Weed-Crop Recognition

Using Deep Learning

Abi Vu

Jerome C. Geronimo

Mariah Anjannelle A. Quinquito

Sheyi Omagene

**Red River College Polytech**

**COMP-3704 Neural Networks and Deep Learning**

Dr. Temitope Mapayi

December 03, 2024

Table of Contents

[Introduction 3](#_Toc184339469)

[Problem Statement 3](#_Toc184339470)

[Dataset 3](#_Toc184339471)

[Data Augmentation 4](#_Toc184339472)

[Model Selection 5](#_Toc184339473)

[Phase 1: Initial Model Exploration 5](#_Toc184339474)

[Phase 2: Pretrained Models 6](#_Toc184339475)

[Image Segmentation 6](#_Toc184339476)

[Experiments and Results 6](#_Toc184339477)

[Basic VGG Model 7](#_Toc184339478)

[VGG16 (Segmented) 9](#_Toc184339479)

[Fine Tuning 10](#_Toc184339480)

[Evaluation Metrics 10](#_Toc184339481)

[Challenges and Solutions 10](#_Toc184339482)

[1. Imbalanced dataset 11](#_Toc184339483)

[2. Preprocessing Inconsistencies 11](#_Toc184339484)

[3. Unexpected Results 11](#_Toc184339485)

[4. Evaluation Metrics & Overfitting 11](#_Toc184339486)

[5. Technical Limitations 12](#_Toc184339487)

[Real-Life Implementation 12](#_Toc184339488)

[Conclusion 12](#_Toc184339489)

[References 14](#_Toc184339490)

## Introduction

The agricultural industry faces significant challenges in maintaining productivity and sustainability while minimizing environmental impact. One critical aspect is the effective management of weeds, which compete with crops for essential resources such as sunlight, water, and nutrients. Traditional weed control methods, such as manual weeding and blanket herbicide application, are labor-intensive, costly, and often environmentally harmful.

In recent years, advancements in computer vision and deep learning have opened new opportunities for precision agriculture. By accurately distinguishing crops from weeds at an early growth stage, farmers can adopt targeted weeding strategies, reducing costs, improving crop yields, and promoting environmentally sustainable practices. This project aims to develop a deep learning model capable of accurately distinguishing crops and weeds using image-based data.

## Problem Statement

Efficiently distinguishing crops from weeds at the early growth stage remains a significant challenge in precision agriculture. The primary issues include:

1. **Complexity of Early-Stage Identification**: At the early stages of growth, crops and weeds often have similar morphological features, making them difficult to differentiate using traditional methods.
2. **Imbalanced Dataset**: Images of certain weed species may be significantly underrepresented in available datasets, leading to biased model performance and reduced generalization.
3. **Environmental Variability**: Factors such as lighting conditions, soil textures, and weather variations can influence the visual appearance of crops and weeds, complicating classification.

This project aims to address these challenges by using advanced deep learning techniques to build a robust, scalable model capable of early-stage crop and weed distinction. The goal is to provide a solution that improves farming efficiency, reduces environmental impact, and contributes to sustainable agricultural practices.

## Dataset

The dataset we chose for this case study is the [V2 Plant Seedlings Dataset](https://www.kaggle.com/datasets/vbookshelf/v2-plant-seedlings-dataset) obtained from Kaggle. The dataset consists of 960 images of seedlings from 12 species, including both crops and weeds. Each class is stored in each folder and folder name acts as a label. The dataset includes variations in lighting, soil conditions, and plant orientation, providing a realistic scenario for precision agriculture tasks.

We chose the dataset for the following reasons:

* **Relevance**: It directly aligns with our goal of distinguishing crops from weeds at an early stage.
* **Species Variety**: Includes a range of both crop and weed species for better model adaptability.
* **Realistic Conditions**: The image backgrounds and lighting reflect real farming environments.
* **Imbalanced Dataset**: It mirrors the challenges of real-world data and provides an opportunity to address class imbalance effectively.

## Data Augmentation

To address the class imbalance in our dataset, where category counts ranged from 253 to 762, we employed data augmentation techniques using ImageDataGenerator. This ensured that all categories were balanced with 762 images each. The augmentation process involved generating synthetic images through transformations such as rotation (up to 20 degrees), width and height shifts (up to 20% of the image), shear, zoom, and horizontal flipping. These techniques enhanced the diversity of our dataset while preserving the key features of the original images, ultimately improving model performance on minority classes.

A screenshot of a computer program

Description automatically generated

A graph of data distribution

Description automatically generatedA graph with text on it

Description automatically generated with medium confidence

## Model Selection

### Phase 1: Initial Model Exploration

A screenshot of a computer

Description automatically generatedThis project is divided into two phases. In the first phase, we aimed to establish a baseline performance using a simple Convolutional Neural Network (CNN). Each team member developed multiple versions of the model, experimenting with different architectures and parameters. Among the tested models, the following CNN architecture achieved the best performance, with a testing accuracy of 78.38%. The details of the architecture are summarized in the table below:

A screenshot of a graph

Description automatically generated

During this phase, team members worked independently, leading to variations in the preprocessing steps applied to the dataset. These differences included resizing, normalization, and data augmentation techniques, which influenced the input pipelines and, consequently, the model's performance. Even with identical architectures, the lack of standardized preprocessing contributed to inconsistent results across experiments.

To ensure more reliable and comparable outcomes in future phases, it is crucial to standardize preprocessing steps. By aligning resizing dimensions, normalization methods, and augmentation strategies, we can eliminate inconsistencies and better evaluate the model's performance.

### Phase 2: Pretrained Models

In Phase 2, we resolved the inconsistencies from Phase 1 by creating a shared preprocessed dataset with standardized resizing, normalization, and augmentation steps. This ensured that all members trained their models on a consistent and uniform dataset, eliminating variability caused by differences in preprocessing pipelines.

We also incorporated pretrained models to evaluate their potential for improving performance. Pretrained architectures, including ResNet50, VGG16, and MobileNet, were fine-tuned and tested with multiple configurations. Among these, VGG16 yielded the best results, outperforming both ResNet50 and MobileNet.

Using a basic VGG16 model, we achieved a test accuracy of 74%. To improve performance, we implemented image segmentation to isolate the plant parts within each image, enhancing feature focus and boosting accuracy to 82.18% for the baseline VGG16 model. Further fine-tuning involved unfreezing the last four layers of VGG16, allowing the model to adapt more effectively to our dataset, resulting in an improved test accuracy of 83.93%.

### Image Segmentation

We used image segmentation for training VGG16 to focus the model on the most relevant parts of the images, improving its ability to differentiate between classes. By isolating specific regions of interest, such as individual plants or weeds, segmentation removes background noise and reduces irrelevant information, enabling the model to learn more effectively. This approach enhances feature extraction and contributes to better accuracy and generalization, especially in datasets with complex or cluttered images.

## Experiments and Results

Given that VGG16 outperformed other pretrained models in Phase 2, the team decided to focus on further experiment with this pretrained model. Each team member developed multiple variations of transfer learning with VGG16 by experimenting with different configurations, including modifications to the number of dense layers, dropout rates, and activation functions. These variations were designed to explore how architectural tweaks could impact the model's performance.

The team then compared the results from these different versions of VGG16, evaluating their performance based on testing accuracy, precision, recall, and overall efficiency. The two best-performing models were selected for further analysis and potential deployment, representing the culmination of collaborative efforts to refine the VGG16 architecture for optimal results in the crop-weed classification task.

### Basic VGG Model

The basic VGG16 model achieved a solid performance with 74% accuracy by leveraging its pre-trained convolutional layers for feature extraction. We added a single dense layer with 256 neurons, enabling the model to effectively learn high-level patterns specific to our dataset. To combat overfitting, we included a dropout layer with a rate of 0.2, which randomly deactivated neurons during training, ensuring better generalization. This straightforward architecture struck a balance between complexity and efficiency, demonstrating the model's capability to adapt to our classification task.

A computer screen shot of text

Description automatically generated

A screenshot of a computer program

Description automatically generated

A graph with blue and orange lines

Description automatically generated

A screenshot of a graph

Description automatically generated

A screen shot of a black screen

Description automatically generated

### VGG16 (Segmented)

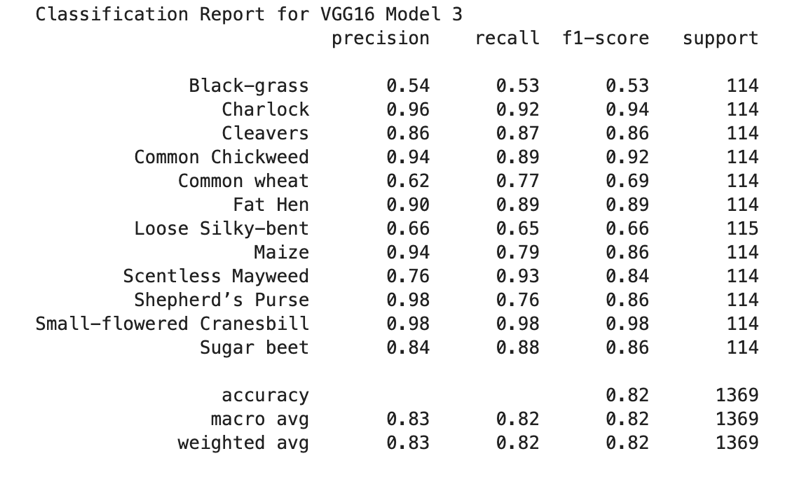
A graph with blue and orange lines

Description automatically generatedA computer screen shot of a math equation

Description automatically generatedA screenshot of a computer

Description automatically generatedUsing VGG16 with a dense layer of 64 neurons on segmented images significantly improved performance, achieving a higher test accuracy of 83%. The segmented images, which focused exclusively on plant parts, enhanced the model’s ability to learn relevant features by reducing background noise. The smaller dense layer provided an effective balance between complexity and regularization, allowing the model to generalize well to unseen data while maintaining computational efficiency.

A graph of a graph with numbers and a bar chart

Description automatically generated with medium confidenceA graph with blue and orange lines

Description automatically generated

### Fine Tuning

A screenshot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generatedTo extract more performance from the VGG16 model, we fine-tuned it by unfreezing the last four layers, allowing the pretrained model to adjust its weights to better suit our dataset. Additionally, we reduced the learning rate to 0.00001, enabling smaller, more precise adjustments during training. This fine-tuning approach yielded a test accuracy of 85%, demonstrating the effectiveness of using transfer learning with targeted modifications for improved model adaptation and performance.

## Evaluation Metrics

To evaluate and select the best model, we relied on key metrics including test accuracy, F1 score, and confusion matrices. Test accuracy provided an overall measure of model performance, while the F1 score balanced precision and recall, making it particularly useful for our imbalanced dataset. Confusion matrices offered detailed insights into class-wise predictions, highlighting areas of strength and misclassification. Together, these metrics ensured a comprehensive assessment of model effectiveness.

## Challenges and Solutions

During this project, we encountered several challenges that required careful problem-solving and collaboration. These challenges can be categorized into five key areas: imbalanced dataset, preprocessing inconsistencies, unexpected results, evaluation metrics and overfitting, and technical limitations.

### Imbalanced dataset

**Challenge**: The dataset used in this project was imbalanced, with some plant species underrepresented compared to others. This imbalance could lead to biased model predictions, where the model favors majority classes and underperforms on minority classes.

**Solution**: To handle this, we used data augmentation techniques to artificially increase the size of minority classes. Methods such as flipping, rotating were applied to create a more balanced dataset.

### Preprocessing Inconsistencies

**Challenge**: In Phase 1, team members worked in silos, which resulted in inconsistencies in dataset preprocessing. Variations in resizing dimensions, normalization methods, and augmentation techniques led to discrepancies in model performance, even when using the same architecture.

**Solution**: To address this, we standardized the preprocessing pipeline in Phase 2. A shared preprocessed dataset was created, with clear guidelines for resizing, normalization, and augmentation. This ensured that all models were trained on consistent input data, leading to more reliable and comparable results.

### Unexpected Results

**Challenge**: While pretrained models like ResNet50 and MobileNet were expected to outperform the simple CNN baseline, their results were underwhelming. Even the best-performing model, VGG16, initially showed a lower accuracy than the baseline, contrary to our initial expectations.

**Solution**: We conducted an in-depth analysis to understand the limited improvement. Factors such as the simplicity of the dataset and the effectiveness of the baseline CNN were identified as possible reasons. To maximize VGG16's potential, we focused on fine-tuning its architecture with multiple variations, ultimately selecting the best-performing model after rigorous comparisons.

### Evaluation Metrics & Overfitting

**Challenge**: Overfitting was particularly apparent when using pretrained models. These models, such as VGG16, ResNet50, and MobileNet, are highly complex and designed to handle large and diverse datasets. However, our dataset was relatively small and specific, making it less suited to the high capacity of these architectures. As a result, the models performed well on the training set but failed to generalize effectively to the validation and test sets.

**Solution**: To address this issue, we applied several techniques to reduce overfitting. These included using dropout layers, L2 regularization, and early stopping during training. Additionally, we experimented with unfreezing some of the pretrained layers to increase the trainable parameters, allowing the models to focus on learning features specific to our dataset. Despite these efforts, we observed that the pretrained models may still have been too complex for the dataset's nature, underlining the importance of selecting architectures that match the complexity of the task and data.

### Technical Limitations

**Challenge**: Limited computational resources presented a constraint, particularly when training large pretrained models. The high memory requirements and longer training times posed difficulties for some team members.

**Solution:** To overcome this, we optimized resource allocation by using batch size adjustments and reducing model complexity for experimentation.

The challenges faced during this project, from preprocessing inconsistencies and imbalanced datasets to overfitting and technical limitations, highlighted key areas requiring adaptability and problem-solving. By standardizing preprocessing, addressing dataset imbalances, fine-tuning pretrained models, and implementing overfitting mitigation techniques, we were able to improve model performance and gain valuable insights. These solutions not only resolved immediate issues but also provided a strong foundation for refining future deep learning projects in similar domains.

## Real-Life Implementation

To apply this model in real-world settings, the following steps can be taken:

1. **Integration with Agricultural Machinery**: The model can be embedded into drones or autonomous tractors equipped with cameras to scan fields in real time.
2. **Mobile Or Web Application**: Develop a user-friendly app for farmers, enabling them to upload images of their fields for instant classification and actionable insights.

There are several other ways the model or solution can be implemented but due to time constraints we were able to put a simple web application which can be deployed and used on the cloud. These implementations can effectively address the identified problems, ensuring farmers can make informed decisions to optimize crop yield and reduce environmental impact.

## Conclusion

This project aimed to develop a robust deep learning model for distinguishing crops from weeds at an early growth stage, contributing to advancements in precision agriculture. The project was divided into two phases, each addressing specific challenges and objectives:

1. **Baseline Development**: In Phase 1, we established a baseline CNN, which provided a strong foundation with a testing accuracy of 78%. However, inconsistencies in preprocessing among team members highlighted the need for standardization.
2. **Pretrained Model Exploration**: Phase 2 focused on addressing preprocessing inconsistencies and leveraging pretrained models. While pretrained architectures like ResNet50, MobileNet, and VGG16 were tested, only VGG16 showed slight improvement over the baseline. The limited gains emphasized the importance of aligning model complexity with the dataset's characteristics.
3. **Optimization and Refinement**: To maximize performance, the team collaboratively refined VGG16, testing various architectural configurations and selecting the best-performing model. This approach demonstrated the value of iterative improvements and teamwork.

Throughout the project, challenges such as imbalanced datasets, overfitting, and technical limitations were addressed through strategic solutions, including data augmentation, regularization techniques, and resource optimization. These efforts underscored the importance of adaptability, collaboration, and critical evaluation in deep learning projects.

The project achieved its goal of building a crop-weed classification model while providing insights for future endeavors. Moving forward, expanding the dataset and exploring simpler or hybrid architectures could further enhance model performance and scalability, ensuring broader applicability in real-world agricultural scenarios.

## References

Kaggle. (n.d.). *V2 Plant Seedlings Dataset*. Retrieved from <https://www.kaggle.com/c/plant-seedlings-classification/data>

Gupta, K., Rani, R., & Bahia, N. K. (n.d.). *Plant-seedling classification using transfer learning-based deep convolutional neural networks*. Deloitte Consulting India Pvt. Ltd., Hyderabad, India, & Department of Computer Science and Engineering, NIT Jalandhar Punjab, India. Retrieved from <https://orcid.org/0000-0003-2104-227X>

Srinivasan, A. (2006). *Precision agriculture: An overview*. In A. Srinivasan (Ed.), *Handbook of precision agriculture: Principles and applications* (pp. 1-18). The Haworth Press.