WHEAT GROWTH STAGE CHALLENGE

by

Navyashree Byregowda, Krishan Kumar, Anaz Nazeer Khan

A report based on wheat growth stage

Course : Data Mining & Machine learning

Dept of Engineering

University of Naples Federico II

Dec 2024

Executive Summary

The project aimed to develop and evaluate an image classification system of wheat growth stage using CNN and pre trained model. This comprehensive dataset comprises 8,253 images with seven different growing stages. Methodologically, the study followed a structured approach including data inception, statistical analysis of pixel insensitive, dimensional analysis upon brightness and contrast of image, Data augmentation using imgaug library, model selection (CNN and pre trained model). Result of real-world **agriculture-based image classification**.

Strengths: A combination of **custom CNN and transfer learning** improved classification accuracy.

Preprocessing techniques like augmentation and undersampling balanced the dataset.

Flask deployment made the model usable in real-world scenarios.

Table of Contents

Methodology

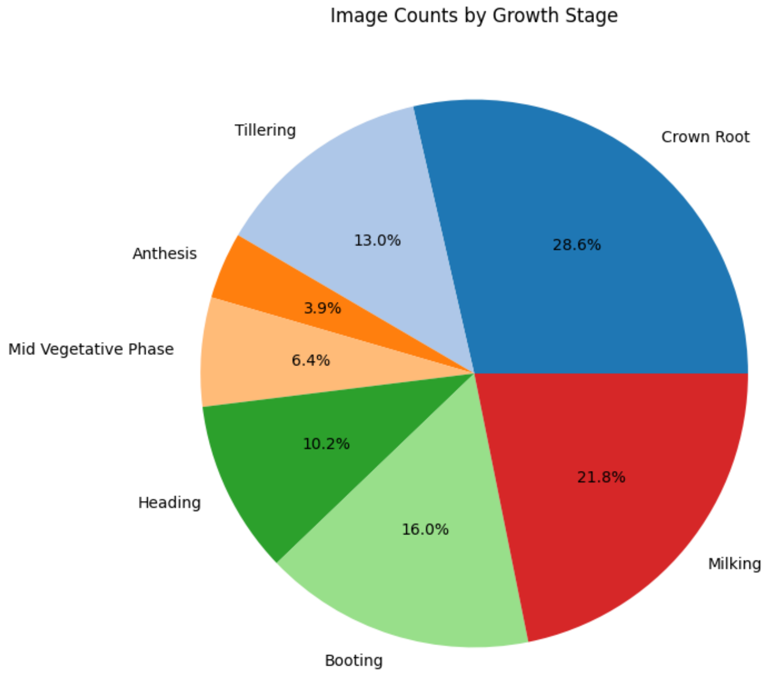
The methodology for this study involves multiple stages, including **data collection, preprocessing, exploratory data analysis, model development, and deployment**.

Wheat is a critical staple crop, and monitoring its growth stages is essential for optimizing yield, resource allocation, and disease prevention. This project leverages computer vision and machine learning techniques to classify wheat growth stages using image data.

The following steps detail the process:

1. Data Collection

- The dataset consists of **8,253 images** of wheat crops collected using the **WheatCam app**.
- Images were captured during the **Rabi season** in **Punjab and Haryana, India**.
- **1,685 farmers** contributed images over different growth stages.
- A **horizon detection algorithm** was used to define the **Region of Interest (ROI)**, anonymizing non-vegetation elements.

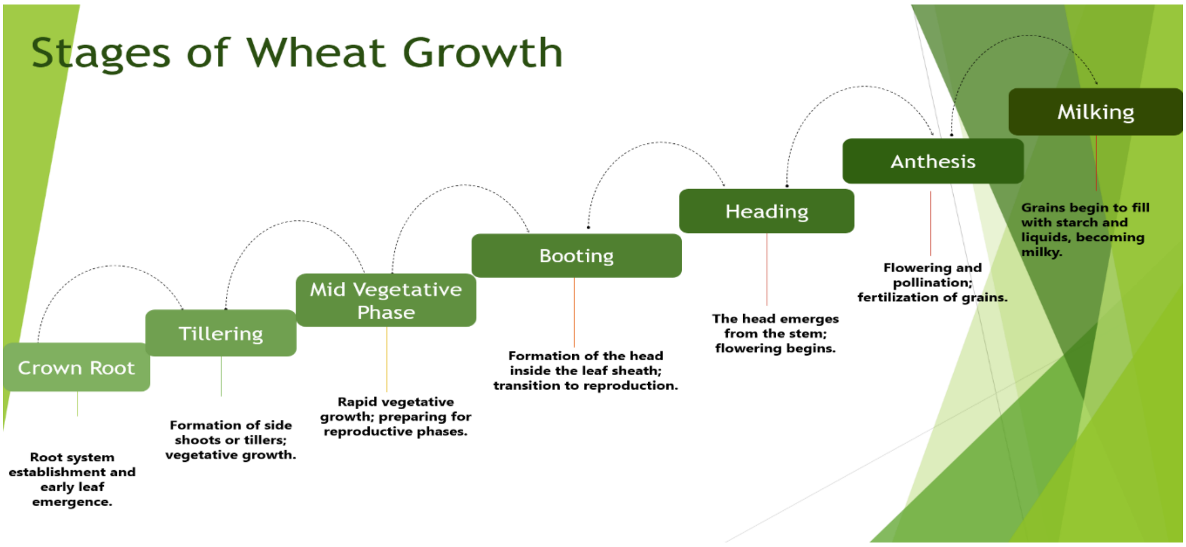


2. Data Preprocessing

To enhance the dataset and ensure model efficiency, the following preprocessing steps were applied:

2.1 Label Validation

- Growth stages were labelled into **seven categories** based on plant maturity:
    - **Crown Root, Tillering, Mid Vegetative Phase, Heading, Booting, Anthesis, Milking**.

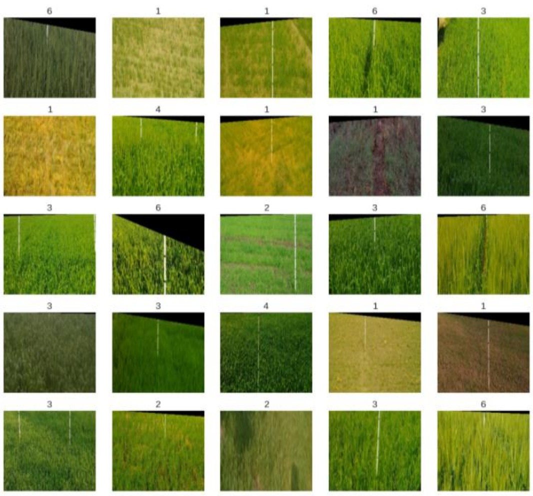


2.2 Handling Imbalanced Data

- **Undersampling:** The majority classes (**Crown Root, Milking, Booting, Tillering**) were reduced to **1,000 images each**.
- **Data Augmentation:** The minority classes (**Heading, Mid Vegetative Phase, Anthesis**) were **increased using rotation, flipping, and contrast variations**.

2.3 Image Normalization

- Pixel values were **scaled between 0 and 1** to improve model performance.
- **Mean and standard deviation of pixel intensities** were computed.

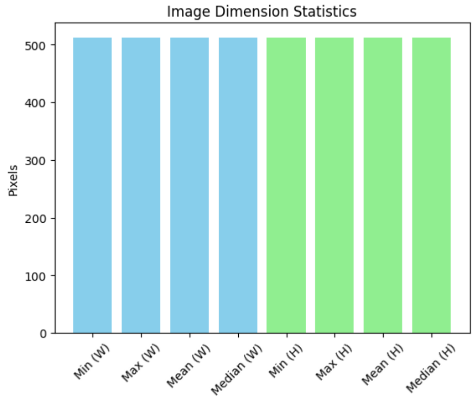


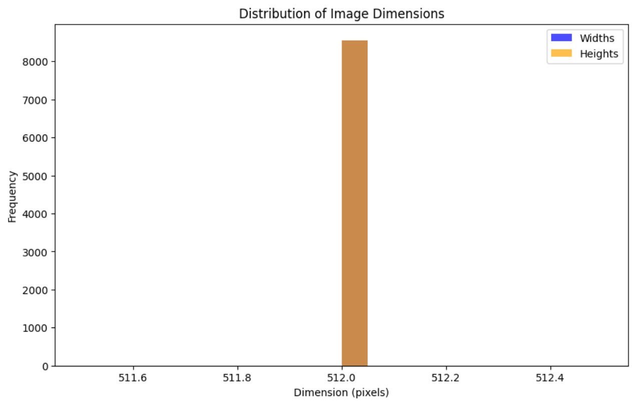
3. Exploratory Data Analysis (EDA)

A thorough analysis of the dataset was performed to understand its characteristics:

3.1 Image Distribution

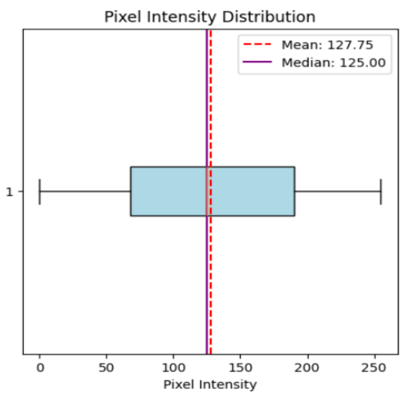
- A **pie chart** was generated to visualize the distribution of images across different growth stages

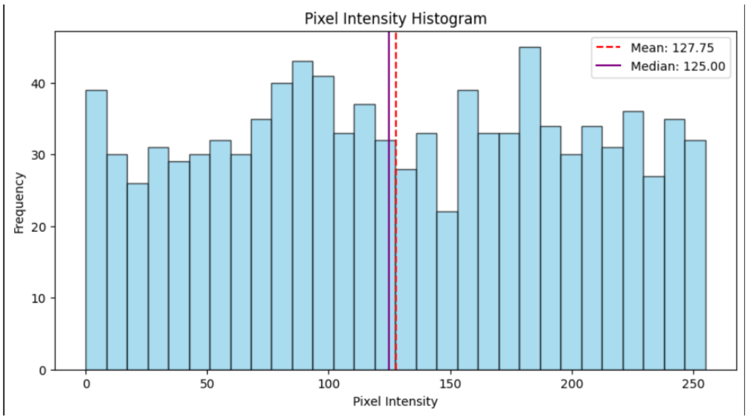




3.2 Pixel Intensity Analysis

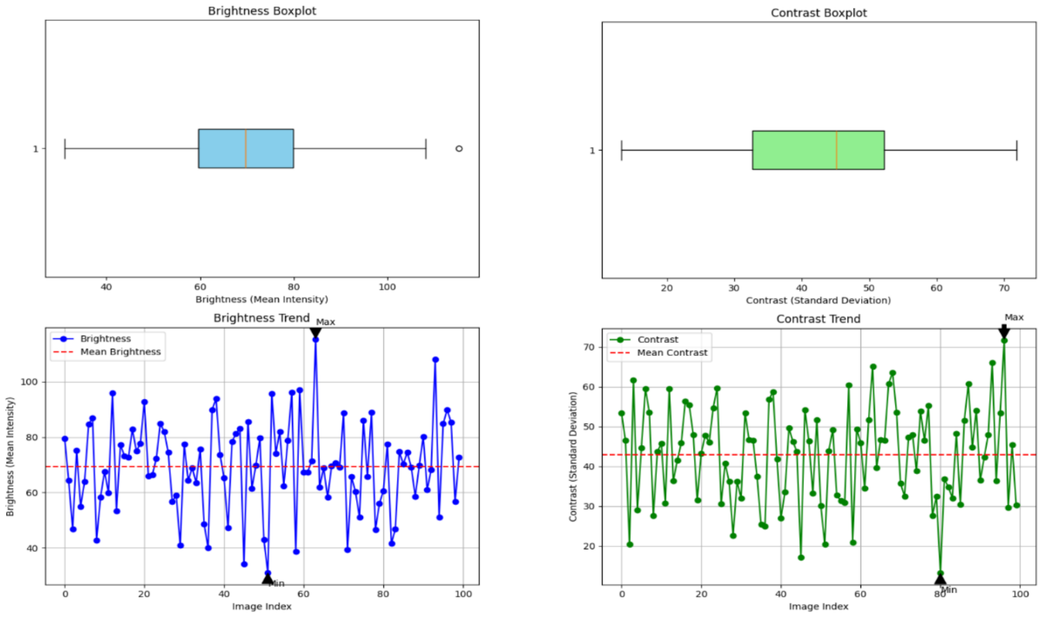
- **Mean, median, standard deviation, and min/max intensity values** were computed.





3.3 Image Brightness and Contrast Analysis

- **Brightness statistics: Min: 31.11, Max: 115.28, Mean: 69.37.**
- **Contrast statistics: Min: 13.27, Max: 71.77, Mean: 42.99.**

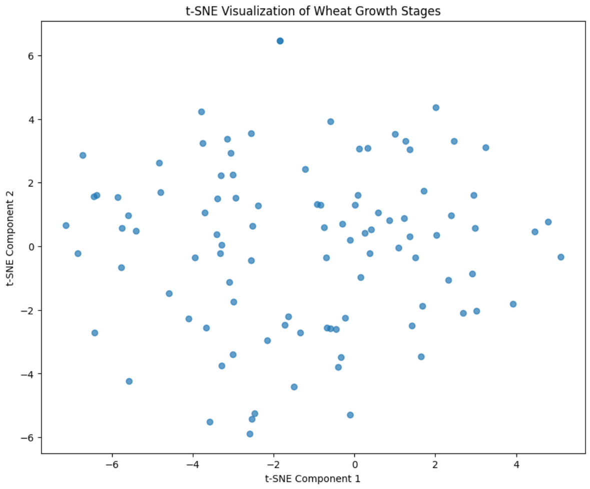


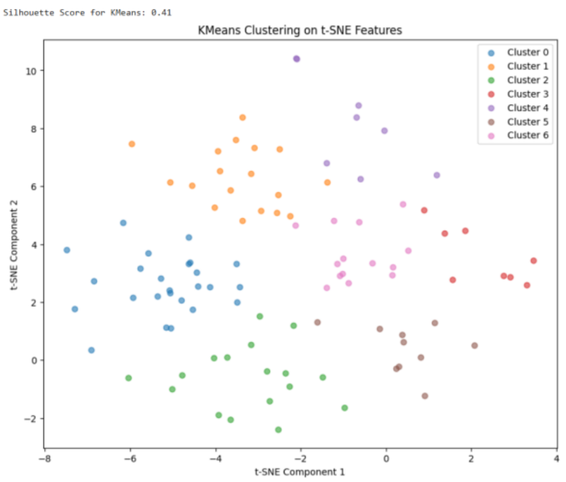
3.4 t-SNE (t-Distributed Stochastic Neighbour Embedding)

- **t-SNE visualization was used to check how different wheat growth stages are grouped in lower-dimensional space.**

3.5 K-Means Clustering

- **7 clusters were identified.**
- **The Silhouette Score (0.41) indicated moderate clustering effectiveness.**





4. Model Development

Two main approaches were used to train machine learning models for growth stage classification.

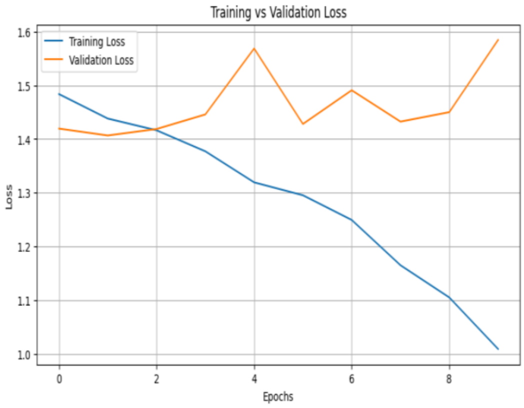
4.1 Custom Convolutional Neural Network (CNN)

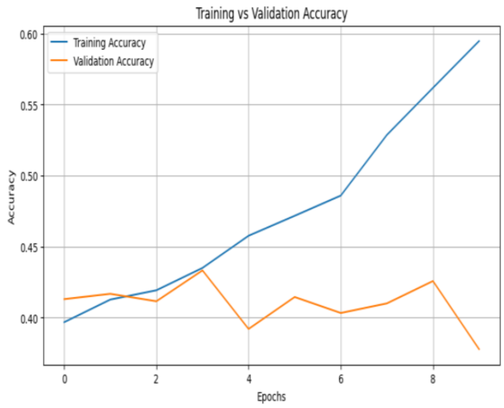
A **CNN model** was designed with the following architecture:

- **Convolutional layers:** Extract features such as edges, textures.
- **Pooling layers:** Reduce image dimensionality.
- **Fully connected (Dense) layers:** Convert extracted features into predictions.
- **ReLU activation function:** Introduced non-linearity.
- **Adam optimizer:** Used for gradient optimization.

CNN Performance

Metric	CNN
Training Accuracy	41.73%
Validation Accuracy	40.40%
Training Loss	1.5279
Validation Loss	1.4854





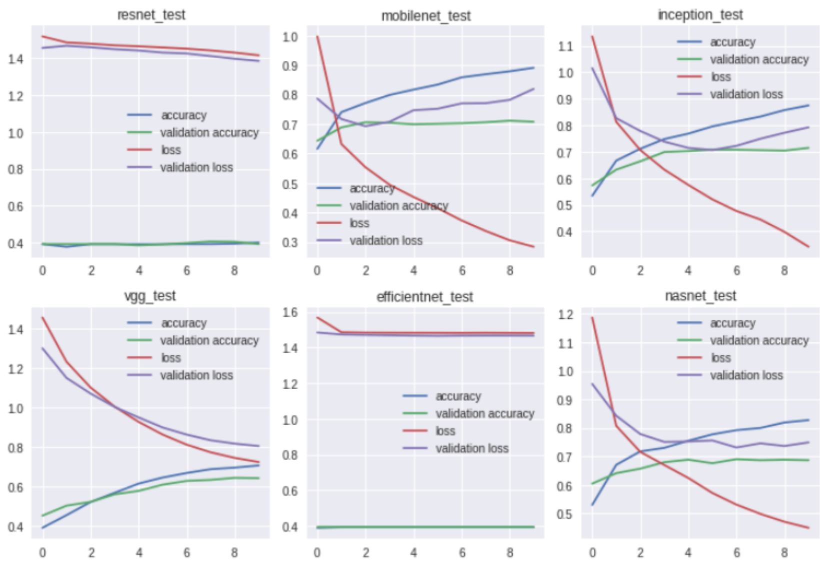
4.2 Transfer Learning (Pre-Trained Models)

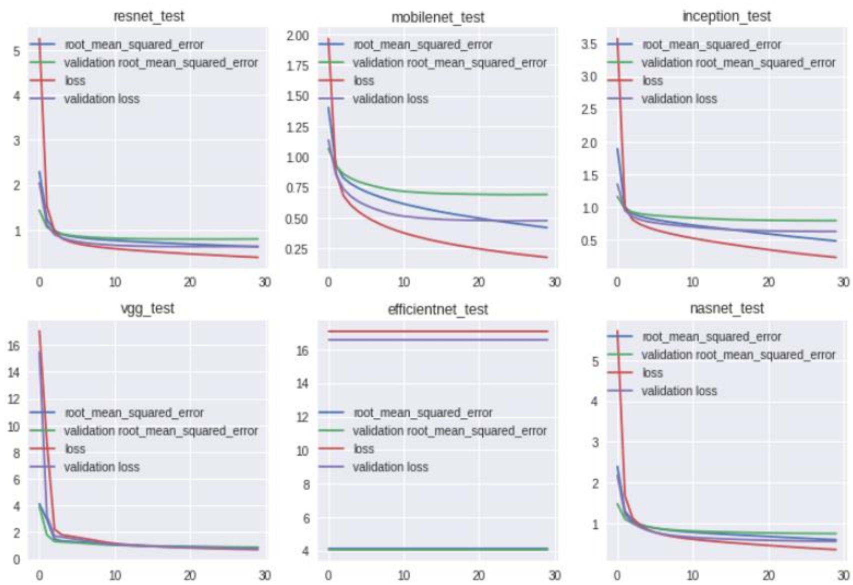
The following **pre-trained deep learning models** were evaluated:

1. **ResNet50**
2. **VGG16**
3. **MobileNetV2**
4. **InceptionV3**

Model	Training Accuracy	Validation Accuracy
ResNet50	25.85%	25.11%
VGG16	41.01%	40.70%
MobileNetV2	95.48%	79.88%
InceptionV3	95.13%	81.50%

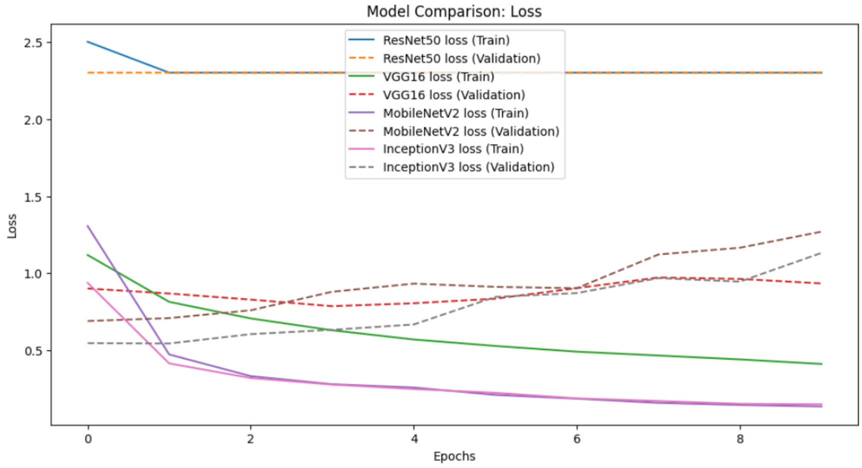
- **MobileNetV2 and InceptionV3 performed the best** in both training and validation.

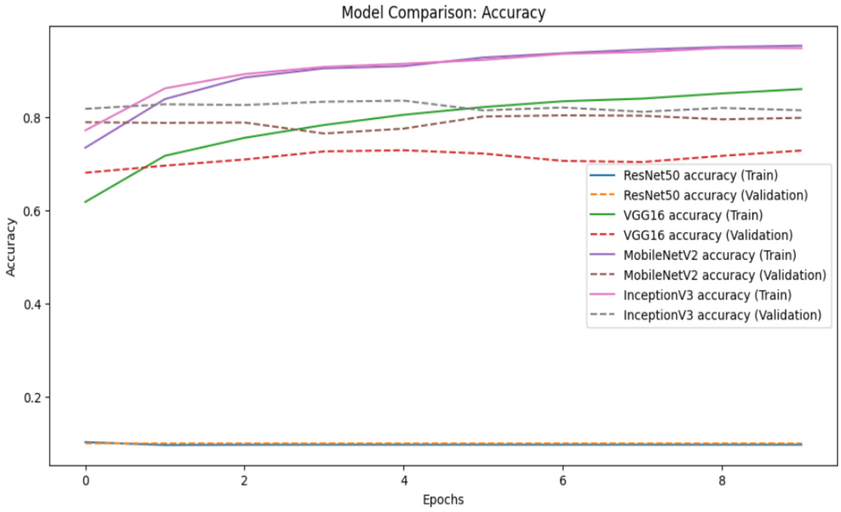




5. Model Evaluation

- **Loss and Accuracy curves** were plotted for each model to monitor training behaviour.
- The **Root Mean Squared Error (RMSE)** was calculated to measure prediction errors.



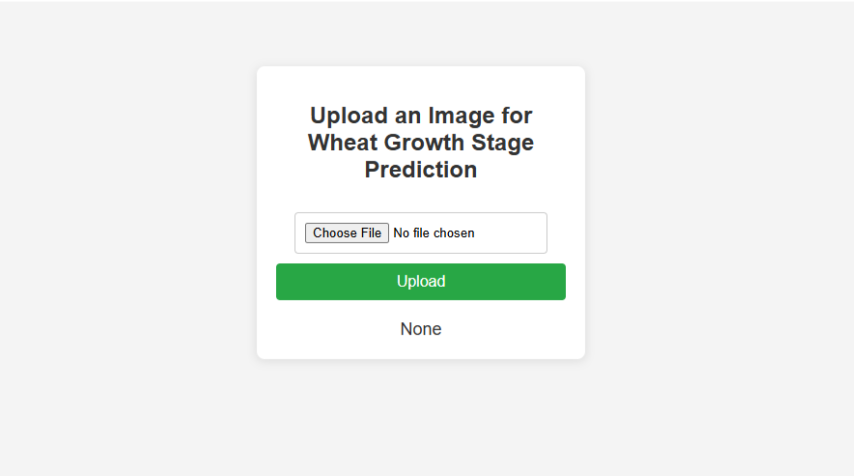


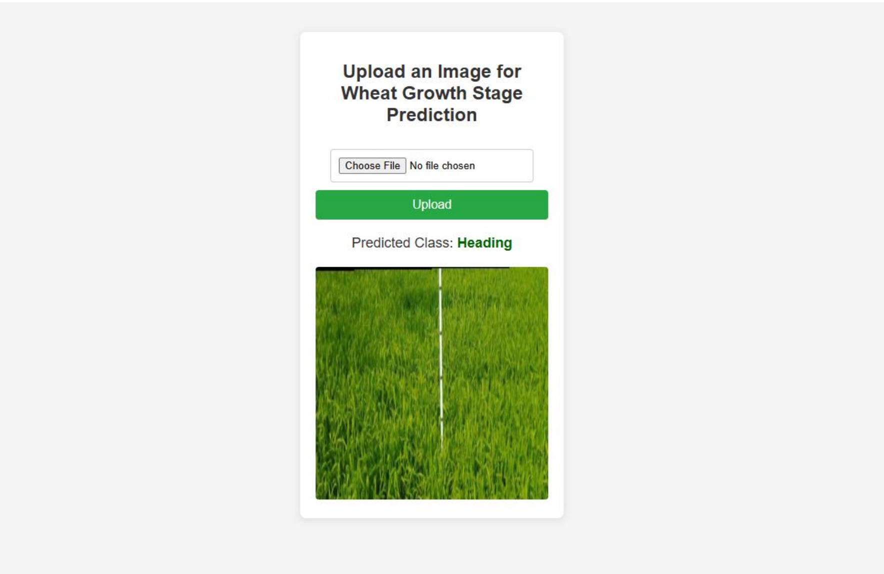
6. Implementation

A **Flask-based web application** was developed to allow real-time wheat growth stage classification.

Key Features:

- **Users can upload images**, and the model predicts the wheat growth stage.
- **Lightweight framework** with a user-friendly interface.
- **Seamless integration with machine learning models** for live predictions.





7. Conclusion

This study demonstrates the **effectiveness of CNN architectures** in **wheat growth stage classification**:

- **Transfer learning outperformed custom-built CNN models.**
- **MobileNetV2 and InceptionV3** provided the **best results**.
- **Data augmentation and undersampling improved model balance**.
- The **Flask-based deployment** enables real-world usability.

Future Work

- **Improve model generalization** by incorporating additional field data.
- **Test on different crop varieties** beyond wheat.
- **Optimize real-time deployment** for mobile applications.

Bibliography

François Chollet, *“Deep_Learning_with_Python”*. Manning Publications. 2021.

Aurélien Géron. *“Hands-On Machine Learning with Scikit-Learn & TensorFlow”*. O’Reille Media. 2019.