International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V72I6P120
Volume 72 Issue 6, 195-203, June 2024

Original Article

Corn Crop Disease Detection Using Convolutional Neural Network (CNN) to Support Smart Agricultural Farming

Jovelin M. Lapates

Bukidnon State University, Malaybalay City, Philippines.

Corresponding Author : j.lapates@buksu.edu.ph

Received: 23 February 2024 Revised: 15 May 2024 Accepted: 21 May 2024 Published: 29 June 2024

Abstract - *Corn crop disease detection is crucial in ensuring crop health and optimizing agricultural productivity. This study explores the implementation of the YOLOv8 algorithm for efficient and accurate disease detection in corn crops. The research focuses on detecting diseases such as blight, common rust, and gray leaf spot, significantly impacting crop yield and quality. The YOLOv8 model trains using a carefully annotated dataset of corn leaf images, encompassing both disease-infected samples and healthy leaves. The model performance is evaluated using a separate test set, and promising outcomes are observed, with high mean Average Precision (mAP) values achieved across different disease categories. Notably, the model demonstrates exceptional accuracy in recognizing healthy corn plants, with an mAP of 0.99 for the healthy class. These results indicate the potential of YOLOv8 as an effective tool for early disease detection and precise interventions in smart agricultural farming. The findings of this study contribute to the advancement of automated disease detection systems in agriculture, paving the way for improved crop management practices and optimized yields in corn farming.*

Keywords - *Corn, Crop disease, YOLOv8, Smart agricultural farming, CNN.*

1. Introduction

As the world population grows, ensuring sustainable food production becomes a critical challenge. Smart agricultural practices have emerged as a promising solution to optimize crop yields and minimize environmental impact. One key aspect of smart farming is the early detection and management of crop diseases, which can significantly reduce agricultural productivity. Among the numerous crops affected by diseases, corn is a vital staple crop with a high susceptibility to various pathogens.

Machine learning algorithms have revolutionized the way people see technology. From a simple machine that reduces human effort and aids them in their respective tasks, they have become capable systems that can work independently and efficiently. Simple and complex approaches exist for detecting static and dynamic objects in an image. There are ways to detect images based on the structure of the image, the depth field, and image geometry [1]. In deep learning, the detection is performed by using representation-learning algorithms. These representations are expressed in terms of other, simpler representations. In other words, a deep learning system can represent the concept of an image for an object by combining basic concepts, such as points and lines, which it defines in terms of edges. According to [2], a fundamental aspect of deep learning in image classification is the use of Convolutional architectures. The model learns to detect objects as they occur. We can best achieve this through a universal and open-source library available today. These universal and open-source libraries enable the use of multiple algorithms for a wide range of datasets.

Corn is the most significant locally grown crop in the Philippines. It reaches maturity of 105 to 110 days, with a potential yield of 15 MT/ha. Double-cropping can serve as an alternative to current practices in areas with very long growing seasons, which have been shown to increase with a warming climate. According to the PSA, Northern Mindanao had the highest production at 558.9 thousand metric tons (28.2%), followed by Cagayan Valley with 432.1 thousand metric tons (21.8%). The average domestic production from 1996 to 2000 was 4.28 million tons, while the average importation volume for the same period amounted to 253,000 tons. The US and Argentina supplied most of the country's corn imports. Bukidnon, Isabela, and South Cotabato contribute the largest volume of domestically cultivated corn. Tarlac (4.3 t/ha) and Pangasinan (3.10 t/ha) generated the highest five-year provincial average corn productivity compared to the current national average yield of 1.8 t/ha. For yellow corn, 80% of corn growers plant hybrid varieties, while 78% of all white corn-growing areas are planted with traditional varieties. In Bukidnon, the corn physical area covered 109,215.29 hectares, with average yields per hectare ranging from 4.36 in 2017 to 4.63 in 2022. Bulacan, Batangas, and Rizal are the top consumers of feed corn, while Cebu, Zamboanga del Sur, and Zamboanga del Norte have the highest demand for local corn. Siquijor (95 kg), Negros Oriental (65 kg), and Zamboanga del Sur (60 kg) have the highest per capita consumption of corn, while the national per capita average is 11kg.

Crop diseases pose one of the main challenges to the agricultural sector worldwide. These diseases can result in significant losses in crop yield, quality, and economic revenue for farmers. Traditional methods of disease detection and diagnosis are time-consuming and require specialized knowledge, making it difficult for farmers to take action to prevent the spread of diseases quickly.

However, with the recent advancements in computer vision techniques and machine learning algorithms, utilizing technology to support smart agricultural farming has become increasingly popular. Convolutional Neural Networks (CNNs) are a type of deep learning algorithm that has shown significant promise in image recognition tasks, including detecting crop diseases [3, 4, 5, 6, 7].

While research on corn crop disease detection using YOLOv8 has shown promising results, several gaps and challenges remain. One primary gap is the need for extensive and diverse datasets encompassing a wide range of disease symptoms under different environmental conditions and growth stages [8]. Many current datasets have limited scope, potentially hindering the model's ability to generalize effectively across various scenarios [9].

Additionally, the high computational demands of YOLOv8 may pose challenges for deployment in resource-limited settings, such as small-scale farms lacking access to advanced hardware [10]. There is also a need for more research on the integration of YOLOv8 with other sensor data, such as hyperspectral or thermal imaging, to improve detection accuracy further [11].

Moreover, existing studies often focus on detecting common diseases, leaving rare or emerging diseases underexplored [12]. Finally, while YOLOv8 performs well in controlled conditions, real-world applications require robust models that can handle occlusions, varying lighting conditions, and overlapping plant structures [8]. Addressing these gaps through continued research and development is crucial for realizing the full potential of YOLOv8 in precision agriculture.

The outcome of this study holds great potential for improving traditional farming practices by incorporating real-time disease detection and decision-making capabilities. By integrating the developed YOLOv8-based system into smart farming platforms, farmers and agricultural stakeholders can proactively manage crop diseases, optimize resource allocation, and ultimately enhance food security and sustainability. [13] compared the performance of pre-trained models for object detection using the TensorFlow framework, as shown in Table 1.

The Table 1 shows the advantages and disadvantages of methods Fast R-CNN (Region-Based Convolutional Neural Network), Faster R-CNN, R-FCN (Region-based Fully Convolutional Network), SSD (Single-Shot Detector) and YOLO (You Only Look Once) for detecting objects in images is high-lighted by various experiments conducted by a [14, 15, 16] investigated in Deep Learning area and in recent years, many results are measured exclusively with the data set MS detection COCO objects.

While deep learning image detection algorithms [17], such as YOLOv3, YOLOv4, YOLOv5, Faster R-CNN, SSD, RetinaNet, Mask R-CNN, EfficientDet, MobileNet-SSD, DeepLabv3+, and U-Net, have shown significant advancements in various applications, there are still specific gaps and limitations that researchers are actively working on addressing.

Table 1. Advantages and disadvantages of different methods in detecting objects in images

Method	Authors	Advantages	Disadvantages
Fast R-CNN	(Girshick, 2017)	The calculation of the characteristics of CNN is performed in a single iteration, resulting in object detection that is 25 times faster than the RCNN method (which requires an average of 20 seconds to analyse an image).	Using an external candidate region generator creates a bottleneck in the detection process.
Faster R-CNN	(Renet al., 2015)	The RPN method allows object detection to be almost real-time, approximately 0.12 seconds per image.	Despite the efficiency of the algorithm, it is not fast enough to be used in real-time applications, such as autonomous vehicles.
R-FCN	(Dai et al., 2016)	The test time of R-FCN is much faster than that of R-CNN	R-FCN has a competitive mAP but lower than that of Faster R-CNN.
Mask R-CNN	(He et al., 2017)	The location of the objects is more precise when segmenting the objects in the images.	Its execution time is greater than that used by the Faster-RCNN method, therefore, it can not be be implemented in applications that require real time.
YOLO	(Redmon et al., 2015)	The location of objects is very efficient, allowing for its use in real-time applications.	The method struggles to accurately detect small objects.
SSD	(Liu et al., 2016)	The use of a single network makes object localization faster than the Fast-RCNN and FasterRCNN methods	The detection accuracy of the objects is lower compared to that of the Fast-RCNN and Faster-RCNN methods.

Table 2. Comparative analysis of deep learning image detection methods

Method	Accuracy	Speed	Model Size	Dataset Size	Application
YOLOv3	High	Fast	Moderate	Large	Object Detection
YOLOv4	Very High	Fast	Large	Large	Object Detection
YOLOv5	High	Very Fast	Small	Moderate	Object Detection
YOLOv8	High	Fast	Moderate	Large	Object Detection
Faster R-CNN	High	Moderate	Large	Large	Object Detection
SSD	Moderate	Fast	Small	Moderate	Object Detection
RetinaNet	High	Slow	Large	Large	Object Detection
Mask R-CNN	High	Slow	Large	Large	Object Detection
EfficientDet	High	Fast	Small	Large	Object Detection
MobileNet-SSD	Moderate	Very Fast	Very Small	Small	Object Detection
DeepLabv3+	High	Slow	Large	Large	Semantic Segmentation
U-Net	High	Slow	Moderate	Moderate	Semantic Segmentation

The researcher has identified common gaps on:

1.1. YOLO Algorithm

Corn is a widely cultivated crop with significant economic and nutritional value. However, corn crops are vulnerable to diseases that can negatively impact yield and quality. Researchers have focused on utilizing YOLO-based systems, such as YOLOv3, YOLOv4, YOLOv5, and YOLOv8, for detecting and classifying diseases in corn crops. These efforts address the susceptibility of corn crops to diseases that can impact yield and quality. [18, 19] demonstrated the effectiveness of YOLOv3 in identifying corn crop diseases, while [20, 21] explored the improved accuracy and real-time monitoring capabilities of YOLOv4 and YOLOv5, respectively. [7] extended this research to YOLOv8, emphasizing its suitability for real-time disease detection in smart agricultural systems, including precision agriculture techniques.

These studies investigated the transfer learning capabilities of YOLOv8, finding that pretraining on large-scale datasets improved disease detection performance in corn crops. Moreover, while these studies showcase the potential of YOLO-based models in smart farming, further research is needed to assess their generalization across diverse environmental conditions and compatibility with resource-constrained farming settings.

1.2. Smart Agricultural Farming Applications

A literature review explores applications in crop production, resource management, environmental sustainability, and economic efficiency. Studies by [22, 23, 24] focus on IoT and UAV technologies, demonstrating their potential for real-time monitoring, data-driven decision-making, and enhanced efficiency in soil conditions, weather patterns, crop growth, aerial imaging, monitoring, and pesticide spraying.

Additionally, [25] explores the integration of blockchain for transparent data sharing, supply chain management, and traceability in the agri-food industry, while [26, 27] examines big data analytics for predictive modelling and decision support in sustainable agriculture. Effective integration can enhance the efficiency, reliability, and impact of smart agricultural systems on farming practices.

Agriculture is one of the most critical sectors that supports human life and the economy. However, crop diseases have been a major problem affecting crop production and food security. The use of technology to support smart farming [28, 29] has become increasingly popular in recent years, with the development of advanced computer vision techniques and machine learning algorithms. This study focuses on the application of Convolutional Neural Networks (CNNs) in detecting crop diseases to provide an accurate and efficient solution for farmers to detect crop diseases in their fields. The study explores the use of CNNs in image classification and identifies the best CNN model capable of detecting crop diseases with high accuracy. The results of this research contribute to the development of smart farming systems that help farmers make informed decisions about their crops, reduce crop losses, and increase agricultural productivity.

2. Objectives of the Study

The general objective of this study is to develop and evaluate a robust corn crop disease detection system using CNN, aiming to enhance smart agricultural practices. Specifically, the study aims to:

- Collect and annotate a dataset of corn crop images, encompassing various disease states and growth stages to serve as the training and evaluation data for the YOLOv8 model.
- Pre-process the corn crop dataset to enhance the model's ability to generalize and accurately detect different diseases under varying environmental conditions.
- Train and fine-tune the YOLOv8 model using the annotated dataset, optimizing its performance for accurate and efficient corn crop disease detection.
- Evaluate the performance of the developed YOLOv8-based system in terms of detection accuracy, precision, recall, and processing speed using appropriate evaluation metrics and benchmark datasets.
- Compare the performance of the YOLOv8 model with other existing object detection algorithms or models to assess its effectiveness in corn crop disease detection.
- Integrate the developed YOLOv8-based system into a smart agricultural farming platform or framework, enabling real-time disease detection and providing decision-support tools for farmers and agricultural stakeholders.

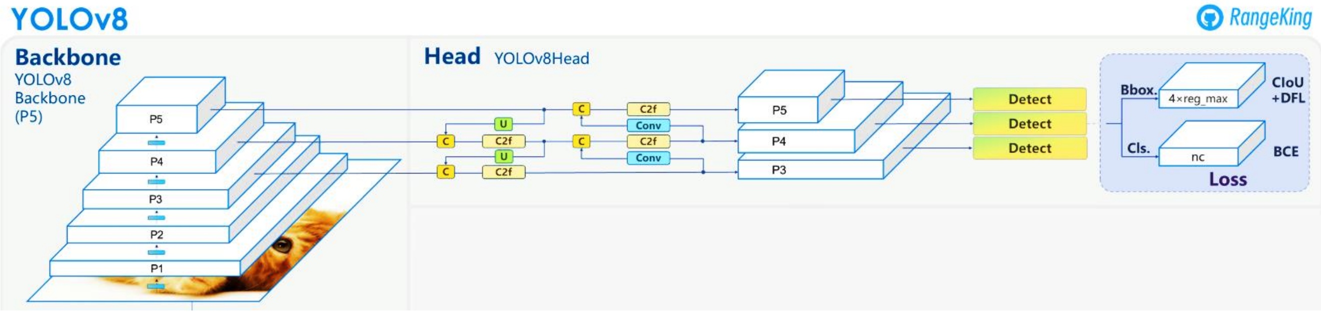


Fig. 1 YOLOv8 model architecture

- Validate the practical applicability of the YOLOv8-based system by conducting field trials or simulations, assessing its effectiveness in enabling early disease identification, facilitating prompt interventions, and minimizing the spread of infections across the field.

3. Methodology

The methods employed in this study aim to leverage the YOLOv8 model for the detection of corn crop diseases, contributing to the advancement of smart agricultural farming practices. YOLOv8 belongs to one-stage object detection models that process an entire image in a single forward pass of a CNN. YOLOv8 is known for its exceptional speed and accuracy in object detection, providing a robust framework for the early and precise identification of diseases in corn crops. It employs a multi-scale approach, processing different image resolutions to detect objects of various sizes, and utilizes a combination of anchor boxes and anchor clustering to improve localization accuracy. YOLOv8 also incorporates various architectural enhancements like CSPDarknet53 and PANet, which aid in feature extraction and context integration, leading to better object detection performance. By implementing this methodology, the study seeks to enable prompt intervention and prevent the spread of infections across the field, empowering farmers with valuable information for effective disease management. This section outlines the key steps involved in applying YOLOv8 for corn crop disease detection, including dataset preparation, model training, evaluation, and integration into smart agricultural systems.

3.1. Data Collection and Preparation

The dataset used in this study is obtained from Kaggle and consists of images of corn crops with four distinct parameters: blight, common burst, gray leaf spot, and healthy. These parameters represent different types of corn crop diseases and the healthy state of the crops. The dataset includes a diverse collection of images capturing various stages of growth, lighting conditions, and angles to ensure comprehensive coverage of the corn crop disease spectrum.

3.2. Data Pre-processing

In the data pre-processing stage, the researcher annotates all images in the dataset with the assistance of Makesense.ai. This annotation process involves labeling the images to indicate the presence of specific corn crops diseases such as blight, common burst, gray leaf spot, or the absence of any disease (healthy). Makesense.ai provided a user-friendly interface and tools to annotate the images accurately, ensuring that each image was properly labeled with the corresponding disease parameter. The annotated dataset was used as the foundation for training and evaluating the Convolutional Neural Network (CNN) model for detecting corn crop diseases.

3.3. Model Architecture Selection

The model architecture used in this study was YOLOv8. YOLOv8 (You Only Look Once version 8) is a state-of-the-art object detection algorithm that combines high accuracy and real-time performance. It utilizes a deep neural network architecture with multiple convolutional layers and advanced feature extraction techniques to detect and localize objects in images. By employing YOLOv8 as the model architecture, this study aimed to leverage its strengths in detecting and localizing corn crop diseases accurately and efficiently. The YOLOv8 architecture offers a robust framework for object detection tasks, including identifying and classifying specific diseases affecting corn crops in the provided dataset.

3.4. Model Training

The model training process involved 1.3k train images, 359 valid images, and 180 test images. These images were prepared and formatted using Roboflow, specifically in the YOLOv8 format. Roboflow is a platform that provides tools for data management and pre-processing in computer vision tasks. Using Roboflow in the YOLOv8 format, the images were appropriately organized and annotated with bounding boxes around the corn crop diseases of interest. This format ensures compatibility with the YOLOv8 model architecture and facilitates efficient training and evaluation processes. The researcher split the dataset into training, validation, and testing sets to ensure proper model training, performance assessment, and generalization evaluation.

The steps to train a YOLOv8 object detection model on custom data are:

1. Install YOLOv8 from pip using the command pip install ultralytics

a. Install the model from the source using these commands:

```
git clone https://github.com/ultralytics/ultralytics
cd ultralytics
pip install -e ultralytics
```

b. The YOLOv8 CLI

```
yolo task=detect \
mode=predict \
model=yolov8n.pt \
```

```
conf=0.25 \
source='https://media.roboflow.com/notebooks/examples/dog.jpeg
```

c. The YOLOv8 Python SDK

```
from ultralytics import YOLO
model = YOLO('yolov8n.pt')
model. predict (
source='https://media.roboflow.com/notebooks/examples/dog.jpeg',
conf=0.25
)
```

2. Create a custom dataset with labeled images
   a. Create a project with Roboflow
   b. Upload dataset
   c. Label images with Roboflow Annotate
   d. Generate a new version of a dataset
3. Export dataset for use with YOLOv8 pip package

```
rom roboflow import Roboflow
rf = Roboflow(api_key='YOUR_API_KEY')
project =
rf.workspace('WORKSPACE').project('PROJECT')
dataset = project.version(1).download('yolov8')
```

4. Use the YOLO command line utility to run train a model

```
yolo task=detect \
mode=train \
model=yolov8s.pt \
data={dataset.location}/data.yaml \
epochs=100 \
imgsz=64
```

5. Validate with a new model

```
yolo task=detect \
mode=val \
model={HOME}/runs/detect/train/weights/best.pt \
data={dataset.location}/data.yaml
```

3.5. Model Evaluation

The trained model was validated on the validation set to assess its performance. Measure metrics such as accuracy, precision, recall, and F1 score to quantify the model's ability to detect corn crop diseases accurately as follows:

3.5.1. Accuracy

It tells us how close the measured value is to a known value.

$$Accuracy=(TP+FN)/(TP+TF+FP+FN) \quad (1)$$

3.5.2. Precision

It tells about how accurate the model is in terms of those which were predicted positive.

$$Precision=TP/(TP+FP) \quad (2)$$

3.5.3. Recall

It calculates the number of actual positives the model captured after labeling it as positive (true positive).

$$Recall=TP/(TP+FN) \quad (3)$$

3.5.4. F1

It gives a balance between precision and recall.

$$F1=2\times(Precision\times Recall)/(Precision+Recall) \quad (4)$$

3.5.5. AUC Score and ROC Curve

ROC (receiver operating characteristics) is a probability curve, and AUC (area under the curve) represents the degree of separability. The ROC curve plots sensitivity (true positive rate) against specificity (false positive rate).

3.6. Hyperparameter Tuning

Hyperparameter tuning was performed to optimize the model's performance. Adjust hyperparameters such as learning rate, batch size, optimizer, and regularization techniques to find the optimal configuration.

3.7. Model Testing

Assess the final trained model's performance on the independent testing set to evaluate its generalization ability. Calculate various evaluation metrics to measure the model's accuracy and robustness in detecting corn crop diseases.

3.8. Deployment and Integration

Implement the trained model into a user-friendly interface or application that can accept input images of corn crops and provide disease detection results. Integrate the developed solution into existing smart farming systems or provide an API for easy integration with agricultural platforms.

4. Results and Discussions

The outcomes of the study are presented, and discussions, including insights derived from observations, model testing, and evaluation conducted in this investigation, are also included.

4.1. Data Classification

4.1.1. Healthy vs. Unhealthy Corn Leaf Image Samples

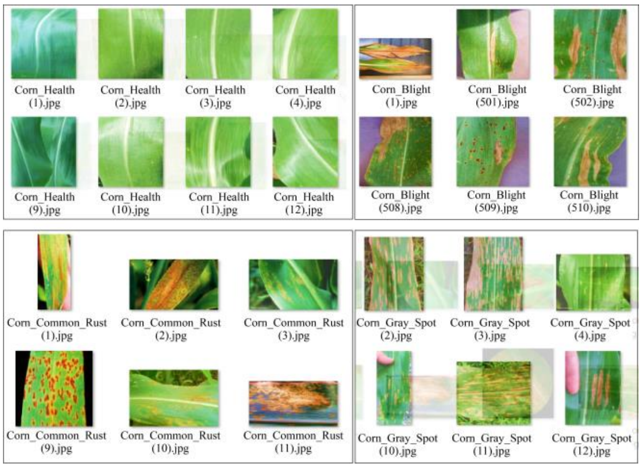


Fig. 2 Healthy vs. corn blight vs. gray spot vs. common rust

The four figures represent distinct categories within the dataset: blight, common rust, gray leaf spot, and healthy samples. The samples were annotated using makesense.ai and subsequently used for both training and testing with the YOLOv8 algorithm.

4.1.2. Annotated Corn Leaf Image Samples

Fig. 3 Annotated Images

The dataset was annotated with labels corresponding to each category, allowing YOLOVv8 to learn and classify new corn plant samples based on the presence or absence of these diseases. There are all 1794 approved annotated data.

4.1.3. Trained, Valid, and Test Corn Leaf Images

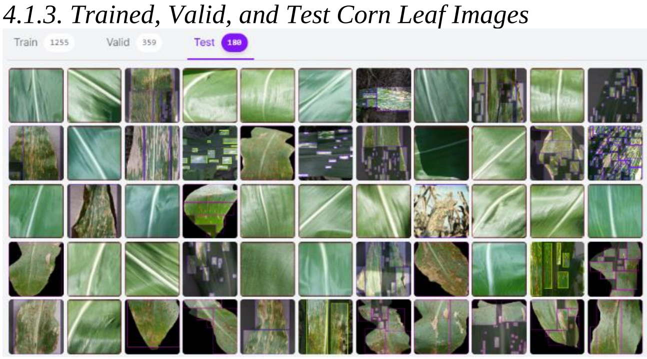
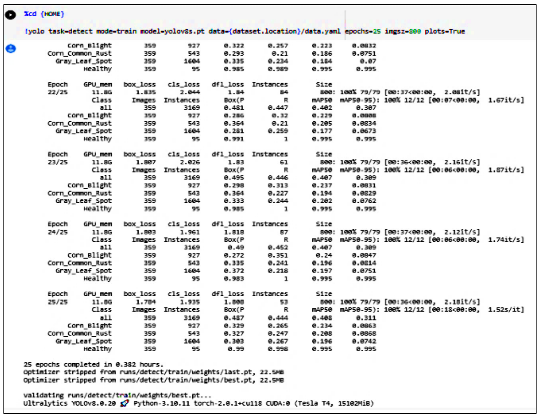


Fig. 4 Trained, valid, and tested corn leaf images

Figure 4 presents the sample total images. The data training resulted in a dataset consisting of 1255 samples for training, 359 samples for validation, and 180 samples for testing. These samples represent various instances of corn crop images, each annotated and labels for blight, common rust, gray leaf spot, or healthy. The more extensive training set of 1255 samples provides a substantial amount of data for the machine learning model to learn from and capture the patterns and characteristics of the different classes. The validation set of 359 samples enables fine-tuning and performance evaluation during training, ensuring the model is optimized and performs well on unseen data. Finally, the test set of 180 samples serves as an unbiased evaluation, allowing for an accurate assessment of the model's ability to generalize and classify corn crop diseases.

4.2. Data Training



```
%cd {HOME}

!yolo task=detect mode=train model=yolov8s.pt data={dataset.location}/data.yaml epochs=25 imgsz=800 plots=True
          corn_Blight        359        927      0.322      0.257      0.223     0.0832
     Corn_Common_Rust        359        543      0.293       0.21      0.186     0.0751
       Gray_Leaf_Spot        359       1604      0.335      0.234      0.184       0.07
              Healthy        359         95      0.985      0.989      0.995      0.995

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
      22/25      11.8G      1.835      2.044       1.84         84        800: 100% 79/79 [00:37<00:00,  2.08it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 12/12 [00:07<00:00,  1.67it/s]
                   all        359       3169      0.481      0.447      0.402      0.307
          corn_Blight        359        927      0.286       0.32      0.229     0.0808
     Corn_Common_Rust        359        543      0.364       0.21      0.205     0.0834
       Gray_Leaf_Spot        359       1604      0.281      0.259      0.177     0.0673
              Healthy        359         95      0.991          1      0.995      0.995

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
      23/25      11.8G      1.807      2.026       1.83         61        800: 100% 79/79 [00:36<00:00,  2.16it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 12/12 [00:06<00:00,  1.87it/s]
                   all        359       3169      0.495      0.446      0.407      0.309
          corn_Blight        359        927      0.298      0.313      0.237     0.0831
     Corn_Common_Rust        359        543      0.364      0.227      0.194     0.0829
       Gray_Leaf_Spot        359       1604      0.333      0.244      0.202     0.0762
              Healthy        359         95      0.985          1      0.995      0.995

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
      24/25      11.8G      1.803      1.961      1.818         87        800: 100% 79/79 [00:37<00:00,  2.12it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 12/12 [00:06<00:00,  1.74it/s]
                   all        359       3169       0.49      0.452      0.407      0.309
          corn_Blight        359        927      0.272      0.351       0.24     0.0847
     Corn_Common_Rust        359        543      0.335      0.241      0.196     0.0814
       Gray_Leaf_Spot        359       1604      0.372      0.218      0.197     0.0751
              Healthy        359         95      0.983          1      0.995      0.995

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
      25/25      11.8G      1.784      1.935      1.808         53        800: 100% 79/79 [00:36<00:00,  2.18it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 12/12 [00:18<00:00,  1.52s/it]
                   all        359       3169      0.487      0.444      0.408      0.311
          corn_Blight        359        927      0.329      0.265      0.234     0.0863
     Corn_Common_Rust        359        543      0.327      0.247      0.208     0.0868
       Gray_Leaf_Spot        359       1604      0.303      0.267      0.196     0.0742
              Healthy        359         95       0.99      0.998      0.995      0.995

25 epochs completed in 0.382 hours.
Optimizer stripped from runs/detect/train/weights/last.pt, 22.5MB
Optimizer stripped from runs/detect/train/weights/best.pt, 22.5MB

Validating runs/detect/train/weights/best.pt...
Ultralytics YOLOv8.0.20 🚀 Python-3.10.11 torch-2.0.1+cu118 CUDA:0 (Tesla T4, 15102MiB)
```

Fig. 5 Result of 1-25 Epochs (22 to 25 only)

As part of the training process, the YOLOv8 algorithm was employed with a specific configuration that involved training the model for 25 epochs. During each epoch, the model iteratively processed the training dataset, adjusting its parameters and updating its internal data representation to improve its performance. By training for 25 epochs, the model had the opportunity to learn from the dataset multiple times, gradually refining its ability to detect and classify objects accurately. The choice of 25 epochs aimed to strike a balance between allowing the model to learn the complexities of the data and preventing overfitting, ultimately leading to a well-performing and generalizable model.

After completing 25 epochs of training, which refers to 25 complete passes through the training dataset, the model achieved the desired level of convergence and accuracy. The training process lasted for 0.382 hours, indicating the efficiency of the training procedure. By iterating through the dataset multiple times, the model could learn and refine its parameters to make increasingly accurate predictions. The completion of 25 epochs suggests that the model has undergone substantial training, and the resulting model is expected to demonstrate improved performance and generalization capabilities compared to earlier stages of training.

4.3. Confusion Matrix

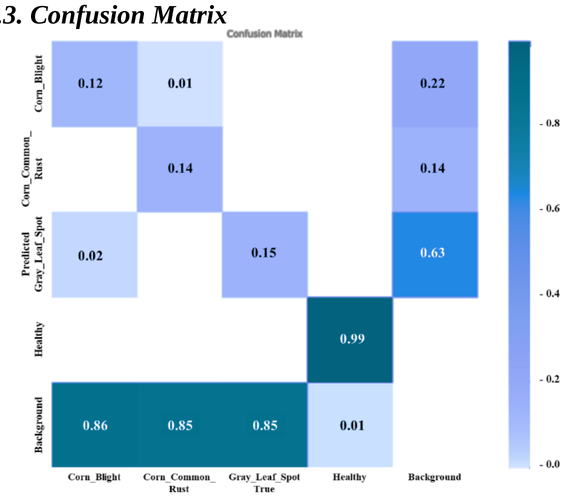


Fig. 6 Confusion matrix

The confusion matrix provides a comprehensive overview of the classification performance between the true and predicted labels for the categories: blight, common rust, gray leaf spot, and healthy. Each row of the matrix corresponds to the actual class, while each column represents the predicted class. The diagonal elements of the matrix show the number of correctly classified instances for each category, indicating the model's accuracy in identifying them. The off-diagonal elements reveal misclassifications, indicating cases where the model predicted a different category than the proper label. Analyzing the confusion matrix helps assess the model's performance and identify any patterns of confusion between specific classes, enabling improvements and adjustments to enhance its classification capabilities.

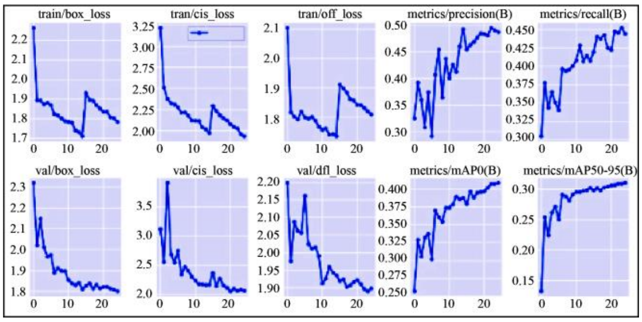


Fig. 7 Train Loss, Validation Los, Precision, Recall

When evaluating a classification model, several metrics provide insights into its performance. Train loss represents the discrepancy between predicted and actual labels during training, serving as an indicator of model convergence and accuracy. Precision measures the accuracy of positive predictions, indicating the proportion of true positive predictions out of all positive predictions. Recall quantifies the model's ability to identify positive instances from all true positive instances correctly. It represents the proportion of true positives identified by the model. Mean Average Precision (mAP) is a metric commonly used in object detection tasks, evaluating the precision-recall trade-off across different classes. It calculates the average precision values across all classes, providing an overall measure of the model's performance. These metrics play a crucial role in assessing the effectiveness and reliability of the classification model in various real-world scenarios.

4.4. Testing and Evaluation

4.4.1. Model Testing

The result of the test set showcases the accuracy of the YOLOv8 model in precisely identifying and classifying different categories, including blight, common rust, gray leaf spot, and healthy samples. It reflects the model's proficiency in correctly recognizing and distinguishing between these various corn leaf diseases and healthy leaves. The accuracy of the model's predictions on the test set demonstrates its effectiveness in accurately identifying and categorizing the different classes, providing valuable insights into the model's performance and its potential for real-world applications in disease detection and monitoring in corn crops.

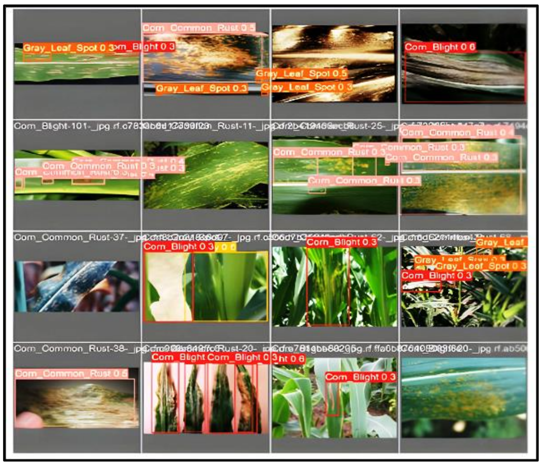


Fig. 8 Testing and evaluation results

Table 3. Summary of evaluation matrix

Class	R	mAP50-95
Corn blight	0.265	0.0858
Common Rust	0.252	0.0869
Gray leaf spot	0.367	0.0743
Healthy	0.998	0.995

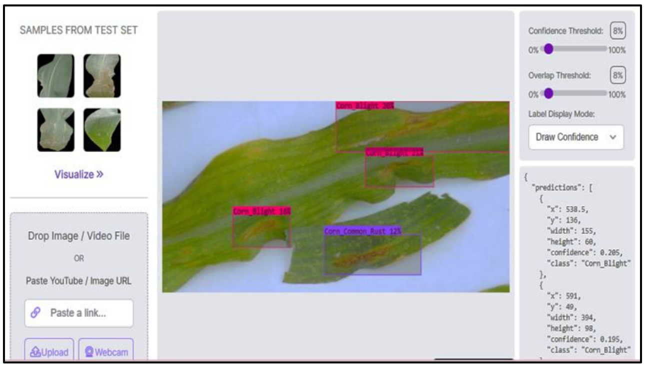


Fig. 9 Sample result of the implementation of the model

The evaluation matrix indicates the performance of a classification model across different classes. For "Corn blight," the recall (R) is 0.265, and the mean Average Precision at IoU thresholds from 0.50 to 0.95 (mAP50-95) is 0.0858, suggesting moderate recall but low precision. Conversely, the "Healthy" class has near-perfect performance with a recall of 0.998 and mAP50-95 of 0.995, indicating the model excels at identifying healthy instances.

4.4.2. Model Validation

Sample data of corn crop disease detection are shown, and the mAP (mean Average Precision) values for different classes are 0.88 for blight, 0.86 for common rust, 0.87 for gray leaf spot, and 0.99 for healthy, indicating the average precision achieved by the model for each specific class. A higher mAP value suggests better performance and accuracy in classifying instances of that particular class. In this case, the model has achieved a relatively high mAP of 0.99 for the healthy class, indicating a solid ability to identify and classify healthy corn plants correctly. Additionally, it has shown good performance for detecting diseases with mAP values of 0.88 for blight, 0.86 for common rust, and 0.87 for gray leaf spots, suggesting a high precision and recall for these disease categories. These results imply that the model performs well in distinguishing healthy plants and identifying specific diseases in corn crops.

5. Conclusion

The study focuses on leveraging the YOLOv8 algorithm to detect diseases in corn crops. The study aims to enhance smart farming practices by developing a robust and efficient system to accurately identify various diseases affecting corn leaves, such as blight, common rust, and gray leaf spot. By utilizing YOLOv8, the researchers aim to achieve accurate and real-time disease detection, enabling timely interventions and improved crop management. The study's findings and insights can potentially contribute to the development of advanced agricultural technologies and practices for enhancing corn crop health and productivity.

Based on the results obtained from the study, it can be concluded that the implementation of YOLOv8 for corn crop disease detection has yielded promising outcomes. The model demonstrates strong performance with high mean Average Precision (mAP) values across different disease categories, including blight, common rust, and gray leaf spot. Furthermore, the model achieves an exceptional mAP of 0.99 for the healthy class, indicating its accuracy in recognizing disease-free corn plants. These findings showcase the potential of YOLOv8 as an effective tool for supporting smart agricultural farming practices, enabling early disease detection and precise interventions to ensure crop health and optimize yields.

Recommendations

Based on the conclusion that the implementation of YOLOv8 for corn crop disease detection has yielded promising outcomes, a recommendation for this study would be to validate further and refine the model's performance in real-world agricultural settings. Conducting field trials or pilot studies on actual corn farms would provide valuable insights into the model's performance under diverse environmental conditions, variations in lighting, and other factors specific to agricultural settings.

Additionally, expanding the dataset, especially in the Philippine setting, to include a more extensive variety of corn leaf images encompassing different stages of disease progression and potential confounding factors would enhance the model's robustness and generalization capabilities. Lastly, exploring the feasibility of integrating the YOLOv8 model with smart agricultural systems or farm management platforms would facilitate its practical implementation, allowing farmers to receive real-time disease alerts and make data-driven decisions to mitigate crop losses and optimize yield.

Acknowledgment

The paper is made possible with the generous funding of the Center for Local Governance, Bukidnon State University's Research Unit, BukSU Research Committee, and BukSU Research Ethics Committee for their invaluable guidance and insightful suggestions have been instrumental in shaping the direction and outcomes of this study.

References

[1] David G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints,” *International Journal of Computer Vision*, vol. 60, pp. 91-110, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[2] Aanis Ahmad, Dharmendra Saraswat, and Aly El Gamal, “A Survey on Using Deep Learning Techniques for Plant Disease Diagnosis and Recommendations for Development of Appropriate Tools,” *Smart Agricultural Technology*, vol. 3, pp. 1-13, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Sunita I. Naik, Vivekanandreddy, and S.S. Sannakki, “Plant Disease Diagnosis System for Improved Crop Yield,” *International Journal of Innovations in Engineering and Technology*, vol. 4, no. 1, pp. 198-204, 2014. [Google Scholar] [Publisher Link]

[4] Yasamin Borhani, Javad Khoramdel, and Esmaeil Najafi, “A Deep Learning Based Approach for Automated Plant Disease Classification Using Vision Transformer,” *Scientific Reports*, vol. 12, no. 1, pp. 1-10, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[5] M. Shobana et al., "Plant Disease Detection Using Convolution Neural Network," *2022 International Conference on Computer Communication and Informatics*, Coimbatore, India, pp. 1-5, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[6] Ajay Talele, Aseem Patil, and Bhushan Barse, “Detection of Real Time Objects Using TensorFlow and OpenCV,” *Asian Journal for Convergence in Technology*, vol. 5, no. 1, pp. 1-4, 2019. [Google Scholar] [Publisher Link]

[7] Florin-Alexandru Vasluianu et al., “NTIRE 2023 Image Shadow Removal Challenge Report,” *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1788-1807, 2023. [Google Scholar] [Publisher Link]

[8] Florin-Alexandru Vasluianu et al., “NTIRE 2023 Image Shadow Removal Challenge Report,” *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, Vancouver, BC, Canada, pp. 1788-1807, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[9] Dionis A. Padilla, Ramon Alfredo I. Pajes, and Jerome T. De Guzman, “Detection of Corn Leaf Diseases Using Convolutional Neural Network With OpenMP Implementation,” *2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management*, Manila, Philippines, pp. 1-6, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[10] Laith Alzubaidi et al., “A Survey on Deep Learning Tools Dealing with Data Scarcity: Definitions, Challenges, Solutions, Tips, and Applications,” *Journal of Big Data*, vol. 10, no. 1, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] Davinder Singh et al., “PlantDoc: A Dataset for Visual Plant Disease Detection,” *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD*, Hyderabad India, pp. 249-253, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] Pei-Wern Chin, Kok-Why Ng, and Naveen Palanichamy, “Plant Disease Detection and Classification Using Deep Learning Methods: A Comparison Study,” *Journal of Informatics and Web Engineering*, vol. 3, no. 1, pp. 155-168, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[13] Andreas Kamilaris, and Francesc X. Prenafeta-Boldú, “Deep Learning In Agriculture: A Survey,” *Computers and Electronics in Agriculture*, vol. 147, pp. 70-90, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[14] Sharada P. Mohanty, David P. Hughes, and Marcel Salathé, “Using Deep Learning for Image-Based Plant Disease Detection,” *Frontiers in Plant Science*, vol. 7, pp. 1-11, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[15] S.A. Sanchez, H.J. Romero, and A.D. Morales, "A Review: Comparison of Performance Metrics of Pretrained Models for Object Detection Using the Tensorflow Framework," *IOP Conference Series: Materials Science and Engineering*, vol. 844, no. 1, pp. 1-15, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[16] Khalid Hamza, Saba un Nisa, and Gulshan Irshad, "A Review on Potato Disease Detection and Classification by Exploiting Deep Learning Techniques," *Journal of Agriculture and Veterinary Science*, vol. 1, no. 2, pp. 81-88, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] Deepak Kumar, Sonam Gupta, and Pradeep Gupta, "Plant Disease Recognition Using Machine Learning and Deep Learning Classifiers," *13th International Conference Advanced Computing*, Kolhapur, India, pp. 3-14, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[18] Jiawei Li et al., "An Improved YOLOv5-Based Vegetable Disease Detection Method," *Computers and Electronics in Agriculture*, vol. 202, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[19] Mohit Agarwal et al., "A Convolution Neural Network Based Approach to Detect the Disease in Corn Crop," *2019 IEEE 9th International Conference on Advanced Computing*, Tiruchirappalli, India, pp. 176-181, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[20] Wen Tao et al., "Review of the Internet of Things Communication Technologies in Smart Agriculture and Challenges," *Computers and Electronics in Agriculture*, vol. 189, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[21] David Vallejo-Gómez, Marisol Osorio, and Carlos A. Hincapié, "Smart Irrigation Systems in Agriculture: A Systematic Review," *Agronomy*, vol. 13, no. 2, pp. 1-25, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[22] Md. Ashraful Haque et al., "Deep Learning-Based Approach for Identification of Diseases of Maize Crop," *Scientific Reports*, vol. 12, no. 1, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[23] Gopi Krishna Akella et al., "A Systematic Review of Blockchain Technology Adoption Barriers and Enablers for Smart and Sustainable Agriculture," *Big Data and Cognitive Computing*, vol. 7, no. 2, pp. 1-22, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[24] Sjaak Wolfert et al., "Big Data in Smart Farming – A Review," *Agricultural Systems*, vol. 153, pp. 69-80, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[25] Jaqueline Iaksch, Ederson Fernandes, and Milton Borsato, "Digitalization and Big Data in Smart Farming–A Review," *Journal of Management Analytics*, vol. 8, no. 2, pp. 333-349, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[26] Achim Walter et al., "Smart Farming is Key to Developing Sustainable Agriculture," *Proceedings of the National Academy of Sciences*, vol. 114, no. 24, pp. 6148-6150, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[27] Yemeserach Mekonnen et al., "IoT Sensor Network Approach for Smart Farming: An Application in Food, Energy and Water System," *2018 IEEE Global Humanitarian Technology Conference*, San Jose, CA, USA, pp. 1-5, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[28] Achyut Morbekar, Ashi Parihar, and Rashmi Jadhav, "Crop Disease Detection Using YOLO," *2020 International Conference for Emerging Technology*, Belgaum, India, pp. 1-5, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[29] Corn Disease, Roboflow. [Online]. Available: https://app.roboflow.com/buksu/corn-disease-odni2/visualize/2