Weed Identification and Classification in wheat crop

Submitted in the partial fulfillment of the requirements

for the degree of B.Tech in Computer Science and Business Systems

by

Pranav Vijay Parle Patil (22CB1027)

Hari Vaghela (22CB1014)

Aditya Pillai (22CB1053)

Rithvik Edupuganti (23CB5004)

Supervisor

Dr. Shamal Salunkhe

Department of Computer Engineering

Ramrao Adik Institute of Technology

Sector 7, Nerul, Navi Mumbai

(Under the ambit of D. Y. Patil Deemed to be University)

November 2024

Ramrao Adik Institute of Technology

(Under the ambit of D. Y. Patil Deemed to be University)

Dr. D. Y. Patil Vidyanagar, Sector 7, Nerul, Navi Mumbai 400 706

CERTIFICATE

This is to certify that, the Mini Project-III report entitled

Weed Identification and Classification in wheat crop

is a bonafide work done by

Pranav Vijay Parle Patil (22CB1027)

Hari Vaghela (22CB1014)

Aditya Pillai (22CB1053)

Rithvik Edupuganti (23CB5004)

and is submitted in the partial fulfillment of the requirement for the degree of

B.Tech in Computer Engineering

to the

D. Y. Patil Deemed to be University

Supervisor	Project Co-ordinator
(Dr. Shamal Salunkhe)	**(Dr. Shamal Salunkhe)**
Head of Department	Principal
(Dr. Amarsinh V. Vidhate)	**(Dr. Mukesh D. Patil)**

Mini Project Report - III Approval

This is to certify that the Mini Project - III entitled ***" Weed Identification and Classification in wheat crop using Deep Learning "*** is a bonafide work done by ***Pranav Vijay Parle Patil (22CB1027)***, ***Hari Vaghela (22CB1014)***, ***Aditya Pillai (22CB1053)***, and ***Rithvik Edupuganti (23CB5004)*** under the supervision of ***Dr. Shamal Salunkhe***. This Mini Project is approved in the partial fulfillment of the requirement for the degree of ***B.tech in Computer Science and Business Systems***

Internal Examiner :

1. ....

2. ....

External Examiners :

1. ....

2. ....

Date : .../.../......

Place : ....

DECLARATION

I declare that this written submission represents my ideas and does not involve plagiarism. I have adequately cited and referenced the original sources wherever others' ideas or words have been included. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action against me by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Date: ____________________

Pranav Vijay Parle Patil (22CB1027)

Hari Vaghela (22CB1014)

Aditya Pillai (22CB1053)

Rithvik Edupuganti (23CB5004)

Abstract

The productivity of agriculture significantly depends on the crop yeild which in turn depends on the effect of weeds on crop health. Our project aims at the efficient identification and classification of weeds from crops. The object detection model we have used is YOLO v5 which due to its efficient and fast execution proves to be an excellent model to use to detect weeds. The model was provided with an image dataset with already classified weeds. The model then trains on it and provides outputs based on user inputs. The model also has incredible robustness as the images are already flipped, rotated, cropped and scaled to help the model .

The Flask framework is used to facilitate the integration of frontend and backend model. The user inputs an image which gets processed by the backend model which places a bounding box, labels and confidence scores on the detected weed. The entire system also includes error handling and CORS for seamless frontend and backend integration. Our project aims to help farmers with scalable software to not only increase crop yield but also maintain crop health. Our project promotes sustainable farming techniques not just for the farmer, but also benefitting the consumers eventually.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

One of the biggest problems in modern agriculture and a problem that has lasted over time, mainly when it comes to growing crops and harvesting them is weed control. Crop production and agricultural output is greatly lowered when plants have to compete with weeds for resources like nutrients from the soil, water and sunlight. Hand labour and pesticides have been used largely in traditional weed management system techniques but these are not necessarily the most effective techniques and often tend to be detrimental to human health

The development of modern technology has allowed for the precise identification and categorisation of weed in fields through technologies like machine learning, deep learning, computer vision and image processing. The advancements in robotics and UAVs(unmanned aerial vehicles) have also fully automated the task of not just identifying these weeds but also eliminating them itself.

1.2 Motivation

The initial motivation for creating an AI weed detection system stems from the increasing need for more sustainable and efficient practices in the field of agriculture. Weeds are a huge threat to crop life, which have an everlasting effects on crop health, productivity and yields. Weeds are not only a direct threat to the crops, but also affect the space management in farms by occupying viable lands where crops could grow. They also soak up water and nutrient from the soil. The traditional weed control methods used worldwide for example manual labour, even though effective, are often costly, time consuming, and mainly a threat to the environment. Not just that, identifying weeds in the first place is also a task in itself for novice as well as experienced farmers. With rising ecological implications and wastage of human hours, there is a need for better weed management system.

The weed detection software combines the power of machine learning and computer vision to help farmers to practice efficient weed detection and control, with minimal effects on the environment. The key driving force behind this software is to revolutionize the agricultural industry for the betterment of not just the farmers but the eventual end customers who want to buy healthy crops. This technology could be a stepping stone for other agricultural related technology in the present or future.

1.3 Problem Statement and Objectives

1.3.1 Problem Statement

The presence of weeds within a farm reduces the overall agriculture output causing plants to compete for resources. For example, Paterson's curse in pastures. The weed we find in plants is also very harmful for live stock and can cause poisoning to both plants and livestock. Within a lot of fields this weed also acts as hosts or shelter for rabbits and other pest animals. The weed is also allergy causing to the humans around it who would work with the plants on the field. Another major factor is soil erosion that occurs when the weed dies. Keeping this problems in mind it is very important that with the help of modern technology we build a system that allows us to automate the detection of these weeds so that their effect on plants, live stock and humans becomes as minimal as possible. It is crucial that removing these weeds becomes a far easier process than it already is so reducing its effects on crop produce is reduced as much as possible. Thus a system that uses computer vision and image processing for weed detection and classification would be a great help for all involved in the agricultural sector.

1.3.2 Objectives

- Weed detection: The main objective of the system is to use the YOLO framework and enable the processing of weeds
- Classification: The next goal is to classify the weeds and crops so that we know what type of weeds grow on the field and the treatment needed for those weeds.
- Sustainability: Reducing depending on just pesticides that are also harmful for consumption is another goal of automating the weed detection process
- Scalability: Ensuring the model can adjust to various image types, fields and detect weeds from across a variety of species to achieve a large scale impact

1.4 Organization of the report

The chapterization provides a structural breakdown of how each chapter is designed to systematically present the key aspects of the project, starting from the problem identification, literature survey, the implementation details and the analysis of the results.

- Chapter 1 – This chapter introduces the topic to the reader. It not only provides relevant knowledge, but also presents the motivation behind the project. The problem statement is also shared to know the need for a solution.
- Chapter 2- This chapter deals with previous technologies that have worked on the same problem, or existing software that deal in problems of a similar kind. Technologies which provided inspirations are also mentioned. Furthermore, this chapter also discusses the limitations and gaps of any previous or current technology working on the same problem.
- Chapter 3- The problem statement is shared with the reader to give context for subsequent solutions. The proposed unique solution to the above problems is shared, which includes the main techniques used in the system to overcome the problem. Following this we have the overall design of the system explained in detail. Concluding the chapter we have

the in-detail information on the hardware and software requirements needed to run this particular software.

- Chapter 4 – This chapter focuses on the implementation details on the software concerned with final execution of the software on a system and gaining valuable results from it. The gained results are analysed to provide information on the final success rate as well as working of the software.

- Chapter 5 – The final conclusion is draw from the project, focusing on the limitation and drawbacks faced during the development of the software. Future possibilities and additions to the software are explored and discussed.

Chapter 2

Literature Survey

Traditionally, farmers were dependant on manual labour for weed control in farming. But with the advent of technology, there are many solutions proposed for weed detection, using advanced techniques ranging from hyperspectral imaging to machine-learning based approaches. Analysing these methods provide valuable insights into highlighting the limitations that need to be addressed. This review also provides a justification for the development of the proposed system which aims to overcome the current limitations.

It was found that although there are several weed detection systems in existence, they each have their own limitation that prevents their widespread adoption. High costs and scalability issues are significant barriers. Our proposed solution using YOLO deep learning model proves to be a feasible alternative. Let us examine this in detail.

2.1 Survey of Existing System

1. **Traditional methods**: Traditional methods of weed control consists of manual labour, cultural practices and herbicide application. Manual labour is the process of detecting and removing unwanted plants by hand in a crop field and preventing damage to cultivating crops. This process is extremely time consuming and taxing but also is an essential part in agriculture. Despite it being time consuming it also is really effective and provides precision in removal of weeds and protecting crops. Traditional practices of weed removal include tilling, hoeing, hand weeding, digging, burning, sickling and mowing etc. which help in disruption of growth of weeds and also prevent seed germination. Manual method is also cost effective compared to the newer methods which consists of using machines

and other latest technology for weed detection. Although initially it is cost effective it can become expensive over time due to repeated wages, when compared to a one-time investment in automated tools. Moreover, there can be seasonal labour shortages, where fewer workers are available during certain seasons to take on strenuous roles during peak times due to which challenges can arise in cultivation of crops.[1]

2. **Machine Vision Based System**: Uses image processing, machine learning and pattern recognition techniques to identify and detect any weeds in an agricultural field. These systems usually involve placing a high resolution camera on tractors, drones or any hand-held devices to distinguish between weeds and crops. The distinguishing is done based on various characteristics of a plant such as, the colour, shape and its texture and also on its growth patterns. The cameras or thermal sensors capture high resolution images which is then analysed and distinguished based on the above mentioned factors, helping differentiate between crops and weeds with precision. This also helps with reducing herbicide usage as these systems detect and spray on individual weeds instead of blanket application all over the field which in turn helps reduce costs and increases efficiency. One example for this is the WeedSeeker 2 automatic spot system.[2] When a weed passes the sensor, it signals the linked spray nozzle to spray herbicide on it. Other than the high initial costs for these systems, these provide high precision and accuracy, increased efficiency, reduced wastage and scalability.[3]

3. **Deep Learning Models**: In recent times, the ability of deep learning models to recognize patterns and gain valuable information from huge datasets, and using this ability to detect weed and other crop abnormalities, have become hugely popular. These powerful models train on vast image datasets, and then are used to differentiate crops from weed, resulting in highly accurate solutions. Not just that but these models have accuracy that increases over time giving them vast use cases. Some of the most recent examples being ”See and Spray” from blue river technology[4] which target weeds in real time by using its deep learning model. These models make sure that crops are not harmed by herbicide and remain healthy and safe.[5]

4. **Thermal imaging**: Thermal imaging, also known as thermography is a widely used technique in weed removal by farmers. It is an advanced technique which is non-destructive and uses no contact and helps in effectively removing weeds by using heat emitted from

objects to monitor crops. There are temperature differences between various types of plants due to various factors such as water content, leaf shape, and metabolic rates allowing weeds to be distinguishable from crops and thermal imaging leverages that to distinguish between weeds and crops in a farm. This method is especially useful in early detection of weeds, which helps farmers to target and work on areas with more weed density. By pinpointing areas with higher weed density farmers can now reduce herbicide usage and labour costs by enabling, more precise and targeted interventions. Some of the tools used in thermal imaging are drones for Ariel Thermal Imaging, thermal scanners for Ground Based Thermal Imaging and also Mapping and Analysis. Although thermal imaging helps in low visibility conditions, its effectiveness can be influenced by certain environmental factors which could limit its accuracy.[6]

5. **Robotics and Autonomous Weeders**: Autonomous robots that sort out weeds from crops which can be done with mechanical or chemical ways is another popularly used system in modern agriculture. These robots make use of computer vision and AI to segregate useful crop from weed. This type of sorting helps the robot avoiding to spray entire fields of crops with herbicide and only target specific weeds. The advantages of having these robots is that they reduce manual and repeated labour, and also promote less chemical usage in crops. An example of such a robot is FarmWise Titan robot, which uses the above said methods to segregate crops from weeds.[7]

2.2 Limitations of Existing System or Research Gap

1. Manual Labour: Manual labour as much as it is prevalent, also comes with its downsides. Manual labour is often time consuming, expensive, and requires a lot of unnecessary hard work which could otherwise have been automated. The workers doing this job also need to have skills and experience which makes it even more demanding. The precision involved in manual labour is also diminished to an extent since human make mistakes often.

2. Machine Vision-Based Systems: These systems might be really useful at identifying weeds from crops based on their physical features like size and colour, but problems start coming up in situations where the weeds look very much alike the crops. Situations

like low light conditions or uneven field can also set back this type of systems. These systems also need technical support from time to time which might not be feasible for budget constrained farms.

3. Thermal imaging: Thermal imaging makes use of the difference in temperature between the weeds and crops but this turns out to be a problem in climatic conditions where the temperature changes quickly, as this could affect the reading and in turns affect the results. Climatic regions where the weather is usually hot and humid might prove difficult for the system to work in.

4. Robotics and autonomous weeders: These robots are known for bringing in high precision and accuracy, but unfortunately come at a high cost. Farmers who want to implement this system in their farms would need to have a huge capital investment in the first place. Mostly these machines are used in well-structured farms, so any amount of terrain or crop diversity may affect the system negatively. Additionally to this, these systems need a lot of supporting architecture, which may cost even more to maintain.

Chapter 3

Proposed System

3.1 Problem Statement

In this era of technological development in agriculture, and in particular in wheat production, weeds are the main problem, as those weeds use the nutrition intended for the crops, cause yield reduction, and can also result in crops of low quality. Traditional weed management methods such as manual weeding or herbicides are more and more recognized as non-sustainable due to their high labor cost, the environmental effect, and the possibility of weed resistance development. Thus, a system that can correctly identify and classify weeds automatically is crucial for farmers to produce better and more food with less impact on the environment.

The proposed solution, which exploits deep learning and the YOLO (You Only Look Once) algorithm, allows real-time exact weed recognition to take place together with wheat in the fields. YOLO's outstanding accuracy in object detection besides its ability to make time-sensitive decisions on edge devices prove to be among the factors that make it the preferred choice for this type of application [Reference to YOLO applications]. The framework intends to eliminate dependence on chemicals, making weed management more eco-friendly, enduring, and cost-saving. It aims to do this through clever image processing techniques and a superbly designed deep-learning model.

3.1.1 Salient Features of the Proposed System:

- **Automated Detection and Classification:** The system implements the YOLO algorithm which facilitates the rapid and correct detection of weeds, thus differentiating the weeds from the wheat and appropriately managing them in the right pattern.

- **Sustainability Focus:** Targeting weeds only, the system enables to cut down the rate of applying pesticides over a wide area, thus chemical use is reduced and environmentally friendly farming practices are adopted.
- **Adaptability and Scalability:** This is a proof-of-concept of a system that has been designed to deal with different physical conditions in a field and it works with different light intensities, soil backgrounds, and weed types which means the system can be used in various agro-ecological settings and it is easily scalable to farms of different sizes.
- **Resource Optimization:** By introducing this machine in the fields, farmers do not have to hire workers and buy more materials for weed control, which means they can instead use the spared resources to increase crop yield.
- **Real-Time Processing:** By virtue of YOLO's real-time processing capability, the system provides farmers with instant data about their work, thereby, facilitating the farmers' timely response to the situation which is also a better way to manage weeds.

3.2 Proposed Methodology/Techniques

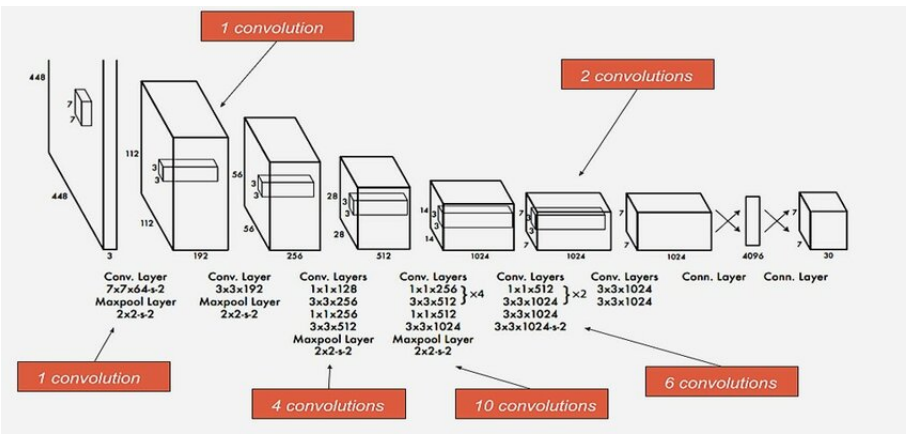

Figure 3.1: Model Architecture of YOLO framework

The YOLO framework is an object recognition framework that analyzes images with a single forward pass. It creates a grid around the cell making various parameter predictions like confidence ratings. By using object detection as a single regression problem YOLO significantly

speeds up the overall detection of the objects. The frameworks architecture allows us to detect a large number of objects at very high speeds, and small runtimes making it one of the best Machine learning frameworks for object detection and autonomous systems
Yolo works by dividing the image into a grid and each grid works by detecting objects whose centres lie in each cell. Yolo also predicts coordinates with reference to the image, confidence scores, and class probabilities. The algorithm then uses non maximal suppression also known as NMS to remove any overlapping boxes, retaining it's most confident decisions
YOLO has many models including the one that has been integrated into this project YOLOv5 i.e. version 5.

3.2.1 Key components of YOLOv5

- The main body of the framework has been designed using the new CSP-Darknet 53 structure a heavy modification from the previous iteration of this model
- Connecting the head to the body it uses a SPFF New and CSP-PAN structure, and it also known as the neck of the project
- The head in the end is that final part which is responsible for providing the end output. YOLOv5 uses the head of its previous iteration the YOLOv3 model

3.2.2 Implementation of YOLOv5

- The model has been trained using two classes derived from the used dataset namely "crop" and "weed". The customization used by YOLOv5 is done using the YAML configuration which allows it to focus only on the classes that are needed to execute the project and improve performance
- The model also sets the epochs (number of times the batch has been processed) to 100 and the batch size (number of samples processed) to 16 which allows for smoother gradient curves updates and more epochs ensure model convergence
- The uses transfer learning which is a ML task that uses knowledge gained from one task to help improve the related task. Transfer learning by YOLOv5 is done by using a pre-trained model and adjusting it to the custom dataset.

3.2.3 Data annotation and Preprocessing

- Images in the output are annotated with bounding boxes around the objects “weed” and “crop” with its coordinates, confidence score(how certain the model is of the objects true nature) and a label indicating what is a crop and weed.

- Flipping and Rotation to help the model generalize different orientation. Scaling and cropping which makes the model accustomed to size variations in crops and weeds

- Colour Jittering which compensates for different images that have been taken under different lighting conditions

3.2.4 Deployment Architecture: Flask API for backend Inference

- The Flask API framework includes an endpoint that is used to handle POST requests. The users input the image via the frontend, which is then processed by the YOLOv5 model and returns coordinates, confidence scores and detection labels

- Once the model provides its predictions and output the user input the results are saved as an image and JSON file, which the frontend uses to display what has been detected.

- The Flask endpoint has also been customized and improved for better error handling, file management and most importantly CORS(cross-origin resource sharing) which allows for seamless connection between frontend and backend

3.2.5 Frontend Interface for user interaction

- The front end has been designed using the trinity of HTML, Javascript and CSS which allows the user to select an image, which is previewed and uploaded. The image is then analyzed and the front end retrieves the results from the server, displaying it for the user

- The Front end offers real time analysis, error messages for wrong inputs and a responsive and interactive User interface for ease of use

3.3 System Design

The system design for this project has been carefully structured to ensure efficient and user-friendly detection of weed in images of crops. A modular approach is adapted where in the system is divided into multiple interacting modules, allowing for flexibility and scalability.

3.3.1 Architectural Overview

The architecture of the system can be broadly categorised into three main layers: the Frontend Interface, the Backend Processing and the Machine Learning Model. These layers are integrated together to function for a seamless workflow

- **Frontend Interface**: This layer is the user facing side which allows the users to upload images of crops and view detection results. The users interact via a responsive and user-friendly interface. It is designed using HTML, CSS and JavaScript to enable interactivity, image preview and effective communication with the backend.

Figure 3.2: User Interface

- **Backend Processing**: This layer is implemented using Flask, a Python based web framework which acts as a bridge between the frontend and the machine learning model. It handles the requests and flow of data in the system. When an image is uploaded, the backend prepares it for model assessment by performing preprocessing and sends to the YOLOv5 model. It also handles the model's output handling for displaying the result on the frontend.

- **Machine Learning Model**: YOLOv5 is the core of the system, it receives the preprocessed images, performs tasks to identify and classify the weed and returns bounding boxes with labels and confidence scores.

3.3.2 Data Flow and Module Interactions

- **Image Uploading and Preprocessing**: The UI allows the user to upload an image of a crop field. Once the image is submitted, it is sent to the backend where it undergoes preprocessing such as resizing and format conversion (if needed) ensuring to meet input requirements of the YOLOv5 model.

- **Model Interface**: After preprocessing, the image is fed to YOLOv5 model for assessment. The model performs feature extraction, object detection and classification using. It predicts the bounding boxes around detected objects and assigns labels to each box, either "weed" or "crop" based on its training.

- **Data Structuring**: The raw output from the model consists of coordinates of bounding boxes, confidence scores and class labels, the data is processed into a more structured format such as JSON to make it easier to interpret and transfer to the frontend.

- **Displaying Results**: The results are displayed on the UI. The frontend uses JavaScript to render bounding boxes on the original image along with confidence scores.

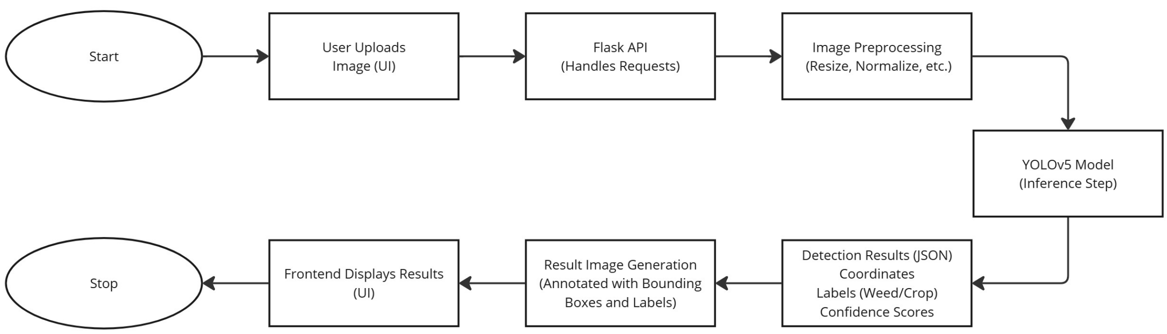

Figure 3.3: Flow Chart for the Weed Detection System

3.3.3 Error Handling

- **Input Validation**: The backend ensures that the uploaded file is a JPEG or PNG. If the input is invalid, the system displays an error message describing the incompatibility of the file.
- **Model Inference Errors**: If errors occur during model inference such as unexpected input dimensions, the backend catches exception and returns an error message which prevents the system from crashing.

3.4 Details of Hardware/Software Requirement

3.4.1 Hardware Specifications

- Processor: Intel Core i5 to i7 or higher, or other equivalents like Ryzen to help work with huge datasets
- RAM: Minimum RAM recommended is 8GB for handling datasets, but for much bigger and complex datasets, a 16GB RAM is recommended.
- GPU: Since the project deals with a lot of images and deep learning models, a good enough GPU is required to help speed up the modelling process
- Internet connection: If working on cloud databases or servers, the need of a fast internet connection is necessary to download and upload data quickly.

3.4.2 Software Specifications

- Python, HTML, CSS, JavaScript: The core language used in this project is Python. The versatility of the language along with its extensive libraries and resources along with excellent machine learning support makes it the perfect choice. Python is used in this project to make the backend as well as the machine learning component. Whereas the frontend, designing and connection with the backend is done with HTML, CSS and JavaScript respectively.
- YOLOv5 : It is the object detector used in the project. In comparison to other models, it provides faster execution and good accuracy.

- Flask: To have a working frontend connected to a backend model, the framework we have used is Flask. Its simplicity and scalability makes the entire software easier to work on
- Torch: The machine learning library with useful tools for deep learning, and user friendly research framework
- OpenCV: The computer vision and image processing library used in the project. It contains necessary tools to work with images, object detection.
- Pandas: To organize and process labelled data in the weed and crop database and helping in perform classifications.
- PIL (Python Imaging Library): A library used to handle image data within Python, allowing the reading, writing, and manipulation of images.

Chapter 4

Results and Discussion

4.1 Implementation Details

4.1.1 Data Preprocessing and Training

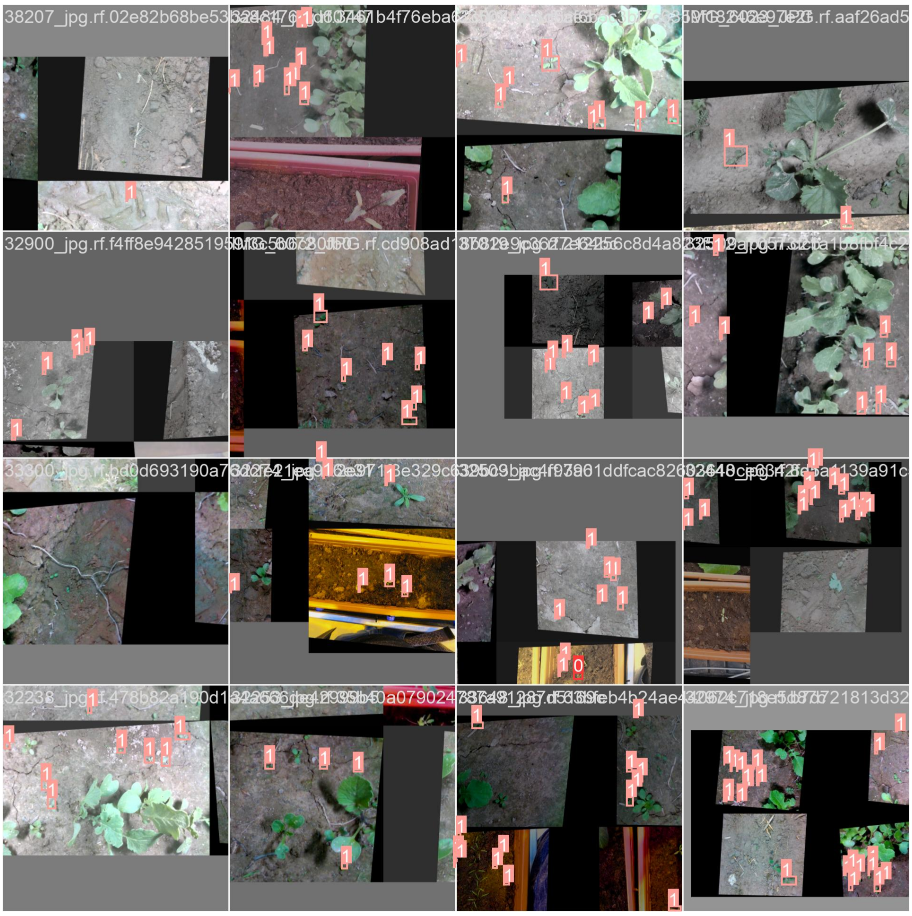

Figure 4.1: Label encoding

Initially, when we were collecting our data, our main focus was on collecting a thorough dataset that portrays real life scenarios farmers have to deal with in order to extract weed from crops. We gathered test images from various farms which were in close proximity to the city and also used kaggle to get additional surplus of images for training and validating purposes. Our training dataset comprises of around more than 2800 images.

After compiling the raw data, we started a thorough cleaning process to detect and remove inconsistencies, improper training images, and anomalies that could impact our results. Then we labelled the data carefully into two classes: weed and crop. Next, we transformed important variables into categorical or numerical attributes to enhance analysis. We used label encoding on categorical variables to convert them into numerical format, which is necessary for training our Yolov5 model on. Refer Figure 4.1 for its implementation

4.1.2 Installing Yolov5 and libraries

Figure 4.2: Installing Yolov5 code

First We have Installed the necessary libraries for the project and cloned yolov5 from its repository. Some important libraries such as torchvision, ultralytics and others necessary for computer vision tasks. This code block will create a Yolov5 folder in which necessary files for training and testing are included. Next we will start training our custom dataset on this Yolov5 by specifying the batch size i.e number of samples processed was set to 16, number of epochs which is the number of times the batch has been processed was specified to 100 and other such factors.

```
# Start training
%cd /content/yolov5

!python train.py \
    --img 640 \
    --batch 16 \
    --epochs 100 \
    --data {UPDATED_YAML_PATH} \
    --weights best.pt \
    --cache \
    --project custom_model \
    --name yolov5_custom \
    --patience 50 \
    --save-period 10 \
    --entity none

# Function to validate training results
def validate_training():
    # Check if CUDA was used
    print(f"CUDA Available: {torch.cuda.is_available()}")

    # Display training results
    print("\nTraining completed. Results saved in custom_model/yolov5_custom")

    # Load and display the training results
    Image(filename='custom_model/yolov5_custom/results.png')

# Validate the training
validate_training()
```

Figure 4.3: Training Yolov5 code

4.1.3 Training Yolov5

We initially used Google Colab notebook's GPU to train our Yolov5 model on the custom dataset to gather the 'best.pt' weights. Later we have used these weights locally to run our custom trained Yolov5 model whenever required without having to retrain them.

```
# Running detect.py to run yolov5 on an image
!python detect.py --weights best.pt --img 640 --conf 0.25 --source "/content/drive/MyDrive/WeedCrop.v1i.yolov5pytorch/test/images/32469_jpg.rf.932e846f4301e65383a16365ecf7b78a.jpg"

while True:
  count += 1
  break;
```

Figure 4.4: Running detect.py

Lastly, we have used the detect.py available by yolov5 to test our trained model. We have used the weights (best.pt) we got after training our model. We have used them here to test it on an image. To run this command we have to specify few things such as the image path, the weights to be used and the minimum confidence level. Figure 4.4 shows that step being executed.

Figure 4.5: Displaying the result

4.2 Result Analysis

4.2.1 Confusion Marix

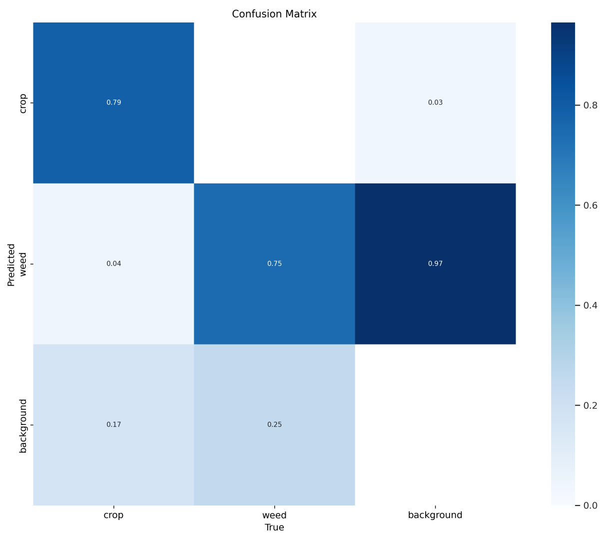

Figure 4.6: Confusion Matrix

The given confusion matrix gives us insight into the accuracy practised by the Weed identification and classification model, as well as the areas of confusion across the three given categories that are: Crop, Weed and Background. The model correctly classified 79% of crops, with

4% of crops misclassified as weeds and 17% as background, showing that the model has a tendency to confuse the crops with the surroundings, which results in some of the crops being overlooked. Similarly for the weeds, 75% of them were accurately detected but as it is shown, there is still confusion between 3% of weeds classified as crops and 25% as background. This might reveal that the model deems it difficult to differentiate between weeds and non-relevant field elements, due to similarities in structure, texture, colour etc. The highest accuracy prediction was done of background elements with 97% correctly classified, whereas 3% we misidentified as crops and 25% were misidentified with weeds. This draws the conclusion that the model has considerable precision in background detection, but needs improvement in other classifications.

4.2.2 F1 Confidence Curve

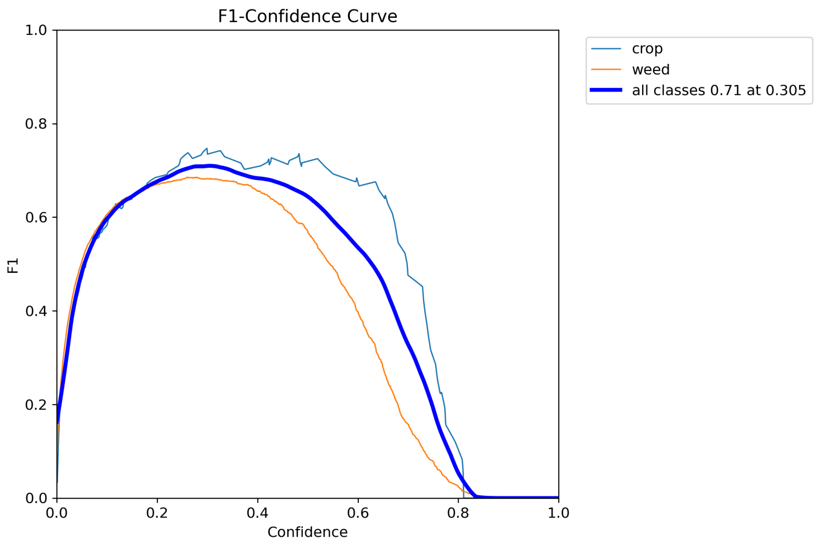

Figure 4.7: F1 Confidence Curve

The F1-Confidence Curve is a valuable visualisation for assessing the performance of our object detection model. It shows how model's F1 score changes across different confidence thresholds taking into account both the precision as well as the recall of a model.

Understanding the graph The X-axis represents the confidence threshold i.e the probability assigned by the model to each correct detection. It ranges from 0 to 1. Lower confidence means the model results in more detections while higher threshold results in fewer detections with high certainty. The Y-axis is the harmonic mean of precision and recall. High F1 score indicates good balance between precision and recall.

The light blue line represents the F1 score for detecting "crop" while the orange line represents the F1 score for detecting "weed". F1 scores for both classes rise indicating model is more confident. After reaching its peak, the score drops likely due to its higher precision but decreased recall. The bold blue line represents the F1 score when considering both weed and crop classes collectively. The peak of this line indicates the optimal confidence for the model when detecting all classes. From the graph we can analyse that at a confidence of 0.305, the combined F1 score for all classes is 0.71.

Key Insights from he Curve

- The optimal confidence threshold for our model's performance is 0.305, providing the best balance between detecting both classes with minimum false poitives or false negatives.

- F1 score for crop is consistently higher than for weed at most confidence levels, suggesting that the model is generally better at detecting crops. A low F1 score for weed can also mean its harder for the model to distinguish the weed from the background noise.

- The F1 score declines after the optimal confidence level likely because recall drops and precision increases, this trade-off results in lower false positives and more accurate results.

4.2.3 Precision-Recall Curve

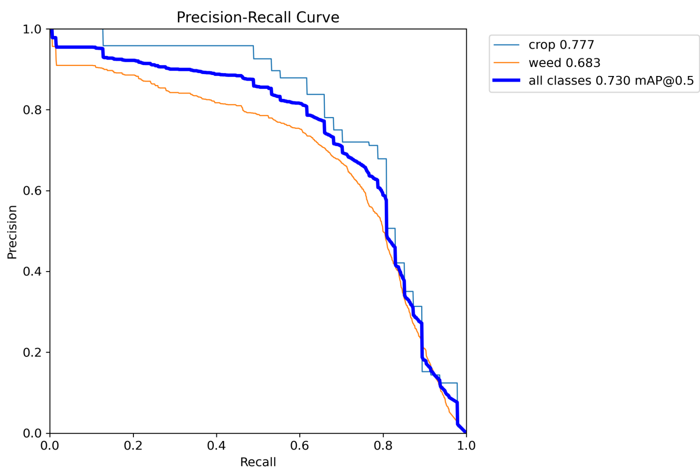

Figure 4.8: Precision-Recall Curve

After analysing the Precision-Recall curve for the model, it is evident how good the model performs in identifying crops which is show by its high average precision(AP) of 0.77, but still lacks in weed detection accuracy where the AP is 0.683. The model struggles with differentiating between weeds and crop or other background elements, which results in reduced precision. The overall mean average precision (mAP) is 0.730 in all classes. This means that the model is generally reliable but still needs improvement, perhaps in weed detection. The conclusion to draw from this is that the model requires more refinement to make it more precise and reliable in the real world, increasing its effectiveness and subsequently its application in the agricultural field. Nonetheless it is still effective in crop identification and will do the job.

Chapter 5

Conclusion

In the future, the model could be refined to reduce misclassifications, and a larger or diverse dataset could be used to improve accuracy under certain conditions. With the advent of advanced technology such as multispectral imaging and post processing, weed types could be differentiated more accurately and targeted interventions can be coordinated more effectively. Lastly, deploying this model on drones or robots could be beneficial for achieving greater precision in agriculture and enhancing sustainability as well.

5.1 Future Scope

- Automated Image Capturing using UAVs: To streamline data capturing our future scope involves using the best of technology in drones and UAVs which will capture images of the field drawing their own boundaries and limits in real time. This allows us to capture high resolution images that will gather critical data without any laboured intervention

- The next goal we aim to achieve is create a 3D, well defined model of the field using these large scale images. The model will offer a precise, spatially accurate representation of the field giving the users a close up as well as birds eye view of the field in digital format

- The next step is to use the 3D model and display each plant and weed using distinct colours like red for weed and green for plants. This colour coded approach will ease the task of locating and identifying weeds from the large fields

- In the last step we aim to help the farmers navigate the field to find and eradicate the weeds in order using a standardized coordinates system, that will accurately track plant location, support management strategies and potentially automate treatment solutions based on exact coordinates

References

[1] “STUDY OF TRADITIONAL METHOD OF WEEDING AND FERTILIZING AND THEIR DRAWBACKS” Mr. Mahesh K.Borbale, Mr. Ajinkya U.Pohekar, Mr. Jayesh R.Nistane, Mr.Nikhil S.Nichit, Mr.Triratna I.Yesankar, Mr.Aditya G.Kaware

[2] “WeedSeeker 2 Spot Spraying System - Vantage NZ,” Vantage NZ, Apr. 13, 2022. https://www.vantage-nz.com/portfolio/weedseeker-2-spot-spraying-system/ (accessed Nov. 08, 2024).

[3] “Weed Detection and Classification with Computer Vision Using a Limited Image Dataset” by László Moldvai , Péter Ákos Mesterházi, Gergely Teschner and Anikó Nyéki

[4] “Our Products - Welcome — Blue River Technology,” Bluerivertechnology.com, 2020. https://www.bluerivertechnology.com/our-products/

[5] J. Chen et al., “Weed detection in sesame fields using a YOLO model with an enhanced attention mechanism and feature fusion,” Computers and Electronics in Agriculture, vol. 202, pp. 107412–107412, Nov. 2022, doi: https://doi.org/10.1016/j.compag.2022.107412.

[6] A. Hruška and Pavel Hamouz, “Verification of a machine learning model for weed detection in maize (Zea mays) using infrared imaging,” Plant Protection Science, vol. 59, no. 3, pp. 292–297, Aug. 2023, doi: https://doi.org/10.17221/131/2022-pps.

[7] “Sensing Breakdown: FarmWise FT35,” www.tangramvision.com. https://www.tangramvision.com/blog/sensing-breakdown-farmwise-ft35

[8] Y. Full, “YOLO Full Model Architecture — Download Scientific Diagram,” www.google.com, 2024. https://images.app.goo.gl/AxDeKAk7w6wuyvZr8 (accessed Nov. 08, 2024).

Appendices

Appendix A

Weekly Progress Report

Figure A.1: Weekly Progress Report

Appendix B

Plagiarism Report

Appendix C

Publication Details / Copyright / Project Competitions

Acknowledgments

We would like to extend our sincere gratitude to everyone who has supported us throughout the journey of this project. We thank everyone who provided us with guidance, pathway and resources to develop this project. We would like to thank our supervisor Dr. Shamal Salunkhe for her valuable guidance in the making of this project. We would also like to thank our Head of Department Dr. Amarsinh Vidhate for providing us with the resources and a conductive environment to study and develop the project. We are thankful to all the faculty that supported us along the way and provide us with immense motivation. We would also like to thank our classmates for their unconditional support and constant encouragement during this entire journey.

Thank you all for your contributions to the success of this project.

Date: ____________________