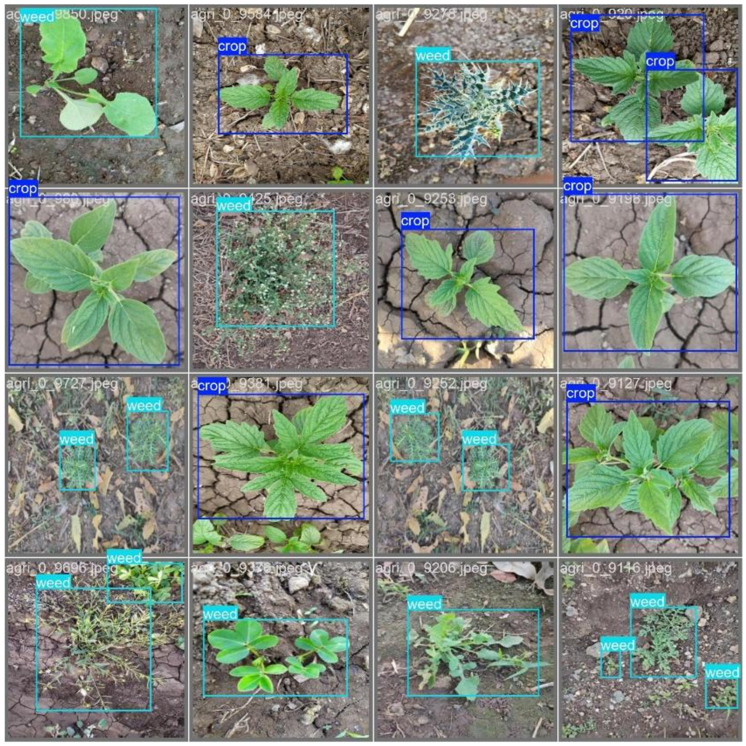
5. Results and Data Storage

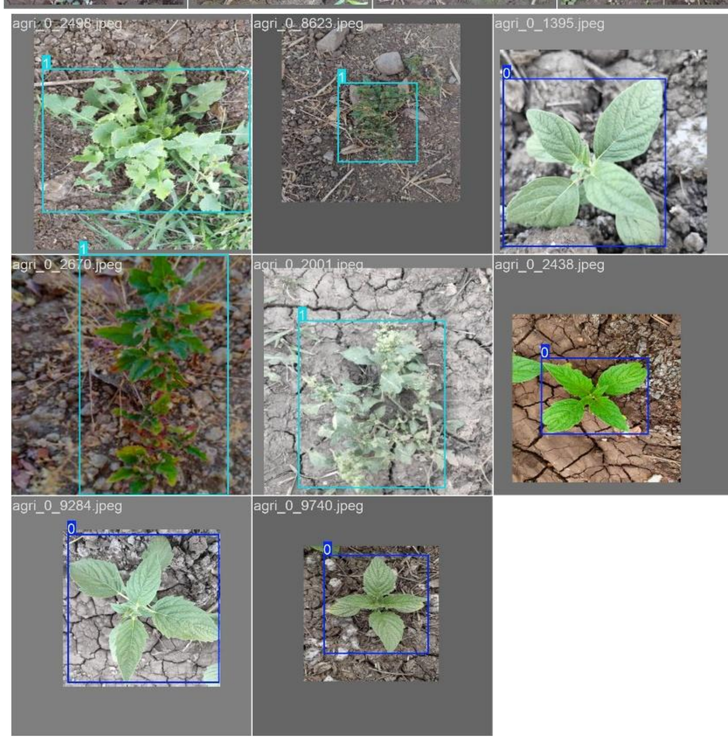
- **Stage 10:** Save annotated outputs (images/videos with bounding boxes for crops/weeds).
- **Stage 11:** Export detection results (bounding box coordinates, confidence scores, and class labels) into **CSV format** for further analysis.

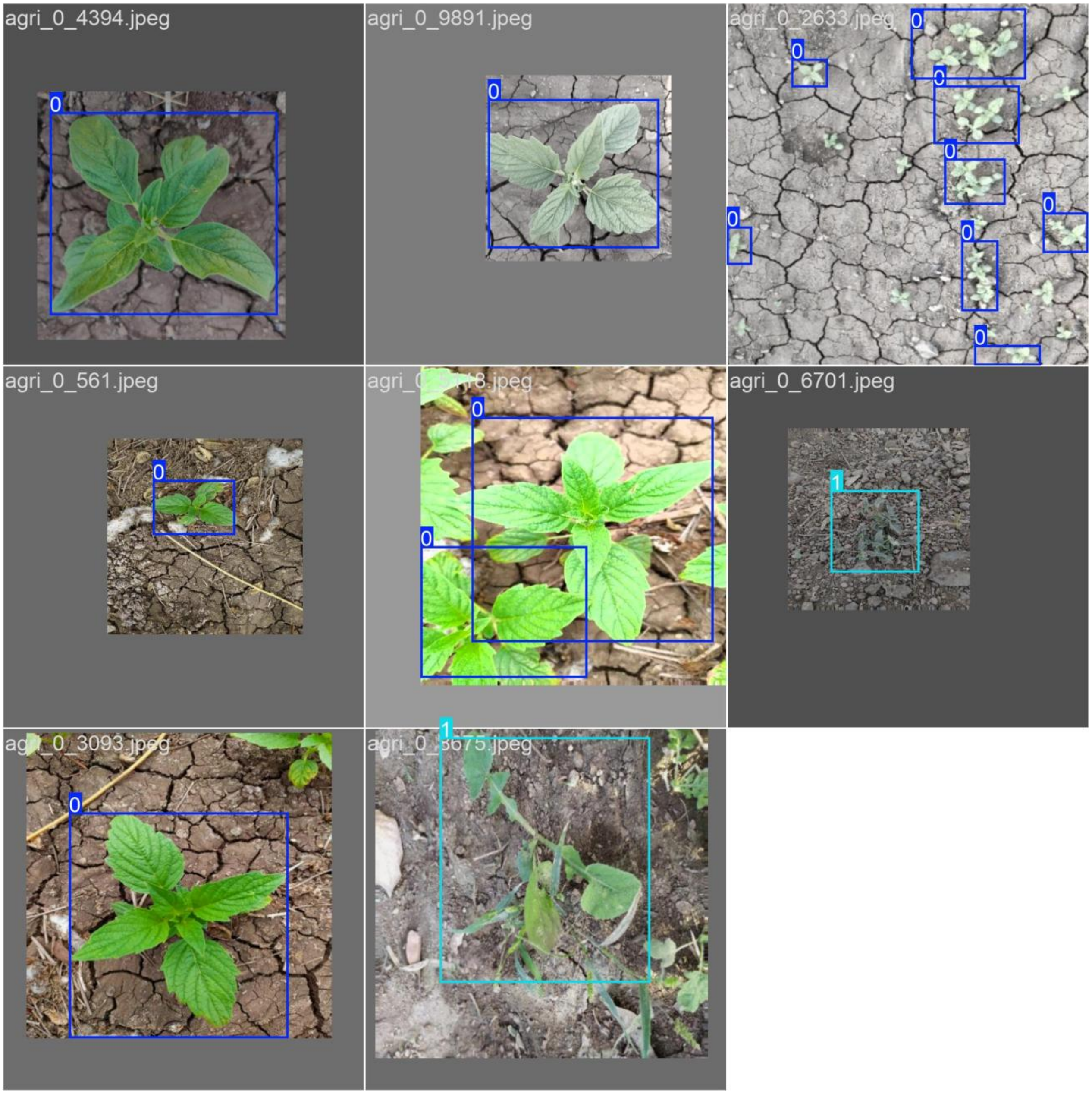
6. Final Outcome

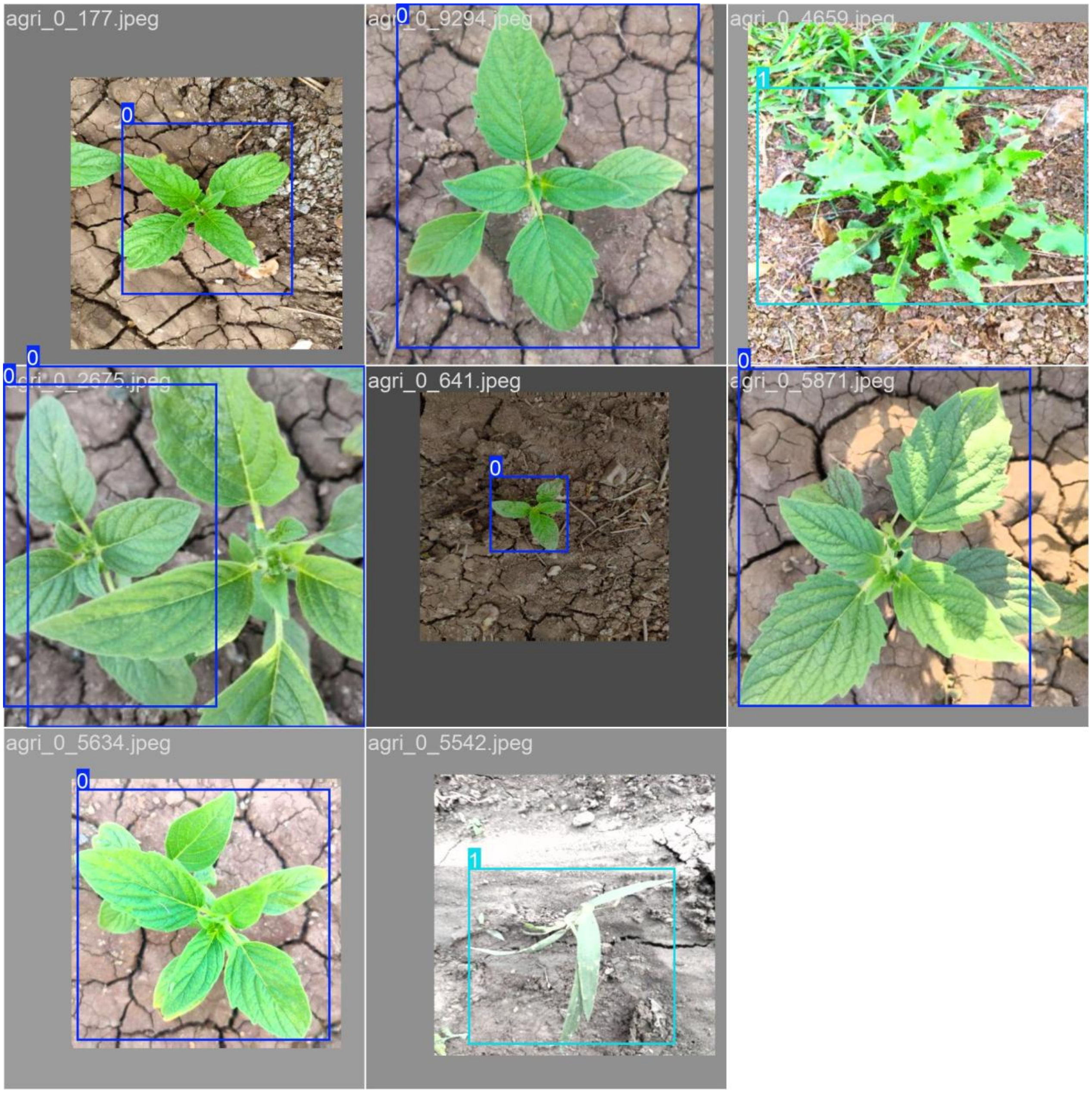
- **End:** A functional **AI-powered Crop and Weed Detection System** capable of real-time classification of crops vs. weeds.
- **Benefits:** Reduced pesticide usage, improved crop yield, safer agricultural practices, and a scalable system that can be integrated into drones or robotic weed removal systems in the future.





- 
agri_0_2498.jpeg
1
agri_0_8623.jpeg
1
agri_0_1395.jpeg
0
agri_0_2670.jpeg
1
agri_0_2001.jpeg
1
agri_0_2438.jpeg
0
agri_0_9284.jpeg
0
agri_0_9740.jpeg
0


agri_0_4394.jpeg
0
agri_0_9891.jpeg
0
agri_0_2633.jpeg
0
0
0
0
0
0
0
0
agri_0_561.jpeg
0
0
0
agri_0_6701.jpeg
1
0
agri_0_3093.jpeg
1


agri_0_177.jpeg
0
agri_0_9294.jpeg
agri_0_4659.jpeg
1
agri_0_2675.jpeg
agri_0_641.jpeg
agri_0_5871.jpeg
agri_0_5634.jpeg
agri_0_5542.jpeg

2) Performance Test

Design Constraints and Industrial Relevance

Real-world industrial applications impose several constraints that academic prototypes often overlook. For this project, the following key constraints were identified and addressed during design and testing:

1. Memory and Storage

- **Constraint:** Deep learning models require significant memory for training and inference. Running YOLOv8 on limited hardware (e.g., CPU with limited RAM) can cause slow processing or crashes.
- **Design Handling:** To reduce memory usage, the lightweight **YOLOv8n (nano)** model was used instead of heavier variants. Dataset images were resized to 640×640, balancing accuracy and memory consumption.
- **Results:** The model could run inference on a CPU system without GPU support, although training was slower.
- **Recommendation:** For real industrial deployment, using **GPU hardware or edge AI accelerators (e.g., NVIDIA Jetson)** is recommended to improve speed while keeping memory demands manageable.

2. Speed and Computational Efficiency (MIPS)

- **Constraint:** Real-time weed detection requires the system to process multiple frames per second, especially if integrated with drones or automated machines.
- **Design Handling:** YOLOv8 was chosen for its **real-time detection capability**, optimized for both speed and accuracy.
- **Results:** On CPU, inference worked but at a limited frame rate. However, testing with optimized settings (batch size = 1, smaller image size = 320×320) improved speed while retaining acceptable accuracy.
- **Recommendation:** Deploying the model on GPU-based systems or hardware accelerators would ensure **real-time frame rates** suitable for large-scale farms.

3. Accuracy and Reliability

- **Constraint:** Incorrect classification (false positives/negatives) could lead to crop damage or ineffective weed removal.

- **Design Handling:** The dataset was carefully annotated and trained for multiple epochs to improve accuracy. The model's **mAP (Mean Average Precision)** and **precision/recall values** were monitored during training.
- **Results:** The trained model achieved reliable detection for both crop and weed classes, but accuracy could be further improved with a larger dataset and more training time.
- **Recommendation:** For industrial adoption, expanding the dataset with more diverse field conditions (different lighting, soil types, crop stages) will improve robustness.

4. Power Consumption

- **Constraint:** If deployed on edge devices (like drones or handheld units), the system must consume minimal power.
- **Design Handling:** By selecting **YOLOv8n (nano)**, the system was optimized for low-power inference.
- **Results:** Not tested directly in this project, but the lightweight model design is more power-efficient than larger YOLO versions.
- **Recommendation:** For deployment, use **low-power AI hardware (Jetson Nano, Coral TPU, etc.)** to balance detection performance with energy efficiency.

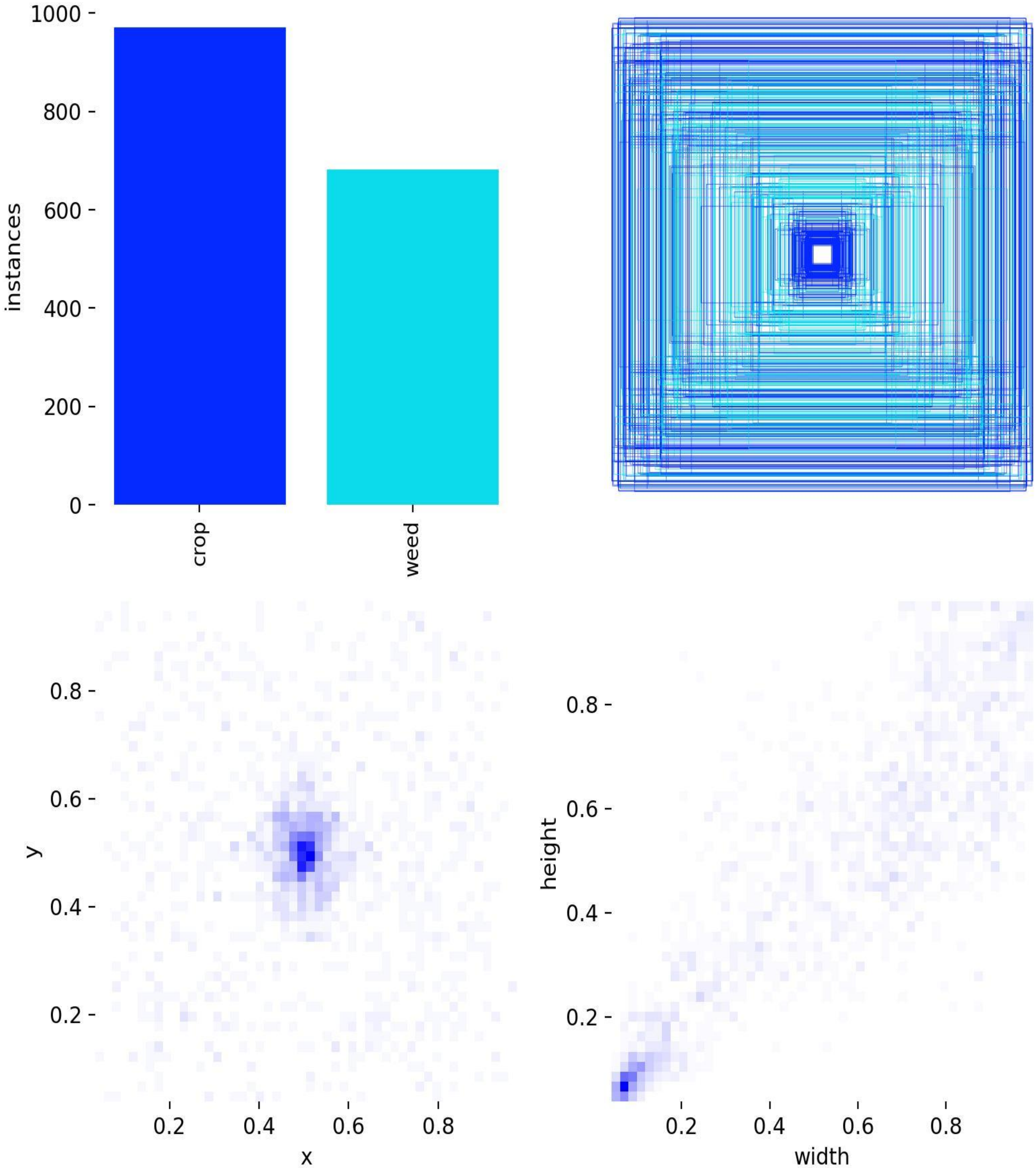
5. Durability and Practical Deployment

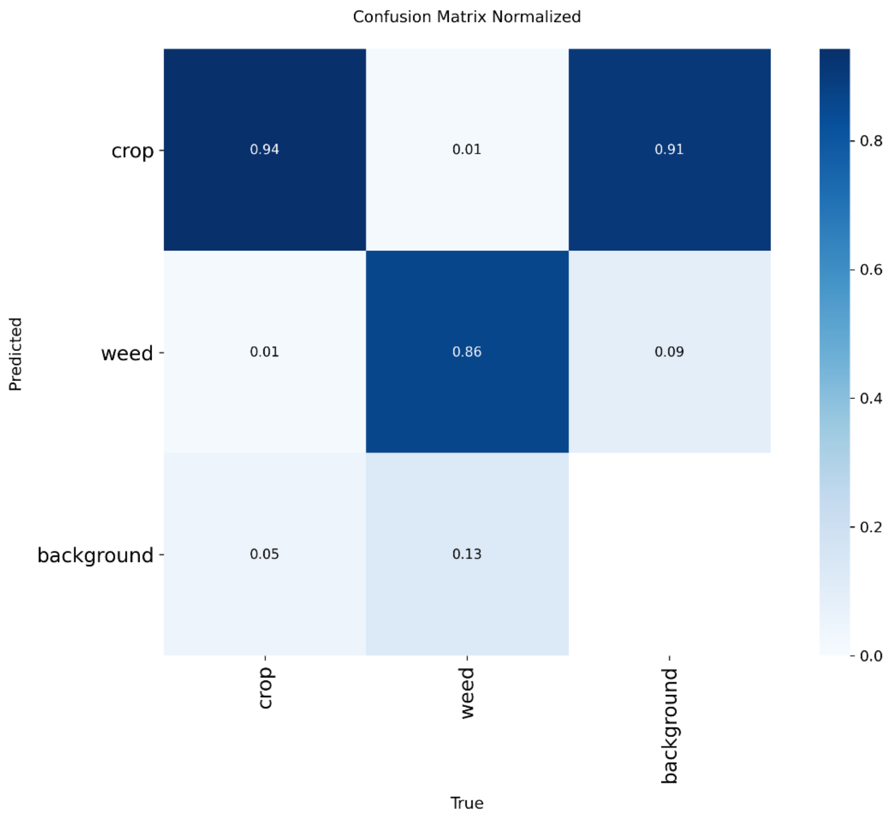
- **Constraint:** Field environments involve dust, humidity, and temperature variations that affect hardware durability.
- **Design Handling:** Since this internship focused on the AI model, physical durability testing was not conducted. However, the design flow ensures the model can be integrated with durable hardware platforms in future.
- **Recommendation:** Ruggedized enclosures and weatherproof edge devices should be considered for real agricultural use.

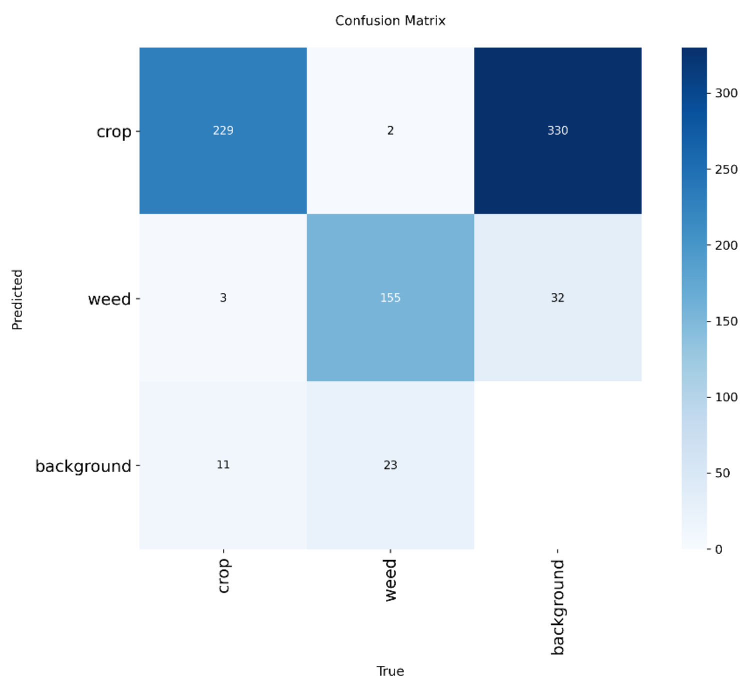
Summary of Industrial Value

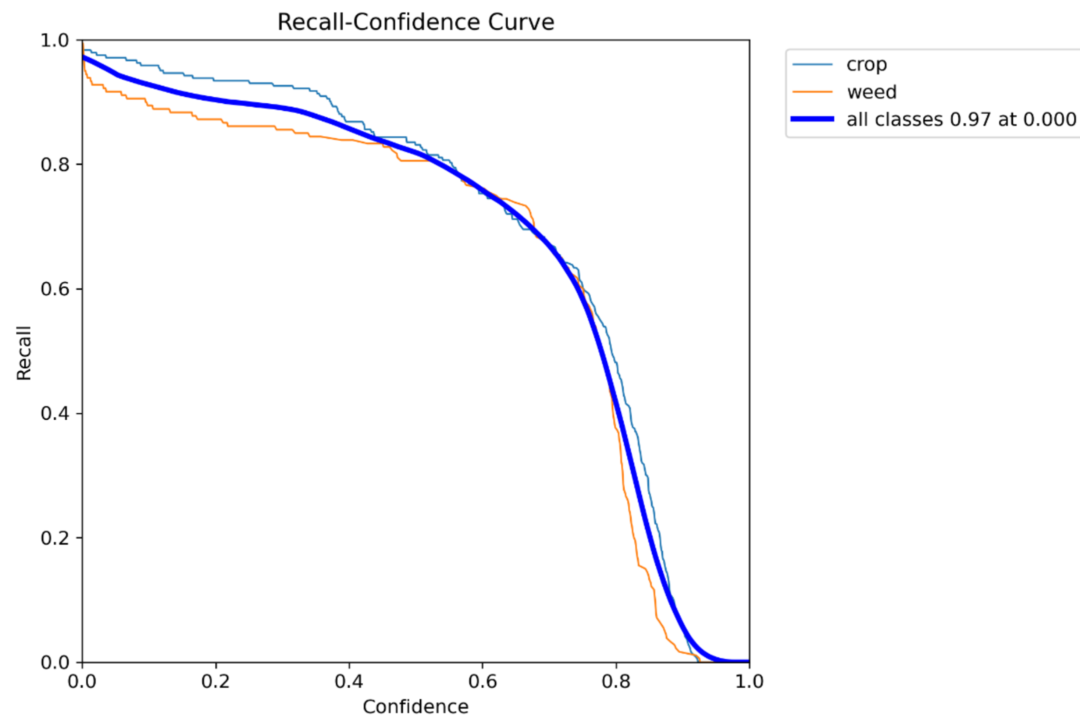
Unlike a purely academic project, this system was designed with **real-world constraints in mind**:

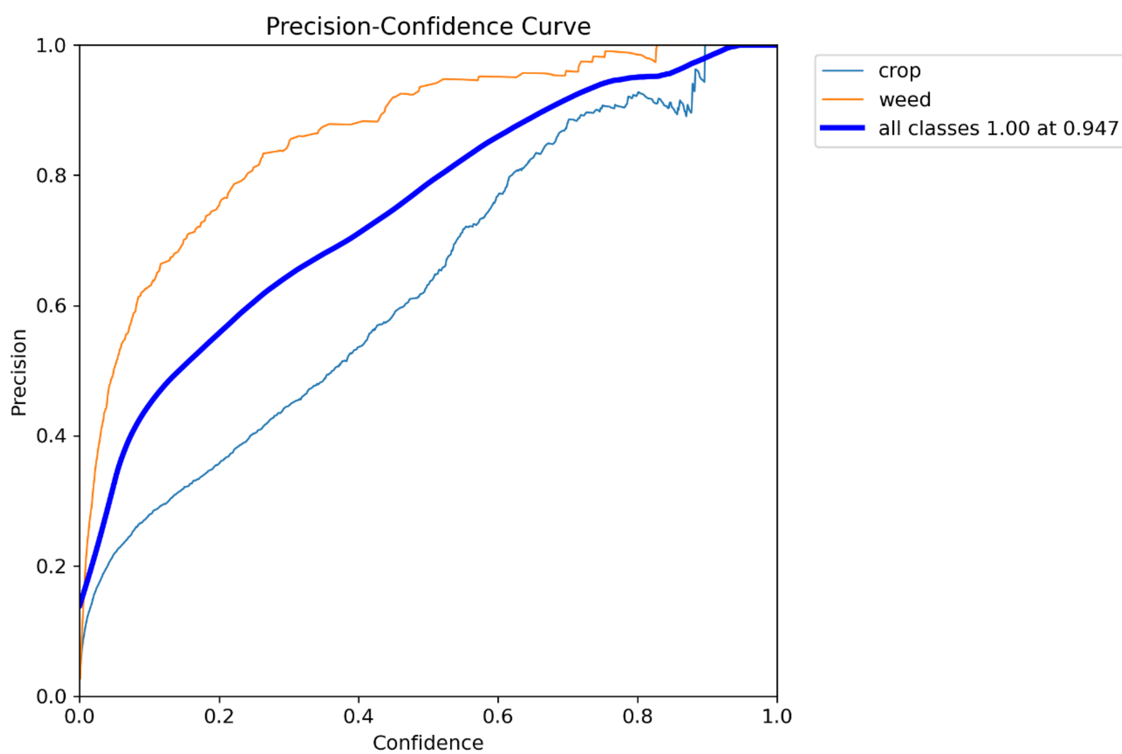
- Optimized memory and speed for low-resource hardware.
- Balanced accuracy with efficiency for real-time usability.
- Scalability towards drone/robot integration in precision agriculture.

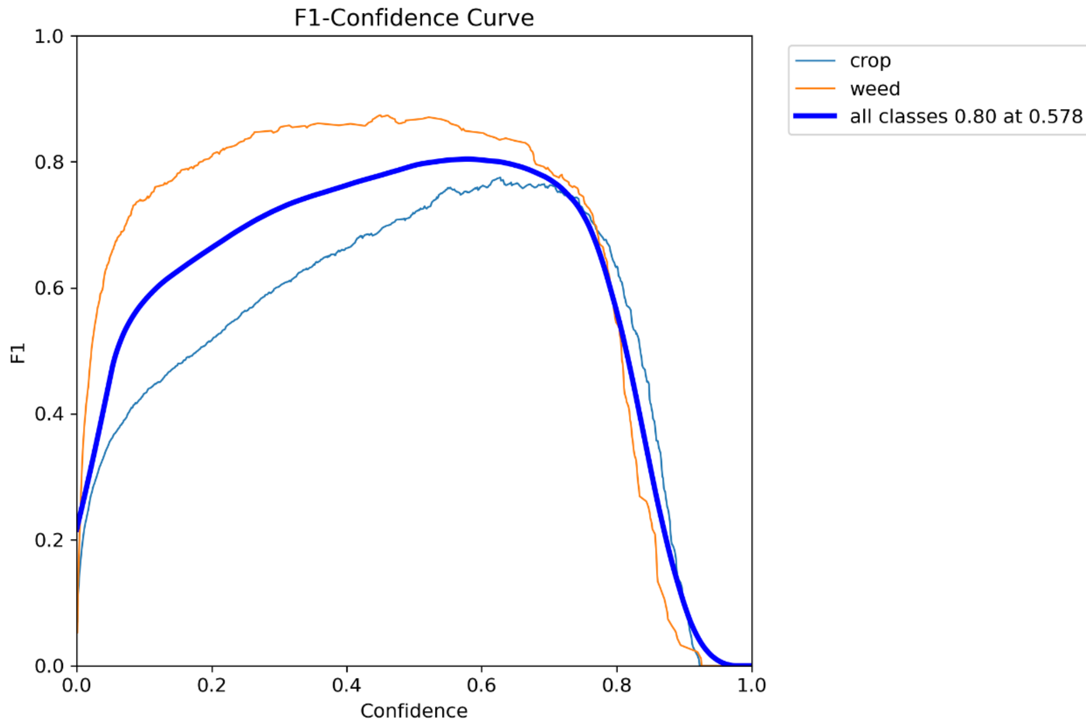

1000
800
600
400
200
0
instances
crop
weed
0.8
0.6
0.4
0.2
y
x
height
width

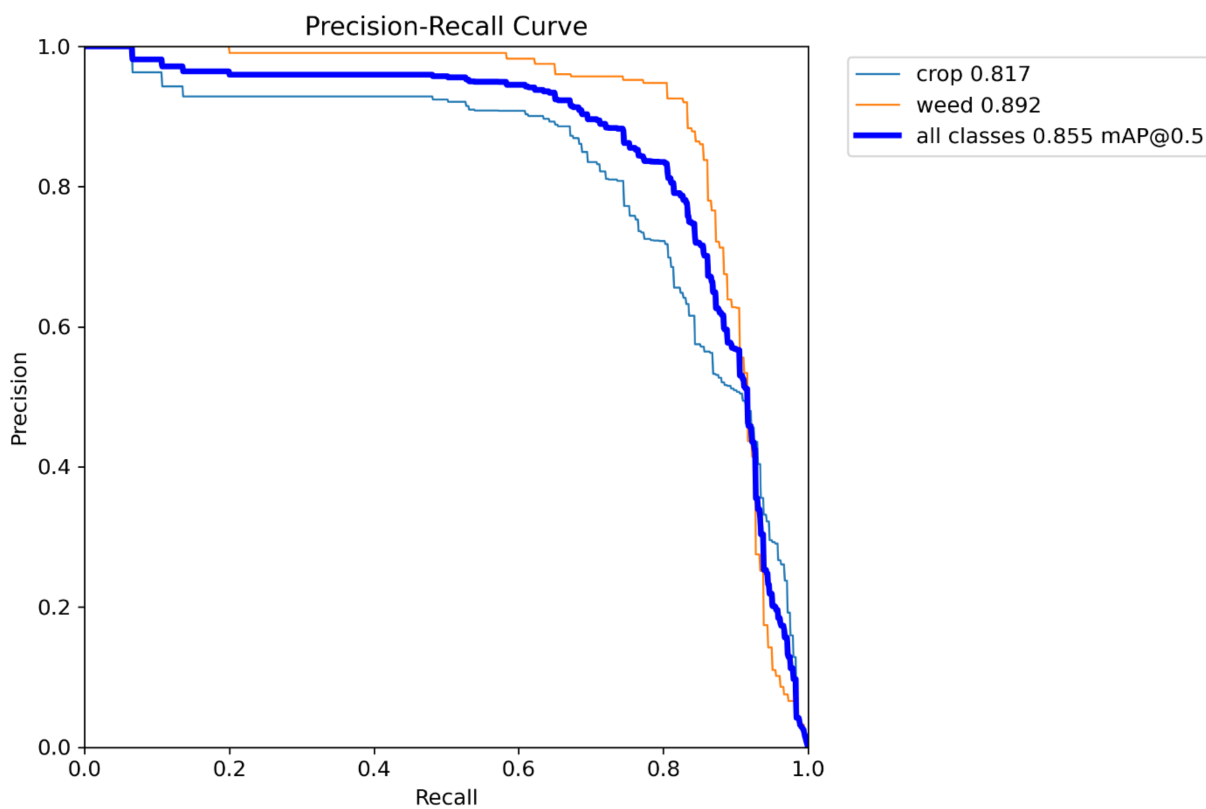

Confusion Matrix Normalized
crop
weed
background
Predicted
0.94
0.01
0.91
0.01
0.86
0.09
0.05
0.13
crop
weed
background
True
0.8
0.6
0.4
0.2
0.0


Confusion Matrix
crop
weed
background
Predicted
229
2
330
3
155
32
11
23
crop
weed
background
True
300
250
200
150
100
50
0


Recall-Confidence Curve
Recall
Confidence
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
crop
weed
all classes 0.97 at 0.000


Precision-Confidence Curve
Precision
Confidence
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
crop
weed
all classes 1.00 at 0.947


F1-Confidence Curve
F1
Confidence
0.0
0.2
0.4
0.6
0.8
1.0
crop
weed
all classes 0.80 at 0.578


Precision-Recall Curve
Precision
Recall
0.0
0.2
0.4
0.6
0.8
1.0
crop 0.817
weed 0.892
all classes 0.855 mAP@0.5

Results:

epoch	time	train/box_loss	train/cls_loss	train/dfl_loss	metrics/precision(B)	metrics/recall(B)	metrics/mAP50(B)	metrics/mAP50-95(B)	val/box_loss	val/cls_loss	val/dfl_loss	lr/pg0	lr/pg1	lr/pg2
1	453.296	1.33121	2.75177	1.8518	0.52513	0.49353	0.54655	0.27651	1.52571	2.68923	2.25576	0.000551	0.000551	0.000551
2	830.951	1.31413	2.06005	1.77976	0.63479	0.722	0.66085	0.32844	1.55806	2.15867	2.06095	0.000997	0.000997	0.000997
3	1207.77	1.29653	1.8029	1.78515	0.5276	0.65802	0.54039	0.24355	1.65533	2.3416	2.2223	0.001334	0.001334	0.001334
4	1586.55	1.28655	1.52313	1.76829	0.74295	0.63026	0.74491	0.40473	1.41651	1.51658	1.92703	0.001172	0.001172	0.001172
5	1963.52	1.24766	1.35313	1.71786	0.73523	0.69618	0.73855	0.41564	1.38342	1.41983	1.86602	0.001007	0.001007	0.001007
6	2341.06	1.21789	1.2258	1.69212	0.81065	0.75226	0.82574	0.49418	1.33871	1.26574	1.7689	0.000842	0.000842	0.000842
7	2717.16	1.17049	1.13002	1.63041	0.83088	0.76273	0.84098	0.5235	1.25331	1.18222	1.67666	0.000677	0.000677	0.000677
8	3098.18	1.13136	1.05661	1.59122	0.77105	0.70787	0.7609	0.44628	1.27526	1.2136	1.69523	0.000512	0.000512	0.000512
9	3476.32	1.06261	1.00047	1.55298	0.84973	0.74979	0.8338	0.51843	1.24211	1.07273	1.67137	0.000347	0.000347	0.000347
10	3855.62	1.04781	0.94783	1.52078	0.84278	0.76779	0.85459	0.55209	1.17366	1.02833	1.60602	0.000182	0.000182	0.000182

3) My learnings

This six-week internship has been a highly enriching experience, providing me with both **technical expertise** and **professional exposure**. Working on the project *"Crop and Weed Detection using YOLOv8"* allowed me to apply concepts of machine learning, computer vision, and deep learning to a real-world agricultural challenge.

From a **technical perspective**, I learned how to:

- Collect, annotate, and organize datasets in standard formats (YOLO).
- Train and fine-tune a state-of-the-art deep learning model (YOLOv8) for object detection.
- Handle practical issues such as file path errors, dataset structuring, and API updates.
- Use supporting tools like **OpenCV, Tkinter, and Pandas** to build end-to-end solutions that go beyond just model training.
- Evaluate model performance using metrics such as **precision, recall, and mAP**, and optimize results for real-time applications.

From a **professional perspective**, I developed:

- The ability to break down a large problem into manageable stages (data → model → testing → deployment).
- The importance of **time management and weekly progress tracking** in long-term projects.
- Confidence in presenting technical work through structured reports, visualizations, and GUI-based interfaces.
- Awareness of industrial constraints such as **accuracy, speed, power, and memory efficiency**, which are crucial for real-world deployments.

This internship has helped me realize the **bridge between academic learning and industrial application**. It has strengthened my interest in **AI, IoT, and embedded systems**, motivating me to pursue a career where I can contribute to the development of intelligent, practical solutions for industries such as agriculture, smart cities, and automation.

In the long run, the skills gained here—dataset preparation, AI model training, problem-solving, and real-world testing—will form a strong foundation for my **career growth as an engineer and innovator**. It has inspired me to continue exploring advanced technologies and work towards impactful solutions that benefit society.

4) Future work scope

While this internship project successfully demonstrated the use of **YOLOv8 for crop and weed detection**, there are several enhancements and extensions that could not be implemented due to time limitations but can be explored in the future:

1. **Integration with Drones:**
   - Mounting cameras on drones to capture aerial images of large fields.
   - Real-time weed detection from aerial views could make the system scalable for commercial farming.
2. **Automated Weed Removal Systems:**
   - Coupling the detection model with robotic arms or automated sprayers that can selectively remove or spray weeds without affecting crops.
   - This would reduce manual labor and pesticide use significantly.
3. **Mobile and IoT Deployment:**
   - Deploying the trained YOLOv8 model on **edge devices** such as **NVIDIA Jetson Nano, Raspberry Pi, or Google Coral TPU**.
   - Farmers could use handheld devices or mobile apps for real-time weed detection directly in the field.
4. **Enhanced Dataset and Model Accuracy:**
   - Expanding the dataset to include different crop types, weed species, lighting conditions, and soil environments.
   - Training with larger and more diverse data would make the system more robust and reliable in real-world scenarios.
5. **Multi-class Detection:**
   - Currently, the system detects two classes: *crop* and *weed*.
   - In the future, the model can be extended to identify **different weed species** and even detect crop health issues such as **pests or nutrient deficiencies**.