Journal of Engineering and Technological Sciences

Research Paper

Comparison Study of Corn Leaf Disease Detection based on Deep Learning YOLO-v5 and YOLO-v8

Nidya Chitraningrum[1], Lies Banowati[2], Dina Herdiana[3], Budi Mulyati[2], Indra Sakti[4], Ahmad Fudholi[5,6], Huzair Saputra[7], Salman Farishi[8], Kahlil Muchtar[9,10,*] & Agus Andria[11]

[1]Research Center for Biomass and Bioproducts, National Research and Innovation Agency (BRIN), Jalan Raya Jakarta-Bogor No. KM 46, Cibinong, Bogor 16911, Indonesia
[2]Department of Aeronatics Engineering, Universitas Nurtanio, Jalan Pajajaran No. 219, Bandung 40174, Indonesia
[3]Department of Electrical Engineering, Universitas Nurtanio, Jalan Pajajaran No. 219, Bandung 40174, Indonesia
[4]Research Center for Telecommunication, National Research and Innovation Agency (BRIN), Jalan Sangkuriang, Kawasan KST Samaun Samadikun Gd. 20, Bandung 40135, Indonesia
[5]Research Center for Energy Conversion and Conservation, National Research and Innovation Agency (BRIN), Kawasan PUSPIPTEK, Kluster Energi Gd. 620-625, Setu – Tangerang Selatan 15314, Indonesia
[6]Solar Energy Research Institute, Level G Research Complex, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia
[7]Master of Artificial Intelligence, Department of Informatics, Universitas Syiah Kuala, Jalan Syech Abdurrauf No. 3, Kopelma Darussalam, Banda Aceh 23111, Indonesia
[8]Research Center for Polymer Technology, National Research and Innovation Agency (BRIN), Kawasan PUSPIPTEK, Gd. 460, Setu – Tangerang Selatan 15314, Indonesia
[9]Department of Electrical and Computer Engineering, Universitas Syiah Kuala, Jalan Syech Abdurrauf No. 7, Kopelma Darussalam, Banda Aceh 23111, Indonesia
[10]Telematics Research Center (TRC), Universitas Syiah Kuala, Jalan Syech Abdurrauf No. 3, Kopelma Darussalam, Banda Aceh 23111 Banda Aceh, Indonesia
[11]The Central Bureau of Statistics (BPS) Kabupaten Simeulue, Jalan Tgk. Diujung Desa Air Dingin, Sinabang, Simelue Timur, Simeulue Regency, Aceh 24782, Indonesia

*Corresponding author: kahlil@usk.ac.id

Abstract

Corn is one of the primary carbohydrate-rich food commodities in Southeast Asian countries, among which Indonesia. Corn production is highly dependent on the health of the corn plant. Infected plants will decrease corn plant productivity. Usually, corn farmers use conventional methods to control diseases in corn plants. Still, these methods are not effective and efficient because they require a long time and a lot of human labor. Deep learning-based plant disease detection has recently been used for early disease detection in agriculture. In this work, we used convolutional neural network algorithms, namely YOLO-v5 and YOLO-v8, to detect infected corn leaves in the public data set called 'Corn Leaf Infection Data set' from the Kaggle repository. We compared the mean average precision (mAP) of mAP 50 and mAP 50-95 between YOLO-v5 and YOLO-v8. YOLO-v8 showed better accuracy at an mAP 50 of 0.965 and an mAP 50-95 of 0.727. YOLO-v8 also showed a higher detection number of 12 detections than YOLO-v5 at 11 detections. Both YOLO algorithms required about 2.49 to 3.75 hours to detect the infected corn leaves. This all-trained model could be an effective solution for early disease detection in future corn plantations.

Keywords: *convolutional neural network; corn leaf disease; deep learning; disease detection; YOLO models.*

Introduction

Corn is one of the primary foods besides rice for Indonesian people because corn contains carbohydrates, which are very important for the human body. Data from the National Food Agency show that domestic corn

ISSN: 2337-5779
J. Eng. Technol. Sci. Vol. 56, No. 1, 2023, 61-70
DOI: 10.5614/j.eng.technol.sci.2024.56.1.5

production in 2023 will reach approximately 16.84 million tons, as stated by Putra [1]. The initial stock for 2023 is 3.29 million tonnes. The demand for corn throughout the year is estimated at 16.44 million tons according to Rozi *et al*. [2]. Due to this high demand for corn every year in Indonesia, the quality of the corn crop must also be maintained. The quality of corn can be known by looking at the health of corn plants.

According to data from the Ministry of Agriculture, national corn production has an increasing trend yearly, as reported in the article by Dihini [3]. However, along with the increase in production, Suriani *et al.* [4] has stated that corn farming is also followed by more widespread plant disease, resulting in reduced corn yields. At a severe level, corn disease can even cause crop failure. In the implementation of corn cultivation, there are obstacles such as corn plants being attacked by pests and diseases. For this reason, to avoid crop failure due to plant disease, diseases that may attack corn plants need to be recognized so the right solutions can be applied to overcome the diseases.

Over the last few years, deep learning algorithms have been developed for agricultural applications as investigated by Wang *et al.* [5], Saleem *et al.* [6], and Zhu *et al.* [7]. To be more specific, these previous applications aimed to detect pests, leaf diseases, and many other plant diseases that may reduce crop yield. Compared to conventional detection by farmers, deep learning methods are expected to be more effective and efficient. The convolution neural network (CNN) is one of the deep learning algorithms to analyze visual images effectively and accurately, according to Alzubaidi *et al.* [8], and Hafifah *et al*. [9]. Recognition system-based CNNs for corn plants, either for the purpose of detection or classification, have been studied by several researchers. The work by Javanmardi *et al.* [10] performed automated detection of corn seed varieties using a deep CNN with several extracted features. The optimum performance was achieved by the CNN-ANN classifier, with accuracy up to 98.1 % and 98.2 % of precision. Among several CNN architectures, the YOLO (You Only Look Once) algorithm is the most popular deep learning method, as studied by Boudjit *et al.* [11], and Dulal *et al.* [12]. YOLO has a single-stage detection approach that can detect an object with high accuracy in real time, as stated by Hussain [13]. The YOLO algorithm has developed from the first version since 2015 [15], from YOLO-v1 to the newest version of YOLO-v8 from 2023. Among all of the YOLO versions, YOLO-v5 and YOLO-v8 are the most popular models for object detection. Both YOLO-v5 and YOLO-v8 were developed by Ultralytics in 2020 and 2023, respectively. Jocher *et al.* [13] conducted a comparison between YOLO-V5, YOLO-v6, and YOLO-v8 at an image resolution of 640 x 640 pixels. With the same number of parameters, YOLO-v8 showed the best performance, owing to the more hardware-efficient architectural modifications. YOLO-v5 provides excellent real-time performance while YOLO-v8 focuses on constrained edge device deployment at high-inference speed [14]. YOLO-v5 offers ease of use, while YOLO-v8 offers faster and more accurate models.

In this work, we compared two variants of the YOLO algorithm, namely YOLO-v5 and YOLO-v8, in detecting corn leaf disease from an open-source data set in the Kaggle repository. The specific aim was to perform a comparative analysis between the YOLO-v5 and YOLO-v8 in detecting corn leaf disease to provide great benefit to farmers. According to the data training, we compared the accuracy of YOLO-v5 and YOLO-v8 based on their mAP score, number of detections, and speed. YOLOv8 had higher accuracy (about 0.965 of mAP 50) and more leaf infection detections (12) compared to YOLO-v5 (about 0.909 of mAP 50 and 11 detections). However, YOLO-v5 was faster than YOLO-v8. Our results showed that YOLO-v5 and YOLO-v8 can both be applied for corn leaf disease detection with high accuracy and speed.

Methods

Figure 1 shows the steps of the corn leaf disease detection method used in this work. The steps consist of data collection, importing the collected data, data pre-processing, building the model and applying the data sets on the YOLO-v5 and YOLO-v8 variants, prediction for the training and testing model, and evaluating the models based on the mAP scores (mAP50 and mAP50-95).

Data Collection

In this paper, we used an open-source data set from the Kaggle repository, namely the 'Corn Leaf Infection Data set' by Acharya [16], which consists of 4,225 images with a high resolution of 3,456 x 4,608 pixels. These images are categorized as 'healthy' and 'infected' images, as shown in Table 1.

DOI: 10.5614/j.eng.technol.sci.2024.56.1.5

Table 1 Corn leaf image data set.

Label	Total	
Healthy	2,000 images	0 annotations
Infected	2,225 images	11,596 annotations

The prepared data set of infected corn leaf images was stored in a dedicated folder. It contained 2,225 images of corn leaves that were used for training and validation. There were 2,225 corresponding labels, explicitly indicating whether a leaf is infected or not. There were 11,596 bounding box annotations, precisely pinpointing the location of infected leaves within the images, facilitating accurate object detection.

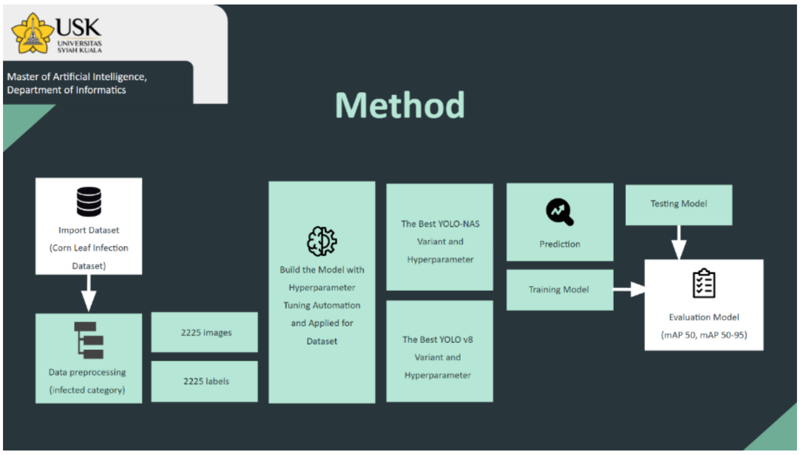

Figure 1 The method steps of corn leaf disease detection.

A total of 2,00 images were labeled as ‘healthy’ and 2,225 images were labeled as ‘infected’. Therefore, the entire data set was imbalanced. There is an explanation file in which the infected images have bounding box coordinates. In total, 11,596 bounding boxes were annotated among the 2,225 images.

Importing Data Set

In importing the data set, a set of infected corn leaves was collected in a folder that contained the images of the corn leaves and labels indicating the infected leaves. The data set may include a total of 2,225 images and 2,225 labels. Raw image samples of the corn leaf data set are shown in Figure 2.

Figure 2 Sample raw images of infected corn plants.

Preprocessing Data (Infected Category)

During the data preprocessing pipeline, a sequence of essential operations is applied to ensure proper formatting of input images for subsequent tasks, likely in the domain of computer vision or deep learning. The process begins by reading each image from a specified input directory and verifying the successful loading of each image. To maintain consistency, the color representation of the images is converted to RGB format.

Further adjustments are made to standardize image orientation, with a 90-degree clockwise rotation applied to images whose width is less than their height. This step ensures a uniform orientation across all images. Then, a scaling factor is calculated based on a predefined maximum dimension. Using this factor, the images are resized while preserving their original aspect ratio, contributing to a standardized input size.

The resized images are saved to an output directory, facilitating organized storage of the preprocessed data. This preprocessing pipeline was designed to homogenize the input data, addressing potential variations in image orientation and size that may arise from diverse sources.

(a) (b)

Figure 3 Images of the original infected leaf (a) and the normalized one (b).

Building Models and Applying to Data Sets

In this step, a model is built to detect infected corn leaves using two different variants of the YOLO (You Only Look Once) algorithm: YOLO-v5 and YOLO-v8. YOLO is an object detection algorithm that can identify and detect objects in an image. By applying the selected YOLO models to the preprocessed data set, these YOLO models will train the data to detect infected corn leaves. Adjusting several parameters for the data sets of the infected corn leaves should be performed to improve the performance of all models, as has been done by Ashtiani *et al.* [17]. The parameters that will be set in the training model are Optimizer, Learning Rate, Epoch, Image Size, Batch Size, and Loss Function as in Table 2.

YOLO (by default) uses BCE (Binary Cross-Entropy) for classification, DFL (Deep Feature Loss) and CIoU (Complete Intersection over Union) losses for bounding box regression, respectively. The final loss is the weighted sum of these three individual losses. The weights control the relative importance or contribution of each loss in the final combined loss that gets optimized during training.

DOI: 10.5614/j.eng.technol.sci.2024.56.1.5

Table 2 Hyperparameters used to train all models.

Hyperparameter	Value
Optimizer	AdamW
Learning Rate	0.01
Epoch	100
Image Size	416
Batch Size	16
Loss Function	BCE, DFL and CIoU

Hardware and Software Specifications

Hardware specifications:

Hardware Description
1 unit PC/laptop equipped with a 12th Gen Intel(R) Core (TM) i5-1235U processor (12 CPUs) at ~1.3GHz, 16.00 GB RAM.
USK GPU Server featuring NVIDIA GeForce RTX 2080 Ti Rev. A with 4 GPUs, each having a capacity of 11019MiB, 64 CPUs, 125.5 GB RAM.
(GPU) backend Google Compute Engine Python 3 Server featuring NVIDIA Tesla T4, 15102MiB with 2 CPUs, 12.7 GB RAM.

Software specifications:

Software Description
Operating System: Windows 11 Pro 64-bit
Python: Python programming language (installable from python.org)
IDE: Visual Studio Code
Deep Learning Frameworks: PyTorch
GPU Acceleration Libraries: CUDA
Jupyter Notebooks: Jupyter (installable using **pip install notebook**)
Additional Libraries and Tools: scikit-learn, pandas, matplotlib, etc.

Prediction Including Training and Validation Model

During the training step using Google Compute Engine (GPU backend) Server, the models learn to detect infected corn leaves by iteratively adjusting the internal parameters based on the provided training data set, which consisted of a total of 2,225 images. The validation set used for model evaluation is not a separate subset but rather a portion of the training data set. Specifically, it may be half of the training data set, and the selection process, whether random or systematic, is a critical aspect for ensuring a fair assessment of the model's generalization performance. After the model is trained, it can predict new or unseen data.

Evaluation model (mAP 50 and mAP 50-95)

We calculated the mean average precision (mAP) score to evaluate the trained model's performance. The mAP score is a commonly used evaluation metric in object detection tasks. 'mAP 50' refers to the average precision over the intersection over Union (IoU) threshold of 0.5, while 'mAP 50-95' means the average precision across the IoU threshold from 0.5 to 0.95. The mAP score indicates how the model performs in detecting the infected corn leaves at various levels of precision.

Result and Discussion

Table 3 shows the mAP score (mAP 50 and mAP 50-95), the number of infected leaves detected, and the speed obtained after training the infected leaf data set. The higher the mAP score, the better the accuracy of detecting the infected leaves (Hong *et al*. [18]).

Table 3 The mAP score of various YOLO models on the infected corn leaf images.

Model	mAP 50-95	mAP 50	Detection (s)	Speed (hours)
YOLOv5n	0.425	0.741	11	2.735
YOLOv8n	0.470	0.792	12	3.137
YOLOv5s	0.527	0.847	9	2.490
YOLOv8s	0.602	0.909	7	3.005
YOLOv5m	0.585	0.895	6	2.684
YOLOv8m	0.666	0.947	5	2.768
YOLOv5l	0.591	0.889	9	3.002
YOLOv8l	0.694	0.953	11	3.377
YOLOv5x	0.617	0.906	8	3.122
YOLOv8x	0.727	0.965	9	3.837

Comparing the YOLO variants, the YOLO-v8x model had the highest mAP 50-95 score at 0.727, followed by the YOLO-v8l model with an mAP 50-95 score of 0.694, as shown in Figure 4. The variant of YOLO-v5 had a lower mAP 50-95 score compared to the YOLO-v8 models, i.e., in the range of 0.425 to 0.617. For the mAP 50 score, the YOLO-v8x model still had the highest accuracy at 0.965, followed by the YOLO-v8l model with an mAP 50 score of 0.953, as shown in Figure 4. The variant of YOLO-v5 had a lower mAP 50 score compared to the YOLO-v8 models, i.e., in the range of 0.741 to 0.906. Therefore, the YOLO-v8x variant showed higher accuracy in the mAP 50-95 and the mAP 50 scores.

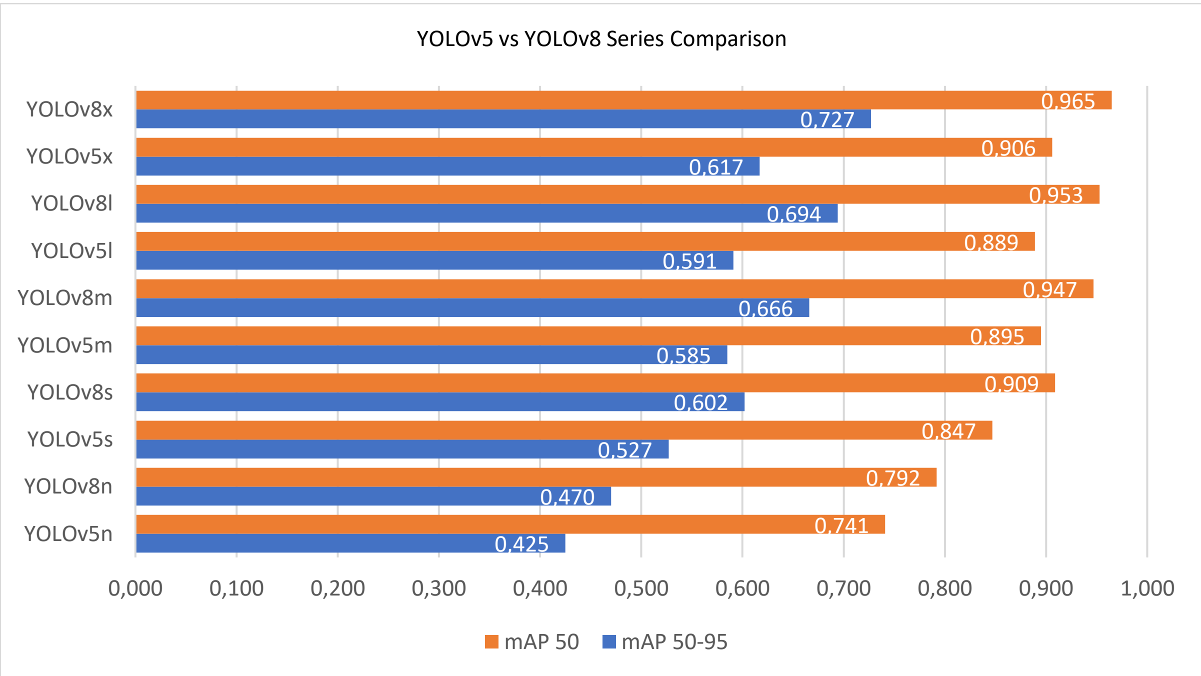

Figure 4 Comparison graph of mAP 50-95 scores using different YOLO algorithms.

DOI: 10.5614/j.eng.technol.sci.2024.56.1.5

The subsequent evaluation of the trained data was based on the number of infected leaf detections. The number of infected leaf detections represents how many detections of infected leaf spots were identified correctly by the models. Figure 5 depicts the model prediction of infected corn leaves using the YOLO algorithm. The number of infected corn leaves detected by each YOLO model is summarized in Table 3. YOLO-v8n had the highest number of detections at 12, followed by YOLO-v5n and YOLO-v8l, with a number of detections of 11 for each. YOLO-v5s, YOLO-v5l, and YOLO-v8x had a number of detections of 9, while YOLO-v8m, YOLO-v5m, YOLO-v8s, and YOLO-v5x had a number of detections of 5, 6, 7, and 8, respectively. Therefore, the YOLO-v8 algorithm had better detection results than the YOLO-v5 algorithm.

The speed evaluation refers to the time required for the YOLO algorithm to detect the infected leaves. According to Table 3, YOLO-v5s had the fastest detection at 2.490 h, followed by YOLO-v5m with a detection time of 2.684 h. YOLO-v8x had the slowest detection time at 3.837 h. Among all variants of the YOLO algorithms, YOLO-v5s was the fastest. In the context of practical applications, these times of YOLO-v5 and YOLO-v8 (in a range from 2.49 to 3.75 h) are feasible and faster than the conventional methods where farmers do plant disease detection manually. In comparison, the conventional method of human-based direct observation for monitoring and detection of plant diseases takes from more than six hours to several days. Therefore, these CNN-based corn leaf detection methods offer a more effective and efficient alternative method for farmers and any other stakeholders.

(a) (b)

Figure 5 YOLOv8x model prediction results from the original infected corn image (a) and the ground truth of the normalized image (b).

YOLO-v5 and YOLO-v8 are two of the most popular deep-learning methods, developed by Ultralytics, where the latest YOLO version, YOLO-v8, offers new features to improve object detection performance. Due to its improved features, YOLO-v8 results in better accuracy than the previous YOLO versions, such as YOLO-v5 according to Ref. [19]. However, YOLO-v5 is fast, easy to use, and accurate in object detection. Hussain *et al.* used YOLO-v1 to YOLO-v8 to predict digital manufacturing and industrial defect detection [13]. They found that the YOLO-v8 algorithm had the highest accuracy, with the highest mAP 50-95 score (about 0.545) compared to its predecessors. Our results in this work agree with previous research where the YOLO-v8 showed better accuracy than the YOLO-v5 algorithm.

Based on the evaluation metrics used in Table 3, specifically the mean average precision (mAP) at different IoU (intersection over union) thresholds (50-95) and at IoU 50, the YOLO-v8x variant outperformed the other models or variants. The model demonstrated consistent improvement in accuracy and loss throughout the training process. The mAP scores, i.e., mAP 50-95 and mAP 50, reached their peak values at 0.727 and 0.965, respectively, showcasing the model's superior performance in detecting infected corn leaves.

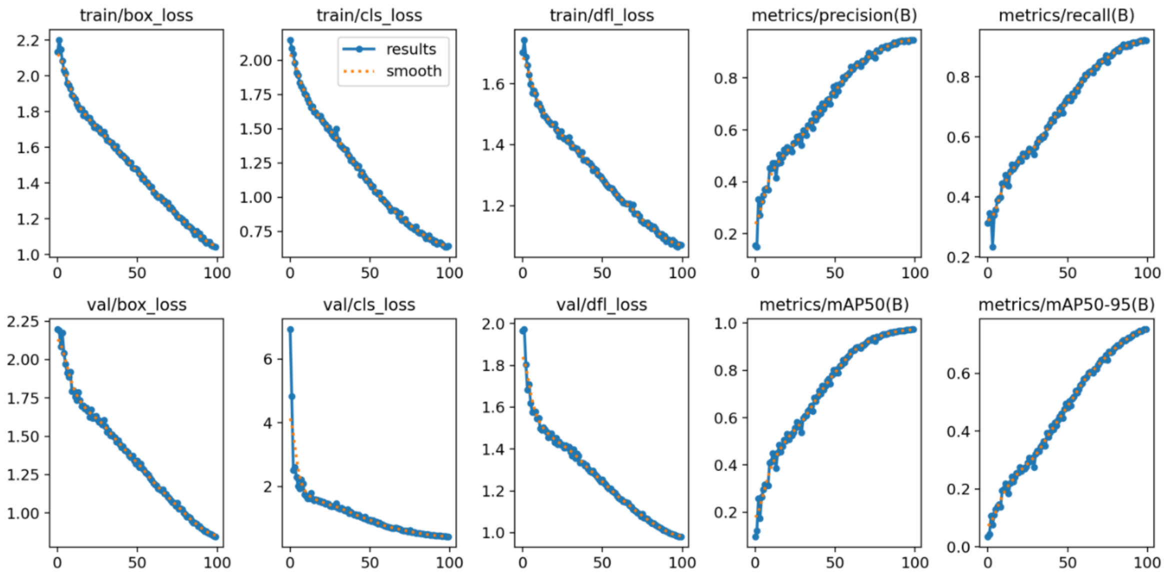

Figure 6 Performance metrics over training epochs.

Figure 6 provides a detailed visualization of the 100-epoch training process for the optimal model, YOLO-v8x, leveraging the computational power of the USK GPU Server equipped with 1 GPU and 8 CPUs. This configuration underscores the efficiency of YOLO-v8x in harnessing parallel processing capabilities for accelerated model training. The training, completed in a commendable time frame of 1.834 hours, further emphasizes the model's ability to efficiently utilize the specified hardware resources. The model achieved impressive final metrics with a precision of 0.94699, recall of 0.92128, mAP50 of 0.97351, and mAP50-95 of 0.75242. Throughout the epochs, the model's loss values consistently decreased, reaching final validation box, class, and DFL losses of 0.84622, 0.42812, and 0.9817, respectively. This trend highlights the robustness and effectiveness of YOLO-v8x in accurately detecting infected corn leaves during the validation process.

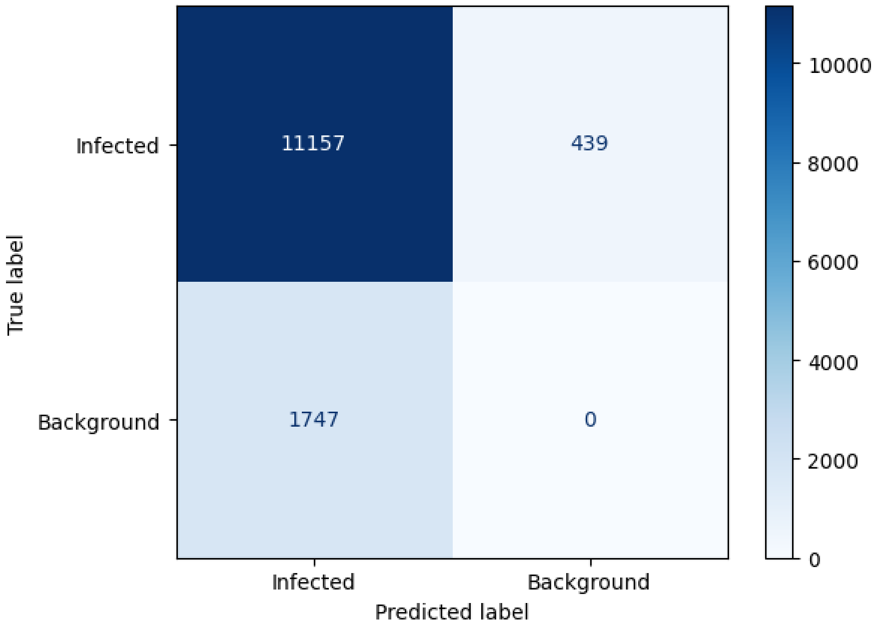

Figure 7 Confusion matrix for infected corn leaf detection.

DOI: 10.5614/j.eng.technol.sci.2024.56.1.5

Furthermore, the confusion matrix for the model, based on the evaluation data set, indicated robust performance with 11,157 true positives, 439 false positives, and 1,747 false negatives. This matrix was visualized as can be seen in Figure 7. The model's ability to correctly identify infected leaves (precision) and capture most infected instances (recall) confirms its effectiveness in practical scenarios.

In summary, the YOLO-v8x model has been proven to be a powerful solution for detecting infected corn leaves, achieving high accuracy, efficiency, and robustness across various evaluation metrics.

Conclusion

In this paper, we used two variants of the YOLO algorithm, YOLO-v5 and YOLO-v8, to identify infected corn leaves caused by plant diseases. The YOLO algorithms, either YOLO-v5 or YOLO-v8, successfully predicted corn leaf infections with high accuracy in terms of mAP scores. The YOLO-v8 algorithm had higher mAP scores compared to the YOLO-v5 algorithm. The YOLO-v8 is better at predicting infected spots than the YOLO-v5 algorithm. However, YOLO-v5 is faster in detecting infected leaves than the YOLO-v8 algorithm. In general, both YOLO algorithms offer relatively fast detection (only about 3 hours of detection time). This prediction by the YOLO algorithm may help farmers to decrease crop failure due to plant disease.

References

[1] Putra, N., *Analysis of the Effect of Harvest Area, Agricultural Credit, Labor, on Food Production (Rice and Corn) in North Sumatra*, Inspirat, **12**(1), pp. 12-22, Jul.2022.

[2] Rozi, F, Santoso, A.B., Mahendri, I.G.A.P., Hutapea, R.T.P., Wamaer, D., Siagian V., Elisabeth, D.A.A., Sugiono, S., Handoko, H., Subagio, H. & Syam, A., *Indonesian Market Demand Patterns for Food Commodity Sources of Carbohydrates in Facing the Global Food Crisis*, Heliyon, **9**(6), e16809, May.2023.

[3] Dhini, V.A., *East Java, the Largest Corn Producer in Indonesia*, KATADATA.co.id, https://www.databoks.katadata.co.id/datapublish/2021/09/28/jawa-timur-produsen-jagung-terbesar-di-indonesia-pada-2020 (28 September 2021). (Text in Indonesian)

[4] Suriani, Patandjengi, B., Junaid M. & Muis, A., *The Presence of Bacterial Stalk Rot Disease on Corn in Indonesia: A Review,* IOP Conference Series: Earth and Environmental Sci., **911**., 012058, 2021.

[5] Wang, D., Cao, W., Zhang, F., Li, Z., Xu, S. & Wu, X., *A Review of Deep Learning in Multiscale Agricultural Sensing*, Remote Sensing **14**(3), pp. 559, Jan.2022.

[6] Saleem, M.H., Potgieter, J. & Arif, K.M., *Automation in Agriculture by Machine and Deep Learning Techniques: A Review of Recent Developments*, Precision Agriculture, **22**(1), pp. 2053–2091, Apr.2021.

[7] Zhu, N.Y., Liu, X., Liu, Z.Q., Hu, K., Wang, Y.K., Tan, J.L., Huang, M., Zhu Q.B., Ji, X. S., Jiang, Y.N. & Guo, Y., *Deep Learning for Smart Agriculture: Concepts, Tools, Applications, and Opportunities*, International Journal of Agricultural and Biological Engineering, **11**(4), pp.32-44, Aug.2018.

[8] Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaria, J., Fadhel, M. A., Al-Amidie, M. & Farhan, L., *Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions*, Journal of Big Data, **8**(1), pp.53, Mar.2021.

[9] Hafifah, Y., Muchtar, K., Ahmadiar, & Esabella, S., *Comparison of Deep Learning Performance in Coffee Bean Damage Detection*, Jurnal Riset Komputer (JURIKOM), **8**(6), pp. 1928-1932, 2022.

[10] Javanmardi, S., Ashtiani, S.H.M., Verbeek, F.J. & Martynenko, A., *Computer-vision Classification of Corn Seed Varieties Using Deep Convolutional Neural Network*, Journal of Stored Products Research, **92**, 101800, 2021.

[11] Boudjit, K. & Ramzan, N., *Human Detection Based on Deep Learning YOLO-v2 for Real-time UAV Applications*, Journal of Experimental & Theoretical Artificial Intelligence, **34** (3), pp. 527-544, 2022.

[12] Dulal, R., Zheng, L., Kabir, M. A., McGrath, S., Medway, J., Swain, D. & Swain, D., *Automatic Cattle Identification using YOLOv5 and Mosaic Augmentation: A Comparative Analysis*, 2022 International Conference on Digital Image Computing: Techniques and Applications (DICTA), pp. 1-8, 2022.

[13] Hussain, M., *YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection*, Machines, **11**(7), 677, Jun.2023.

[14] Jocher, G., Chaurasia, A. & Qiu, J., *YOLO by Ultralytics,* GitHub, https://www.github.com/ultralytics/ultralytics, (1 January 2023).

[15] Redmon, J., Divvala, S., Girshick, R. & Farhadi, A., *You only look once: Unified, Real-Time Object Detection*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779-788, 2016.
[16] Acharya, R., *Corn Leaf Infection Data Set, Version 1*, https://www.kaggle.com/qramkrishna/corn-leaf-infection-data set. (Oktober 2020).
[17] Ashtiani, S.H.M., Javanmardi, S., Jahanbanifard, M., Martynenko, A. & Verbeek, F.J., *Detection of Mulberry Ripeness Stages Using Deep Learning Models*, IEEE Access, **9**, pp. 100380-100394, Feb.2021.
[18] Hong, S.J., Kim, S.Y., Kim, E., Lee, C.H., Lee, J.S., Lee, D.S, Bang. J. & Kim, G., *Moth Detection from Pheromone Trap Images Using Deep Learning Object Detectors*, Agriculture, **10**(5), 170, May.2020.
[19] *YOLO-v8 vs. YOLO-v5: Choosing the Best Object Detection Model*, Augmented.A.I., https://www.augmentedstartups.com/blog/yolov8-vs-yolov5-choosing-the-best-object-detection-model, (20 February 2023).

Manuscript Received: 29 October 2023
Revised Manuscript Received: 17 January 2024
Accepted Manuscript: 1 February 2024