International Journal on Advanced Science Engineering Information Technology

Vol.13 (2023) No. 4
ISSN: 2088-5334

Deep Learning Model for Identification of Diseases on Strawberry (*Fragaria sp.*) Plants

Setyo Pertiwi [a,*], Dandi Handoko Wibowo [b], and Slamet Widodo [a]
[a] *Department of Mechanical and Biosystem Engineering, Faculty of Agricultural Engineering and Technology, IPB University, Indonesia*
[b] *Agricultural and Biosystem Engineering Study Program, Faculty of Agricultural Engineering and Technology, IPB University, Indonesia*
Corresponding author: [*]pertiwi@apps.ipb.ac.id

***Abstract*— Plant diseases can significantly affect crop productivity if not effectively managed. Accurate disease identification is critical for disease control and yield enhancement. Addressing these concerns, the potential application of deep learning techniques for plant disease identification is promising in Indonesia. This research aims to formulate a deep-learning model to detect strawberry (*Fragaria* sp.) plant diseases. The study encompasses several key phases, including (1) collecting datasets, (2) preprocessing datasets, (3) annotating datasets, (4) configuring and training deep learning models, and (5) validating and evaluating the model. The developed model employs YOLOv7 and YOLOv7-X algorithms, utilizing a dataset of 7337 instances across three disease categories: tip burn, leaf scorch, and anthracnose. These datasets were obtained from publicly accessible repositories. The evaluation of the deep learning model's performance in detecting plant diseases involved using 717 in-field plant images. The outcomes of the evaluation, employing YOLOv7 and YOLOv7-X algorithms, demonstrated accuracy rates of 92.5% and 92.3%, precision levels of 94.5% and 95.1%, and recall values of 90.5% and 89.6%, respectively. These results emphasize the effectiveness of the deep learning model in accurately and precisely identifying diseases in strawberry plants, and it could lay the groundwork for further studies using deep learning to detect diseases in various crops.**

***Keywords*— Deep learning; CNN; YOLO algorithm; diseases detection; strawberry.**

Manuscript received 17 May 2023; revised 29 Jun. 2023; accepted 15 Jul. 2023. Date of publication 31 Aug. 2023.

I. INTRODUCTION

Strawberry is an important fruit crop, providing essential nutrients and antioxidants to consumers worldwide. In Indonesia, the development of strawberry farming started in the early 2000s, and since then, it has been expanding rapidly due to the high demand for fruit in the local market. Strawberry production and popularity have been growing steadily in recent years. The fruit has a relatively high selling price due to increasing demand, but production is still limited because this plant can only grow optimally in high-altitude areas with cooler temperatures.

However, strawberry plants are susceptible to various diseases that can severely impact the yield and quality of the fruit. The warm and humid climatic conditions make this commodity susceptible to fungal, bacterial, and viral diseases. In their research, [1], [2] show a plant's susceptibility to disease. Early detection and identification of these diseases are crucial to prevent their spread and minimize the damage they cause. Conventional disease detection methods in plants rely on visual assessment, which can be both sluggish and inefficient, incurs substantial costs, is prone to subjectivity, exhibits low accuracy, and lacks timeliness [3].

One of the technologies that can be used to detect diseases in plants early is monitoring using a camera integrated with deep learning models. Deep learning (DL) is a subfield of machine learning model with a model complexity that makes it possible to represent data hierarchically through several levels of abstraction [4]. DL focuses on training artificial neural networks with multiple layers to learn hierarchical representations of data. It consists of various components that differ depending on the network architecture used such as Unsupervised Pre-trained Networks, Convolutional Neural Networks (CNN), etc. [5]. The highly hierarchical structure with the great learning capacity of the DL model allows for excellent classification and prediction, more flexibility, and adaptability for very complex problems in terms of data analysis such as audio, speech, and even advanced data such as weather and soil chemistry [6]. Recent advances in deep learning techniques, particularly convolutional neural

networks (CNNs), have shown great potential for disease identification in crops. Herein, leaves are important indicators of plant health, and several studies have explored the potential of leaf image analysis for disease identification in various crops [7], [8]. A number of deep learning models based on specific CNN architectures have been developed for classifying both crop species and identifying the presence of diseases on leaf images of healthy or diseased plants [9]–[12]. From those models, it has been discovered that deep learning models can analyze large datasets of images and learn to identify patterns and features indicative of particular diseases.

In 2016, Redmon et al. [13] introduced YOLO (You Only Look Once), an object detection algorithm that relies on deep neural networks for its object detection capabilities. YOLO utilizes a deep CNN as its backbone to process the input image and predict bounding boxes and class probabilities. The network is typically pre-trained on a large dataset, such as ImageNet, to learn general features and then fine-tuned on object detection tasks using labeled data. Over the years, several variations and improvements have been introduced to YOLO, such as YOLOv2, YOLOv3, and up to YOLOv8, addressing some of its limitations and further enhancing its performance and accuracy.

The YOLO algorithm has been widely applied in agriculture, such as research conducted by Liau and Wang [3] utilizing the modified YOLOv3 algorithm for detecting diseases and pests in tomato plants with an accuracy rate of up to 92.4%. YOLOv4 has been modified by Li et al. [14] to detect grapes under complex background conditions with an excellent accuracy of 95.2%, which meets the requirements of grape-picking robots for real-time detection of multiple varieties of table grapes in complex situations, and then Zhang and Li [15] used the YOLOv5s algorithm to detect the survival rate of seeds in plant factories with an accuracy of up to 94.5%. YOLOv5 was also used for detecting the bacterial spot disease in bell pepper plants from the symptoms seen on the leaves and was able to detect even a small spot of disease with considerable speed and accuracy [16]. Recently, YOLOv7 was used in the model to detect and identify different types of tea leaf disease in tea gardens. The model automatically detected five distinct types of tea leaf diseases and differentiated between healthy and diseased leaves with an overall classification accuracy is 97.30%, while recall and precision are 96.4% and 96.7%, respectively [17]. Considering these developments, deep learning methods, especially those with the YOLO algorithm, can potentially be developed to detect pests and plant diseases in Indonesia.

This study aims to develop a deep learning model tailored for the detection of strawberry (*Fragaria* sp.) plant diseases through leaf image analysis. In this study, YOLOv7 and YOLOv7-X are applied to extract and analyze the features of leaf images. The results of this study are expected to have significant benefits for strawberry cultivation in Indonesia, as they provide a fast and accurate method for detecting diseases that attack strawberry plants. The timely identification of the diseases will enable farmers to take appropriate action to prevent their spread and minimize the damage they cause. Furthermore, this study has the potential to establish a basis for subsequent studies on using deep learning techniques for disease identification in other crops, which can greatly benefit the agricultural sector.

II. Materials and Methods

The study was carried out by following the procedure depicted in Fig. 1.

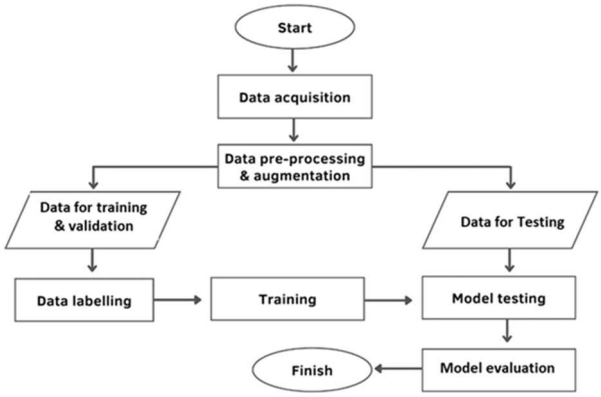

Fig. 1 Flowchart of the study

A. Dataset Acquisition

The datasets used were the images of diseased strawberry leaves and fruit with different angles and backgrounds downloaded from the Kaggle open repository (https://www.kaggle.com/) and images taken directly from strawberry fields in the Lembang and Ciwidey area, West Java. The datasets include tip burn [18], leaf scorch [19], and anthracnose [20]. Those three diseases are known as the common diseases of strawberries in the field or after postharvest. Tip burn is a major problem for most vegetable cultivation under a controlled environment [21]. It is a physiological disorder that results from environmentally induced localized calcium deficiency [22]. The most noticeable symptom of tip burn is the browning or yellowing of the tips or edges of the plant's leaves. Leaf scorch, also known as leaf burn or sunscald, is another common plant disorder characterized by the browning and drying leaf edges or entire leaves. Like tip burn, leaf scorch is a physiological condition rather than a result of a pathogenic infection. Anthracnose disease caused by *Colletotrichum* sp. affects many plants, including trees, shrubs, fruits, and vegetables. It is one of the major pathogens of strawberries [23], [24]. Dark, sunken lesions on leaves characterize anthracnose, stems, fruits, and other plant parts. Fig. 2 shows examples of the collected images.

The data collected was divided into data for training, validation, and testing, where training and validation use images originating from open repositories that have been processed, while testing uses a combination of open repository data and images taken directly from the field.

B. Data Preprocessing and Augmentation

The dataset collected was then preprocessed to get a more varied and larger number of images with objects of interest so that the resulting model performance would be even better. Processing was undertaken manually using Photoshop and automatically using the Roboflow platform (https://roboflow.com/). Some of the images were images of a single object (leaf, fruit), so modifications need to be made to resemble the original conditions of the plants in the field. Images with a single object were processed into synthetic data

to improve model performance and reduce training time compared to training on all single data without preprocessing. Making synthetic data requires images of healthy strawberry leaves taken from a strawberry field. It was processed manually using Photoshop (Fig. 3). Augmentation was carried out on the data that will be used for training so that the images used are more varied in terms of rotation (0°, 90°, 180°, 360°), horizontal flip, brightness (+/- 20%), exposure (+/- 20%), and background conditions which aim to adjust to original conditions on different land and to increase system generalization capabilities. From data preprocessing and augmentation, 492 image datasets have been obtained.

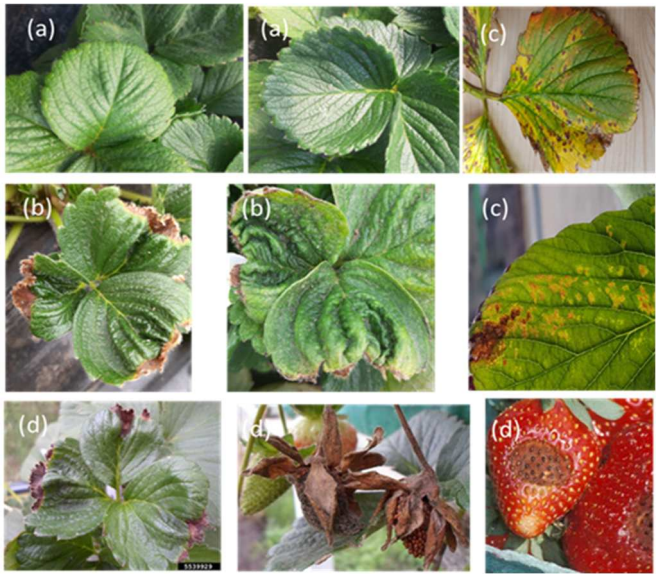

Fig. 2 Sample images of strawberry leaves: (a) healthy leaf; (b) tip burn; (c) leaf scorch; (d) anthracnose

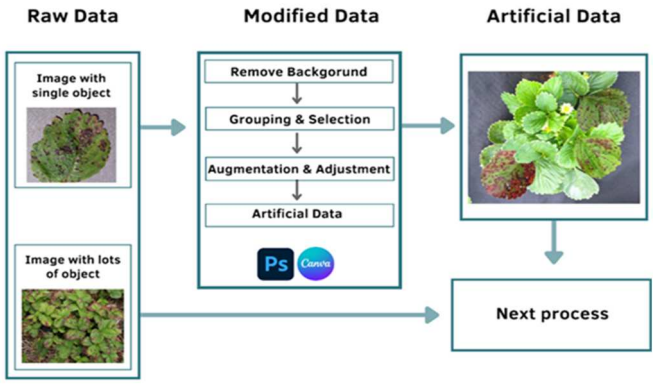

Fig. 3 Synthetic data creation process

C. Data Annotation

Data annotation is crucial in developing and training deep learning models, particularly for tasks like object detection using deep learning algorithms. The dataset annotation process was carried out by labeling the objects with bounding boxes and assigning appropriate class labels using Roboflow software. Roboflow can identify several kinds of deep learning algorithms as annotation files to be processed. In processing this dataset, the YOLO algorithm was selected so that the processed annotation files can be compatible with the YOLO deep learning algorithm. The Roboflow software labeling procedure is simply undertaken by pulling the bounding boxes to cover the objects that need to be detected. In this case, each leaf or fruit that is identified as diseased will be given bounding boxes. From this process, using 492 image data sets, a total of 7337 objects have been labeled or annotated. The output of this object labeling process is in the form of *.txt and *.xml files, which will be used in the training data process. Table 1 shows the results of data preprocessing, augmentation, and annotation.

TABLE I
DATASETS OBTAINED FROM DATA PREPROCESSING, AUGMENTATION, AND ANNOTATION

No	Label	Number of Objects
1	Leaf scorch	2400
2	Tip Burn	2485
3	Anthracnose	2452
	Total	7337

D. Deep Learning Model Configuration

Model configuration was undertaken by using YOLOv7. By considering the number of parameters, the amount of computation, and the computational density, the architecture of YOLOv7 was configured, as shown in Fig. 4.

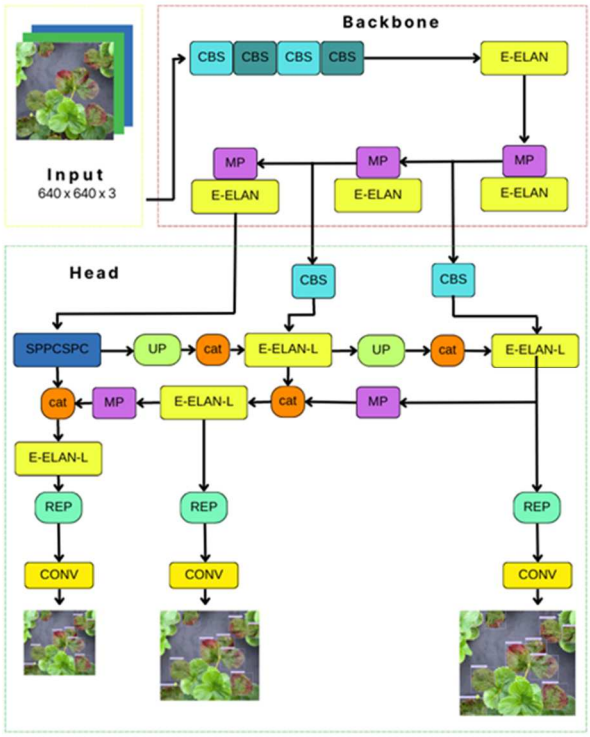

Fig. 4 Architecture of YOLOv7

The image detection process with YOLO has three steps: changing the size of the input image, running a single convolutional network on the input image, and applying a threshold based on the confidence value. According to wang et al. [25], using the same dataset and computational capabilities, YOLOv7 performs better than previous YOLO versions and other object detection algorithms, namely MS COCO and GPU V100. It was reported that YOLOv7 has a better AP (Average precision) value than other algorithms, with the highest score of 56.8% for the YOLOv7-E6E model. YOLOv7 also has better accuracy and detection time than YOLOR, YOLOX, Scaled-YOLOv4, YOLOv5, and other object detection algorithms.

Based on the architecture shown in Fig. 4, YOLOv7 consists of three parts: the input, backbone network, and head for detection. YOLOv7 changes the input size to 640×640×3 pixels, which will then be forwarded to the backbone network. The backbone network is a trained network consisting of a CBS composite module for convolution, batch normalization, and activation function. Then the Extended Efficient Layer Aggregation Networks (E-ELAN) module and MP module, composed of MaxPool and CBS module, are used to extract important features from an image, help reduce the spatial resolution of an image, and increase its feature resolution alternately. To continuously increase the learning rate without destroying the original gradient path, the ELAN module expands, shuffles, and combines each set to improve network accuracy. In this network, the convolution process is carried out to produce a feature map.

Furthermore, the feature map resulting from multiplication matrix convolution with filters of different sizes will be forwarded to the head network. The input from the backbone network spreads out to multiply the feature map and merge the learned information, improving target detection performance and the model's ability to recognize important features. The SPPCSPC module is used to increase the network reception field by obtaining multi-scale object information without changing the feature map size. After the input image has been through feature extraction, this algorithm will divide the image into s x s sized grids, which then in each grid will predict the bounding box and the class map of each grid. Each bounding box has parameters: x, y, w, h, and confidence. Parameters x and y are the coordinates of the bounding box. Parameters w and h are the height and width of the bounding box. Confidence is the value of the intersection over union from calculating the predicted box and ground-truth box on the grid which will predict the class probability. The confidence value will predict the confidence level in detecting an object [13].

E. Training

Training, also known as the learning phase, involves instructing an algorithmic model to analyze the gathered data. This training procedure utilizes annotation files derived from the labeling results of the dataset. The training process aims to train the model by processing images that have been annotated or labeled so that patterns or characteristics of each class are formed, which will be taken into consideration by the computer in reaching a decision or prediction. In this study, the training process was carried out using Google Collaboratory with an Epoch value of 200, batch size 8, input 640×640. The pre-trained models to be used are YOLOv7 and YOLOv7-X. YOLOv7-X is the variant of YOLOv7 that differs from YOLOv7 in terms of layer count, with YOLOv7-X having 467 layers while YOLOv7 has 415 layers.

F. Model Testing and Evaluation

The validity test process is carried out to check the accuracy and precision of the model that the learning process has carried out. The validity test is carried out using the weight value obtained from the learning process. The input data used is in the form of images of healthy and diseased plants, then the output generated in this process is in the form of boxes that limit the plants and have been identified according to their conditions. Data is separated into positive (object) and negative (non-object) in binary classification. The validation test results must measure the model's accuracy in detecting objects. This process is carried out by manually calculating the correct identification results from the results of identifying plant conditions.

In this study, the confusion matrix is used to calculate precision, recall, mAP, and F1 scores to evaluate model performance. The confusion matrix shown in Fig. 5 is used to generate actual classification results and predicted results by a classification system. The results of the classification are evaluated using a data matrix. The model evaluation formula can be seen in the equations below.

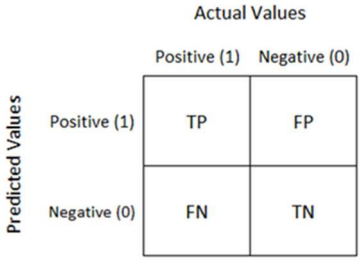

Fig. 5 Confusion Matrix

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$F1 - Scores = \frac{2 \text{ x Precision x Recall}}{TP + \text{Precision} + \text{Recall}FN} \quad (3)$$

$$\text{mAP} = \frac{1}{n}\sum_{k=1}^{k=n} AP_k \quad (4)$$

$$AP_k = Average\ Precision\ of\ Class\ k \quad (5)$$

In which,

True Positive (TP): The predicted value is positive and actually it is positive.

False Positive (FP): The predicted value is positive when actually belongs to the negative.

False Negative (FN): The predicted value is negative when actually belongs to the positive.

True Negative (TN): The predicted value is negative and actually it is negative.

III. Results and Discussion

A. Model Development

The training process aims to produce models or programs that have been trained using datasets to recognize patterns or characteristics in new data to make decisions or predictions. The resulting model is a weight file (.pt) containing mathematical calculations to estimate output based on input and training. The datasets employed during the training procedure are categorized into three segments: datasets allocated for training, validation, and testing, adhering to a distribution ratio of 70% for training, 20% for validation, and 10% for testing. In general, there are no specific provisions

for the data used, but it is recommended that the datasets for training are larger than the data for validation and testing of the total dataset so that learning outcomes are maximized. In this study, the algorithms used are YOLOv7 and YOLOv7-X. Each training process uses an epoch value of 200, a batch size of 8, and an input of 640 × 640. A comparison of the training performance of each YOLOv7 model is described in Table 2.

TABLE II
COMPARISON OF TRAINING PERFORMANCE DATASET FOR EACH YOLOV7 MODEL

Model	Epochs	Runtime (hour)	P (%)	R (%)	F1-Score (%)	mAP @5 (%)
YOLOv7	200	1.360	94.00	89.40	91.60	91.70
YOLOv7-X	200	1.815	93.00	88.90	90.90	90.20

The training results show that the highest precision (P), recall (R), mean average precision (mAP @ 0.5), and fastest training time are owned by the YOLOv7 model, with the largest difference found in the mAP value of 1.5% with YOLOv7-X. In addition to performance results in the form of P, R, F1-Score, and mAP values, training performance results can also be seen from the graph of the model development results. Each parameter on the graph describes the performance of each passed epoch, starting from graph precision, recall, mAP, and loss in objects and classes. The graphical results of each model's performance represented by the loss graph can be seen in Fig. 6.

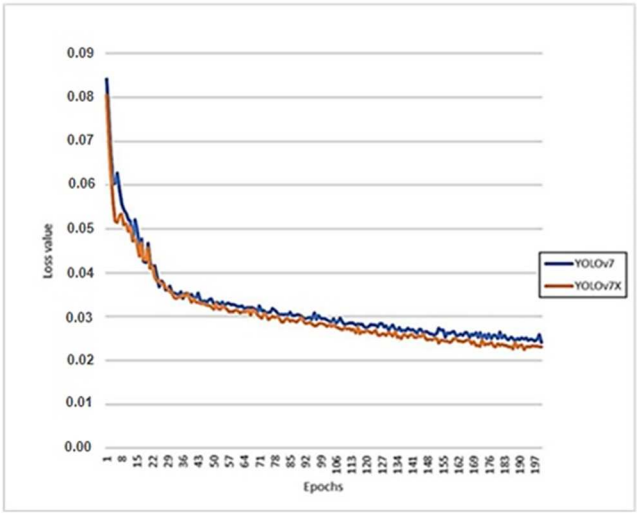

Fig. 6 The graph of loss during training

Based on the results of the YOLOv7 and YOLOv7-X training performance. It shows that the box loss curve for the train decreases in value from the first epoch to the 200th epoch; it shows a stable decline until the lowest value is less than 0.03. Loss values close to zero suggest that the model created can detect the targeted object. In the process of creating a dataset, bias in the data may not be avoidable. Even with a careful review of the algorithms and data sets, it may not be possible to delete all unwanted biases, particularly because the systems learn from historical data, which encodes historical biases [26]. Model and Shamir have comprehensively discussed the possible dataset bias factors and the necessary action to reduce dataset bias [27]. The bias in the data may be causing loss and affecting the ability to detect objects. Therefore, data quality has a considerable influence on object detection performance. The data shows that both models have a fairly low loss and are good to use.

B. Detection Results

The results of the weight or weight values obtained are re-tested by direct testing on the land. This is because the model was created using plant data that is not from Indonesia, and the proportion of artificial data is higher than the original image on unprocessed land. The test aims to ensure that the model can be applied directly to land in Indonesia. Tests were again carried out on land in Lembang and Ciwidey, West Java. Weight testing on images is carried out using dataset testing to test detection performance and determine the value of precision, recall, F1-Score, mAP, and average IOU of the model that has been made so that it can be compared with the model's computational output and can be evaluated later. An example of a detection result image can be seen in Fig. 7.

Fig. 7 The results of disease detection in the image obtained from the field

Fig. 7 shows the prediction results made by the model on the field and test data, where the model can predict the results very well with little error. As seen in the figure, the model can predict leaf scorch disease very well even though there are some leaves that are infected with the disease but not detected. This can be caused by several factors, including the confidence value that is below the threshold (0.4), the object being too small, the image being blurry, or there are objects attached to other objects that have been predicted by the bounding box so that the model considers the objects as the same leaf. The prediction results for the tip burn class also showed very good results where diseased leaves could be detected correctly, with the lowest confidence value of 69%. A comparison of the detection results for the anthracnose class in fruit shows that both models are very good at detecting it, but the YOLOv7-X model makes a slight error where healthy objects adjacent to diseased objects are detected as diseased objects. Direct testing on land shows that the model that has been developed not only has excellent performance in the training process but also has good performance when applied to land. This is evidenced by the detection results where the model that has been developed can distinguish objects in the form of healthy leaves from diseased leaves with little error.

C. Evaluation

The performance results are evaluated by calculating the value of precision, recall, F1-Score, mAP, and average IOU. Performance evaluation is carried out using a confusion matrix table and formulas for precision, recall, F1-Score, mAP, and average IOU from the resulting values in the confusion matrix table. The data used for evaluation is data outside the training dataset, which consists of 717 detection objects. The results of calculating the evaluation matrix for each weight model algorithm can be seen in Table 3 and Table 4.

TABLE III
EVALUATION RESULTS OF THE YOLOV7 MODEL USING TESTING DATA

Class	P	R	F1-Score	mAP @0.5	IoU
Anthracnose	91.7%	91.7%	91.7%	90.0%	75.6%
Leaf Scorch	96.4%	95.3%	95.8%	96.0%	72.6%
Tip burn	95.4%	84.5%	89.6%	89.8%	71.9%
Total	94.5%	90.5%	92.5%	91.9%	73.4%

TABLE IV
EVALUATION RESULTS OF THE YOLOV7-X MODEL USING TESTING DATA

Class	P	R	F1-Score	mAP @0.5	IoU
Anthracnose	93.0%	92.1%	92.5%	91.9%	72.5%
Leaf Scorch	96.5%	94.9%	95.7%	95.1%	72.2%
Tip burn	95.7%	81.9%	88.3%	89.7%	68.6%
Total	95.1%	89.6%	92.3%	90.9%	71.1%

Based on these results, precision, recall, F1-Score, mAP, and average IOU values were obtained. The YOLOv7-X model has the highest precision value at 95.1% (Table 4). For tip burn, the results are better than the CNN model developed with YOLO5 for detecting the tip-burn stress on lettuce grown in an indoor environment, achieving the best accuracy with 84.1% mAP [28]. For leaf scorch, the results are comparably as higher than the field experiment results of the 4 CNN models, SqueezeNet, EfficientNet-B3, VGG-16, and AlexNet developed by Abbas et al. [29] for identifying strawberry leaf scorch disease in a real-time strawberry field which achieved the highest classification accuracy of 0.80 and 0.86 for initial and severe disease stages, respectively. Regarding anthracnose, the deep learning model developed by Anagnostis et al. [30] achieves an average precision of 63% for object detection and an accuracy of 87%. Yet, comparing these results here is less precise because it uses tree canopy images as the input. A high precision value shows that the deep learning model has a high consistency level in terms of object detection because the precision value indicates the model's consistency level in conducting classification [31]. The highest Recall value with a value of 90.5% (Table 3) for YOLOv7 has a very small difference with the YOLOv7-X model, which is 0.9%. The recall value reflects the model's proficiency in classification consistency. The distinction between precision and recall arises from their calculation methods. Precision measures the proportion of accurate predictions relative to the overall intended predictions, whereas recall is the ratio of accurate predictions to the total count of assigned actual classes. The recall values obtained in the two models do not differ much, indicating that the two models have a fairly high success rate in differentiating each class. F1-score is an alternative for accuracy that describes the comparison of the weighted average precision and recall. The F1-score value in the test data shows that both models can predict an object very accurately, where the YOLOV7 model is superior (92.5%) compared to YOLOv7-X (92.3%). YOLOV7 exhibits the highest mAP and average IoU values, reaching 91.9% and 73.4%, respectively. The average IoU value serves as an indicator of the confidence level of the model in object detection. These findings suggest that the deep learning model manifested as a weight file (.pt) demonstrates notable consistency in accurate object detection. This observation is underlined by the higher number of correctly detected objects compared to those incorrectly detected (Fig. 7). Model errors in detecting objects occur because the images are unclear and have a high degree of similarity, so the model is unable to distinguish properly.

Given the above-discussed results, the deep learning model that has been developed can be implemented into an application program, preferably web-based. This is because the website is a platform that is cheap, stable, and easy to access. Furthermore, it is realistic to consider that this study could lay the groundwork for further studies using deep learning to detect diseases in various crops, offering significant benefits for agriculture.

IV. CONCLUSION

The deep learning model was successfully developed using the YOLOv7 and YOLOv7-X algorithms. The model is in the form of a weight file with mathematical calculations based on input and training for disease detection on strawberry fruit leaves. Evaluation of model performance for precision, recall, F1-Score, mAP, and average IOU both received high scores from the results of validation data calculations and field testing. Based on the values obtained from the evaluation using dataset testing, YOLOv7 with less weight has the best performance with a precision value of 94.5%, Recall 90.5%, F1-score 92.5%, mAP 91.9%, and average IOU 73.4%. Both models have performance results that do not have significant differences in terms of precision, recall, F1-Score, mAP, and average IOU values. This suggests that both models have successfully identified the target object, as demonstrated by the loss value approaching zero (<0.3). Direct testing on land also proves that the model has good performance during the training process and can be implemented on land with very little error. These results could lay the groundwork for further studies using deep learning to detect diseases in various crops.

REFERENCES

[1] J. A. Diaz-Pendon, M. C. Canizares, E. Moriones, E.R. Bejarano, H. Czosnek, J. Navas-Castillo, "Tomato yellow leaf curl viruses: Menage a trois between the virus complex, the plant and whitefly vector," *Mol. Plant Pathol*., vol. 11, no. 4, pp 414–450, 2010, doi:10.1111/j.1364-3703.2010.00618.X.

[2] R. L. Gilbertson, O. Batuman, "Emerging viral and other diseases of processing tomatoes: biology diagnosis and management," *Acta Hortic*. no.1, pp. 35–48, 2013, doi:10.17660/ActaHortic.2013.971.2.

[3] J. Liu and X. Wang, "Tomato diseases and pests detection based on improved Yolo V3 convolutional neural network," *Front. Plant Sci.,* vol. 11, pp. 898, 2020, doi:10.3389/fpls.2020.00898.

[4] J. Schmidhuber, "*Deep learning* in neural networks: An overview," *Neural Netw.,* vol. 61, pp. 85-117, 2015.

[5] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep Learning in Agriculture: A Survey," *Comput. Electron. Agric.,* vol. 147, pp. 70-90, doi:10.1016/j.compag.2018.02.016.

[6] X. Song, G. Zhang, F. Liu, et. al., "Modeling spatio-temporal distribution of soil moisture by *deep learning*-based cellular automata

model," *J. Arid Land.,* vol. 8, pp. 734–748, 2016, doi:10.1007/s40333-016-0049-0.

[7] S. Arivazhagan, R. N. Shebiah, S. N. Ananthi, and S. V. Varthini, "Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features," *Agric. Eng. Int.: CIGR J* vol. 15, no. 1, pp. 211-217, 2013.

[8] V. Singh, and A. K. Misra, "Detection of plant leaf diseases using image segmentation and soft computing techniques," *Inf. Process. Agric.,* vol. *4, no.* 1, pp. 41-49, 2017, doi:10.1016/j.inpa.2016.10.005.

[9] S. P. Mohanty, D .P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Front. Plant Sci.,* vol. 7, pp. 1419, 2016.

[10] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, D., and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Comput. Intell. Neurosci,* vol. 2016, pp 3289801, 2016.

[11] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Comput. Electron. Agric.,* vol. 145, pp. 311-318, 2018, doi: 10.1016/j.compag.2018.01.009.

[12] V. Tiwari, R.C. Joshi, and M. K. Dutta, "Dense convolutional neural networks based multiclass plant disease detection and classification using leaf images," *Ecol. Inform.,* vol. 63, pp. 101289, 2021.

[13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 779-788, 2016, doi:10.1109/CVPR.2016.91.

[14] H. Li, C. Li, G. Li, and L. Chen, "A real-time table grape detection method based on improved YOLOv4-tiny network in complex background," *Biosyst. Eng.,* Vol 212, pp. 347-359, 2021, doi:10.1016/j.biosystemseng.2021.11.011.

[15] P. Zhang and D. 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.,* vol. 193, pp. 106714, 2022, doi:10.1016/j.compag.2022.106714.

[16] M. P. Mathew, and T. Y. Mahesh, "Leaf-based disease detection in bell pepper plant using YOLO v5," *SIViP,* vol. 16, pp. 841–847, 2022, doi:10.1007/s11760-021-02024-y.

[17] M. J. A. Soeb, M. F. Jubayer, T. A. Tarin, *et al.,* "Tea leaf disease detection and identification based on YOLOv7 (YOLO-T). *Sci. Rep.*. vol. 13, pp. 6078, 2023, doi:10.1038/s41598-023-33270-4.

[18] M. Hariri, and E. Avşar, "Tipburn disorder detection in strawberry leaves using convolutional neural networks and particle swarm optimization," *Multimed. Tools. Appl.,* Vol. 81, no. 8, pp. 11795-11822, 2022, doi:10.1007/s11042-022-12759-6.

[19] D. Hughes, and M. Salathé, "An open access repository of images on plant health to enable the development of mobile disease diagnostics", *arXiv preprint arXiv:1511.08060,* 2015.

[20] U. Afzaal, B. Bhattarai, Y.R. Pandeya, and J. Lee, "An Instance Segmentation Model for Strawberry Diseases Based on Mask R-CNN," *Sensors,* vol. 21, pp. 6565, 2021.

[21] E.F. Cox, J. M. T. McKee, "A Comparison of Tipburn Susceptibility in Lettuce Under Field and Glasshouse Conditions," *J. Hortic. Sci.* vol. 51, pp. 117–122, 1976.

[22] M. Kroggel, and C. Kubota, "Controlled environment strategies for tipburn management in greenhouse strawberry production," In *VIII International Strawberry Symposium* vol.1156, pp. 529-536, August 2016.

[23] G. N. Agrios, "*Plant Pathology",* 5th Ed., San Diego California, Academic Press. I. N. 635 p, 2005.

[24] B. J. Smith, "Epidemiology and pathology of strawberry anthracnose: a North American perspective," *HortScience*, vol. 43, no. 1, pp. 69-73, 2008.

[25] C. Y. Wang, A. Bochkovskiy, and H. Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.,* pp. 7464-7475, 2022, doi:10.48550/arXiv.2207.02696.

[26] D. Rosell, J. Matthews, and N. Talagala, "Managing bias in AI," *In Companion Proc. of 2019 TheWebConf.,* pp. 539-544, May 2019.

[27] I. Model and L. Shamir, "Comparison of Data Set Bias in Object Recognition Benchmarks", *IEEE Access*, vol. 3, no. 1, pp. 1953-1962, 2015.

[28] M. H. Hamidon and T. Ahamed, "Detection of Tip-Burn Stress on Lettuce Grown in an Indoor Environment Using Deep Learning Algorithms." *Sensors,* vol. 22, pp. 7251, 2022, doi:10.3390/s22197251.

[29] I. Abbas, J. Liu, M. Amin, A. Tariq, and M. H. Tunio, "Strawberry fungal leaf scorch disease identification in real-time strawberry field using deep learning architectures," *Plants,* vol. 10, no. 12, pp. 2643, 2021.

[30] A. Anagnostis, A. C. Tagarakis, G. Asiminari, et. al., "A deep learning approach for anthracnose infected trees classification in walnut orchard," *Comput. Electron. Agric.,* vol. 182, pp. 105998, 2021, doi:10.1016/j.compag.2021.105998.

[31] O. Natan, A. I. Gunawan, and B. S. B. Dewantara, "Grid SVM: Machine Learning Applications in Aquaculture Data Processing," (in bahasa), *Jurnal Rekayasa Elektrika* vol. 15, no. 1, pp. 7-17, 2019, doi:10.17529/jre.v15i1.13298.