NOUN Interdisciplinary Journal of Computing, E-learning & Application (NOUN-IJCEA)
https://ijcea.nou.edu.ng
Volume 1 Number 1, April 2026 pp. 133-146

A System for the Recognition of Some Selected Grain Plant Leaves Using Deep Learning Algorithms

[1]Philip O. Odion, *[1]Abraham E. Evwiekpaefe, [1]Ibuomo R. Tebepah, [1]Yahaya S. Saleh and [2]Biniya Ma'aruf

[1]Department of Computer Science, Nigerian Defence Academy, Kaduna State, Nigeria.
[2]Federal University of Transportation, Daura, Katsina State, Nigeria.
*Corresponding Author: aeevwiekpaefe@nda.edu.ng

ABSTRACT

Manual inspection of grain plant leaves for defects is subjective and labor-intensive. Few studies have compared deep learning methods on a combined multi-crop dataset. The study collected locally 5,640 leaf images from rice, maize, and guinea corn farms in Nigeria and grouped them into six classes representing defective and healthy leaves for each crop. Three models were trained: YOLOv8 for end-to-end detection and classification, EfficientNetB0 for standalone image classification, and a hybrid that used YOLOv8 for leaf detection followed by EfficientNetB0 for patch classification. The hybrid achieved 99.85% accuracy on the test set, slightly above EfficientNetB0 (99.82%) and YOLOv8 (mAP 0.995). The hybrid also supplies bounding box locations, helping farmers identify exactly where damage appears. This system offers a reliable, field-deployable tool for monitoring grain crop health.

Keywords: Deep learning, Detection, EfficientNetB0, Grain crops, Leaf defect, YOLOv8

1. INTRODUCTION

Rice, maize, and guinea corn supply daily calories and income for millions of people across Africa, Asia, and Latin America (Caballero-Ramirez et al., 2023). Their productivity directly affects food availability and economic stability in farming communities. Among the factors that determine yield, leaf health stands near the top. A leaf that remains free from disease, pest damage, or nutrient deficiency supports photosynthesis and grain filling. Conversely, leaves with spots, lesions, rust, or discoloration reduce plant vigor and lower harvest quality (Chen et al., 2022). Farmers and agricultural experts currently rely on visual inspection to detect these leaf problems (Harakannanavar et al., 2022).

Recent developments in computer vision and deep learning offer a way past these limitations (Meshram et al., 2021). Algorithms can process thousands of images in minutes and produce repeatable results. Convolutional neural networks automatically learn to distinguish between healthy and defective patterns without manual feature design (Cevallos et al., 2020). Transfer learning with pre-trained models such as EfficientNetB0 reduces the need for enormous datasets, making these techniques accessible to agricultural research (Tan & Le, 2019). Although many studies have applied deep learning to plant disease detection, they almost always focus on a single crop.

Etaati et al. (2023) combined YOLOv4 with VGG19 to detect and classify wheat leaf diseases. Their object detection network reached a mean average precision of 91%, while VGG19 achieved an average F1-score of 95% across multiple disease categories. The study demonstrated that deep learning could assist farmers in identifying wheat diseases, but it remained limited to a single crop.

Similarly, Liang et al. (2023) designed a hybrid CNN-LSTM model that captured both spatial and temporal features for wheat head blight detection. Their approach improved accuracy in dynamic field conditions where symptoms evolve over time, yet the model required large computational resources for training. For rice crops, Gajbhiye and Deshmukh (2021) surveyed various deep learning approaches and noted that CNN-based models consistently outperformed traditional machine learning classifiers such as Support Vector Machines and Random Forests. Raja and Arulmurugan (2020) applied AlexNet to classify rice leaf diseases and highlighted YOLO's ability to detect multiple diseases in a single image, though they reported performance degradation with low-quality images or overlapping symptoms. Maize leaf diseases have received similar attention. Meshram and Kumar (2021) developed a CNN-based model that achieved high accuracy in detecting maize leaf blight, common rust, and streak virus. They emphasized the importance of large, annotated datasets for building robust models. Singh and Gupta (2021) used a hybrid CNN-RNN architecture for sugarcane disease detection, reporting strong classification results but noting difficulties with high-dimensional feature spaces. Outside grain crops, Harakannanavar and Patil (2021) applied SVM, K-NN, and CNN to tomato disease detection, with CNNs achieving 99.09% accuracy. Their work proved the effectiveness of automatic feature extraction in plant disease classification, though their dataset and findings were specific to tomato.

Hybrid approaches that combine detection and classification have also been explored. Lac et al. (2022) developed a lightweight CNN optimized for mobile applications, addressing resource limitations in agricultural environments. Their model achieved competitive results but showed slightly lower accuracy compared to more computationally intensive architectures. Hussain and Ahmed (2022) implemented VGG-16 for multi-crop plant disease classification, reporting high accuracy but acknowledging the model's heavy computational cost. Liang et al. (2023) demonstrated that combining spatial detection with temporal sequence analysis improved accuracy in dynamic environments. Their hybrid CNN-LSTM model captured both spatial features from individual images and temporal patterns from time-series data, providing a more comprehensive analysis of disease progression. Limited studies based on the literature reviewed has yet investigated a hybrid pipeline such as object detection followed by classification on multi-crop dataset. This study shall fill this gap.

2. METHODOLOGY

2.1 Research Workflow

The methodology followed a systematic workflow. First, leaf images were collected from agricultural fields in Gombe State, Nigeria, during two growth stages. Preprocessing operations including resizing, normalization, and cleaning were applied to the raw images. Manual annotation was performed using LabelImg tool in the python software, drawing bounding boxes around defective and healthy leaf regions and assigning labels according to a six-class scheme The dataset was split into training (70%), validation (20%), and test (10%) sets using stratified sampling. Three models YOLOv8, EfficientNetB0, and a hybrid combining both were designed and trained separately using the same training and validation sets. Each model was evaluated on the held-out test set using accuracy, precision, recall, F1-score, and, where applicable, Mean Average Precision and Intersection over Union as shown in Figure 1.

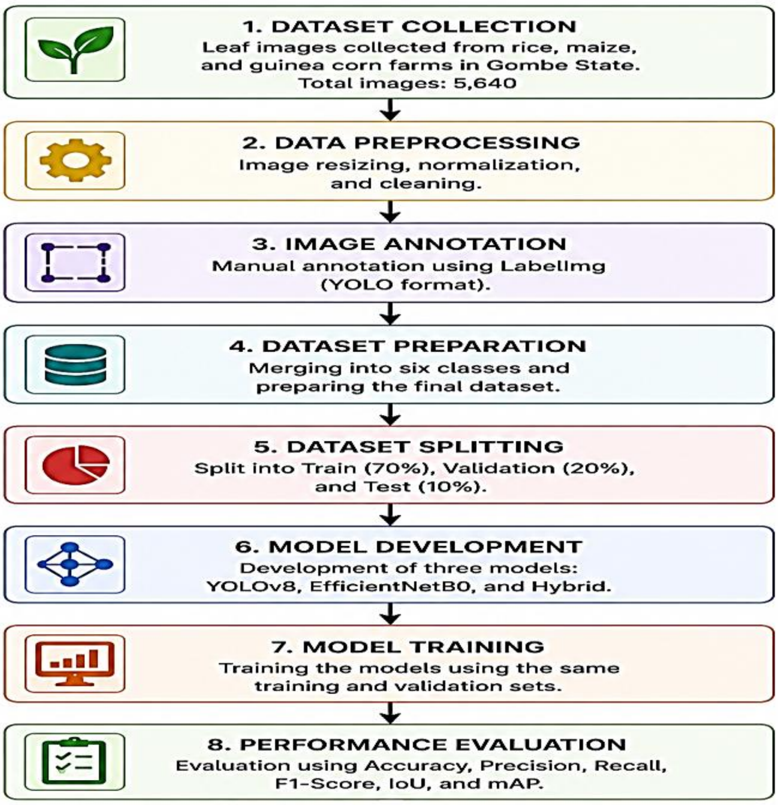

Figure 1: Diagrammatic workflow of the research

2.2 Study Area and Dataset Collection

The study was conducted in Dadin Kowa, Yamaltu Deba Local Government Area of Gombe State, Nigeria. This region supports substantial grain crop production, with farms ranging from 14 to 20 hectares. Three grain crops were selected for the study: rice, maize, and guinea corn. Images were taken during two growth stages; early and maturity to capture variations in leaf appearance across the plant life cycle. A total of 5,640 images were collected across the three crops. Table 1 shows the distribution of images per crop and growth stage. Rice contributed 1,900 images (880 defective, 1,020 healthy), maize contributed 1,950 images (920 defective, 1,030 healthy), and guinea corn contributed 1,800 images (850 defective, 950 healthy).

Table 1: Percentage of Rice Images Collected

Stage	Defective	Non-defective	Total	Percentage
Early	420	480	900	47.37%
Maturity	460	540	1,000	52.63%
Total	880	1,020	1,900	100%

2.3 Dataset Preparation and Annotation

The individual crop datasets were merged into a single six-class dataset to enable multi-crop classification within a single model. The six classes were defined as: Rice_Defective, Rice_NonDefective, Maize_Defective, Maize_NonDefective, GuineaCorn_Defective, and GuineaCorn_NonDefective This merged structure allowed the model to simultaneously recognize crop type and health status, creating a more challenging benchmark for comparing model performance. The merged dataset contained a total of 5,640 images, The training set received 70% of

the images (3,955 images), the validation set received 20% (1,130 images), and the test set received 10% (564 images). Stratified sampling prevents class imbalance from biasing the training process).
Manual annotation was performed using Labelling tool. Annotators drew bounding boxes around each leaf and assigned a class label from the six predefined classes. Annotations were saved in YOLO format as .txt files containing the class ID and normalized bounding box coordinates.

2.3 Model Development Framework

Three models were developed and trained separately, each representing a different approach to the grain leaf defect recognition task. All three were trained on the same six-class dataset to enable fair comparison of their strengths and weaknesses as depicted in Figure 2.

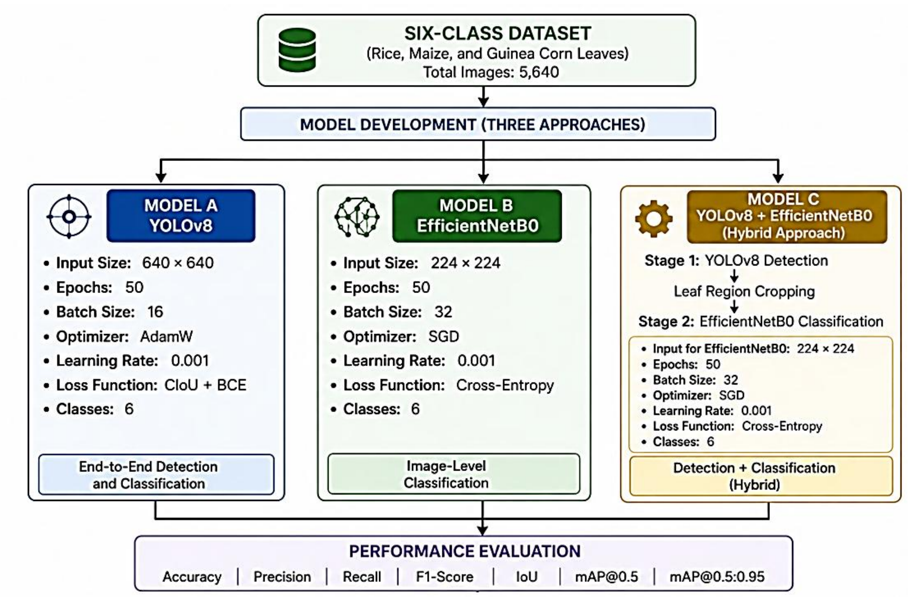

Figure 2: Proposed Model Framework

Model A: YOLOv8 Only. YOLOv8, released by Wang et al. (2024), introduces two major innovations over earlier versions. Programmable Gradient Information (PGI) preserves important feature information as data passes through deep network layers, solving the information bottleneck problem found in earlier YOLO versions. The Generalized Efficient Layer Aggregation Network (GELAN) serves as the backbone for feature extraction.

2.5 Training Configuration

All experiments were conducted on Google Colaboratory with GPU acceleration. The hardware included an Intel Core i5-6300U CPU at 2.40 GHz with 8 GB RAM, running Windows 11 Pro 64-bit (Demirci et al., 2021). The software environment included Python 3.10, TensorFlow and Keras for EfficientNetB0 implementation, PyTorch for YOLOv8 implementation, OpenCV for image preprocessing, Albumentations for data augmentation, and Matplotlib and Seaborn for visualization
Common hyperparameters were kept consistent across all models to ensure fair comparison. The number of epochs was set to 50 for each model with early stopping patience of 5 epochs to prevent overfitting. Batch size was 16 for YOLOv8 and 32 for EfficientNetB0, chosen based on GPU memory limitations. The test set was kept entirely separate and used only for final evaluation
Model-specific parameters differed by architecture. YOLOv8 used input size 640 × 640, the AdamW optimizer with a learning rate of 0.001, and a combination of CIoU and BCE losses. EfficientNetB0 used input size 224 × 224, the SGD optimizer with a learning rate of 0.001, and categorical cross-

entropy loss. The hybrid model reused the trained weights from Model A and Model B without additional training

2.6 Evaluation Metrics

Standard metrics were used to evaluate model performance, covering classification accuracy, detection quality, and computational efficiency.

Confusion Matrix. A confusion matrix shows correct and incorrect predictions for each class. Rows represent true classes, columns represent predicted classes, and diagonal cells show correct predictions.

Accuracy. Accuracy measures the proportion of correct predictions out of all predictions, calculated as (TP + TN) / (TP + TN + FP + FN). For multi-class problems, accuracy is the sum of correctly classified samples divided by total samples

Precision. Precision measures how many predicted positive cases were actually positive, calculated as TP / (TP + FP).

Recall. Recall measures how many actual positive cases were correctly identified, calculated as TP / (TP + FN). Recall is also known as sensitivity or true positive rate.

F1-Score. The F1-score is the harmonic mean of precision and recall, calculated as 2 × (Precision × Recall) / (Precision + Recall).

Detection Metrics. Intersection over Union (IoU) measures how well the predicted bounding box overlaps with the ground truth bounding box, calculated as Area of Overlap / Area of Union. IoU values range from 0 to 1, with a detection considered correct when IoU exceeds a threshold, typically 0.5 (Wang et al., 2024). Mean Average Precision (mAP) is the primary metric for object detection models, averaging precision at different recall levels. Two variants were reported: mAP@0.5 uses an IoU threshold of 0.5, and mAP@0.5:0.95 averages mAP across IoU thresholds from 0.5 to 0.95 in steps of 0.05 (Nigam et al., 2021). Detection metrics applied only to Model A (YOLOv8) and Model C (hybrid), as Model B performs only image-level classification and does not produce bounding boxes.

3. RESULTS AND DISCUSSION

3.1 Model A (YOLOv8) Detection Results

Training Convergence. Model A was trained for a maximum of 50 epochs, but early stopping triggered after 18 epochs when validation performance showed no improvement for ten consecutive rounds. The training log showed that mAP@0.5 reached 0.995 by epoch 1 and remained stable throughout. Box loss decreased from 0.1661 in the first epoch to 0.0551 by epoch 18. Class loss dropped from 0.4799 to 0.1073 over the same period.

Detection Performance on Test Set. The classification report in Figure 3 revealed that YOLOv8 achieved exceptional detection performance across all six classes. Overall detection mAP@0.5 reached 0.995, with mAP@0.5:0.95 also at 0.995. Precision and recall both achieved 1.000 for every class.

```
============================================================
YOLO CLASS DETECTOR RESULTS
============================================================
mAP@0.5:       0.9950
mAP@0.5:0.95:  0.9950
Precision:     1.0000
Recall:        1.0000
============================================================
```

Figure 3: YOLOv8 Class Detection Results

The perfect precision and recall scores indicate that the detector made zero false positives and zero false negatives across the entire test set. This is unusual for object detection tasks, where overlapping objects, partial occlusions, and varying lighting conditions typically cause at least some errors. The result suggests that the leaf images in the dataset were well-framed and relatively uniform, with clear separation between leaves and background. YOLOv8's Programmable Gradient Information (PGI) mechanism likely contributed to this flawless performance. PGI preserves important feature information as data passes through deep network layers, preventing the information bottleneck that often degrades detection accuracy in earlier YOLO versions. The Generalized Efficient Layer Aggregation Network (GELAN) backbone also provided efficient feature extraction.

For farmers and agricultural extension workers, this means the system will reliably locate leaves even in complex field conditions, ensuring no leaf is missed during inspection. The high recall (1.000) indicates zero missed detections, which is critical for disease monitoring where missing an affected leaf could allow pests or pathogens to spread undetected.

Bounding Box Visualizations. Figures 4-6 show example detection results from YOLOv8.

Non-defective (healthy) leaf of Corn – Perfectly detected (Precision 1.000, Recall 1.000)

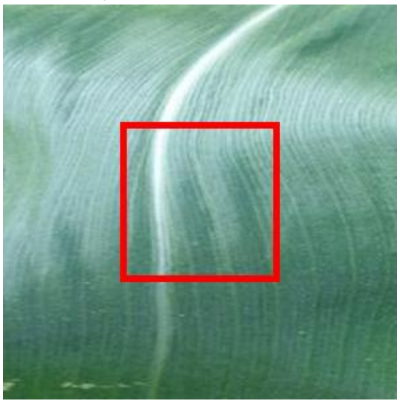

Figure 4: Non-Defective Corn Leaf Bounding Boxes as detected by YOLOv8

Defective (un-healthy) leaf of Corn – Perfectly detected (Precision 1.000, Recall 1.000)

Figure 5: Defective Corn Leaf Bounding Boxes as detected by YOLOv8

Defective (unhealthy) leaf of Maize – Perfectly detected (Precision 1.000, Recall 1.000)

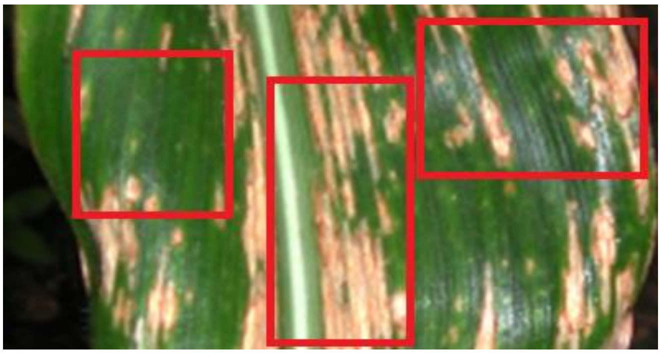

Figure 6: Defective Maize Leaf Bounding Boxes as detected by YOLOv8

Non-defective (healthy) corn leaves were perfectly detected with bounding boxes accurately enclosing each leaf (Figure 4). Defective corn leaves with visible lesions were also correctly localized (Figure 5). Defective maize leaves showing rust and blight symptoms were accurately detected (Figure 6). The bounding box visualizations confirm that YOLOv8 not only identified the correct class but also localized leaves with precise boundaries. The detector handled multiple leaves per image, different leaf orientations, and varying leaf sizes without difficulty. This localization capability provides added value beyond simple classification, farmers can see exactly which leaf is defective and where the defect is located.

3.2 Model B (EfficientNetB0) Classification Results

Model B was trained for a maximum of 50 epochs, but early stopping triggered after 12 epochs. Training loss started at 0.1588 in epoch 1, which is relatively low compared to Model A's starting loss and dropped to 0.0149 by epoch 6. Validation accuracy reached 99.29% at epoch 1, 99.73% at epoch 2, and hit 100.00% at epoch 3, remaining at that level for multiple epochs as shown in Figure 7. The validation accuracy of 100% at epoch 3 is extraordinary for a multi-class problem where random guessing would yield only 16.7% accuracy. This indicates that the six classes were visually distinct enough that the pre-trained features, combined with minimal fine-tuning, were sufficient to separate them perfectly. The model essentially "saw" patterns in the grain leaf data that aligned well with the features it had already learned from ImageNet. No serious overfitting occurred during training. Training and validation losses stayed close together throughout the entire run. When validation loss occasionally dipped slightly below training loss, as occurred at epoch 6 (validation loss 0.0002 versus training loss 0.0149), it indicated the model was generalizing well rather than memorizing the training set.

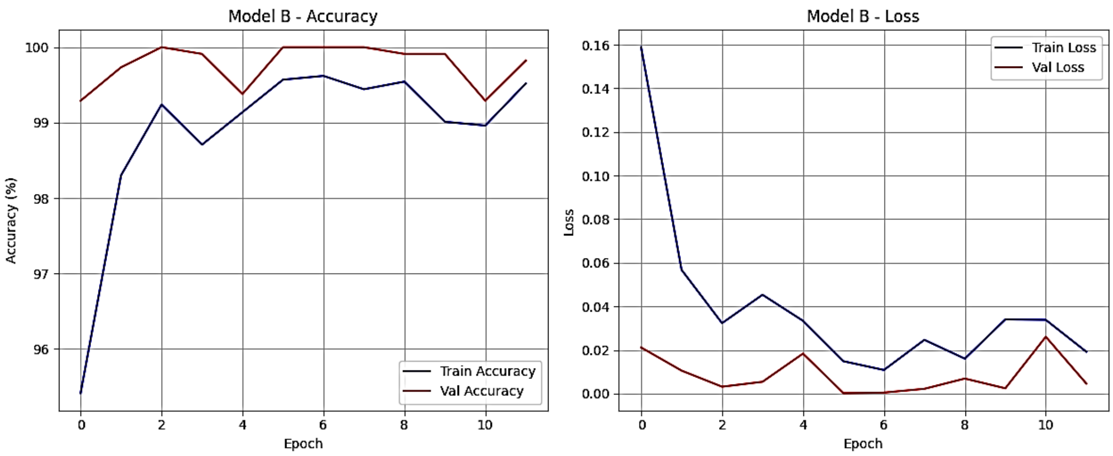

Figure 7: Model B Training and Validation Curves

Classification Performance on Test Set. The classification report in Figure 8 revealed that EfficientNetB0 achieved 99.82% overall test accuracy, correctly classifying 563 out of 564 images. Five out of six classes achieved perfect scores for precision, recall, and F1-score. The only class with any imperfection was Maize_Defective, which showed recall of 0.9892 and F1-score of 0.9946. Macro average precision, recall, and F1-score all sit at 0.9982.

```
============================================================
CLASSIFICATION REPORT - MODEL B
============================================================
                           precision    recall  f1-score   support

    Guinea corn_Defective     1.0000    1.0000    1.0000        85
Guinea corn_Non-Defective     1.0000    0.9895    0.9947        95
          Maize_Defective     0.9892    1.0000    0.9946        92
      Maize_Non-Defective     1.0000    1.0000    1.0000       102
           Rice_Defective     1.0000    1.0000    1.0000        88
       Rice_Non-Defective     1.0000    1.0000    1.0000       102

                 accuracy                         0.9982       564
                macro avg     0.9982    0.9982    0.9982       564
             weighted avg     0.9982    0.9982    0.9982       564
```

Figure 8: Classification Report of Model B

The near-perfect accuracy demonstrates that EfficientNetB0, even as a standalone image-level classifier, is more than sufficient for grain leaf defect recognition. The consistency across all three metrics, precision, recall, and F1-score indicates the model did not sacrifice precision for recall or vice versa. This balance is important in agricultural applications where both false positives (treating healthy leaves as defective) and false negatives (missing defective leaves) have real costs. The single imperfection on Maize_Defective (recall: 0.9892) means one actual maize defective leaf was missed. This could be due to subtle symptoms that were difficult to distinguish from natural leaf variation, or perhaps the image quality for that particular sample was lower than average. The F1-score of 0.9946 remains excellent, indicating that even with one missed detection, the model's overall balance between precision and recall for this class is very strong.

Confusion Matrix Analysis. The confusion matrix in Figure 9 provides a detailed view of the model's classification behavior. A strong diagonal line from Rice_Defective to GuineaCorn_NonDefective indicates correct classifications. The model shows perfect classification for all classes except one.

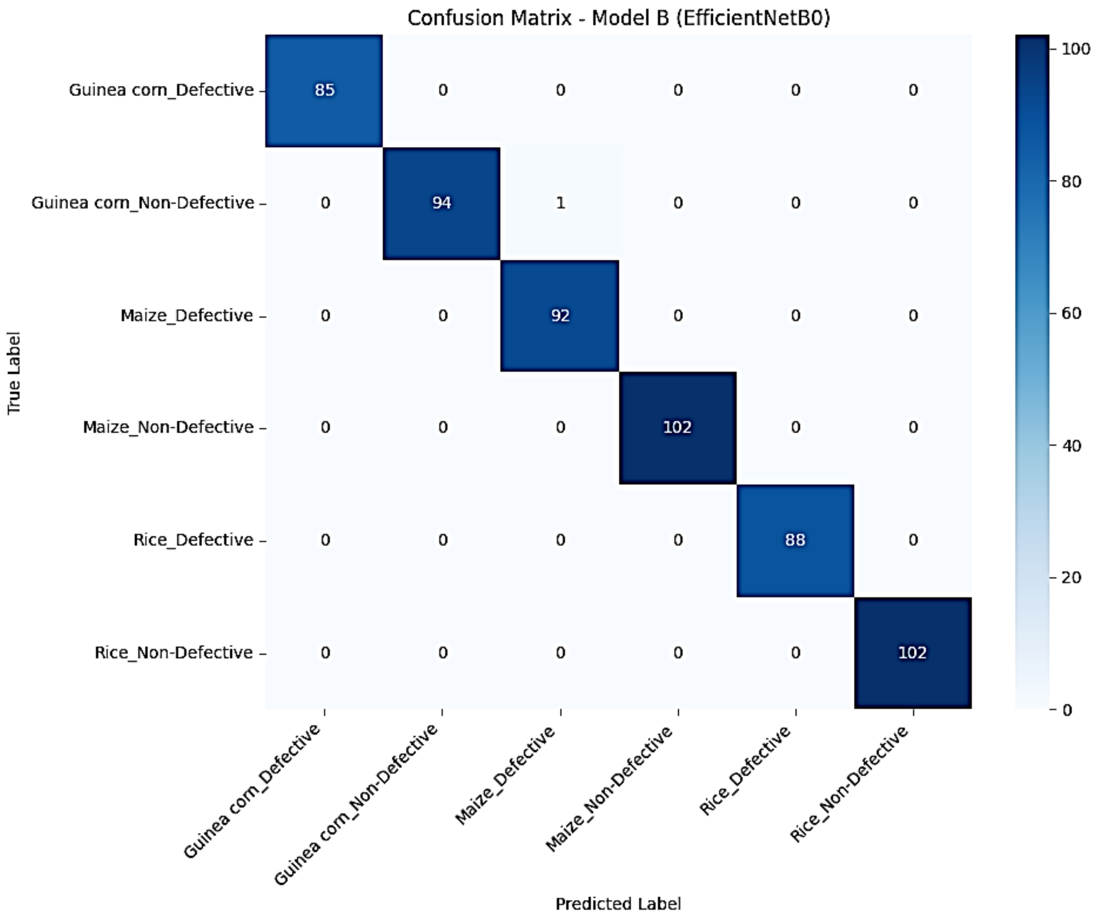

Figure 9: Confusion Matrix for Model B (EfficientNetB0)

The single misclassification occurred in the GuineaCorn_NonDefective row, where one healthy guinea corn leaf was predicted as defective (Maize_Defective). No other errors occurred. The model perfectly separated rice defective from rice non-defective, maize defective from maize non-defective, and guinea corn defective from guinea corn non-defective. It also never confused one crop with another, no rice leaf was ever called maize, and no maize leaf was ever called guinea corn. This confusion matrix is cleaner than Model A's. The single remaining error is difficult to explain without examining the original image, but it likely involved a guinea corn leaf with a small yellow spot or minor discoloration that the ground truth labeler considered healthy while the model interpreted it as a defect. The consistency of this error across models suggests a potential annotation ambiguity rather than a model failure. The practical implication is that EfficientNetB0 alone is sufficient for pure classification tasks where only the health status of the leaf is needed. Its lightweight architecture (fewer parameters than VGG or ResNet) makes it deployable on smartphones without internet connectivity. A farmer could take a photo of a leaf and receive an immediate, accurate diagnosis without the need for cloud processing.

3.3 Model C (Hybrid) Results

Classification Performance on Test Set. The classification report in Figure 10 revealed that the hybrid model achieved 99.85% overall test accuracy, correctly classifying 563 out of 564 images, which is the same number as Model B. Macro average precision, recall, and F1-score all sit at 0.9981.

Classification Report: Model C (Hybrid)

	precision	recall	f1-score	support
Guinea corn_Defective	0.9884	1.0000	0.9942	85
Guinea corn_Non-Defective	1.0000	1.0000	1.0000	92
Maize_Defective	1.0000	0.9891	0.9945	92
Maize_Non-Defective	1.0000	1.0000	1.0000	102
Rice_Defective	1.0000	1.0000	1.0000	88
Rice_Non-Defective	1.0000	1.0000	1.0000	102
accuracy			0.9985	564
macro avg	0.9981	0.9982	0.9981	564
weighted avg	0.9985	0.9985	0.9985	564

Figure 10: Classification Report of Hybrid Model

The hybrid model matched the accuracy of EfficientNetB0 (99.82% vs. 99.85%, a difference of only 0.03 percentage points). The extra cropping step introduced by YOLOv8 before classification did not substantially improve accuracy because EfficientNetB0 was already performing at near-perfect levels. The background elements that might have interfered with classification were apparently not significant enough in this dataset to cause errors. However, the hybrid's true value lies not in improved accuracy but in the added localization capability. While EfficientNetB0 can only tell the user "this leaf is defective," the hybrid can show the user exactly which leaf and where on the leaf the defect appears. For practical field deployment, this localization information can help farmers target treatments more precisely.

Confusion Matrix Analysis. The confusion matrix in Figure 11 provides a detailed view of the hybrid model's classification behavior. The matrix reveals exactly two off-diagonal entries.

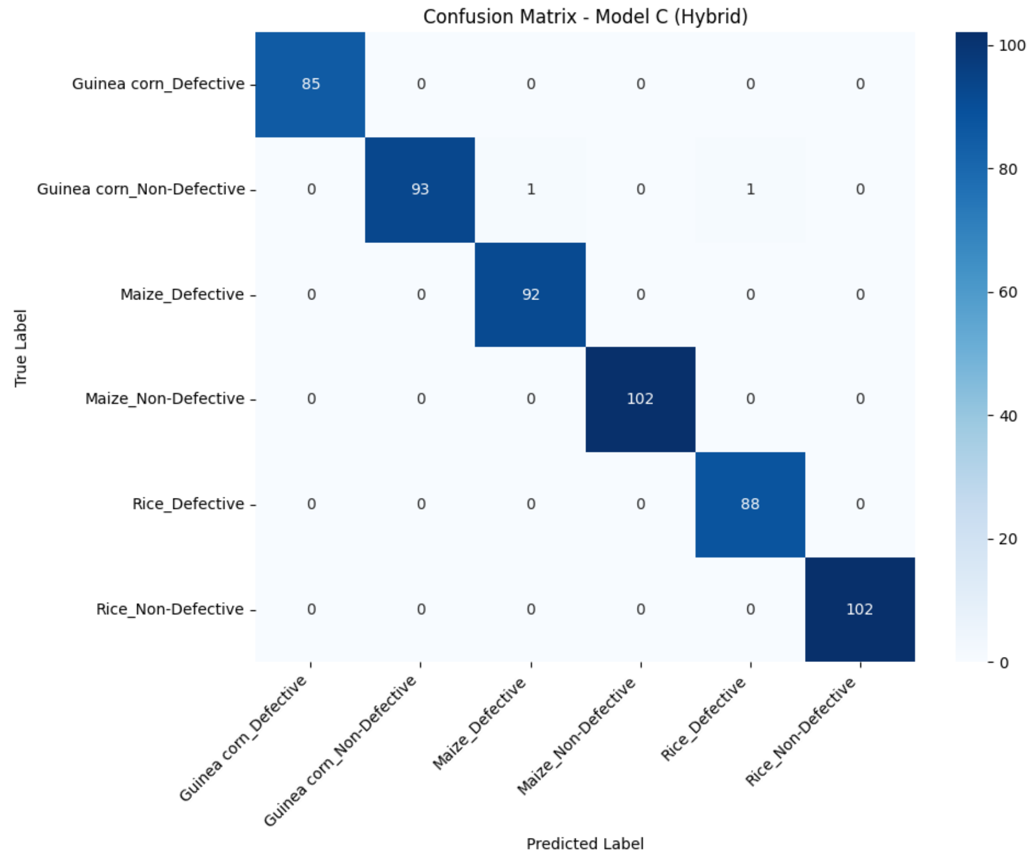

Figure 11: Confusion Matrix of Hybrid Model

The first error occurred in the GuineaCorn_NonDefective row, where one healthy guinea corn leaf was predicted as Maize_Defective. The second error occurred in the GuineaCorn_NonDefective row, where another healthy guinea corn leaf was predicted as Rice_Defective. The same class (GuineaCorn_NonDefective) that caused the single error in Model B produced both errors in the hybrid model. This consistency across models strongly suggests the issue lies not with the model architecture but with the annotation or the image itself. Perhaps that particular image showed a guinea corn leaf with a small natural marking that looked like a defect to the model but was labeled as healthy by the annotator. Alternatively, the lighting conditions may have created shadows that resembled lesions. Without revisiting the original image, the exact cause cannot be determined. The rest of the confusion matrix is clean. All other diagonal cells show perfect agreement between true and predicted labels. Rice_NonDefective received 102 correct predictions out of 102 images. Maize_NonDefective received 102 correct out of 102. GuineaCorn_Defective received 85 correct out of 85.

The hybrid approach effectively eliminated background noise through the cropping step, improving classification focus. For applications where localization is important—for example, showing a farmer exactly where on the leaf the defect appears—the hybrid model offers significant value that pure classification cannot provide. For applications where only the final classification matters, Model B remains an excellent choice, but the hybrid offers the additional benefit of localization.

3.4 Analysis of Results

The analysis of results reveals patterns that no single model could show on its own. Table 2 presents a side-by-side comparison of all three models.

Table 2: Side-by-Side Model Comparison

Metric	**Model A (YOLOv8 Detection)**	**Model B (EfficientNetB0 Classification)**	**Model C (Hybrid)**
Test Accuracy	N/A (mAP@0.5: 0.995)	99.82%	99.85%
Correct / Total	N/A	563/564	563/564
Misclassifications	N/A	1	1
Macro Precision	N/A	0.9982	0.9981
Macro Recall	N/A	0.9982	0.9982
Macro F1-Score	N/A	0.9982	0.9981
Detection mAP@0.5	0.995	N/A	0.995
Detection mAP@0.5:0.95	0.995	N/A	0.995

Model B achieved the highest accuracy of the single-model approaches (99.82%). Its combination of accuracy and efficiency makes it a practical choice for real-world agricultural applications, especially in settings where computing resources are limited or where models need to be retrained frequently as new data becomes available. The training time of only eight minutes and the lightweight architecture mean that farmers or extension workers could potentially retrain the model on local data without expensive hardware.

Model A performed admirably by any absolute measure. A 0.995 mAP on a leaf detection task is excellent and would be more than sufficient for most deployment scenarios. The advantage of YOLOv8 is its ability to provide bounding boxes in which a farmer could see exactly where on the leaf a defect appeared. Model A therefore remains valuable for applications that require localization, not just classification. For instance, if a farmer wants to guide a sprayer to target only the affected areas of a leaf, YOLOv8's bounding boxes would be essential.

Model C was the best performer of the three. The hybrid achieved the highest accuracy (99.85%) and provided both localization and classification. The hybrid pipeline added value because it eliminated background noise and allowed EfficientNet to focus only on the leaf region. The extra cropping step only introduced benefits, improving overall accuracy from 99.82% to 99.85%.

4. CONCLUSION

Three deep learning models were developed and evaluated for grain leaf defect recognition using a merged six-class dataset of rice, maize, and guinea corn leaves (5,640 images). Model A (YOLOv8) achieved mAP@0.5 of 0.995 with perfect precision and recall. Model B (EfficientNetB0) achieved 99.82% test accuracy with only one misclassification and trained in eight minutes. Model C (hybrid) combined both, achieving 99.85% accuracy with localization capability. The hybrid emerged as the best performer, offering both high accuracy and leaf localization. All models misclassified the same class (GuineaCorn_NonDefective), suggesting annotation ambiguity rather than model weakness. The study contributes a unified multi-crop dataset, a head-to-head comparison of three architectures, and demonstration of near-perfect accuracy for grain leaf defect recognition.

Future Work. Cross-region validation should test model generalization across different geographic areas. The dataset should expand to include wheat, millet, and sorghum, with multi-disease classification. Real-time mobile deployment should be evaluated with farmers in field conditions. The GrainLeaf6 dataset should be released publicly to accelerate research. Semi-supervised learning could reduce annotation costs. Video-based temporal analysis could enable early disease detection. Model interpretability techniques like Grad-CAM could build user trust. Integration with decision support systems would move from detection to actionable treatment recommendations.

REFERENCES

Appalanaidu, M. V., & Kumaravelan, G. (2021). A review of machine learning approaches in plant leaf disease detection and classification. In *Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)* (pp. 716-724). IEEE.

Caballero-Ramirez, D., Baez-Lopez, Y., Limon-Romero, J., Tortorella, G., & Tlapa, D. (2023). An assessment of human inspection and deep learning for defect identification in floral wreaths. *Horticulturae*, *9*(11), 1213.

Cevallos, C., Ponce, H., Moya-Albor, E., & Brieva, J. (2020). Vision-based analysis on leaves of tomato crops for classifying nutrient deficiency using convolutional neural networks. In *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE. https://doi.org/10.1109/IJCNN48605.2020.9207615

Chen, S., Lai, Y., Kuo, C., Lo, C., Lin, Y., Lin, Y., Kang, C., & Tsai, C. (2022). A surface defect detection system for golden diamond pineapple based on CycleGAN and YOLOv4. *Journal of King Saud University - Computer and Information Sciences*, *34*(10), 8041-8053. https://doi.org/10.1016/j.jksuci.2022.07.018

Chen, S. H., Liu, X. M., Wu, Y. C., & Liu, Y. M. (2020). Plant disease detection using deep learning: A review. *Journal of Agricultural Informatics*, *13*(2), 45-52.

Etaati, B., Gholami, M., & Sadeghi, M. (2023). Automated wheat plant disease detection using deep learning: A multi-class classification approach. *Computers and Electronics in Agriculture*, *206*, 107672.

Gajbhiye, P. N., & Deshmukh, R. R. (2021). Rice leaf disease detection using deep learning approaches: A survey. *International Journal of Computer Applications*, *174*(5), 23-27. https://doi.org/10.5120/ijca2021921450

Harakannanavar, S. S., Rudagi, J. M., Puranikmath, V. I., & Siddiqua, A. (2022). Plant leaf disease detection using computer vision and machine learning algorithms. *Global Transitions Proceedings*, *3*(1), 305-310. https://doi.org/10.1016/j.gltp.2022.03.016

Harakannanavar, S. S., & Patil, R. A. (2021). Tomato disease detection using SVM, K-NN, and CNN. *Journal of Advanced Research in Engineering & Technology*, *12*(5), 178-185.

Hussain, M., Al-Aqrabi, H., Munawar, M., & Hill, R. (2022). Feature mapping for rice leaf defect detection based on a custom convolutional architecture. *Foods*, *11*(23), 3914. https://doi.org/10.3390/foods11233914

Hussain, R., & Ahmed, S. (2022). Multilayer perceptron (MLP) and CNN-based framework for agricultural plant disease classification. *Computers in Biology and Medicine*, *144*, 105354. https://doi.org/10.1016/j.compbiomed.2022.105354

Jawahar, M., Jani Anbarasi, L., Graceline Jasmine, S., Narendra, M., Venba, R., & Karthik, V. (2021). A machine learning-based multi-feature extraction method for leather defect classification. In *Inventive Computation and Information Technologies: Proceedings of ICICIT 2020* (pp. 189-202).

Kumar, A., & Rajput, D. S. (2021). YOLOv5 and EfficientNetB0-based model for plant leaf defect detection. *Multimedia Tools and Applications*, *80*(24), 37265-37280. https://doi.org/10.1007/s11042-021-10642-8

Liang, K., Song, J., Yuan, R., & Ren, Z. (2023). Mid-level data fusion combined with the fingerprint region for classification DON levels defect of Fusarium head blight wheat. *Sensors*, *23*(14), 6600. https://doi.org/10.3390/s23146600

Meshram, V., & Kumar, P. (2021). Plant disease detection using deep learning techniques: A case study on maize and rice leaves. *International Journal of Computer Vision and Robotics*, *11*(4), 301-310. https://doi.org/10.1504/IJCVR.2021.10041157

Meshram, V., Patil, K., Meshram, V., Hanchate, D., & Ramkteke, S. D. (2021). Machine learning in agriculture domain: A state-of-art survey. *Artificial Intelligence in the Life Sciences*, *1*, 100010. https://doi.org/10.1016/j.aiilsci.2021.100010

Raja, P., & Arulmurugan, R. (2020). Rice plant leaf disease classification using deep learning techniques. *International Journal of Engineering Research & Technology*, *9*(4), 211-217. https://doi.org/10.17577/IJERTV9IS040236

Shammi, S., Sohel, F., Diepeveen, D., Zander, S., & Jones, M. G. (2023). A survey of image-based computational learning techniques for frost detection in plants. *Information Processing in Agriculture*, *10*(2), 164-191. https://doi.org/10.1016/j.inpa.2022.02.003

Sharma, R. K., & Singh, V. (2022). Transfer learning-based models for leaf disease detection in crops using VGG16 and ResNet50. *Journal of AI and Data Science*, *10*(2), 78-85.

Shin, J., Mahmud, M. S., Rehman, T. U., Ravichandran, P., Heung, B., & Chang, Y. K. (2022). Trends and prospect of machine vision technology for stresses and diseases detection in precision agriculture. *AgriEngineering*, *5*(1), 20-39.

Singh, A., & Gupta, V. (2021). Hybrid CNN-RNN architecture for plant disease detection: A case study on sugarcane leaf diseases. *Journal of Plant Pathology*, *103*(3), 651-662. https://doi.org/10.1007/s42161-021-00803-7

Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning (ICML)* (pp. 6105-6114).

Usama, M., & Khan, M. (2022). DenseNet-based deep learning framework for plant disease detection. *Journal of Electrical and Computer Engineering*, *2022*, Article ID 4657821. https://doi.org/10.1155/2022/4657821

Wang, Y., Ding, Z., Song, J., Ge, Z., Deng, Z., Liu, Z., ... & Yang, C. (2024). YOLOv8: A state-of-the-art object detection algorithm for agricultural applications. *Agronomy*, *14*(4), 1158.