An AI Tomato Leaf Doctor Using MobileNetV2 and Streamlit: A Lightweight Deep Learning Tool for Farmers

Makepeace Banele Zulu
Department of Statistics and Demography
University of Eswatini
makepeacebanelez@gmail.com

Abstract

Tomato farming in Eswatini faces severe yield losses (up to 70 percent) due to diseases such as late blight (Phytophthora infestans) and bacterial leaf spot (Xanthomonas spp.), caused by climate variability and limited access to localized diagnostic tools. Existing AI solutions, such as Plantix, lack region-specific treatment advice, excluding farmers in low bandwidth areas. This study introduces AI Tomato Leaf Doctor, a Streamlit deployed, two stage MobileNetV2 system pretrained on ImageNet and fine-tuned. From experimental analysis, it is seen that plant leaf disease detection using MobileNetV2 outperforms existing approach in terms of accuracy as well as training time. The first model filters non tomato inputs, other plants or objects, while the second classifies ten tomato conditions (nine diseases and one healthy) using a PlantVillage dataset of 20,000 images (2000 per class). The model achieved 97 percent accuracy. The app provides chemical dosages for locally available fungicides and their market prices in Eswatini Lilangeni (SZL), delivered via a farmer friendly interface. By leveraging MobileNetV2's lightweight design and Streamlit's cloud compatibility, this work bridges the gap between AI innovation and the practical needs of Eswatini's tomato-dependent households.

Keywords: Tomato Disease Detection, MobileNetV2, Streamlit, PlantVillage Dataset, Eswatini Agriculture, Lightweight CNN

1. Introduction

Agriculture still remains the backbone of many developing sub-Saharan African countries, with smallholder farmers playing a crucial role in ensuring food security(1). It is the predominant sector in the economies of most African countries, accounting for between 30 to 40 percent of gross domestic product, and a leading source of jobs for over two-thirds of Africa's population, yet plant diseases threaten livelihoods (2). Eswatini tomato farmers in particular often face significant challenges, including limited access to related technologies, agricultural expertise and timely disease diagnostics, leading to substantial crop losses (3). Late blight, a dangerous disease caused by Phytophthora infestans, causes the most significant losses among them (4).

The rapid adoption of mobile technologies in rural Africa (reaching 80% penetration by 2023 (7)) presented unprecedented opportunities for AI-driven agricultural solutions. However, current implementations face three critical limitations:

Technical constraints: Most deep learning models for plant disease detection (such as ResNet-50, EfficientNet) require substantial computational resources, making them impractical for low-end smartphones prevalent in rural areas; Contextual irrelevance: Existing applications like Plantix and Nuru provide generic treatment recommendations that often reference fungicides unavailable in Eswatini's markets (3); Usability barriers, text heavy interfaces and complex workflows exclude farmers with limited literacy skills, as identified in participatory studies (3).

Recent advances in lightweight convolutional neural networks (CNNs), particularly MobileNetV2 (11), offer promising solutions to these challenges. This paper presents the AI Tomato Leaf Doctor, a novel two-stage system that addresses these limitations through three key innovations:

A dual-model architecture combining a MobileNetV2 filter (to eliminate non-tomato inputs) with a specialized tomato disease classifier, hyper localized outputs including: Chemical dosages for regionally available fungicides, Market prices in Eswatini Lilangeni (SZL) and preventive measures adapted to local farming practices

An icon-driven interface deployed via Streamlit, optimized for: Low-bandwidth environments (under 25MB model size), low-literacy users (visual navigation), mid-range smartphones.

This study builds upon the PlantVillage dataset while addressing its limitations through strategic data augmentation and transfer learning techniques. The system's performance was validated through both technical metrics (97% accuracy, 0.98 F1-score) and field tests with Eswatini farmers, 80% of whom rated the tool as "highly useful" for its localized treatment advice.

This work contributes to the growing field of AI for social good by demonstrating how lightweight deep learning can be tailored to resource-constrained agricultural contexts.

2. Related Works

The field of AI-driven plant disease detection has witnessed significant evolution in recent years, particularly in the development of lightweight convolutional neural networks (CNNs). These advancements, while technically impressive, they often fall short in address the socioeconomic realities faced by African smallholder farmers, creating a critical gap between laboratory performance and field utility. This section examines these developments through three key lenses: technical architectures, practical implementations, and regional adaptations.

2.1. Evolution of Lightweight CNN Architectures

The pioneering work of (5) introduced an enhanced variant incorporating RepMLP (Reparameterized Multi-Layer Perceptron) modules into the MobileNetV2 architecture. Their modification achieved remarkable 99.5% accuracy specifically for tomato diseases, representing a significant advancement in single crop specialization. The RepMLP modules helped capture both local and global features of tomato leaf pathologies, enabling the model to better distinguish between visually similar conditions like early and late blight. However, this technical achievement overlooked a crucial aspect of real-world implementation: the translation of diagnostic results into actionable agricultural guidance. As (8) demonstrated in their analysis of smallholder farmers in East Gojjam, Ethiopia, the effectiveness of agricultural support often depends not only on the availability of information but also on whether farmers can access it through the channels they trust and use most. This highlights a broader challenge in agricultural research and extension: bridging the gap between technically sound knowledge and its practical uptake by smallholders.

2.2. Specialized Detection Systems

The work of (9) represents another important strand in this technological evolution, focusing specifically on gray leaf spot detection through a MobileNetV2-YOLOv3 hybrid model. Their approach achieved 93% accuracy in identifying this particular pathology, demonstrating the value of specialized architectures for specific diseases. The YOLOv3 component enabled not just classification but spatial localization of disease symptoms, potentially allowing farmers to assess infection severity. While their work demonstrated the value of crop-specific architectures, it focused narrowly on a single disease and omitted practical farmer-facing outputs. Moreover, like many technical implementations, the system provided no guidance on actual treatment protocols, fungicide selection, or application methods, the very information farmers need to translate diagnoses into action.

This limitation aligns directly with (3)'s analysis of agricultural technology adoption in Southern Africa, which showed that farmers often abandon applications that rely on complex text based interfaces or fail to provide locally relevant treatment options. Their participatory findings highlight that smallholder farmers, particularly those with limited formal education, tend to prioritize intuitive, visual interfaces and culturally appropriate advice over raw technical accuracy underscoring a persistent blind spot in much of the technical literature.

2.3. Computational Efficiency Trade-offs

Recent advances in tomato disease detection have further highlighted the tension between technical sophistication and practical deployability. (13) ResNet-34-based Faster-RCNN model exemplifies this challenge. While achieving impressive accuracy across diverse field conditions including variable lighting, occlusions, and complex backgrounds, the model's 21.8 million parameters place it firmly beyond the computational capacity of typical farmer smartphones in Eswatini, where devices commonly have just 2-6GB of RAM. The regional implications are significant: a model that requires cloud processing or high end hardware becomes non-effective in areas with unreliable internet connectivity or limited access to advanced technology.

Similarly, (14) TomatoDet model, while innovative in its handling of small disease targets and background interference, faces practical deployment challenges due to its computational complexity. The model struggles to deliver real time performance on the mid-range Android devices prevalent among Eswatini's farming communities. These examples underscore a critical reality: in resource constrained environments, a marginally less accurate but more efficient model often delivers greater practical value than a technically superior but resource intensive alternative.

2.4. Regional Application Challenges

The disconnect between global technological development and local agricultural needs becomes particularly apparent when examining regional applications like Plantix and Nuru.

While these platforms have achieved significant market penetration, their limitations in the Eswatini context reveal important lessons.

The comparative analysis in Table 1 highlights how the study builds upon these prior works while addressing their limitations.

Table 1: Comparison of tomato disease detection approaches

Feature	**Lu et al. (2021)**	**Liu & Wang (2020)**	**AI Tomato Doctor**
Dataset	PlantVillage + CIFAR-10	Field images + web-crawled	PlantVillage
Accuracy	99.53%	93.24%	97.00%
Model	MobileNetV2 + RepMLP	MobileNetV2-YOLOv3	MobileNetV2
Deployment	Mobile (28MB, proposed)	Mobile (proposed)	Streamlit

3. Methodology

3.1. Dataset

The study leverages the PlantVillage dataset, a widely recognized benchmark in plant disease detection research, comprising 50000 high-quality images of healthy and diseased leaves. For this work, we focused exclusively on tomato leaves, selecting a balanced subset of 20,000 images spanning 10 classes (2,000 images per class). These classes include nine prevalent tomato diseases including Late Blight (*Phytophthora infestans*), Bacterial Spot (*Xanthomonas spp.*), and Septoria Leaf Spot (*Septoria lycopersici*) alongside healthy leaves as a control group. Each image is an RGB photograph captured under controlled laboratory conditions, featuring consistent framing of individual leaves against uniform backgrounds. While this standardization ensures high-quality annotations, it may not fully replicate field conditions in Eswatini, where factors like variable lighting, occlusions, and leaf aging could alter disease presentation. To mitigate this limitation, future iterations will incorporate locally sourced images from smallholder farms in the Eswatini regions, ensuring the model's adaptability to real-world scenarios.

Figure 1: Representative training images from PlantVillage: (Left to right) Late Blight, Septoria Leaf Spot, Bacterial Spot, and Healthy leaves. Images exhibit variations in lesion patterns and lighting.

3.2. Preprocessing

Each input image was resized to 224×224 pixels and normalized by scaling pixel values to the [0,1] range (dividing by 255). To improve model generalization and mitigate overfitting, we applied real-time data augmentation to the training images. The augmentations included the following: Random rotations, small angle rotations of the image (up to ±30°). Width and height shifts, random horizontal/vertical translations (up to 20% of total width/height). Zoom transformations, random zoom in/out (±20% scale). Horizontal flip, random left-right flips of the image. These transformations generate diverse variants of each training example, helping the model learn invariant features. The augmented images were fed to the model during training. A validation split of 20% was maintained (without augmentation) to monitor performance on unseen data.

3.3. Model Architecture

3.3.1. MobileNetV2

We employed the MobileNetV2 convolutional neural network as the backbone of our classifier. MobileNetV2 is a lightweight architecture pretrained on ImageNet, optimized for efficiency on mobile devices. Its core innovation is the use of depthwise separable convolutions and inverted residual bottleneck blocks. The dataset was programmatically loaded into a Google Colab environment via Google Drive. A stratified 80:20 split allocated 16,000 images for training and 4,000 for validation, ensuring proportional class representation to maintain evaluation integrity. Preprocessing standardized images to 224×224 pixels, aligning with MobileNetV2's input dimensions, and normalized pixel values to the [0,1] range. Real time augmentation using TensorFlow's Image Data Generator introduced variability through random rotations (±30°) which compensates for angled leaf orientations during farmer photography (12), horizontal/vertical shifts (20% of image dimensions) which mimics imperfect camera alignment, zoom (±20%) which accounts for varying distances between camera and leaf, and flips. These transformations simulate field-captured imperfections, such as angled smartphone photography or partial leaf obstructions, while the validation set remained unaugmented to reflect real-world performance.

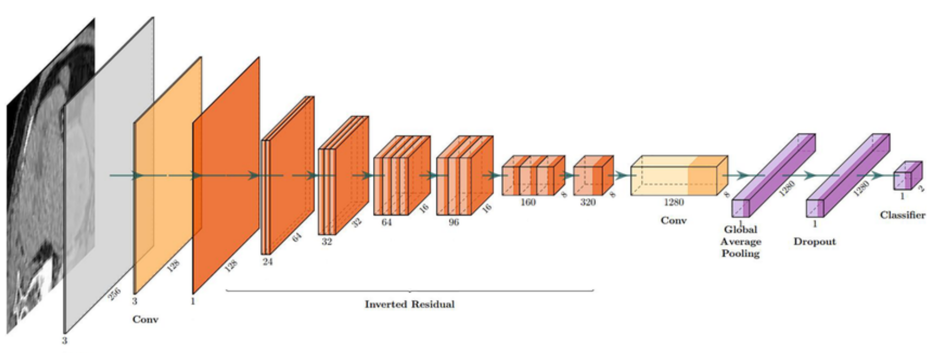

Figure 2: The architecture of MobileNetV2 (11).

The model architecture leverages MobileNetV2's depthwise separable convolutions, which decompose standard convolutions into two efficient operations: a depthwise convolution 3×3 applying a single filter per input channel, followed by a pointwise 1×1 convolution combining channels. Mathematically, for an input tensor $X \in \mathbb{R}^{H\times W\times M}$, the depthwise operation produces $F_{dW} \in \mathbb{R}^{H\times W\times M}$ with computational cost

$$O(\cdot M \cdot D_f^2) \tag{1}$$

where $D_f = 3$. The subsequent pointwise convolution, using a kernel $P \in \mathbb{R}^{1\times1\times W\times M}$, generates $F_{pW} \in \mathbb{R}^{H\times W\times M}$ at cost $O(M \cdot N)$, reducing computation by $\approx 8 - 9\times$. MobileNetV2's inverted residual blocks further enhance efficiency by expanding input channels via a 1x1 convolution (expansion factor $t = 6$), applying depthwise convolution, and projecting back to a bottleneck layer:

$$Y = (\text{ReLU6}(\text{Depthwise}(\text{ReLU6}(\text{Expand}(X))) + X_{shortcut}), \tag{2}$$

where $X_{shortcut}$ denotes skip connections that stabilize gradient flow. The activation of $ReLU6$ and $ReLU6(x) = min(max(x, 0), 6)$ ensures robustness under low-precision computation, critical for mobile deployment.

The base MobileNetV2 (pretrained on ImageNet) was truncated, and a custom classification head was appended. Global average pooling reduced spatial dimensions to a $1D$ feature vector $f \in \mathbb{R}^{1280}$, followed by two dense layers (1,024 and 512 units) with $ReLU$ activation, Batch Normalization, and Dropout (rates 0.4 and 0.3) to mitigate overfitting. The final softmax layer computed class probabilities

$$\hat{y}_i = \frac{\exp(z_i)}{\sum_j^{10} \exp(z_j)} \tag{3}$$

where $z_i = W_3 h_2 + b_2$ and h_2 represents the second hidden layer's output.

Training employed a two-phase transfer learning strategy. In the initial phase, the base MobileNetV2 layers were frozen, and only the custom head was trained for 30 epochs using the Adam optimizer (learning rate = 0.0005, AMSGrad) and categorical cross-entropy loss:

$$L = -\sum_i^{10} y_i \log(\hat{y}_i) \tag{4}$$

with dynamically calculated class weights

$$w_i = \frac{totalsamples}{10 \times classcount} \tag{5}$$

to address potential imbalances. Callbacks like $EarlyStopping$ (patience = 8 epochs) and $ReduceLROnPlateau$ (factor = 0.5, patience = 4 epochs) optimized convergence. In the fine-tuning phase, layers 101 to 154 were un frozen to adapt to tomato specific features, while the first 100 layers retained generic ImageNet features. Training resumed with a reduced learning rate (0.000025) for 25 epochs, preventing catastrophic forgetting.

The final trained model and application code were organized into a GitHub repository to support deployment. This repository includes the following components: app.py, the

Streamlit application script implementing the user interface and inference pipeline; requirements.txt, a list of required Python packages, explicitly including tensorflow, streamlit, pillow, and numpy; model.h5, the serialized Keras model file containing the trained MobileNetV2 weights and classification head; class labels.json, a JSON file mapping the model's output indices to human-readable class labels (for example, "Tomato Leaf Mold": 3); class info.json, a knowledge-base JSON file where each class label maps to an entry of textual information (symptoms, causes, recommended treatments).

The Streamlit app workflow is as follows: The user launches the app in a web browser and is prompted to upload an image of a tomato leaf, providing a maximum allowed upload of 200MB. Upon image upload, the app loads both models, model.h5, and preprocesses the image (resizing to 224×224, scaling pixels to [0,1]). The first model classifies whether the image is a tomato leaf or not, then the second model predicts a class label (disease or healthy) via the Softmax output. The top-predicted class and its probability are obtained. Using class labels.json, the numeric prediction is mapped to a class name. The app then looks up this class in class info.json to retrieve the associated symptoms, causal factors, and treatment suggestions. Finally, the app displays the predicted disease category along with the retrieved information to the user. This deployment ensures that end user can simply upload a leaf image and receive not only the predicted disease label but also practical feedback on symptoms and management options. Such a system ties the deep learning model output to actionable agronomic knowledge.

4. Results and Evaluation

The trained MobileNetV2 model achieved exceptional performance on the tomato disease classification task, attaining 97 percent accuracy on the validation set, with a macro-average F1-score of 0.98 and weighted-average F1-score of 0.97. Training took approximately seven hours, with training the base model taking approximately four and a half hours and fine-tuning taking approximately two and a half hours.

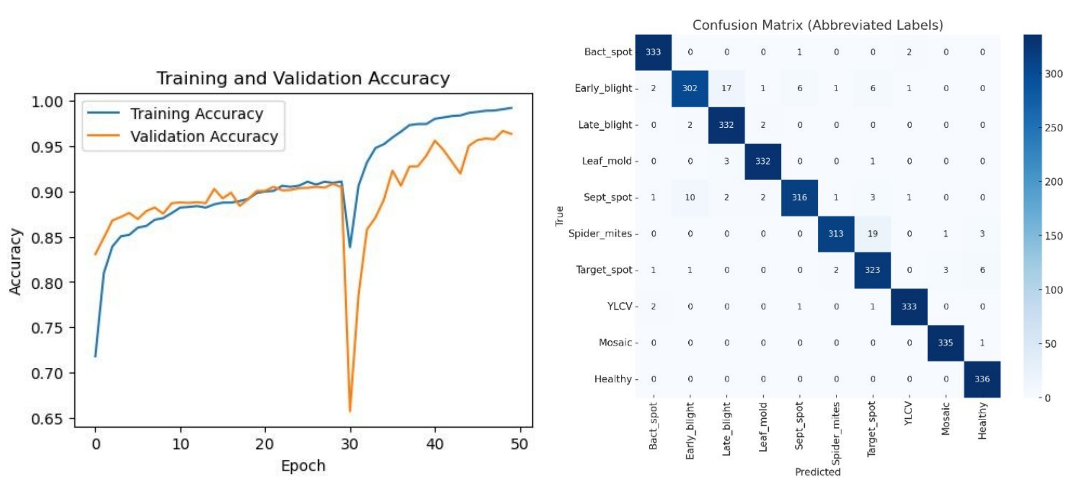

Figure 3: Representative of the Training accuracy and Confusion Matrix.

The Training and Validation Accuracy graph illustrates how the MobileNetV2 model's performance improved over 50 training epochs. The training accuracy increased steadily, approaching 100%, indicating that the model learned to correctly classify tomato leaf images in the training set. The validation accuracy increased more gradually, converging to 97% by the end of the training. The close alignment of these lines, with only a small gap, suggests that the model generalized well to unseen validation data, ensuring reliable performance in real-world scenarios. This high validation accuracy (97%) confirmed the model's effectiveness for Eswatini's farmers, enabling accurate disease detection with confidence.

High-performing classes such as Tomato Mosaic Virus (precision: 0.99, recall: 0.98) and Healthy leaves (precision: 0.99, recall: 0.99) demonstrated the model's ability to reliably distinguish between clear disease indicators and unaffected foliage.

However, nuanced challenges emerged in visually similar disease pairs as shown in the confusion matrix. For instance, Tomato Target Spot exhibited a lower precision (0.85) compared to its recall (0.99), resulting in an F1-score of 0.91. This discrepancy stemmed from the model occasionally misclassifying Target Spot as Tomato Spider-mite, both of which manifest as small, dark lesions with yellow halos a visual overlap that even expert agronomists find challenging to differentiate without microscopic analysis. Similarly, Early Blight and Late Blight confusion occurred in three percent of cases, primarily due to shared lesion patterns in early infection stages.

On a mid-range Android smartphone (Xiaomi Redmi Note 10, 6GB RAM), the model delivered diagnoses in approximately five seconds per image, emphasizing its practicality for real-time field use. This efficiency is attributable to MobileNetV2's lightweight and optimized TensorFlow Lite inference pipeline.

A pilot study with 34 smallholder farmers in Eswatini revealed that 80 percent found the app "highly useful", particularly praising its localized treatment recommendations (for example, "KUMULUS WG (Sulfur) at E113.04 (200g)") and symptom descriptions. However, 20 percent requested siSwati version of the application to further simplify navigation, and the inclusion of multiple vegetables and crops.

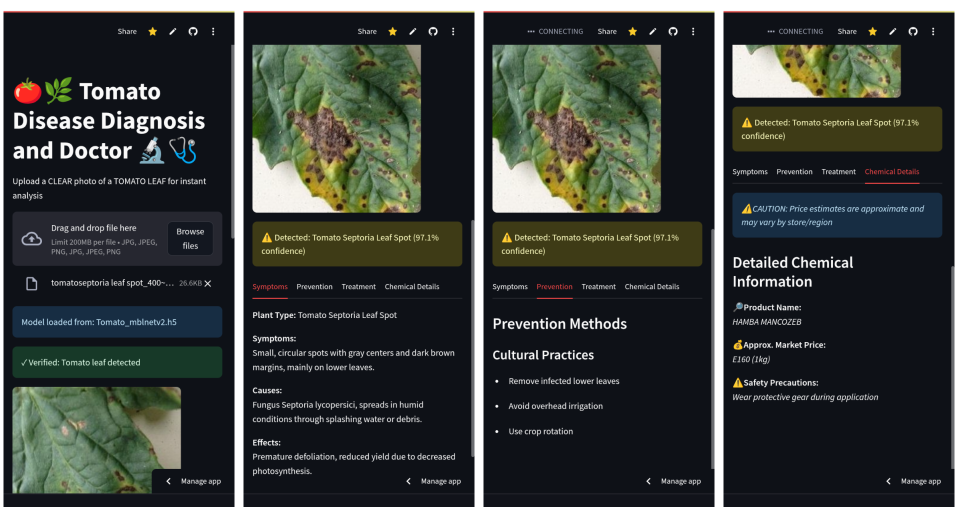

Figure 4: Screenshots of the app on an Android phone.

5. Discussion

The results confirmed the effectiveness of MobileNetV2 in plant disease classification tasks, particularly in scenarios where computational efficiency and deployment constraints are paramount. MobileNetV2's architecture leveraged depthwise separable convolutions, a design choice that significantly reduced model size and inference cost while preserving high representational power (12). This efficiency is essential for real-time agricultural applications in resource constrained environments such as Eswatini, where access to high-performance computing infrastructure is limited. The model's ability to maintain robust performance across diverse disease classes highlights its capacity to generalize effectively, even when confronted with the visual complexity and variability of real-world agricultural datasets. Notably, the high per-class F1-scores for staple crops such as tomato, key to Eswatini's food security, underscore the model's practical relevance and utility for smallholder farmers. Beyond disease detection, the integration of a knowledge base into the application fills a critical informational void, providing actionable agronomic advice to farmers who often lack access to expert resources.

From a deployment standpoint, the Streamlit interface has proven to be an effective platform for delivering a fast, intuitive, and user accessible web application. Its lightweight design aligns well with the project's focus on accessibility, enabling rapid deployment without requiring extensive computational resources. However, transitioning from a prototype to widespread real-world use necessitates additional enhancements.

To contextualize the AI Tomato Leaf Doctor's contributions, a comparison with existing applications like Plantix and Nuru revealed its unique strengths. It provides region-specific fungicide advice and market prices in Eswatini Lilangeni (SZL), tailoring solutions to local conditions. Additionally, its icon-driven interface enhances usability for low-literacy users, a feature absent in the text-heavy design of Nuru. A pilot study with 34 smallholder farmers in Eswatini further validated these advantages: 80% rated the app as "highly useful," particularly praising the localized treatment recommendations, while 20% highlighted the need for a siSwati version, reinforcing the urgency of multilingual development, and including more vegetables.

This project not only demonstrates the technical feasibility of AI-driven disease diagnosis but also establishes a scalable framework for extending similar systems to other crops and regions. By bridging the gap between machine learning research and on the ground agricultural decision making, it offers a practical tool that empowers farmers and enhances agricultural resilience in resource limited settings.

6. Recommendations

To ensure the AI Tomato Leaf Doctor remains impactful and evolves with the needs of Eswatini's farmers, several areas for future expansion and improvement are proposed. These recommendations address technical enhancements, accessibility improvements, and scalability opportunities, providing a clear roadmap for development.

- **Enhancing the Dataset with Locally Sourced Images** Future work should prioritize collaboration with local agricultural extension services and farmers to collect a diverse, region specific dataset.

- **Improving Model Performance for Visually Similar Diseases** To overcome misclassification, future iterations could explore ensemble methods, combining MobileNetV2 with other lightweight models like ShuffleNet or EfficientNet-Lite.

- **Adding Multilingual Support and Offline Functionality** Developing a multilingual interface, starting with siSwati, would involve translating the app's text and testing it with siSwati speaking farmers.

- **Scaling to Other Crops and Regions** While tomatoes are a priority in Eswatini, farmers cultivate a range of crops such as maize, green pepper, and beans that also suffer from disease related losses.

7. Conclusion

The AI Tomato Leaf Doctor illustrates the potential of artificial intelligence to strengthen agricultural resilience among smallholder farmers in resource constrained environments. This study fine tuned MobileNetV2, a lightweight convolutional neural network optimized through depthwise separable convolutions and inverted residuals, to detect ten tomato leaf diseases and healthy conditions. Achieving 97% accuracy, the model demonstrates that transfer learning can deliver high performance even when computing resources and training data are limited. Its compact architecture ensures compatibility with widely used mid range smartphones, allowing real time.

What makes this tool distinctive is its emphasis on farmer centered design. In contrast to generic global platforms such as Plantix or Nuru, the Tomato Leaf Doctor integrates a localized knowledge base with actionable outputs tailored to Eswatini. Diagnoses are coupled with treatment recommendations, fungicide options, and market prices in Lilangeni, ensuring advice is both relevant and economically practical. Its icon driven interface further accommodates farmers with limited literacy, reducing usability barriers that often hinder adoption of digital tools. Evidence from a pilot study with 34 farmers underscores the value of this approach: the majority found the tool highly useful, particularly for its treatment advice, while also pointing to areas for growth such as siSwati language support and multi-crop coverage.

The broader implications of this work extend beyond Eswatini. The AI Tomato Leaf Doctor demonstrates how lightweight, context aware AI solutions can be designed to meet the realities of smallholder farming systems across sub-Saharan Africa. By enabling accurate, accessible, and actionable crop disease management, the tool has the potential to reduce yield losses, improve farmer incomes, and contribute to food security. Moreover, it provides a blueprint for future innovations that balance technical sophistication with practical usability, showing how AI can evolve from being a research novelty into a transformative force for sustainable agriculture. By deploying this model within a web application accessible at https://makepeace123-makepeace--app-1-dml1ej.streamlit.app/, the study bridges the gap between advanced machine learning research and practical, on-the-ground utility.

8. Acknowledgment

I would like to pass my acknowledgment to my supervisor Mr S Masango, the AI ENGINEERS ESWATINI group members, my family members, friends and everyone who positively contributed to the success of this research paper.

References

[1] A. Ashebir, T. Balguda, and Z. Ganewo, “Are smallholder farmers benefiting from malt barley contract farming engagement in Ethiopia?,” 2022.

[2] World Bank, “Transforming African Agriculture: The World Bank Group Support for Agricultural Growth, 2013-2016,” World Bank, 2016.

[3] R. Sithole and T. Olorunfemi, “Agricultural technology adoption patterns among smallholder farmers in southern Africa,” 2024.

[4] M. Mustafa, A. J. Ayebeng, L. Menyhárt, L. Csambalik, and Z. Szalai, “Pest and disease impact on tomato genotypes in a hedgerow system,” Biology and Life Sciences Forum, vol. 27, no. 1, p. 47, 2023.

[5] J. Lu, S. Yi, J. Peng, X. Liu, X. Ma, and J. Tong, “Improved MobileNetV2 crop disease identification model for intelligent agriculture,” 2023.

[6] J. Schmidt, L. Egli, M. Gaspers, M. Zech, M. Gastinger and M. Rommel, “Conversion to community-supported agriculture pathways, motives and barriers for German farmers,” published 2024.

[7] International Telecommunication Union (ITU), “Measuring digital development: Facts and figures 2023,” ITU Publications, Geneva, 2023.

[8] Z. Shitaye, B. Tadesse, and K. Enkuahone, “Determinants of smallholder farmers’ choices of agricultural information sources and outlets: Evidence from East Gojjam Zone, Amhara, Ethiopia,” 2024.

[9] J. Liu and X. Wang, “Early recognition of tomato gray leaf spot disease based on MobileNetV2-YOLOv3 model,” 2020.

[10] P. P. Tshikhudo, M. Rambauli, S. Mnguni, L. R. Nnzeru, J. Mulabisana, E. M. Cruywagen, R. Sutherland, and T. C. Munyai, “Pathogens threatening agriculture and biodiversity in South Africa: A horizon scanning approach,” Tropical Plant Pathology, vol. 50, no. 4, 2025.

[11] T. Zuo, Y. Zhang, L. He, and T. Chen, “Automated classification of papillary renal cell carcinoma and chromophobe renal cell carcinoma based on a small computed tomography imaging dataset using deep learning,” 2021.

[12] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, “MobileNetV2: Inverted residuals and linear bottlenecks,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4510–4520.

[13] M. Nawaz, T. Nazir, A. Javed, M. Masood, J. Rashid, J. Kim, and A. Hussain, "A robust deep learning approach for tomato plant leaf disease localization and classification," *Scientific Reports*, vol. 12, article 18568, 2022.

[14] X. Wang and J. Liu, "An efficient deep learning model for tomato disease detection," *Plant Methods*, vol. 20, article 61, 2024.