Chapter 1
Introduction

1.1 Overview of plant phenotyping and the importance of leaf counting.

At the forefront of agricultural research, plant phenotyping provides a robust framework for meticulously measuring and analyzing various plant traits, akin to conducting a comprehensive "health check-up" for plants. This includes evaluating traits like leaf size, shape, and color, as well as the number of leaves, which collectively influence plant performance and adaptability. Leaf counting, in particular, serves as a pivotal aspect in understanding plant growth dynamics and developmental processes. By quantifying leaf numbers, researchers gain insights into crucial stages of plant life cycles, such as flowering, fruit set, and senescence.

The significance of leaf counting extends beyond numerical assessment, playing a critical role in diverse realms of agricultural research and crop improvement. Precise quantification of leaf numbers enables researchers to monitor growth trajectories, phenological events, and biomass accumulation, ultimately enhancing crop yield potential and resource use efficiency. Moreover, leaf counting serves as a foundational element in various phenotypic analyses, facilitating genotype-phenotype associations and trait discovery in crop breeding programs.

In essence, leaf counting emerges as a cornerstone in plant phenotyping, offering unparalleled insights into plant biology, agronomic practices, and global food security. Its multifaceted role as a quantitative descriptor and a phenotypic marker underscores its importance in advancing our understanding of plant physiology and optimizing agricultural strategies to meet the challenges of a rapidly evolving environment.

1.2. Motivation

The motivation behind this project stems from the critical need to address challenges in agricultural productivity and food security amidst a rapidly changing global landscape. As the world's population continues to grow, coupled with escalating environmental pressures and climate variability, there is an urgent imperative to enhance agricultural resilience and sustainability.

At the heart of agricultural advancement lies the optimization of crop productivity, which necessitates a deep understanding of plant biology, genetics, and environmental interactions. Plant phenotyping emerges as a crucial discipline in this endeavor, offering a holistic approach to unraveling the complex interplay between genotype and phenotype—the genetic makeup of plants and their observable traits.

Traditional methods of plant phenotyping, such as manual measurement and observation, are labor-intensive, time-consuming, and often prone to subjectivity. By leveraging automated techniques, such as convolutional neural networks (CNNs) and explainable AI, we aim to streamline and expedite the phenotyping process, enabling rapid and scalable data collection across diverse plant populations.

In the face of climate change and evolving environmental conditions, resilient crop varieties are paramount for ensuring food security and agricultural sustainability. Automated leaf counting serves as a vital tool for characterizing plant resilience traits, such as drought tolerance, disease resistance, and yield stability, enabling breeders to develop climate-smart crops capable of thriving in challenging agroecosystems.

1.3 Objectives of the project

The primary objective of the project is to develop an automated leaf counting system for plant phenotyping using deep learning techniques. This system aims to address the inefficiencies of manual leaf counting methods by automating the process, thereby streamlining agricultural research and improving crop management practices. By leveraging advanced technologies like convolutional neural networks (CNNs) and the GradCAM explainable AI technique, the project seeks to accurately determine leaf counts for input plant images. The system's objective is to provide quick and non-invasive assessments of plant traits, aiding in informed farming decisions and enhancing global food sustainability.

1.4 Scope of the Project

The scope of this project encompasses the development and implementation of a novel automated leaf counting system using convolutional neural networks (CNNs) and explainable AI techniques. The project will focus on:

1. Designing and training a CNN model from scratch to accurately count the number of leaves in sorghum plant images taken at 5 different angles.
2. Integrating explainable AI methods, such as GradCAM, to provide insights into the features contributing to leaf count predictions.

1.5 Functional and Non-Functional Scope of the Project

<u>Functional Scope:</u>

1. **Leaf Count Prediction**: The primary function of the system is to accurately predict the number of leaves present in sorghum plant images taken at 5 different angles using the trained CNN model.
2. **Explainable AI Integration**: The system will incorporate explainable AI techniques, such as GradCAM, to provide visual explanations for the leaf count predictions. The system will provide explanations for its predictions using heat maps, enhancing the interpretability of the CNN model's decision-making process.

Non-Functional Scope:

1. **Accuracy**: The system aims to achieve high accuracy in leaf count predictions for sorghum plant images captured at different angles, ensuring reliable results for agricultural applications.
2. **Robustness**: The system will demonstrate robustness in handling variations in image perspectives, lighting conditions, and plant morphology, ensuring consistent performance across diverse datasets and environmental settings.
3. **Scalability:** The system will be designed to handle an increasing number of images and plants without significant performance degradation.
4. **Availability:** Ensuring the system is accessible and operational whenever required, with regular maintenance and monitoring to address potential issues promptly.

Chapter 2
Literature Review

Leaf counting plays a crucial role in plant phenotyping, facilitating the assessment of plant growth and yield estimation. Recent advancements in computer vision and deep learning techniques have revolutionized leaf counting methodologies, leading to improved accuracy and efficiency in the phenotyping domain [1].

Yotam Itzhaky et al. from Ben Gurion University of the Negev proposed innovative approaches for leaf counting using deep convolutional neural networks (CNNs). Their study introduced two methods: direct regression and counting via leaf center point detection, both outperforming previous techniques [1].

Object detection techniques have emerged as promising solutions for leaf counting tasks. Modified versions of popular frameworks like YOLOv3 have shown remarkable performance improvements in leaf counting accuracy [2]. By treating leaf counting as an object detection problem, these methods offer a flexible and effective approach to automate the process [3].

In addition to object detection, regression-based models have been developed for leaf counting in monocot plants like sorghum and maize. These models leverage deep neural networks and skeleton structure extraction to overcome challenges such as occlusion and unclear plant structures [4]. The inclusion of leaf tips as significant features enhances the accuracy of regression models [5].

Real-time leaf counting systems have practical implications in precision agriculture, enabling timely decision-making for farmers and greenhouse practitioners. Autonomous ground robotic devices equipped with deep object detection networks can efficiently capture and analyze plant images, providing valuable insights for crop management [6]. High-throughput phenotyping platforms coupled with deep learning-based approaches offer scalable solutions for leaf counting in diverse plant species. These platforms contribute to the development of large-scale datasets, fostering algorithm comparison and improvement in leaf counting techniques [7].

The continuous evolution of leaf counting methodologies underscores the importance of interdisciplinary collaboration between computer vision researchers, plant scientists, and agricultural experts. Future research directions may focus on refining deep learning models, addressing challenges such as occlusion and noise, and exploring novel techniques for multi-scale leaf counting [8].

Chapter 3
Methodology

3.1 Dataset Description

The dataset used in this project is Sorghum image data taken from a previously published plant phenotyping dataset. This dataset consisted of 27,770 images collected from 343 unique sorghum plants representing 295 inbred lines from the sorghum association panel. Images were photographed from 26 July to 31 Aug, 2017, over a period of 37 days spanning vegetative and reproductive development for the majority of genotypes in the population. On each imaging date, sorghum plants were photographed from five different viewing angles, including 0°, 36°, 72°, 108°, and 144°.

3.2 Technologies Used

Technology	Description
Python	Programming language used for coding the project.
Keras	Deep learning library for building and training neural networks.
Pandas	Python library for data manipulation and analysis, used for handling data from Excel files.
NumPy	Library for numerical computing, used for array operations and mathematical functions.
OpenCV	Open-source computer vision library, used for image processing tasks.
TensorFlow	Deep learning framework, used as a backend for Keras and for various machine learning tasks.
Scikit-learn	Machine learning library, used for data splitting and performance evaluation.
Matplotlib	Library for creating static, animated, and interactive visualizations in Python.

Table 3.2.1 Techstack used

3.3 Convolutional Neural Network (CNN) Architecture

The Convolutional Neural Network (CNN) utilized in the project is a deep learning architecture specifically designed for image classification and feature extraction tasks.

1. Convolutional Layers

- The CNN consists of multiple convolutional layers, each responsible for learning and extracting features from the input images.
- Convolutions involve sliding a filter (also known as a kernel) over the input image to perform element-wise multiplication and aggregation, resulting in feature maps that capture spatial patterns.
- In the provided code, Conv2D layers with varying filter sizes (e.g., (3, 3)) are employed to convolve the input images and extract hierarchical features.

2. Activation Functions

- Rectified Linear Unit (ReLU) activation functions are applied after each convolutional layer to introduce non-linearity and enable the network to learn complex relationships within the data.
- ReLU activation functions set negative values to zero while preserving positive values, aiding in feature representation and model training.

3. Pooling Layers

- Max-pooling layers are interspersed between convolutional layers to downsample feature maps and reduce spatial dimensions.
- Max-pooling operates by selecting the maximum value from a predefined window, effectively preserving the most significant features while discarding redundant information.
- Pooling layers help in achieving translation invariance, making the model more robust to variations in object position and scale.

4. Flattening Layer

- Following the convolutional and pooling layers, a flattening layer is employed to reshape the 2D feature maps into a 1D feature vector.
- This transformation facilitates the integration of spatial information and prepares the data for input to the dense (fully connected) layers.

5. Dense Layers

- Dense layers are responsible for feature integration and non-linear transformations, enabling the network to learn complex mappings between input features and output labels.
- In the provided code, multiple dense layers with varying numbers of neurons (e.g., 128, 64, 32) are utilized to capture and encode higher-level abstractions from the flattened feature vector.
- The output layer comprises a single neuron, representing the predicted leaf count, with no activation function applied, as leaf count is treated as a regression task.

6. Regularization

- L1 regularization is applied to the dense layers to mitigate overfitting by penalizing large weight coefficients.
- Regularization techniques help in improving the model's generalization performance by discouraging overly complex representations and reducing the risk of memorizing noise in the training data.

3.4 Implementation

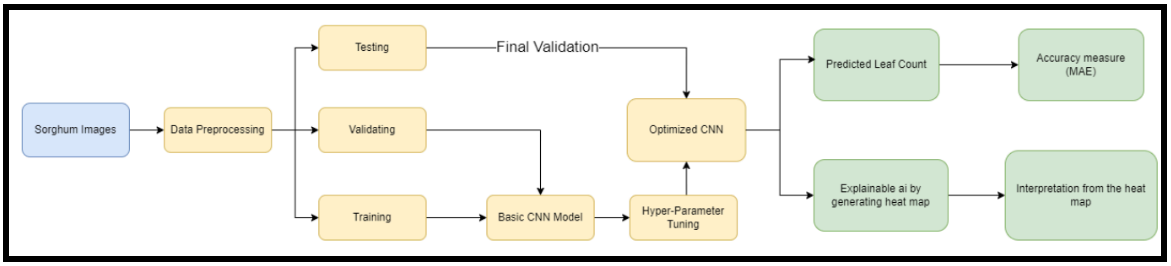


Fig.3.4.1 Block diagram

1. Data Collection and Preprocessing:

- Data collection commenced with the extraction of annotations containing leaf counts from the "sorghum_leaf_number.xlsx" Excel file.
- Images corresponding to the annotations were sourced from the "sorghum_leaf_number" folder, ensuring alignment between image data and leaf count labels.
- Each image underwent preprocessing using the `preprocess_image` function, which included loading, resizing to (128, 128) pixels, and normalizing pixel values to the range [0, 1].

2. Dataset Splitting:

- The dataset was divided into three subsets: training, validation, and testing.
- The `train_test_split` function from sklearn was utilized to split the data, with 80% allocated to training, 10% to validation, and 10% to testing.

3. CNN Architecture Design:

- A Convolutional Neural Network (CNN) architecture was meticulously crafted using the Keras Sequential API.
- The architecture comprised four convolutional layers, each followed by a rectified linear unit (ReLU) activation function.

- Three max-pooling layers were strategically placed to downsample feature maps and reduce computational complexity.
- A flattening layer was employed to convert the 2D feature maps into a 1D feature vector.
- Three dense layers were incorporated for feature integration and non-linear transformations.
- The output layer consisted of a single neuron for leaf count prediction.

4. Integration of GradCAM:

- GradCAM (Gradient-weighted Class Activation Mapping) was integrated into the CNN architecture to provide visual explanations for model predictions.
- GradCAM highlights regions of the input images that contribute most significantly to the predicted leaf count, offering interpretability and insights into model decision-making.

5. Hyperparameter Tuning and Model Compilation:

- Key hyperparameters, including learning rate, optimizer, loss function, and regularization, underwent meticulous tuning.
- L1 regularization was applied to the dense layers of the CNN model to mitigate overfitting.
- The model was compiled using the Adam optimizer and mean squared error loss function.
- A lower learning rate (0.0001) was chosen to facilitate smoother convergence during training.

6. Model Training:

- Training commenced with the CNN model being trained on the training set for 50 epochs.

- A batch size of 32 was employed to balance computational efficiency and model convergence.

7. Model Evaluation:

- The trained model's performance was evaluated on the testing set using mean absolute error (MAE) as the primary evaluation metric.
- Root mean squared error (RMSE) was additionally computed to gauge overall model performance.

8. Results Visualization:

- Predicted vs. actual values were visually compared through scatter plots to assess model performance and identify any discrepancies.
- The distribution of mean absolute error (MAE) was visualized using histograms to gain insights into prediction accuracy.
- Learning curves plotting training and validation loss over epochs were generated to analyze model convergence and potential overfitting.

9. Model Saving:

- Upon successful training and evaluation, the trained CNN model was saved as "leaf_counting_model.h5" for future deployment and use.

Chapter 4
Results

4.1 Test Mean Absolute Error: 0.5642

The mean absolute error (MAE) measures the average magnitude of errors between predicted and actual values. In this case, a MAE of 0.5642 indicates that, on average, the model's predictions deviate from the actual values by approximately 0.5642 units. A lower MAE suggests better accuracy and performance of the model in predicting plant phenotyping attributes.

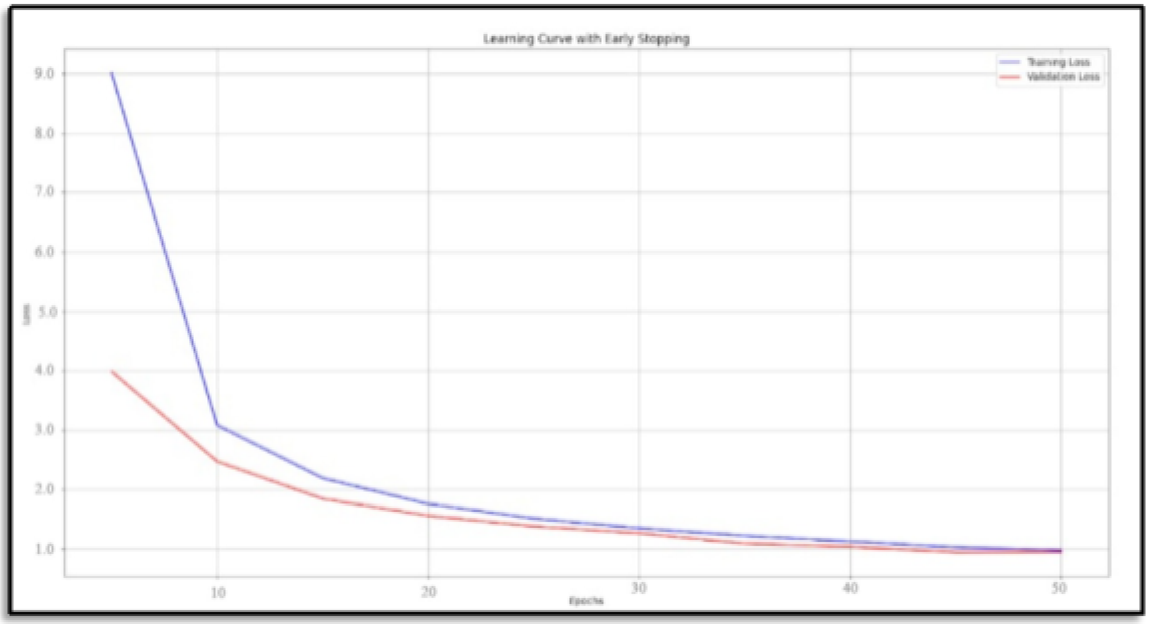
4.2 Test Root Mean Squared Error: 0.6203

The root mean squared error (RMSE) is another measure of the model's prediction accuracy, which penalizes larger errors more heavily than smaller ones. An RMSE of 0.6203 implies that, on average, the model's predictions deviate from the actual values by approximately 0.6203 units. Similar to MAE, a lower RMSE indicates better performance of the model in capturing the variability in plant phenotyping attributes.

4.3 Final Loss: 0.945

The final loss, often referred to as the validation loss or test loss, represents the overall discrepancy between the model's predictions and the actual values. A loss of 0.945 suggests that the model's performance in minimizing prediction errors, as measured by the chosen loss function, reached this value after training. Lower loss values indicate better model performance, as the model learns to make predictions that are closer to the ground truth.

4.4 Learning Curve

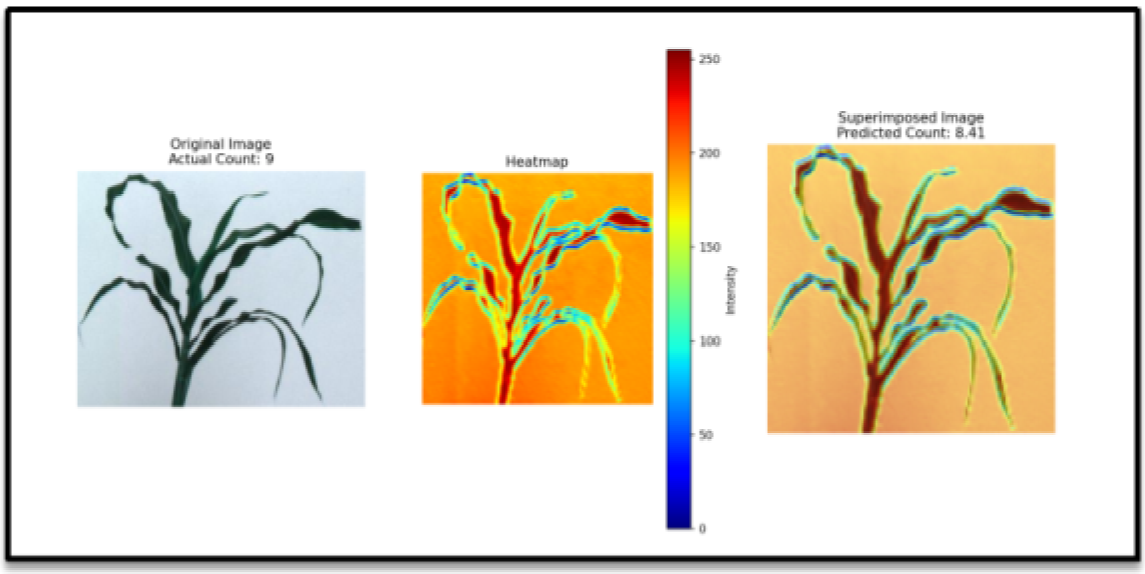


4.4.1 Learning Curve

The learning curve graph depicts the progression of the model's performance over the course of training, typically across multiple epochs.The x-axis represents the number of epochs, while the y-axis represents the loss (e.g., mean squared error, categorical cross-entropy).The learning curve graph illustrates how the training and validation losses change as the model iterates through training epochs.A convergence of the training and validation losses indicates that the model has effectively learned from the training data and is not overfitting.

4.5 Heatmap Analysis

The heatmap provides a visual representation of the regions of interest identified by the model within an image.Each pixel in the heatmap corresponds to a specific area in the image, with varying intensity indicating the degree of importance assigned by the model to that particular region.Brighter areas in the heatmap suggest higher importance or relevance to the model's predictions, while darker areas indicate lower importance.By analyzing the heatmap, we can gain insights into which parts of the image the model focuses on when making predictions related to plant phenotyping attributes.The Grad-CAM heatmap analysis provides insights into the areas of the leaf image that are most relevant for the model's prediction of leaf count. By visualizing the intensity of activation within the neural network's convolutional layers, we can interpret different regions of the image based on their color representation in the heatmap.



4.5.1 GradCam Output

1. **Red Areas:**

The red areas within the leaf represent regions of high intensity or activation. These areas are where the model is focusing its attention and where the features most relevant to leaf count are detected.

Typically, the central regions of the leaf exhibit a higher intensity of red coloration, indicating that the model considers these regions crucial for its prediction.

2. **Blue Areas:**

Blue areas, often observed along the edges of the leaf, represent regions of low intensity or activation.
The presence of blue along the leaf edges suggests that the model is assigning less importance to these areas when determining leaf count.
This could be because the edges may contain less distinct features or may vary more in appearance across different images, making them less reliable for counting leaves.

3. **Yellow Areas:**

Yellow areas, which may appear at the tips or extremities of the leaves, indicate moderate activation levels.While not as intense as the red regions, yellow areas still contribute to the model's prediction to some extent.The presence of yellow at the tips could suggest that certain features or patterns at the ends of the leaves are somewhat indicative of leaf count.

4. **Orange Background:**

The orange background surrounding the leaf serves as a reference point for contrasting the activation levels within the leaf.The absence of strong coloration (red, blue, or yellow) in the background indicates that the model is not focusing its attention on these regions for leaf counting.

Chapter 5
Conclusion

5.1 Summary of key findings and conclusions drawn from the project.

Here are the key findings and conclusions drawn from the project, based on the introduction and results:

1. **Importance of Plant Phenotyping and Leaf Counting:**

Plant phenotyping is crucial for understanding plant development and optimizing crop production. Leaf count, a fundamental aspect of plant phenotyping, correlates with essential plant traits such as height and biomass yield.
Manual leaf counting is labor-intensive and prone to errors, highlighting the need for automated solutions.

2. **Performance Metrics:**

Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) were used to evaluate prediction accuracy.A final loss value of 0.945 indicated the overall performance of the model in minimizing prediction errors.

3. **Learning Curve Analysis:**

The learning curve depicted the progression of the model's performance over training epochs .Convergence of training and validation losses indicated effective learning without overfitting.

4. **Heatmap Analysis (GradCAM):**

GradCAM provided insights into regions of interest within leaf images for leaf count prediction.Red areas represented regions of high activation, typically found in central leaf regions.Blue areas indicated low activation, often observed along leaf edges.

Yellow areas showed moderate activation, typically at leaf tips.
The absence of strong coloration in the background indicated areas not relevant for leaf counting.

5. **Potential Impact:**

The system has the potential to streamline agricultural research processes and improve crop management practices.Quick and non-invasive assessments of plant traits contribute to informed farming decisions and global food sustainability efforts.

5.2 Discussion

Our comparison with the referenced research highlights key differences in how we approached model development for sorghum leaf detection. While the referenced study used a model pre-trained on maize data, we opted to train our model from scratch using a large dataset of 27,770 sorghum images from 5 different angles.

Their method, while efficient, might not fully capture the unique features of sorghum leaves due to the differences between maize and sorghum. This is reflected in their reported RMSE of 1.14 ± 0.05, indicating some inaccuracies in leaf detection.

In contrast, our approach, although more resource-intensive, led to a lower RMSE value of 0.623. Training our model specifically on sorghum data allowed us to better understand the intricacies of sorghum leaves, resulting in improved accuracy.

However, it's important to consider practical implications. While our method offers higher accuracy, it requires more resources and time. On the other hand, the referenced approach is more practical for scenarios with limited resources, despite its slightly lower accuracy.

Chapter 6
Acknowledgements

We extend our sincere gratitude to our Head of the Department, Mrs. Nilkamal More, for her unwavering support and encouragement throughout the duration of this project. Her guidance and leadership have been invaluable in shaping our efforts towards achieving our goals.

We would also like to express our heartfelt thanks to our esteemed Mentor, Professor Suchtira Patil, for her expertise, mentorship, and continuous motivation. Her insightful inputs and strategic direction have played a pivotal role in steering this project towards success.

Additionally, we extend our appreciation to all faculty members and staff who provided assistance and resources, contributing to the fruition of this endeavor.

Lastly, we acknowledge the support of our peers and colleagues for their collaboration and constructive feedback, which enriched the project experience.

Thank you all for your invaluable contributions.

Chapter 7

References

[1] Yotam Itzhaky, G. Farjon, Faina Khoroshevsky, Alon Shpigler, and Aharon Bar-Hillel, "Leaf counting: Multiple scale regression and detection using deep CNNs.," p. 328, Jan. 2018.

[2] Y.-L. Tu, W.-Y. Lin, and Y.-C. Lin, "Toward automatic plant phenotyping: starting from leaf counting," *Multimedia tools and applications*, vol. 81, no. 9, pp. 11865–11879, Jan. 2022, doi: https://doi.org/10.1007/s11042-021-11886-w.

[3] Y. Tu, W. Lin, and Y.-C. Lin, "Automatic Leaf Counting Using Improved YOLOv3," Nov. 2020, doi: https://doi.org/10.1109/is3c50286.2020.00058.

[4] C. Miao, A. Guo, A. M. Thompson, J. Yang, Y. Ge, and J. C. Schnable, "Automation of leaf counting in maize and sorghum using deep learning," *The Plant Phenome Journal*, vol. 4, no. 1, Jan. 2021, doi: https://doi.org/10.1002/ppj2.20022.

[5] M. Buzzy, V. Thesma, M. Davoodi, and J. Mohammadpour Velni, "Real-Time Plant Leaf Counting Using Deep Object Detection Networks," *Sensors*, vol. 20, no. 23, p. 6896, Dec. 2020, doi: https://doi.org/10.3390/s20236896.

[6] M. Deb, K. G. Dhal, A. Das, A. G. Hussien, L. Abualigah, and A. Garai, "A CNN-based model to count the leaves of rosette plants (LC-Net)," *Scientific Reports*, vol. 14, no. 1, p. 1496, Jan. 2024, doi: https://doi.org/10.1038/s41598-024-51983-y.

[7] R. Pieruschka and U. Schurr, "Plant Phenotyping: Past, Present, and Future," *Plant Phenomics*, vol. 2019, pp. 1–6, Mar. 2019, doi: https://doi.org/10.34133/2019/7507131.

[8] X. Xie, Y. Ge, H. Walia, J. Yang, and H. Yu, “Leaf-Counting in Monocot Plants Using Deep Regression Models,” *Sensors*, vol. 23, no. 4, p. 1890, Feb. 2023, doi: https://doi.org/10.3390/s23041890.

Chapter 8
Appendices

9.1 Additional information, data, or supplementary materials.

Dataset Description:The dataset used in this project is Sorghum image data taken from a previously published plant phenotyping dataset by Miao et al. (2020). This dataset consisted of 27,770 images collected from 343 unique sorghum plants representing 295 inbred lines from the sorghum association panel. Images were photographed from 26 July to 31 Aug, 2017 over a period of 37 days spanning vegetative and reproductive development for the majority of genotypes in the population. On each imaging date, sorghum plants were photographed from five different viewing angles, including 0°, 36°, 72°, 108°, and 144°.

GradCAM Explainable AI Technique:

GradCAM (Gradient-weighted Class Activation Mapping) is an explainable AI technique used to visualize and interpret CNN predictions by highlighting the regions of input images that contribute most to the model's decision-making process. GradCAM is integrated with the CNN leaf counting model to provide insight into which features or regions of plant images are most influential in predicting leaf counts accurately. GradCAM generates heat maps overlaid on input images, indicating the areas where the model focuses its attention when making leaf count predictions. By visualizing the relevant features identified by GradCAM, we gain transparency into the decision-making process of our leaf counting model, enhancing its interpretability and trustworthiness.