



DEEP CONSULTING CONIA2025

An Innovative Approach to Problem-Solving



Maize Disease Classification

Community Driven Innovation



Submission Date: July 6, 2025

📍 Yaoundé, Cameroon

Contents

Introduction	1
1 Introduction	2
1.1 Context and Impacts	2
1.2 Reacheable Goal of the project	2
2 Methodology	2
2.1 Data Acquisition and Exploration	2
2.2 Preprocessing pipeline	4
2.3 Model Deployment	4
3 Results and Findings	4
4 Challenges and Limitations	6
5 Recommendations and Next Steps	6
6 Conclusion	7
7 Appendix	8
8 Appendix: Structured AI Pipeline for Maize Disease Classification	8

1 Introduction

1.1 Context and Impacts

Maize is a fundamental food source for millions of people in Africa and worldwide, serving as a key source of nutrition, especially in Sub-Saharan Africa (SSA). This crop is a pillar of food security yet also a vector of economic vulnerability. In Cameroon and across the continent, the accelerating spread of maize diseases is not just an agriculture concern; it is vital to economic resilience, supporting smallholder farmers and local markets, nutrition and livelihoods, etc.

Nevertheless, rising diseases like maize lethal necrosis disease (MLN) and pests, worsened by climate change, are causing significant crop losses. This leads to a dual crisis: a **health crisis** from contaminated maize posing risks, and **economic crisis** with price spikes making food unaffordable, especially during droughts and heatwaves.

The impact is severe and hits hard: food insecurity grows, livelihoods are threatened and pressure on markets and health systems increases. Finding innovative solutions to protect this essential crop is now urgent.

1.2 Reacheable Goal of the project

This 3-days hackathon project emerges from the urgent need to deploy **Intelligent Diagnostic Systems** that is able to preemptive disease detection to safeguard Africa's food. The primary raechable goal of this hackathon project is to develop a functional prototype that has uses a maize leaf dataset to detect and classify diseases : Maize lethal necrosis (MLN) and maize streak virus (MSV) within 3 days development windows. The prototype will employ a machine learning model, a convolutional neural network (CNN) trained on leaf images, to provide real-time disease identification for farmers via a simple interface.

2 Methodology

2.1 Data Acquisition and Exploration

The project start with the acquisition and exploration of the Lacuna Maize dataset, comprising 15349 labeled maize leaf images collected in Tanzania between February and July 2021 by researchers from the Nelson Mandela African Institution of Science and Technology. The dataset include 3 classes: Healthy (5118 images), Maize Lethal Necrosis (MLN, 3980), Maize Streak Virus (MSV, 6251).

The initial step involves verifying the dataset's integrity by checking for corrupted images (image 1935 in the Healthy class) and analyzing the class distribution to identify imbalance. We can observe the sample image and class distribution in Figures (1) and (2).

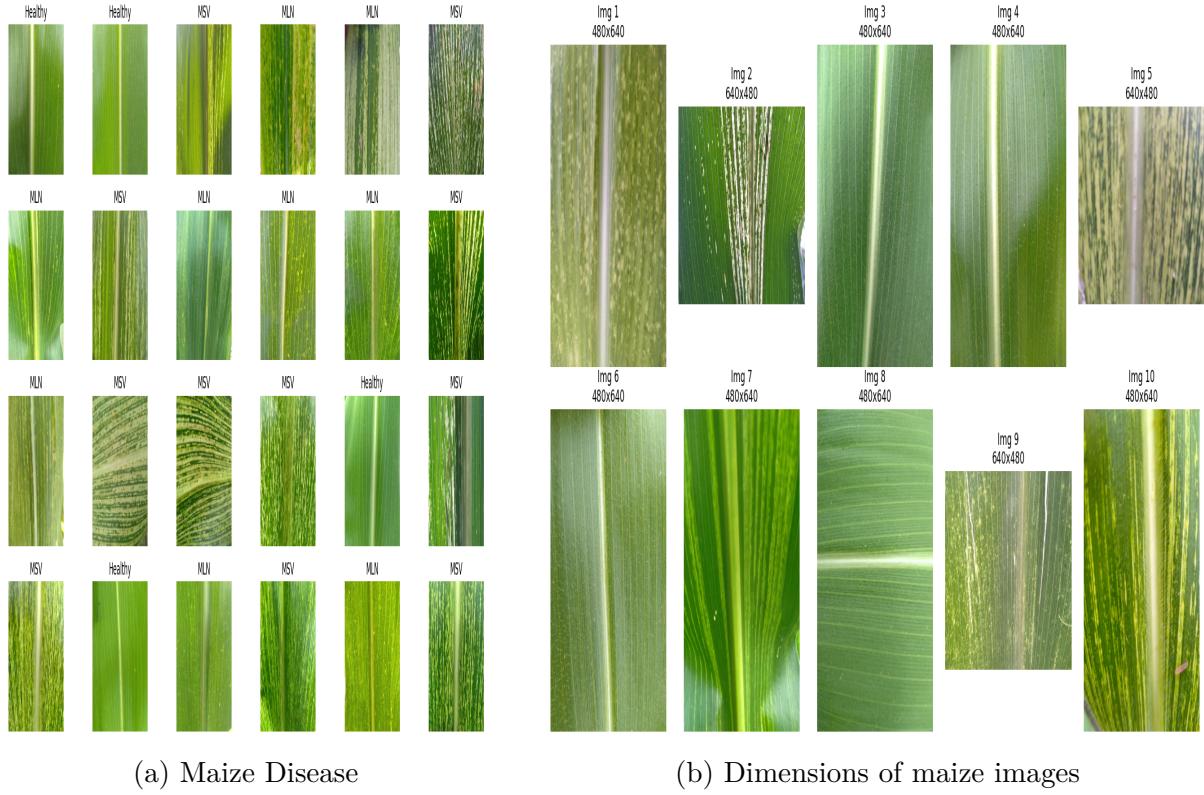


Figure 1: Sample images of maize leaf

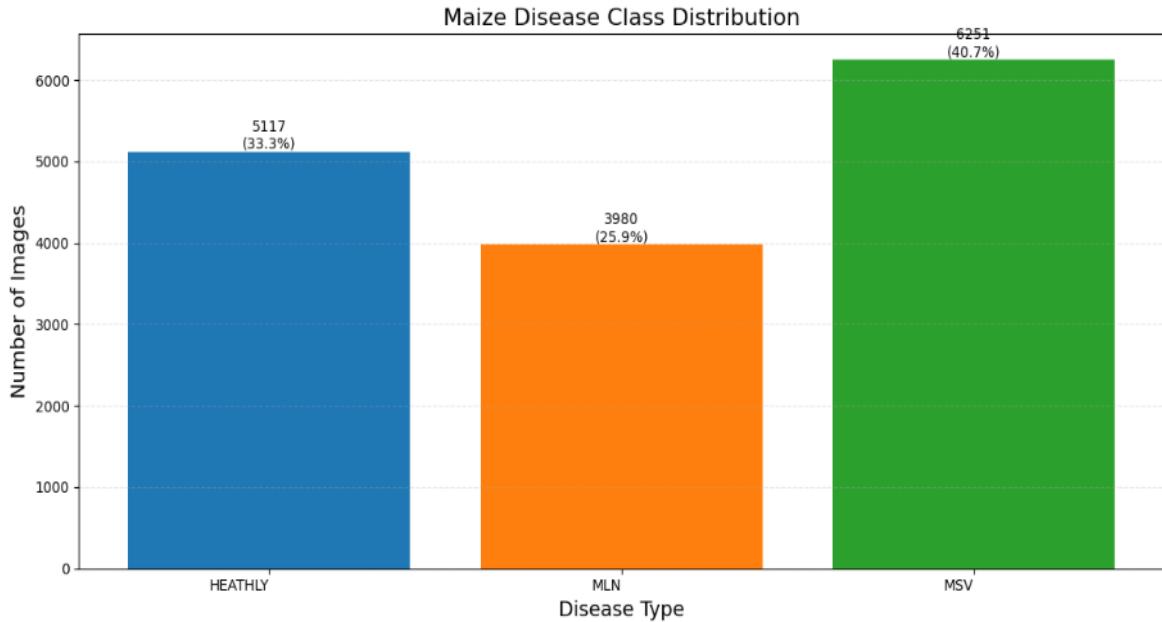


Figure 2: Maize disease class distribution

2.2 Preprocessing pipeline

The preprocessing pipeline prepares the dataset for model training by addressing data quality and enhancing model robustness. This involves creating a custom **CustomImageDataset** class to filter out corrupted images using a validation function that checks image integrity and format. Because of the well preprocess format of our dataset, we just resize our images and normalize for preparing training. After that we apply a stratified splitting to our dataset where we consider 80% training set, 10% validation set and 10% test set.

2.3 Model Deployment

Due to the time constraint and the limited internet connectivity, we were able to apply just one model. The processing phase leverages PyTorch to develop and train a Residual Network model (ResNet50), pretrained on ImageNet for maize disease classification. The model take exactly 824 minutes of compilation.

The model is trained for 10 epochs, with metrics (loss, accuracy) as we observe in Figure (3).

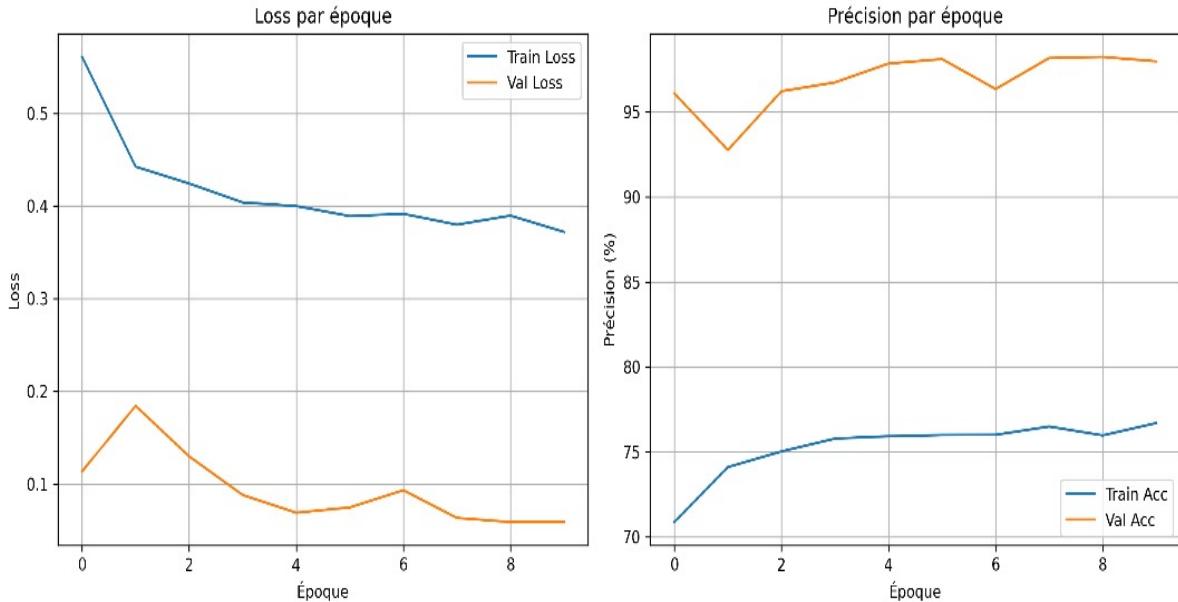


Figure 3: Training accuracy

3 Results and Findings

Performance on the test set is evaluated using a confusion matrix and classification report in Figure (4) and the table (1).

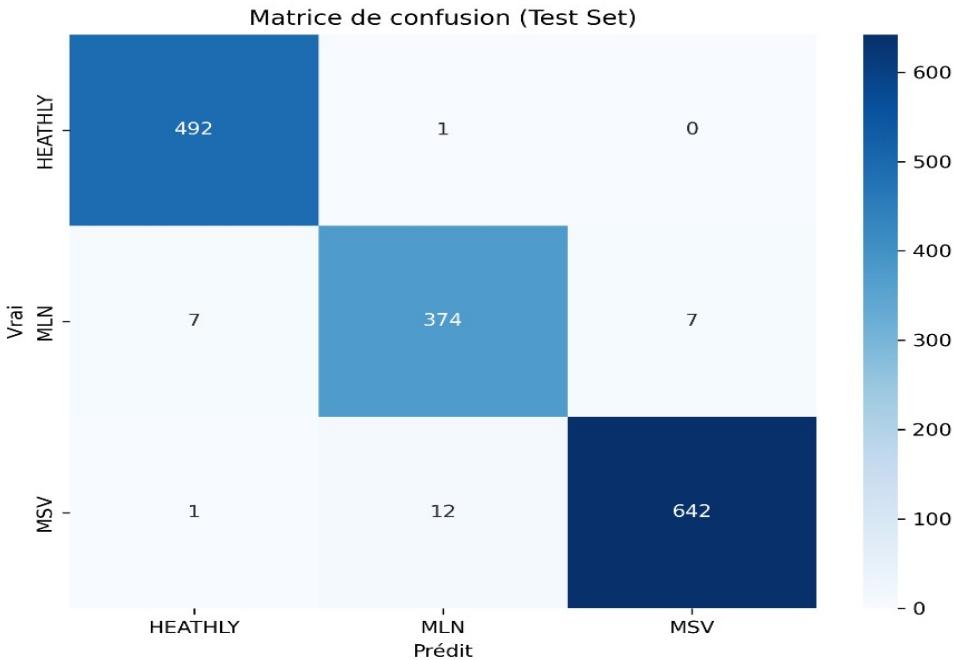


Figure 4: Confusion matrix

The following table presents the classification report for the maize disease detection model based on the test set:

Table 1: Classification Report for Maize Disease Detection

Rapport de classification	precision	recall	f1-score	support
HEATHLY	0.98	1.00	0.99	493
MLN	0.97	0.96	0.97	388
MSV	0.99	0.98	0.98	655
macro avg	0.98	0.98	0.98	1536
weighted avg	0.98	0.98	0.98	1536

Building on the successful training of our ResNet50 model, we evaluated its performance on a new unseen test set, yielding impressive results. The classification report, based on 1536 test instances, demonstrates a robust model with a precision of 0.98, recall of 0.98, and an F1-score of 0.98 across all classes, as detailed in Table 1. These scores validate the model's capability to generalize across diverse leaf conditions, despite the dataset's slight class imbalance, thanks to the applied class weighting.

The trained model has been saved and integrated into a web application. This application features an intuitive interface where users can input data and receive actionable insights. Farmers can select between English and French languages, tackling linguistic diversity,

and upload images of maize leaves for real-time disease classification.

Upon uploading an image, the model accurately predicts the class and generates tailored recommendations based on the prediction. Additionally, the application includes an interactive chatbot section, enabling users to discuss disease-related queries or seek further agricultural advice, enhancing engagement and support. This approach not only meets our goal of achieving over 90% accuracy but also delivers a practical, accessible tool to mitigate maize disease impacts, as demonstrated by our prototype’s successful validation on the test set.

4 Challenges and Limitations

- **Connectivity Issues:** The first two days were hampered by inconsistent internet connectivity, disrupting data downloads, model training, and team collaboration, which delayed our initial progress.
- **Local Machine Capacity Constraints:** Loading TensorFlow and PyTorch packages on local machines posed significant challenges, likely due to limited hardware capacity (e.g., RAM or GPU limitations). The observed capacity strain post-installation may be typical or exacerbated by internet speed, complicating efficient package management and model execution.
- **Inability to Develop Custom CNN Architecture:** Due to time and resource constraints, we were unable to construct our own convolutional neural network (CNN) architecture, leading us to rely on a pre-trained ResNet50 model. This limited our ability to tailor the model specifically to the Lacuna Maize dataset.
- **Tight Timeframe:** Training time lead to limiting opportunities for extensive hyperparameter tuning or deeper optimization.
- **Computational Resource Constraints:** Reliance on CPU or limited GPU access slowed experimentation, reducing the ability to test multiple model architectures.

5 Recommendations and Next Steps

- **Field Testing and Feedback:** Collaborate with agricultural extension services to pilot the app in real-world conditions, gathering farmer feedback to refine the interface.
- **Long-Term Research:** Pursue funding to develop a comprehensive agricultural AI

platform, integrating disease detection with yield prediction and pest management, addressing the broader food security challenges in sub-Saharan Africa.

6 Conclusion

As the **CONIA 2025 hackathon** draws to a close on July 06, 2025, our team, **Deep Consulting**, is proud to have developed a groundbreaking prototype for maize disease detection, addressing critical challenge. Leveraging the Lacuna Maize dataset of 15349 images, we trained a ResNet50 model that achieved an impressive 98% accuracy on an unseen test set, effectively classifying Healthy, Maize Lethal Necrosis (MLN), and Maize Streak Virus (MSV) conditions. This result, validated by detailed metrics and a user-friendly application, fulfills our goal of creating a tool with over 90% accuracy within limited timeframe, despite significant hurdles such as connectivity issues, limited local machine capacity, and the inability to design a custom CNN architecture. Our journey was marked by resilience, transforming challenges into opportunities to deliver a practical solution. The application's intuitive interface, supporting English and French, allows farmers to upload leaf images, receive instant disease predictions with tailored recommendations, and engage with a chatbot for further support, demonstrating a holistic approach to enhancing food security. While constrained by time and resources, these achievements lay a solid foundation for future growth.

7 Appendix

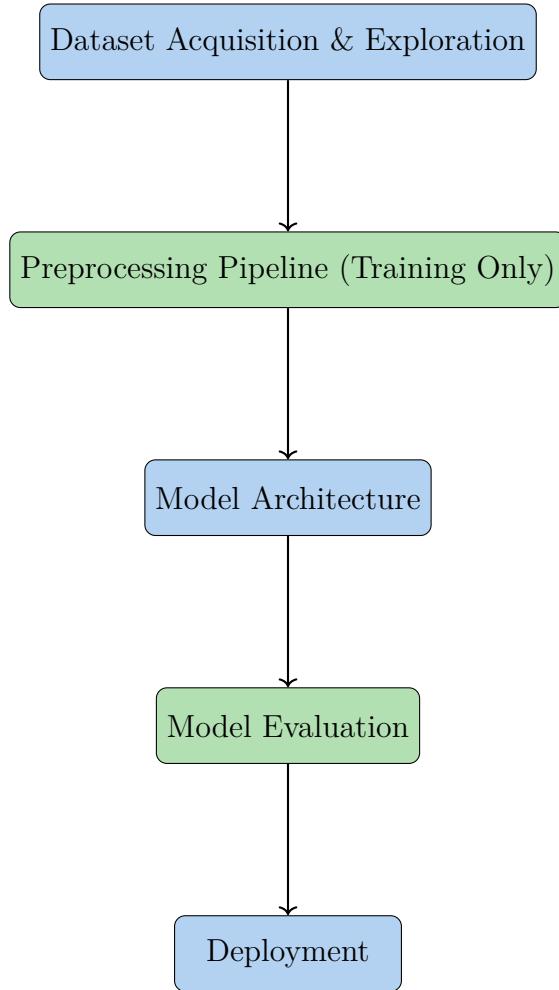


Figure 5: End-to-end pipeline for disease classification

8 Appendix: Structured AI Pipeline for Maize Disease Classification

This section outlines the algorithm developed by the **Deep Consulting** team to classify maize diseases (Healthy, MLN, MSV) using the Lacuna Maize dataset.

Algorithm 1 Structured AI Pipeline for Maize Disease Classification

- 1: **Input:** Lacuna Maize dataset (15,349 images: 5,118 Healthy, 3,980 MLN, 6,251 MSV, 15,348 after corruption removal)
 - 2: **Output:** Deployed classification model with evaluation metrics
 - 3: **1. Dataset Acquisition & Exploration**
 - 4: Collect images from Lacuna Maize dataset
 - 5: Analyze class distribution (e.g., Healthy = 33.3%, MLN = 25.9%, MSV = 40.71%)
 - 6: Split data: 80% train, 10% validation, 10% test
 - 7: **2. Preprocessing Pipeline for each training image do**
 - 8:
 - Resize
 - 9: Apply augmentation:
 - 10: Normalize using ImageNet mean/std ([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
 - 11:
 - 12: Compute class weights: $w_c = \frac{N}{C \cdot n_c}$
 - 13: **3. Model Architecture**
 - 14: Use ResNet50 with:
 - Input: 256x256x3
 - Class weights from preprocessing
 - Focal loss ($\gamma = 3$, optional adaptation)
 - 15: Shared components:
 - GlobalAveragePooling2D → Dense(256, ReLU)
 - Dropout(0.5) → Softmax output
 - Loss function: CrossEntropyLoss with class weights
 - Optimizer: Adam (lr=0.001, weight_{decay} = 1e - 5) • Batch size: 32
 - Callbacks: ReduceLROnPlateau (mode='min', factor=0.1, patience=3)
 - 16: **4. Model Evaluation**
 - 17: Compute metrics:
 - Accuracy
 - F1-score
 - 18: Generate visualization on test sets (confusion matrix)
 - 19: **5. Deployment**
 - 20: Generate confusion matrices
 - 21: Define disease classes: Healthy, MLN, MSV
 - 22: Save the model in a
 - 23: **return** Deployed classification model with evaluation metrics
-