Nigerian Journal of Technology (NIJOTECH)
Vol. 43, No. 4, December, 2024, pp.807 - 817
www.nijotech.com

Print ISSN: 0331-8443
Electronic ISSN: 2467-8821
https://doi.org/10.4314/njt.v43i4.21

SMART MULTI-PURPOSE FARM DISEASE MONITORING AND NOTIFICATION MODEL

AUTHORS:
N. I. Ezeani[1,*], S. S. Usoro[2], N. B. Okoye[3], D. O. Oyeka[4], and O. N. Iloanusi[5]

AFFILIATIONS:
[1,2,3]Electronics Development Institute Eldi-Naseni Awka, Nigeria.
[3,4]Department of Electronic Engineering University of Nigeria Nsukka, Nigeria.

***CORRESPONDING AUTHOR:**
Email: nnekaokafor516@gmail.com

ARTICLE HISTORY:
Received: 05 March, 2024.
Revised: 15 August, 2024.
Accepted: 23 October, 2024.
Published: 31 December, 2024.

KEYWORDS:
Plant, Disease, Farm, Yolov-5 Algorithm, Training, Validation, False Alarm.

ARTICLE INCLUDES:
Peer review

DATA AVAILABILITY:
On request from author(s)

EDITORS:
Ozoemena Anthony Ani

FUNDING:
None

Abstract

Disease has remained a threat to the existence of every living thing on earth. Particularly in agriculture, disease has remained a major constraint to the success of crop yield and demands urgent solutions, starting with early detection. While many studies have been presented on plant disease detection and control, despite their success, the research gap resides in creating a balance between plant disease detection and farm disease detection. This is because detecting disease in a plant does not necessarily imply that the farm is infected with diseases, and this has resulted in issue of false alarm in the existing system. To address this challenge, YOLOV-5 model was trained with a plant disease dataset considering diverse classes of plants such as corn, waterleaf, tomato, pepper, and cassava. The plant disease model generated was used to develop a farm disease monitoring, detection and notification algorithm, which was converted into mobile application software using Python programming for real-time monitoring notification of farm diseases. This multi-purpose system when tested, reported an average precision mean of 0.95. In addition, experimental validation of the model in maize and watermelon farms reported real-time disease detection and notification which facilitates rapid response and control of the disease by the farmer.

1.0 INTRODUCTION

Man's ability to cultivate the land and grow food is a testament to our resilience, creativity, and connection to nature. In the act of nurturing the soil, we nurture our own well-being and sustain our journey on this Earth. In this context, nurturing the soil involves the traditional preparatory steps undergone by farmers such as clearing the land, tilling of soil, treatment, and then cultivation of plants, which then yield the harvest to sustain life. However, pests and diseases remained the most disruptive factors that affect the growth of plants with an annual loss of 10-28% [1]. In addition, the impact of climate change, and natural disasters, coupled with the exponential increase in the population of man, have all presented a major challenge to sustainable global food supply [2]. According to [3], while all these factors have the potential to cause food scarcity, plant diseases stand as a major contributing factor affecting sustainable food production through the agricultural process, and hence form the purpose of this paper tailored towards providing a reliable, easy-to-use, cost-effective real-time plant disease monitoring, and detection using Transfer Learning Technique (TLT).

HOW TO CITE:
Ezeani, N. I., Usoro, S. S., Okoye, N. B., Oyeka, D. O., and Iloanusi, O. N. "Smart Multi-Purpose Farm Disease Monitoring and Notification Model", *Nigerian Journal of Technology*, 2024; 43(4), pp. 807 – 817; https://doi.org/10.4314/njt.v43i4.21

Plant diseases are organisms such as viruses, fungi, bacteria, early blight, yellow curl virus, septoria, molds, and environmental conditions that cause infection in plants [3]. Currently, the traditional approaches to plant disease detection are pathogen analysis and expert diagnosis [4]. The latter refers to the expertise of farmers to detect plant disease based on previous experiences, while the former involves a microscopic analysis of plant pathogens to detect diseases [3]. While the expert analysis approach can be prone to human error, and classification of farm disease as homogenous in many instances, the pathogen analytical approach is more reliable, but time-consuming and cumbersome, hence making it not suitable for large farm disease monitoring [4].

In recent years, artificial intelligence approaches, specifically TLT have been applied to help improve plant disease monitoring and control. TLT is pre-trained deep learning algorithms such as Alex.Net, GoogleNet, ResNet, You Only Look Once Version (YOLOV) among many others carefully designed and trained with large volumes of datasets to solve real-time object classification problems [2, 4]. Among these algorithms [5, 6, 7, and 8] revealed that YOLOV algorithms stand out as one of the most adopted to solve real-time object classification problems and have also been applied for plant disease detection due to their high accuracy potential and speed of classification. For instance [10] applied YOLOV-4 for the detection of apple diseases, while the model was improved to YOLOV-5 in [11] and applied for the detection of the apple diseases. The result demonstrated successful detection of apple diseases but lacked trustworthiness as it was not practically validated.

Authors in [12, 13, 14] applied YOLOV-5 for the classification of (Citrus, grapes, and tomato) plant diseases and reported an average overall mean absolute precision of over 80%. While these studies have all demonstrated high precision values in detecting diseases, there is no balance between plant diseases and farm disease detection. This is because, the detection of plant disease does not necessarily mean that the entire farm is infected and when deployed results in issue of false alarm, as it is almost impractical to have a farm with 100% healthy plants. To correctly label a farm as infected with disease, at least a certain small proportion of the farm has to be infected, which presents the need for an algorithm that determines this infected proportion of the farm using intelligent computation as proposed in this research. Other part of the paper includes more literature review, material used for the research, and methods used for the development of the farm disease monitoring and notification algorithm. The highlights of this paper are as follows;

i. A clarity differentiating plant disease from farm disease was presented
ii. A new algorithm for farm disease monitoring and notification was presented
iii. Propose a multi-purpose system for diverse plant diseases monitoring
iv. Proposed a model that is tested and experimentally validated in an actual farm

2.0 LITERATURE REVIEW

Over the years, many studies have applied YOLOV algorithms for the classification of plant diseases. This is because YOLOV has consistently recorded high classification performance in terms of speed, accuracy, and precision. Among the studies include [4] who trained YOLOV-4 with 200 samples of tomato pants spanned across three classes of unripe fruits, ripe and disease infected fruits, and reported an average precision of 81.28%. Similarly, [9] trained YOLOV-5 on 5700 samples of apple data to classify calyxes, stems, and defects classes of the data. In the study, model pruning was applied to compress the YOLOV-4 model and then tested to report a precision of 93.74%, which when compared with [4] implied that increased data size improves learning performance. In the case of [14], YOLOV-2, 3, and 4 were trained using 30,059 images and then compared. The result reported that YOLOV-4 achieved the best average precision value of 90.8%. What this means is that the denser the YOLOV model, the better the performance.

In this vein, [10], improved the YOLOV-5 model, adding two extra layers and an attention layer to consider the importance of different channels and then trained with 1600 images of kiwifruit spanned across four classes, the result reported 94.7% precision, which was improved by [12] who pruned the model and trained on 3165 samples of apple fruit and reported 95.8% precision. In addition, [15] added an attention mechanism to replace the PANet multiscale features fusion network of YOLOV-5 and trained with 16000 images generated with data augmentation. The result reported a 98.4% precision rate, which is good, thus demonstrating the effectiveness of YOLOV-5 in plant disease detection; however, while these studies all reported high classification precision scores for plant disease detection, it is not clear the definition of success in classifying an infected farm. This is because these models while able to detect disease in a plant, do not necessarily imply that the farm is infected and hence present an issue of false alarm.

https://doi.org/10.4314/njt.v43i4.21

[16] Presented a customized CNN based predictive intelligence for the monitoring and detection of disease infected millet. In achieving this, 3000 images of millet with rust and blast diseases were collected and applied to train CNN customized with three convolutional layers and generate model for the classification of millet diseases. The results reported 98.8% and training time o 67secs. [17] Applied IoT and A.I for smart cattle health monitoring. Data of castle behavior considering heart rate, skin temperature and accelerometer was collected and used to train XGboost and random forest classifier and generate model for the monitoring of cattle behavior. Model was converted to mobile application software and deployed in android operating system. The results reported for R 96% accuracy and XG 100% accuracy. [18] Used logistic regression to improve Wifi network and apply for the monitoring of livestock. In addition, cryptograph and stenography was applied to protect the network. These studies have successfully demonstrated the application of A.I and IOT for smart agriculture; however more studies are needed to explore how deep learning can improve real time monitoring [19].

Diving more into deep learning [20], applied it for the optimal selection of soil sampling sites. In the paper, the first approach involves utilizing encoder–decoder architecture with a self-attention mechanism with the CNN backbone, while the second is to innovate a deep-learning design grounded in the concepts of transformer and self-attention. In the decoder, the system introduces convolution networks to concatenate, fuse the extracted features, and then export the optimal locations for soil sampling. The results after training reported a mean accuracy of 99.52%, a mean Intersection over Union (IoU) of 57.35%, and a mean Dice Coefficient of 71.47%, while the performance metrics of the state-of-the-art CNN-based model are 66.08%, 3.85%, and 1.98%, respectively. The result despite the success leaves room for improvement especially in the dice coefficient but provides a good foundation for DL studies on soil sampling.

In [21] deep learning was applied for precision agriculture with focus on cereal plant head detection. The work discussed the various types of deep learning algorithms capable of counting cereal heads using image segmentation and object detection strategies. Research gap was identified in the need for a diverse sophisticated dataset and image processing algorithm which can adapt to dynamic nature of cereal plants. In addition, the application of unmanned vehicles was recommended as a tool to improve precision agriculture in the future. In [22] a comparative study on the application of deep learning and IoT was presented for improved precision agriculture. The study surveyed the diverse application of deep learning and IoT in agriculture, considering areas such as crop yield, weed detection, pest, diseases and soil prediction. In addition, the architecture of IoT for precision agriculture was presented constituting of the physical layer, network layer, middleware layer, processing layer and application layer. Sensors such as electrochemical, location, optical, acoustic, airflow were all discussed in the study. Types of datasets to facilitate training of DL were discussed and then the challenges of IoT and DL were also highlighted. In the IoT challenges issues such as hardware failure, cost, scalability, interference are some of the issues pointed out which needs immediate attention. For the DL approaches issues of over-fitting, increased training time, varnishing and exploding gradient and huge data requirements for model success.

Suggestions to solve these issues such as application of dropout regularization, network pruning, back-propagation algorithms were presented to help address these challenges. In [23] DL and semi-automated image labeling as applied for wheat stripe detection. ResNet-18 was fine-tuned on data collected with UV from multiple wheat conditions such as winter, spring, irrigated wheat, non-irrigated wheat. The generated model was validated across sensor platforms, location and types of wheat. The area under curve recorded 0.72 to 0.87, while an independent validation approach applied reported 0.79 to 0.86. While this work achieved good score for the respective classification metrics, there is still room for improvement in the metrics. In addition, the model is limited to the classification of wheat, and therefore can be improved considering other plants to improve application diversity. In addition, the model lacks clear definition of plan disease detection success, because as we have already stated that plant disease is not farm disease. Therefore, there is an urgent need for a model that computed the fitness of the entire farm to determine the health or unhealthy status of the farm. In addition, the applications of the existing studies are mostly limited to single plants, which are good but can be improved considering many other plants to expand their practicability and also make the marketing very easy.

3.0 MATERIAL AND METHODS

The materials used for the research are YOLOV-5 model, Techno Samsung Galaxy A12, android operating system, 5G, internet enabled SIM card, battery, python programming language. The methods used are data collection of plant diseases, considering diverse classes as discussed in Section 3.1. Data processing to automatically label the plant disease data according to their classes, training the YOLOV-5 model for plant disease detection, development of the farm disease monitoring and notification algorithm, integrated of the algorithm as mobile application software for real time farm disease surveillance and notification using python programming; finally, experimental demonstration of the model's applicability in a real farm for diseases monitoring and detection as a validation strategy.

3.1 Data Collection

The primary sources of data collection are 300 x 120 meter infected maize farm and 150 x 120 meter infected water leaf farm respectively identified at Agric center, Aninri Local Government Area, Enugu State, Nigeria. The instrument used for the data collection are a 550mm quadcopter integrated with Tapot100 camera sensor, which was remotely controlled with the quadcopter fight control system and flew around the farm at a vertical distance of 4ft above the infected plant area, while the capturing of the plant disease images in a resolution format of 640x640pixels. The maize diseases considered are bacteria spot, early blight and late blight, while the cassava diseases considered are mosaic virus and yellow curl virus. During the data sampling of these plant diseases by the drone, diverse lightening condition was also taken into consideration, to capture the effect of lightening condition on the model sensitivity.

For each plant, was sampled at early morning hours (7am to 8am), afternoon period of (1pm to 2pm) and night period of (6:30 to 7pm). After each period, of data sampling, it was automatically stored in a 250MBSD memory cards system and collected inserting the SD card reader system into a USB 3.0 iPad SD Card reader linked to an iPad computer. From the iPad, each folder of the data collected was copied and saved in the new plant disease dataset. The population sample size of maize plant disease images copied is 800 incorporating the diverse diseases aforementioned for maize, while that of cassava plant diseases is 755 images respectively of diverse cassava plant diseases, thus presenting a total population sample of 1555 images. The Figure 1 presented the experimental setup of the instrument used for the data capturing at the testbed. To prepare the data and make it perfect for integration with a secondary dataset which will be collected shortly, first it was divided into training (1200sample) and test (355sampels) sets respectively.

The test class was not processed, as it was mapped out for the model testing, while the training samples were automatically annotated through the assignment of bounding box around the images and assigning the appropriate class name in a .text file and saved. This labeling was in-line with YOLOV-5 requirement, with the labeled represented in c,x,w,h formats respectively, where c is the object class index, x and y are the bounding box coordinated, w and h represented the height and weight of the boxes. This was achieved using the python-based Labelling [83] graphic annotation tool, which is a special toolbox for the labeling of images to train YOLOV-5. The Figure 2 presents the annotated images of plant diseases ready for integration.

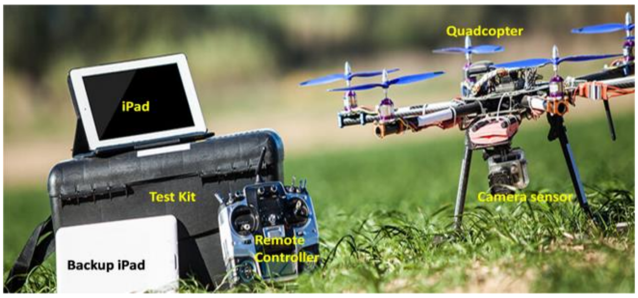

Figure 1: Experimental setup for data collection

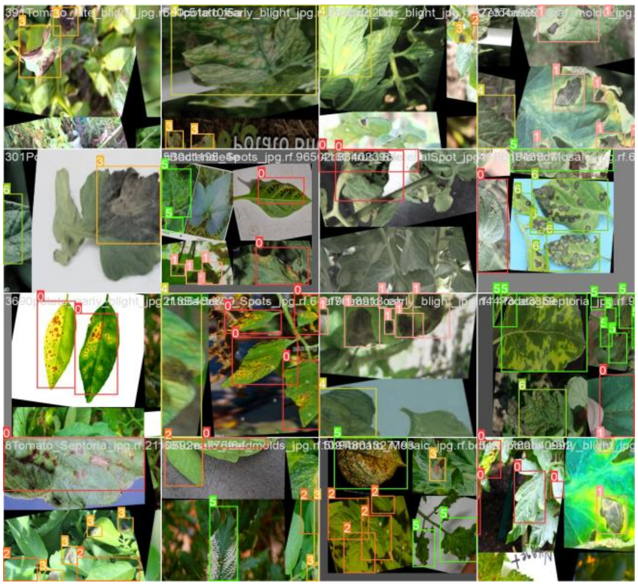

Figure 2: Outcome of the primary data annotation process

The secondary data set used for this study is the plant-village dataset collected from Kaggle repository which contained 54303 plant species span across nine classes of peach, potato, strawberry, cherry, blue berry, tomato, pepper, and apple and their respective diseases and grouped into two classes of health and unhealthy. The link for the data source was reported in the Appendix A. These data were already processed

https://doi.org/10.4314/njt.v43i4.21

with annotation labels [24, 25] and are grouped into training, validation and test sets in the ratio of 80:10:10. The 80% was used to train the classifier which understands the intricate of the plants with diverse disease; the test was used to evaluate the training performance of the classifier, while the validations set validated the reliability of the classifier to detect plant diseases. The Figure 2 presents the annotated plants in the dataset. Collectively the primary and secondary data were integrated with the Google data fusion tool, with a total sample size of 55858 images.

3.2 YOLOV-5 Model

The typical YOLOV-5 architecture is made up of three parts which are the backbone, neck and output [15]. The backbone is used for the extraction of features from input image, the neck is applied for fusion of extracted features and generation of feature maps, while the output is used for the plant detection from these generated feature maps. The Figure 3 presents the architecture.

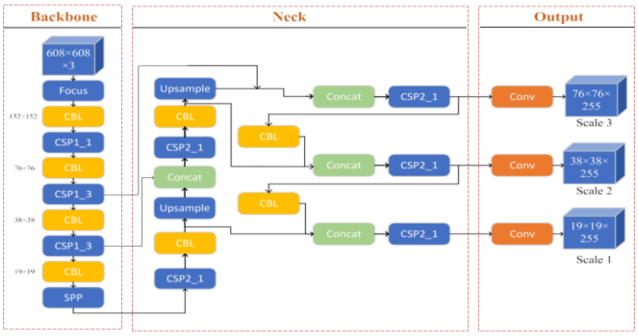

Figure 3: Architecture of YOLOV-5

The backbone network is made of convolutional neural network which extracts the features maps from the various image sizes of the input data, through the multiple convolutions and pooling. The backbone is made of input layer dimensioned as 608x608x3 (height, weight and color channel), then a focus layer was applied for spatial dimensionality reduction of the input, by aggregation of the input information across different spatial location of the image. In addition, four layers of feature maps which are 152x152 pixels, 76x76 pixels, 38x38 pixels and 19x19 pixels respectively are individually consisted of Convolutional, Batch Normalization, and Leaky ReLU (CBL) which role is to improve speed of training and introduction of nonlinearity. Then Cross Stage Partial (CSP)was applied to improve the flow of information and address vanishing gradient problem during over-fitting and also improve training performance. In the last layer, Spatial Pyramid Pooling (SPP) was applied for multi-scale information extraction from the feature maps and then applying maximum pooling approach to each pixel region.

It involves dividing the input feature map into regions of different sizes and pooling each region separately, making the network more robust to plants of different sizes. The neck part of the YOLOV-5 is responsible for fusion process. This is achieved using the CSP which in this case employs a pyramid structure of Path Aggregation Network (PAN) and Feature Pyramid Network (FPN) for the localization of pixels concentrated at the concat and then fusion together in three output layers with a color channel of 255. The up-sampling was used to recover spatial information loss, thus allowing the model to make high resolution predictions. In the output section of the model, three layers of new scales of feature maps with sized of 76x76x255 for the layer one; 38x38x255 and then 19x19x255. What this means is that the model was able to perform plant disease detection for diverse plant sizes and classification in real time.

3.3 Training of the YOLOV-5 Model

Training of the YOLOV-5 model [26] was performed on Google colab platform, using the prepared plant disease dataset, Stochastic Gradient Descent Optimization (SGDO) technique [27], and then training parameters in table1. The data already prepared and splitted into training, test and validation sets of ratio (80:15:5) were imported to the YOLOV-5 model adopted from ultralytic environment and then trained. During the training process in batches, the learning rate, weights, bias, momentum is adjusted automatically while the loss function is monitored using the Equation 1 [28].

$$Loss = \alpha_1 L_{cls} + \alpha_2 L_{obj} + \alpha_3 L_{clou} \qquad (1)$$

Where, $\propto$ is the classes of detected objects, $ciou$ is complete intersection over union, $obj$is the object loss, L is the location of loss and cls is the classification loss. The test and validation sets are also applied to evaluate the model, and the hyper-parameters continuously adjusted at each epoch iteratively until the neurons converges, then the plant disease detection model generated and reported in the algorithm 1.

Table 1: Training parameters

Items	Specification
Momentum	0.937
Epoch; batch size	150; 16
Weight decay	0.0005
Input size	608x608
Learning rate	0.01
Optimizer	SGDO
Bias and momentum	0.8; 0.1
Warmup epoch	3

Table 2: Implementation materials

Items	Specification
CPU	AMD-8-Core processor
RAM	8GB
Python	3.7.11
Pytorch	1.7
Mobile phone	Samsung Galaxy A12
GPU	RTX3070-Ti
Operating system	Windows 10; Android
Video Memory	8G

Algorithm 1: Plant disease detection

```
1. Start
2. Initialization of trained YOLOv5 model
3. camera initialization
4. def capture_plant_image(608Hx608W)pixels% define plant image size captured
5. while true;
6. plant_image = capture_image()%capture plant images
7. defYOLOV-5 detect_diseases_model (Plant_image)% model YOLOv5for classification
8. Return the detected bounding boxes and confidence scores
9. detections = detect_diseases(plant_image)% detect diseases in the captured image
10. Repeat step (6)
End
```

Algorithm 2: New Farm Disease Monitoring and Notification system

```
1. Start
2. Initialize plant disease detection algorithm
3. Initialize email configuration (SMTP server, steverolans@gmail.com)
4. Set the total_farm_area (TF_A)in square meters % Make sure the camera covers the area
5. Set time interval for notifications% how often the email will be sent per day/week
6. Set the threshold_percentage (T_p) for disease > = 10%
7. For     detections = detect_diseases(plant_image) = True
8. def calculate_percentage(detections, total_area)% Calculate Percentage of Diseases
9. Identify TF_A value
10. percentage_diseases = calculate_percentage(detections, TF_A) %Calculate disease
    percentage
11. def send_email_notification(recipient, Farm notification, "This email is a demonstration
    to prove that your farm has been affected by disease")
12. If percentage_diseases >T_p
13. Recipient_email = " Enter Recipient email address"
14. email_subject = "Farm notification"
15. email_body = f" This email is a demonstration to prove that {percentage_diseases}% of
    your farm has been affected by disease;"
16. send_email_notification(recipient_email, email_subject, email_body)
17. Elseif
18. Return to step (2)
19. # End if: End
```

4.0 THE FARM DISEASE DETECTION (FDD) SYSTEM

FDD is a system proposed in this paper to address the issue of false positive identified as major issue with the existing systems. This is because; while there are numerous models which can correctly detect real time diseases in plants, this does not exactly imply that the whole farm or major portion of the farm is infected with disease and hence presents an issue of false alarm and lack of integrity with the existing system. This paper proposes a new algorithm which considers the area of the farm, percentage of disease detected by the classification model and then make an informed decision on the farm status, through a simple rule based which determines disease on the farm when >= 10% of the entire farm area is infected. The detected disease is relayed to the farmer using Email notification. The farm disease detection algorithm is presented in algorithm 2; while the assumptions used in the development model was presented as;

i. Farmer's email is cjeneral@yahoo.com
ii. Host email is stevrolas@gmail.com
iii. Farm size is measured in square meters
iv. Threshold for disease is 10%

5.0 PERFORMANCE EVALUATION PARAMETERS

The evaluation of the YOLOV-5 model trained considers parameters such as Precision (P), Recall (R), Average Precision (AP_i), and mean average precision (mAP). These Equations of the parameters are;

$$P = \frac{TP}{TP+FP} \quad (2)$$

$$R = \frac{TP}{TP+FN} \quad (3)$$

$$AP_i = \int_0^1 P(R)d(R) \quad (4)$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \quad (5)$$

Where, TP is true positive plant disease detection, FP is negative samples which are positively detected, FN are positive samples which are not detected.

6.0 RESULTS AND DISCUSSION

The performance of the trained YOLOV-5 model was evaluated considering the loss function model in Equation 1 which was used to measure the gradient loss which occurs during the training process. The model Equations 2 - 5 were also engaged to measure the ability of the model to detect healthy and unhealthy plant correctly. The results were reported in the Figure 4;

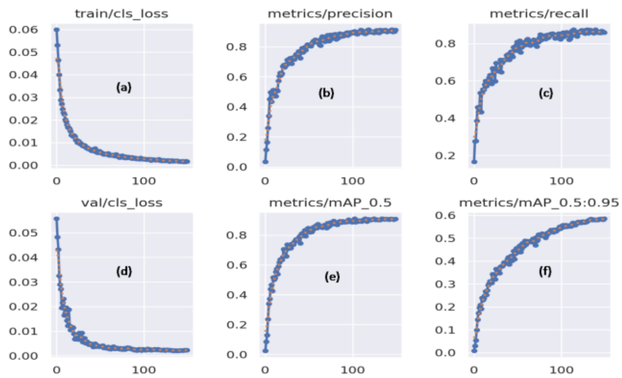

Figure 4: Result of the YOLOV-5 Training performance

From the Figure 4, the training and validation performance of the model were reported as shown considering the loss function during the training (a) and also validation (d). From the result it was observed that at both evaluations, the loss depreciated from each epoch until it converges after epoch 100. What this means is the SGDO was also able to optimally adjust the hyper-parameters while the YOLOV-5 model learn the plant diseases features correctly. The precision evaluation of the model was reported in (b) and it was observed that at each epoch the probability to correctly predict the true class for each disease increases. Similarly, the mean average precision was measured using figure (e) and the overall mAP for the model is 0.95 as reported in figure (f), while the recall in figure (c) also reported an average of 0.963. In addition, confusion matrix was used to evaluate the performance of the model, showing the relationship between the true class and the predicted using the Figure 5.

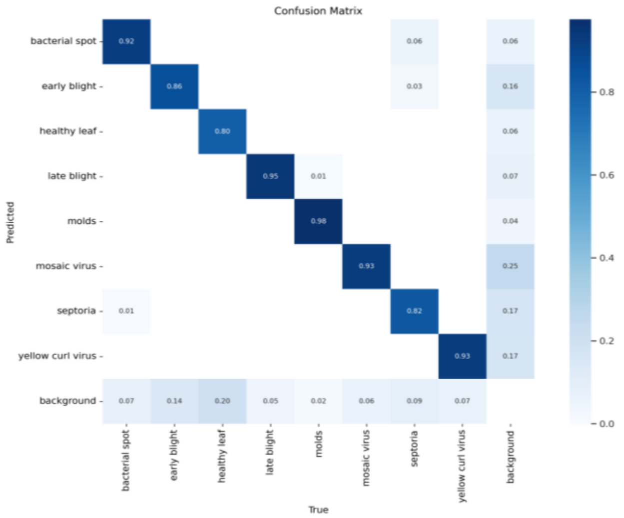

Figure 5: Confusion Matrix analysis of the plant disease detection model

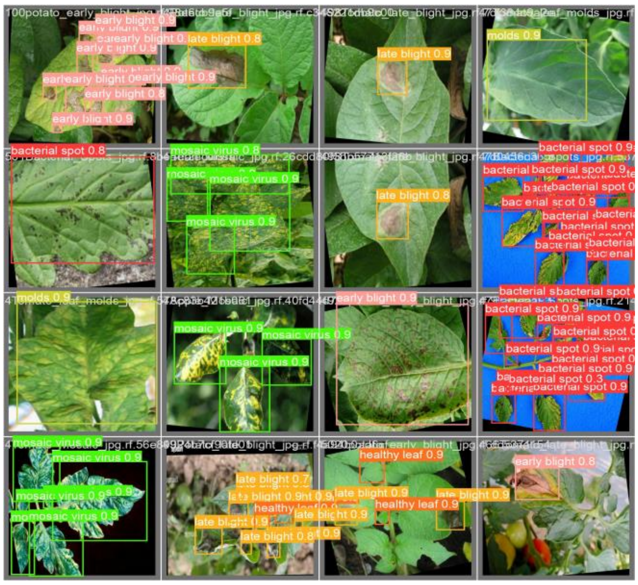

Figure 6: Testing Result of the plant disease detection model

From the result in Figure 5, it was observed that bacterial spot class testing reported a positive predicted value of 0.92, what this means is that when the bacterial spot was used to test the YOLO-5 trained model for plant disease detection, it was able to correctly classify 92% of the plants with the disease successfully. Similarly, early light reported 0.86, healthy plant reported 0.80, molds reported 0.98, late blight reported 0.95, septoria reported 0.82 and yellow curl virus reported 0.93. Overall, it was observed that the plant disease detection model was good, because the prediction of diseases for all classes reported an average of 0.90 positive prediction values. The Figure 6, reported the testing result of the model on various plants.

The Figure 6 presented the evaluation of the plant disease detection model considering some of the test plant disease samples. From the result, it was observed that the trained YOLOV-5 model was able to correctly classify the diseases on the plants, and assign annotations and confidence score for the disease detection.

6.1 Experimental Results on Different Plants to Show Application Diversity

This section showed the performance of the model when evaluated on different plant types considering maize and water leaf at two separate farms. The Figure 7 showed the mobile interface where user can set the notification frequency and email address to received information on the farm status. In the Figure 8, the result of the model when tested on a waterleaf farm infected with various disease was reported.

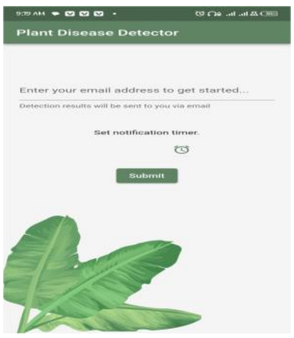

Figure 7: Mobile interface of the system

Figure 8: Experimental validaiton on water leaf

The Figure 7 showed the user interface for the configuration of the system to the farmers requirements, while the Figure 8 demonstrated the practical testing of the model in a waterleaf farm. The classificaiton results as served in the Figure 8 showed the ability of the model to correctly classify different waterleaf diseases in real-time. From the results, it was osberved that while the traiend YOLOV-5 was able to classify the disease, the farmer did no receive any notificaiton, which is why no email results was reported for this case. The reason was because, the notifcaiton algorithm which makes the decision for farm disease was not able to classify disease in the farm. This mean that the portion of the waterleaf farm with disease is tolerable (less then 10% infeacted) compared to the entire farm size, hence, no email notificaiton was sent to the farmer. Nevertheless, when applied to another maize farm as shown in Figure 9, the model was able to detect disease in the major portion of the farm and when overall disease percentage compared to the area of the farm is over 10%, the farmer was notified of the problem through email as reported in the Figure 10;

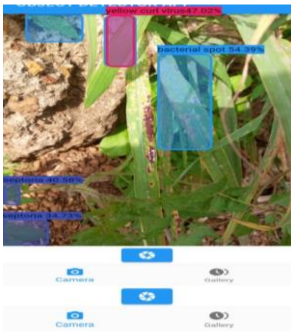

Figure 9: Experimental validatin on maize farm

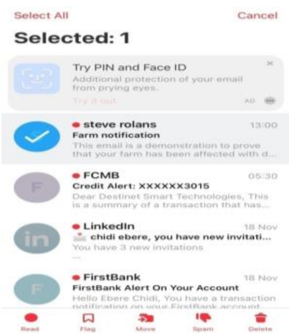

Figure 10: Result of email notification

The Figure 9 demonstarted the performance of the farm disease detection system on small maize farm. From the result it was observed that the YOLOV-5 was able to detect the disease on the farm in real time, while the percentage of disease detected was determined considering the area of the farm, and when the decision outputs diasese, an email notification is sent to te farmer as in Figure 10, reporting the problem. In addition a comparative model validation with existing algorithms was performance and reporte in Table 3;

Table 3: Comparative analysis

Reerence	Technique	mAP
[4]	YOLOV-4	81.28
[9]	YOLOV-5	93.74
[14]	YOLOV-4	90.80
[10]	YOLOV-5	94.70
[12]	YOLOV-5	95.80
[15]	YOLOV-5	98.40
New system	YOLOV-5	95.00

From the comparative analysis in Table 3, it was observed that YOLOV-5 reported an average mAP of over 94% or all models, however while [12, 13] reported the best mAP, the lack of clear definition of farm disease detection success affects the trustworthiness of the model. in addition, the new mode which reported a good mAP score of 95%, is also the most reliable, as the modeling considered and addresses the issue of false positive which is a major challenge in the existing system.

Figure 11: Results with low light

Figure 12: Result with light

6.2 Experimental validaiton considering diverse lightening condition

In the Figure 11 and Figure 12, the result of the model was validated in diverse lightening condition considering the water leaf farm was presented. The Figure 11 presented the model perormance at low light condition, while the Figure 12 presented the performance during the day with normal sun rays.

From the result in 11, it was observed that desptie the low light in the environment, the model was able to correctly classify the disease on each farm and also classify the health plants. The reason was because the YOLO-V5 model was able to extract inticate feaures of the plant diseases using the SPP which facilitates multi-scale informaiton extraction and improve classificaiton accuracy. In the Figure 12, with normal light condition, it was observed that the model was also able to function effectively with correct classificaiton of disease in the farm.

7.0 CONCLUSION

This paper has successfully presented a model which in real-time, monitors disease in a farm and then uses the information obtained during the monitoring to determine when the farm is infected with diseases. This was achieved using the YOLOV-5 algorithm, trained with the dataset containing multiple plant diseases. The farm disease detection model generated was converted to a mobile application and experimentally validated considering watermelon and maize farms respectively. The result demonstrated the ability of the model to detect disease in a farm and then notify the farmer of the problem for rapid response. The benefits include increased crop yield, increased food production, food supply sustainability due to better farm performance, and more effective response to issue of diseases on the farm. The broader effects of FDD extend to agricultural productivity, food security, and economic stability. This leads to increased yields, healthier crops, and more efficient use of resources. Improved disease control ensures a consistent supply of agricultural products, which helps farmers maintain their livelihoods and reduces the possibility of food shortages. This contributes to greater food security overall. In terms of economics, it raises the sustainability and profitability of agricultural methods by lowering the costs associated with pest control and crop failure. Moreover, these systems support the global clamor for climate change through increase and sustainable farm practices to help reduce flood and erosion. The system is limited to farmers who have internet-enabled mobile phones, only, this is because the email notification of the farm diseases requires internet for access.

APPENDIX A (SECONDARY DATA SOURCE)

https://www.kaggle.com/datasets/abdallahalidev/plantvillage-dataset

REFERENCE

[1] Syed-Ab-Rahman, S. F., Hesamian, M. H., and Prasad, M. "Citrus disease detection and classification using end-to-end anchor-based deep learning model", *Applied Intelligence*, [ISSN: 2352-9148] 2022; 52(1), 927–938. http://dx.doi.org/10.1007/s10489-021-02452-w

[2] Cardellicchio, A., Solimani, F., Dimauro, G., Petrozza, A., Summerer, S., Cellini, F., and Renò, V. "Detection of tomato plant phenotyping traits using YOLOv5-based single stage detectors", *Computers and Electronics in Agriculture*; 2023; 207; 107757.

[3] Shueil, L., Kaiyu L. et al. "A multi scale cucumber disease detection methods in natural scene based on YOLOv5", *Computr and electronics in agriculture*, 2022; Vol 202; 107363

[4] Ruparelia, S., Jethva, M., and Gajjar, R. "Real-time tomato detection, classification, and counting system using deep learning and embedded systems", In: Thakkar, F., Saha, G., Shahnaz, C., Hu, Y.-C. (Eds.), Proceedings of the International EConference on Intelligent Systems and Signal Processing. In: *Advances in Intelligent Systems and Computing, Springer, Singapore*, 2022; pp. 511–522. http://dx.doi.org/10. 1007/978-981-16-2123-9_39.

[5] Yao, J., Qi, J., Zhang, J., Shao, H., Yang, J., and Li, X. "A real-time detection algorithm for kiwifruit defects based on YOLOv5", *Electronics*, 2021; 10 (14), 1711. http://dx.doi.org/10.3390/electronics10141711, URL: https://www.mdpi.com/2079-9292/10/14/1711 Number: 14 Publisher: Multidisciplinary Digital Publishing Institute.

[6] Zhang, P., Li, D. "EPSA-YOLO-v5s: A novel method for detecting the survival rate of rapeseed in a plant factory based on multiple guarantee mechanisms", *Comput. Electron. Agric*, 2022; 193, 106714. http://dx.doi.org/10.1016/j.compag.2022.106714, URL: https://linkinghub.elsevier.com/retrieve/pii/S016816992200031X.

[7] Zheng, T., Jiang, M., Li, Y., Feng, M. "*Research on tomato detection in natural environment based on RC-YOLOv4*", Comput. Electron. Agric. 2022; 198, 107029. http://dx.doi.org/10.1016/j.compag.2022.107029, URL: https://linkinghub.elsevier.com/ retrieve/pii/S0168169922003465.

[8] Chidi, E., Udanor C., Anoliefo E. "Exploring the Depths of Visual Understanding: A Comprehensive Review on Real-Time Object of Interest Detection Techniques", *Preprints*,

2024; 2024020583. https://doi.org/10.20944/preprints202402.0583.v1

[9] Fan, S., Liang, X., Huang, W., Jialong Zhang, V., Pang, Q., He, X., Li, L., and Zhang, C. "Real-time defects detection for apple sorting using NIR cameras with pruningbased YOLOV4 network", *Comput. Electron. Agric*. 2022; 193, 106715. http://dx.doi.org/10.1016/j.compag.2022.106715, URL: https://www.sciencedirect.com/science/article/pii/S0168169922000321

[10] Wang, D., He, D. "Channel pruned YOLO v5s-based deep learning approach for rapid and accurate apple fruitlet detection before fruit thinning", *Biosyst. Eng*. 210, 271–281. http://dx.doi.org/10.1016/j.biosystemseng.2021.08.015, URL: https://www.sciencedirect.com/science/article/pii/S1537511021001999.

[11] Qi, J., Liu, X., Liu, K., Xu, F., Guo, H., Tian, X., Li, M., Bao, Z., Li, Y. "An improved YOLOv5 model based on visual attention mechanism: Application to recognition of tomato virus disease", *Comput. Electron. Agric.* 2022; 194, 106780. http://dx.doi.org/10.1016/j.compag.2022.106780, URL: https://www.sciencedirect.com/science/article/pii/S0168169922000977.

[12] Yao, J., Qi, J., Zhang, J., Shao, H., Yang, J., Li, X. "A real-time detection algorithm for kiwifruit defects based on YOLOv5", *Electronics* 10 (14), 2021; 1711. http://dx.doi.org/10.3390/electronics10141711, URL: https://www.mdpi.com/2079-9292/10/14/1711. Number: 14 Publisher: Multidisciplinary Digital Publishing Institute.

[13] Sozzi, M., Cantalamessa, S., Cogato, A., Kayad, A., Marinello, F. "Automatic bunch detection in white grape varieties using YOLOv3, YOLOv4, and YOLOv5 deep learning algorithms", *Agronomy* 12 (2), 2022; 319. http://dx.doi.org/10.3390/agronomy12020319, URL: https://www.mdpi.com/2073-4395/12/2/319. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

[14] Mirhaji, H., Soleymani, M., Asakereh, A., Abdanan Mehdizadeh, S. "Fruit detection and load estimation of an orange orchard using the YOLO models through simple approaches in different imaging and illumination conditions", *Comput. Electron. Agric*. 191, 2021; 106533. http://dx.doi.org/10.1016/j.compag.2021.106533, URL: https://www.sciencedirect.com/science/article/pii/S0168169921005500.

[15] Liu, X., Li, G., Chen, W., Liu, B., Chen, M., Lu, S. "Detection of dense citrus fruits by combining coordinated attention and cross-scale connection with weighted feature fusion", *Appl. Sci*. 12 (13), 2022; 6600. http://dx.doi.org/10.3390/ app12136600, URL: https://www.mdpi.com/2076-3417/12/13/6600. Number: 13 Publisher: Multidisciplinary Digital Publishing Institute.

[16] Adimora, K. C., Aru, O. E., Ilo, F. S., and Ezeh, C. M-E. "Optimization of Wi-Fi Throughput for IoT Livestock Monitoring Using Machine Learning", *Advances in Engineering Design Technology*, 4(1), 2022; 17-26. ISSN: 2682-5848. You can access the paper via: Optimization of Wi-Fi Throughput for IoT Livestock Monitoring Using Machine Learning

[17] Alonso, R. S., Sittón-Candanedo, I., García, Ó., Prieto, J., and Rodríguez-González, S. "An intelligent Edge-IoT platform for monitoring livestock and crops in a dairy farming scenario. Ad Hoc Networks", 98, 102047. ISSN: 1570-8705; 2020. https://doi.org/10.1016/j.adhoc.2019.102047.

[18] Darvesk K., Khande N., Avhad S. Khemchandani M. "IOT and AI based smart cattle health monitoring", *Journal of Livestock Science* (ISSN online 2277-6214), 2023; 14: 211-218

[19] Eneh P., Ene I., Egoigwe V., Ebere U. "Deep Artificial Neural Network Based Obstacle Detection and Avoidance for a Holonomic Mobile Robot", *International Research Journal of Applied Sciences, Engineering and Technology*, 2019; Vol.5, No.1; ISSN (1573-1405); p –ISSN 0920-5691

[20] Pham, T, Acharya P., Bachina S., Osterloh K., and Nguyen K. "Deep-learning framework for optimal selection of soil sampling sites", *Computers and Electronics in Agriculture* 217 108650, 2024; https://doi.org/10.1016/j.compag.2024.108650

[21] Dogra R., Rani S., Singh A., Albahar M., Barrera M., and Alkhayyat A. "Deep learning model for detection of brown spot rice leaf disease with smart agriculture", *Computers and Electrical Engineering* 109 108659, 2023; https://doi.org/10.1016/j.compeleceng.2023.108659

[22] Sanaeifar A., Guindo M., Bakhshipour A., Fazayeli H., Li X., Yang C. "Advancing precision agriculture: The potential of deep learning for cereal plant head detection" *Computers and Electronics in Agriculture* 209 107875, 2023.

[23] Sarany T., Deisy C., Sridevi S. , Kalaiarasi S. "A comparative study of deep learning and

Internet of Things for precision agriculture", *Engineering Applications of Artificial Intelligence* 122; 2023, 106034.
[24] Stefanics, D., and Fox, M. "COCO Annotator: Web-Based Image Segmentation Tool for Object Detection, Localization, and Key points", *ACM SIG Multimedia Records*, 2022; 13(3), 1.
[25] Khorram, S., Nelson, S., Cakir, H., Van Der Wiele, C. "Digital Image Processing: Post-processing and Data Integration", In: Pelton, J.N., Madry, S., Camacho-Lara, S. (eds) Handbook of Satellite Applications. Springer, New York, NY. 2013; https://doi.org/10.1007/978-1-4419-7671-0_92
[26] Zhu, L., Geng, X., Li, Z., Liu, C. "Improving YOLOv5 with Attention Mechanism for Detecting Boulders from Planetary Images", *Remote Sens.*, 2021; 13, 3776. https://doi.org/10.3390/rs13183776
[27] Kingma, D., Ba, J. "Adam: A method for stochastic optimization", 2014; arXiv, arXiv:14 12.69 80.
[28] Colombo, H. "YoloV-5 Model architecture", 2023; https://iq.opengenus.org/yolov5/; Accessed; 11/19/2023.