**Capstone Project Concept Note and Implementation Plan**

**Project Title: AI-Powered Early Detection of Crop Diseases for Smallholder Farmers**

**Team Members**

1. Mubarek Jemal
2. Nahom Tibebu
3. Natnael Meles

Table of Contents

[**Concept Note** 1](#_Toc195900827)

[**1.** **Project Overview** 1](#_Toc195900828)

[**2.** **Objectives** 2](#_Toc195900829)

[**3.** **Background** 2](#_Toc195900830)

[**3.1.** **The Significance of the research** 4](#_Toc195900831)

[**4.** **Methodology** 4](#_Toc195900832)

[**4.1.** **Model Architecture and Algorithm Selection** 5](#_Toc195900833)

[**4.2.** **Data Preprocessing and Augmentation** 5](#_Toc195900834)

[**4.3.** **Model Training and Evaluation** 6](#_Toc195900835)

[**4.4.** **Model Deployment** 7](#_Toc195900836)

[**4.5.** **Explain ability and User-Centered Features** 7](#_Toc195900837)

[**5.** **Architecture Design** 8](#_Toc195900838)

[**6.** **Data Sources** 9](#_Toc195900839)

[**7.** **Literature Review** 9](#_Toc195900840)

[**Implementation Plan** 10](#_Toc195900841)

[**1.** **Technology Stack** 10](#_Toc195900842)

[**2.** **Project Timelines** 11](#_Toc195900843)

[**3.** **Project Milestones** 11](#_Toc195900844)

[**4.** **Challenges and Mitigations** 12](#_Toc195900845)

[**i.** **Data Quality Issues** 12](#_Toc195900846)

[**ii.** **Model Performance & Generalizability** 13](#_Toc195900847)

[**5.** **Ethical Considerations** 13](#_Toc195900848)

[**References** 14](#_Toc195900849)

# **Concept Note**

# **Project Overview**

Agriculture remains the backbone of economic development and food security in many developing regions, particularly in sub-Saharan Africa, where smallholder farmers dominate food production. Smallholder farmers, who constitute the majority of agricultural producers in regions such as sub-Saharan Africa, face a numerous of challenges that delay productivity and threaten their livelihoods. One of the most pressing issues is the inability to detect crop diseases early and accurately. Crop diseases not only reduce yield quality and quantity but also jeopardize food availability and household income (Foysal, 2024). While advanced agricultural technologies exist to assist in disease detection and management, they are often inaccessible to smallholder farmers due to cost, lack of infrastructure, or limited technical knowledge.

In recent years, artificial intelligence (AI), particularly deep learning techniques, has shown remarkable potential in solving complex problems across various domains, including agriculture. Image-based disease detection using AI-powered models such as Convolutional Neural Networks (CNNs) has proven to be highly accurate in laboratory settings (Asani E. O., 2023). However, there remains a gap in creating practical, scalable solutions that can be deployed in the field by resource-constrained farmers. This research aims to bridge that gap by developing a lightweight, mobile-compatible AI model that can detect crop diseases from images of plant leaves. By integrating such technology into accessible mobile applications, smallholder farmers can be equipped with a powerful tool to make timely and informed decisions, thereby reducing losses and improving agricultural productivity (Foysal, 2024).

To bridge this gap, our project proposes the development of a lightweight, mobile-compatible AI model that leverages deep learning techniques, particularly Convolutional Neural Networks (CNNs), to identify crop diseases from leaf images. By deploying the model through a user-friendly mobile application, farmers can receive real-time disease diagnosis without needing internet access, enabling timely interventions and reducing reliance on costly agricultural experts.

This research not only aligns with the goals of precision agriculture but also contributes directly to multiple Sustainable Development Goals (SDGs), including No Poverty (SDG 1), Zero Hunger (SDG 2), and Good Health and Well-being (SDG 3) (Asani E. O., 2023). The proposed solution will enable farmers to take early preventive actions against crop diseases, decrease dependence on costly agricultural experts, and minimize the usage of harmful pesticides. Overall, this research envisions a future where AI becomes a catalyst for agricultural transformation, especially for underserved farming communities.

# **Objectives**

The primary objective of this project is to develop and deploy an AI-powered, mobile-compatible solution that enables smallholder farmers to detect crop diseases early and accurately using images of plant leaves. The specific objectives are as follows:

* Design and Train a Deep Learning Model for Crop Disease Detection
* Ensure Mobile Compatibility and Offline Functionality
* Develop an intuitive mobile app interface that allows farmers to capture or upload leaf images and receive instant diagnostic feedback.
* Reduce unnecessary pesticide use by helping farmers target treatments based on accurate diagnoses.

# **Background**

In many developing countries, agriculture remains the foundation of both the economy and food security, particularly in rural communities where a significant portion of the population relies on small-scale farming. Smallholder farmers, especially those in sub-Saharan Africa, play a pivotal role in food production but often operate under resource-constrained conditions. Among the various challenges they face, crop diseases pose a major threat, frequently leading to reduced yields, lower incomes, and increased food insecurity (Foysal, 2024). The early detection and accurate diagnosis of crop diseases are crucial for mitigating these impacts, but traditional methods of identification such as visual inspection by agronomists or lab-based testing are often inaccessible or unaffordable for these farmers (Khan, 2020).

In recent years, researchers and development organizations have made strides toward addressing this gap through digital agricultural tools. For instance, the PlantVillage initiative developed a public image dataset and mobile-based disease detection system to assist farmers in recognizing plant diseases using visual data. Similarly, (Ramcharan, 2019) deployed a Convolutional Neural Network (CNN)-based model for cassava disease detection directly on Android smartphones, demonstrating the feasibility of offline, real-time diagnosis in rural areas. However, many existing solutions still fall short of meeting the unique needs of smallholder farmers. Some models are trained on clean, lab-collected datasets and struggle with generalizing to noisy, real-world conditions, while others lack intuitive user interfaces or multilingual support, limiting widespread adoption (Too, 2019); (Awan, 2021).

Crop diseases can lead to significant yield losses, threaten household food security, and reduce income for rural farmers who already operate with limited resources. Studies show that up to 40% of global crop yields are lost annually to pests and diseases, with disproportionate effects on developing countries where access to modern farming tools and professional agronomic support is scarce. Traditional disease detection methods typically require expert diagnosis, laboratory testing, or expensive equipment—resources that are inaccessible to most smallholder farmers due to financial, logistical, and infrastructural barriers. In some cases, delayed or inaccurate diagnosis forces farmers to apply chemical pesticides indiscriminately, further harming the environment and human health.

This is where machine learning—particularly deep learning approaches like CNNs—becomes essential. CNNs have demonstrated remarkable accuracy in identifying plant diseases from leaf images by learning complex patterns and features directly from raw data (Mohanty, 2016); (Ferentinos, 2018). Unlike traditional image analysis methods that rely on handcrafted features, CNNs can automatically detect and differentiate between subtle disease symptoms, even in varied environmental conditions. Moreover, CNN models can be optimized for deployment on mobile devices through tools such as TensorFlow Lite, allowing real-time, offline disease detection that is both scalable and accessible for farmers with limited digital infrastructure (Lu, 2021)

Additionally, the integration of CNN-based models into mobile applications not only democratizes expert-level diagnostics but also encourages timely intervention and reduces the misuse of chemical pesticides—thereby promoting environmentally sustainable practices and improving human health outcomes (Asani E. O., 2023). Such solutions align closely with the Sustainable Development Goals, particularly SDG 1 (No Poverty), SDG 2 (Zero Hunger), and SDG 3 (GoodHealth and Well-being). By leveraging machine learning in a mobile-compatible format, this project aims to overcome the limitations of previous tools and bring impactful, low-cost innovation directly into the hands of smallholder farmers.

## **The Significance of the research**

The significance of this research lies in its potential to create practical, impactful change for millions of smallholder farmers who are at the frontline of food production. Crop diseases are a major constraint to agricultural productivity, often resulting in severe yield reductions and economic hardships for small-scale farmers (Asani, 2023). The inability to detect and manage these diseases in a timely manner worsens poverty, increases food insecurity, and continues the cycle of underdevelopment in rural communities (Khan, 2020). Traditional disease management approaches rely on the intervention of agronomists or the use of sophisticated laboratory equipment, both of which are inaccessible to most smallholder farmers due to logistical and financial limitations.

By leveraging the power of AI and mobile technology, this research seeks to democratize access to crop disease diagnostics. A mobile-compatible AI model can bring expert-level analysis directly into the hands of farmers, allowing them to take quick and effective action. Such empowerment is critical for enhancing their resilience to agricultural risks and for improving overall productivity and sustainability. Furthermore, early detection can reduce the indiscriminate use of chemical pesticides, contributing to better environmental and human health. In this way, the research supports not only economic development and food security but also the promotion of environmentally sustainable agricultural practices.

# **Methodology**

This project utilizes a deep learning approach to develop a mobile-compatible crop disease detection tool that empowers smallholder farmers with timely, accurate, and accessible diagnostics. The methodology integrates several key phases: model selection, data preprocessing, training and evaluation, mobile deployment, and explain-ability features each tailored to address real-world agricultural constraints.

## **Model Architecture and Algorithm Selection**

The primary machine learning approach used in this project is **Convolutional Neural Networks (CNNs)**, which are well-suited for image classification tasks due to their ability to extract hierarchical spatial features from images. After evaluating multiple architectures, we have selected **MobileNetV2** as the base model. MobileNetV2 is a lightweight, efficient CNN architecture specifically designed for mobile and embedded vision applications. It offers a favorable balance between accuracy and computational cost, making it ideal for deployment on low-end smartphones (Sandler, 2018).

To train the model, we will use **supervised learning** techniques. The dataset will consist of labeled images of healthy and diseased plant leaves, sourced primarily from the **PlantVillage dataset**, supplemented by field images collected using smartphones to ensure real-world diversity. The model will be trained to classify images into disease categories based on the crop type.

## **Data Preprocessing and Augmentation**

To enhance the model's generalizability and robustness under varied field conditions, several **data preprocessing and augmentation techniques** will be applied. These include resizing all images to 224x224 pixels (standard input for MobileNetV2), normalization of pixel values, and augmentation techniques such as rotation, flipping, zooming, and brightness adjustments. These augmentations will simulate real-world variability and address class imbalances in the dataset.

#### **Data Preprocessing**

The preprocessing phase involves transforming raw images into a standardized format that the model can interpret effectively. For this project, all images are first **resized to 224×224 pixels**, which aligns with the input requirements of the MobileNetV2 architecture. Standardizing the image size ensures consistency across the dataset and reduces the computational load during training and inference.

Next, **pixel values will be normalized** typically scaled to a range between 0 and 1 or adjust to have a zero mean and unit variance. Normalization helps the model converge faster during training by ensuring that all input features contribute equally to the learning process. Additionally, any **noisy, mislabeled, or low-resolution images** will be manually review and clean from the dataset to enhance overall data quality. To increase model performance under diverse field conditions, preprocessing also includes **image enhancement techniques** such as **contrast adjustment** and **background noise reduction.** These methods help highlight key disease-related features (e.g., leaf spots, discoloration) while minimizing irrelevant background elements that could confuse the model.

#### **Data Augmentation**

While preprocessing focuses on cleaning and standardizing the data, **augmentation aims to artificially increase dataset size and diversity** especially important for underrepresented disease classes and field images. Augmentation also addresses overfitting, where a model performs well on training data but fails to generalize to new, unseen data.

Several **real-time augmentation techniques** will be applied to each training batch using image augmentation libraries such as Keras’ Image Data Generator or Albumentations. These techniques include:

* **Horizontal and vertical flipping**: Reversing the image to mimic variations in how farmers might photograph leaves.
* **Zooming and cropping**: Slight zoom-in/out operations to replicate images taken at different distances.
* **Brightness and contrast shifts**: Simulating different lighting conditions in the field (e.g., shade, direct sunlight).

## **Model Training and Evaluation**

The model training process is based on **supervised learning**, where the system learns to map input images of leaves to predefined disease labels using annotated data. Since training a deep learning model from scratch requires massive datasets and computational resources, we apply **transfer learning** a powerful technique that uses a model pre-trained on a large benchmark dataset (i.e. ImageNet) as a starting point.

Specifically, we employ **MobileNetV2**, a Convolutional Neural Network (CNN) architecture designed for mobile and embedded vision applications. MobileNetV2 is pre-trained on large of general images and is then **fine-tuned** on our custom dataset, which includes labeled images of healthy and diseased plant leaves. Fine-tuning allows the model to adapt its learned features to the specific task of crop disease classification, significantly improving performance with less training data and computational effort.

To optimize training performance, we use the **Adam optimizer**, known for its adaptive learning rate and stability during training. The **categorical cross-entropy loss function** is used because the task involves multi-class classification each image belongs to exactly one class (i.e. healthy, tomato early blight, apple scab, etc.). The training will use **cross-entropy loss** and **Adam optimizer,** and the model will be evaluated using metrics such as **accuracy, precision, recall, and F1-score.** A **confusion matrix** will also be used to visualize the model's performance across different classes.

## **Model Deployment**

Model deployment is the final and most crucial step in translating the trained AI model into a usable tool for smallholder farmers. While model training occurs in a controlled, high-compute environment, deployment ensures that the model can be accessed in **real-world, low-resource environments,** especially via **mobile devices** that do not always have access to the internet or cloud infrastructure. In this project, deployment focuses on **embedding the trained model into an Android mobile application** using lightweight tools and frameworks optimized for performance, speed, and usability.

## **Explain ability and User-Centered Features**

To build trust and encourage adoption among farmers, the application will include **explainable AI (XAI)** features such as **Grad-CAM (Gradient-weighted Class Activation Mapping)**, which visualizes the parts of the leaf image the model used for prediction. Additionally, the app will support **multilingual interfaces**, offline functionality, and voice-based input/output where feasible, ensuring accessibility for users with low digital literacy.

# **Architecture Design**

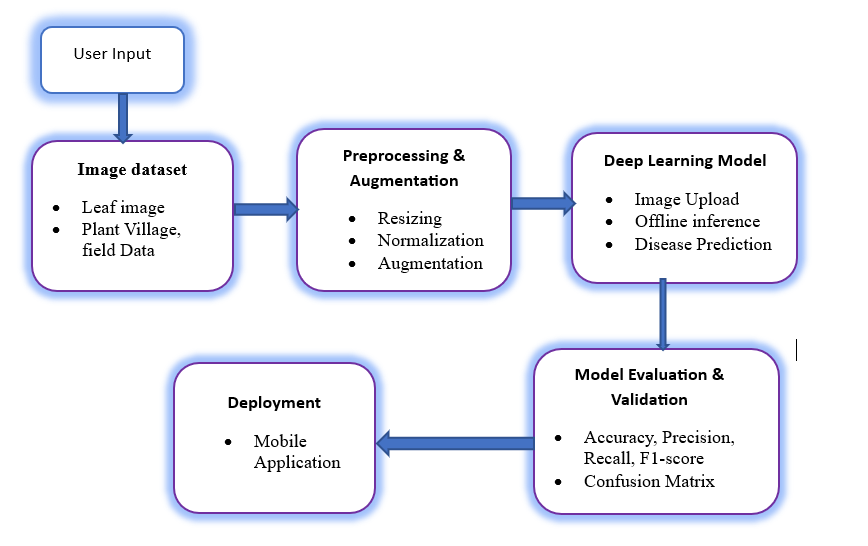


Figure 1 Architecture design diagram

The user captures an image of the crop leaf through the mobile app’s camera or selects one from the photo gallery. Ensures real-time data collection directly from the farm. Handles basic input validation (ensuring an image is selected). Then, ensuring the image dimensions match the model input requirements (i.e. 224×224 pixels) and scales pixel values between 0 and 1 to enhance model performance then apply techniques such as rotation, zoom, and brightness variation to improve the model’s robustness.

Uses we use a deep learning model with a lightweight MobileNetV2 model fine-tuned and converted to TensorFlow Lite (.tflite) format for mobile compatibility. Performs on-device inference to classify the input image into one of several disease categories or labels it as healthy. Works offline without needing cloud servers, making it ideal for rural use cases. Finally, we Converts predictions into actionable insights for the user and deploy the model that displays disease name, Confidence score, Visual feedback and make results accessible even to users with low literacy.

# **Data Sources**

The primary data source for this project is the **PlantVillage dataset**, a publicly available and widely recognized repository containing over **20,654 high-resolution images** of healthy and diseased plant leaves across 14 different crop species and 16 disease categories. To enhance real-world applicability and improve model generalizability, the dataset is further **augmented with field-collected images** taken under diverse conditions such as varying lighting, angles, and backgrounds using smartphone cameras. This dual-source approach ensures that the training data represents both controlled and natural environments, aligning with the mobile deployment goals of the project. Preprocessing steps include **resizing images to 224×224 pixels** for compatibility with MobileNetV2, **normalization** of pixel values, and **data augmentation techniques** such as flipping, rotation, and brightness adjustment to address class imbalances and improve model robustness. This rich and diverse dataset serves as the foundation for developing a lightweight, reliable AI model for crop disease detection, specifically tailored for deployment in rural agricultural settings.

# **Literature Review**

The literature reviewed for this project highlights the significant progress made in using **Convolutional Neural Networks (CNNs)** for plant disease detection, with studies by(Mohanty, 2016)and(Ferentinos, 2018)demonstrating accuracies above 99% in controlled environments. However, a recurring limitation across these works is their reduced effectiveness in real-world conditions, where image quality varies due to factors like lighting, background clutter, and inconsistent leaf presentation. While researchers such as (Ramcharan, 2019) have made strides in deploying lightweight CNN models on mobile devices for offline disease detection, most models still struggle to generalize well outside lab settings. The literature also emphasizes the importance of **image preprocessing and augmentation** like rotation, flipping, and contrast adjustments for improving robustness (Sandler, 2018). Moreover, social factors such as **user adoption, digital literacy, and local language support** are shown to be critical for successful implementation among smallholder farmers. These insights collectively inform this project’s approach to building a **mobile-compatible, accessible, and explainable AI solution** tailored for rural agricultural contexts.

# **Implementation Plan**

# **Technology Stack**

To effectively develop a mobile-compatible AI model for early crop disease detection, this project utilizes a carefully selected set of technologies and tools. These are chosen to ensure the system is accurate, lightweight, accessible, and suitable for deployment in low-resource environments, particularly for smallholder farmers.

1. **Programming Languages**

Python will be use as core programming language for model development, data preprocessing, training, and evaluation and for developing the native Android mobile application and integrating the AI model via TensorFlow Lite we use Java or Kotlin.

1. **Libraries and Packages**

* TensorFlow & TensorFlow Lite: For building and training the deep learning model (MobileNetV2), then converting and optimizing it for mobile deployment.
* Keras: High-level neural network API (built on TensorFlow) used for model training and architecture definition.
* OpenCV: For basic image preprocessing (e.g., resizing, color normalization).
* Matplotlib & Seaborn: For visualizing model performance and evaluation metrics.
* Keras ImageDataGenerator: For data augmentation such as flipping, rotation, and brightness adjustment.
* Grad-CAM (Gradient-weighted Class Activation Mapping): For adding explain-ability features to the AI model (visual heatmaps).

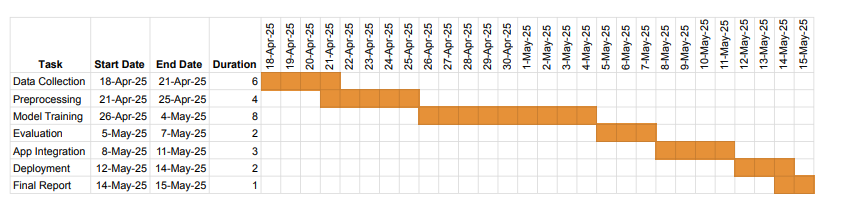
1. **Deployment Tools and Platforms**

* **TensorFlow Lite Converter**: Converts trained TensorFlow models into .tflite format for mobile inference.
* **Android Studio**: Integrated development environment (IDE) used to build and test the Android app.

1. **Hardware Requirements**

* **Development Machine**: At least 8 GB RAM and GPU support for model training and testing.
* **Smartphones (Android)**: Mid-range devices used for model testing and deployment; must support TensorFlow Lite(optional).

# **Project Timelines**



| **Task** | **Mubarek ✔** | **Nahom ✔** | **Natnael ✔** |
| --- | --- | --- | --- |
| Data Collection & Cleaning | ✔ | ✔ |  |
| Data Preprocessing & Augmentation |  | ✔ | ✔ |
| Model Selection & Architecture | ✔ |  | ✔ |
| Model Training & Tuning | ✔ | ✔ |  |
| Model Evaluation & Testing | ✔ |  |  |
| Deployment | ✔ |  | ✔ |
| Final Report & Documentation | ✔ | ✔ | ✔ |

# **Project Milestones**

**Milestone 1: Dataset Preparation Completion**

Successful collection, cleaning, and organization of all relevant image datasets from PlantVillage and field sources. Basic exploratory data analysis (EDA) completed to check class balance, image quality, and metadata integrity.

**Milestone 2: Data Augmentation & Preprocessing Finalizing**

All image preprocessing steps (resizing, normalization) and augmentation techniques (rotation, flipping, contrast adjustments) implementing and validating.

**Milestone 3: Model Architecture Selected and Initialized**

Final decision and setup of the deep learning architecture (MobileNetV2) using transfer learning, including input/output layers and loss function configuration.

**Milestone 4: Model Training Completed**

Model trained with optimized parameters and validated using holdout test sets. Overfitting managed and performance tuned.

**Milestone 5: Model Evaluation and Deployment Completed**

Model evaluation using different metrics and then deploying the developed model.

**Milestone 6: Final Report and Documentation Submitted**

Full project report completed, including literature review, methodology, results, deployment steps, and future work recommendations.

# **Challenges and Mitigations**

1. **Data Quality Issues**

**Challenge:**

* Inconsistent image quality due to variable lighting, angles, and backgrounds in field-collected data.
* Class imbalance (some diseases underrepresented).
* Mislabeling or noise in open-source datasets like PlantVillage.

**Mitigation Strategies:**

* Perform data cleaning and manual inspection to remove low-quality or mislabeled images.
* Apply data augmentation techniques (rotation, flipping, brightness adjustment) to increase diversity and balance classes.
* Collect additional field images or generate synthetic samples using GANs to fill gaps in underrepresented categories.
* Use normalization and standard resizing during preprocessing to maintain input consistency.

1. **Model Performance & Generalizability**

**Challenge:**

* High performance in lab settings may not translate to real-world field conditions.
* Risk of overfitting to clean, controlled images.
* Poor prediction accuracy on images with noise or background clutter.

**Mitigation Strategies:**

* Use transfer learning with MobileNetV2 and fine-tune on both controlled and field datasets.
* Introduce real-world augmentation in training (simulate shadows, blur, varied zoom levels).
* Use cross-validation and holdout sets for robust performance evaluation.
* Regularly monitor confusion matrix and per-class metrics to refine underperforming categories.
* Implement Grad-CAM visualizations to verify model attention and explain decisions to users.

# **Ethical Considerations**

Implementing an AI-based crop disease detection system, especially in vulnerable and rural agricultural communities, requires careful attention to **ethical principles**. While the technology promises to empower smallholder farmers and improve food security, its design and deployment must consider implications related to **data privacy, algorithmic bias, and community impact** to ensure fair, inclusive, and responsible innovation.

# **References**

Asani, E. O. (2023). mPD-APP: a mobile-enabled plant diseases diagnosis application using convolutional neural network toward the attainment of a food secure world. *Front. Artif. Intell*, 6, 1227950. https://doi.org/10.3389/frai.2023.1227950.

Asani, E. O. (2023). mPD-APP: a mobile-enabled plant diseases diagnosis application using convolutional neural network toward the attainment of a food secure world. *Frontiers in Artificial Intelligence*, 6, 1227950. https://doi.org/10.3389/frai.2023.1227950.

Awan, M. J. ( 2021). Smart agriculture: An approach towards the integration of AI and IoT for better crop management. *Sustainability*, 13(8), 4489. https://doi.org/10.3390/su13084489.

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture,*, 145, 311–318. https://doi.org/10.1016/j.compag.2018.01.009.

Foysal, M. A. (2024). Multi-Class Plant Leaf Disease Detection: A CNN-based Approach with Mobile App Integration. *International Journal of Computer Applications Volume 186, No.41*.

Khan, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, 53(8), 5455–5516. https://doi.org/10.100.

Lu, Y. Y. (2021). Intelligent agriculture: Detection and diagnosis of plant diseases using deep learning. *Frontiers in Plant Science*, 12, 690305. https://doi.org/10.3389/fpls.2021.690305.

Mohanty, S. P. ( 2016). Using deep learning for image-based plant disease detection. . *Frontiers in Plant Science*, 7, 1419. https://doi.org/10.3389/fpls.2016.01419.

Ramcharan, A. B. (2019). Deep learning for image-based cassava disease detection. *Frontiers in Plant Science*, 10, 272. https://doi.org/10.3389/fpls.2019.00272.

Sandler, M. H.-C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* , 4510–4520. https://doi.org/10.1109/CVPR.2018.00474.

Too, E. C. (2019). A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*, 161, 272–279. https://doi.org/10.1016/j.compag.2018.03.032.