Improving YOLOv8 Deep leaning model in rice disease detection by using Wise - IoU loss function

Cong Dong Trinh[1], Tra My Do Le[2], Thu Ha Do[2], Nhat Minh Bui[2], Thanh Huong Nguyen[2], Quang Uoc Ngo[3], Phuong Thuy Ngo[3], Dang Thanh Bui[1,2*]

[1]*Institute of Automation and Control Technology - ACT, Hanoi University of Science and Technology, Vietnam*
[2]*School of Electrical and Electronics Engineering, Hanoi University of Science and Technology, Vietnam*
[3]*Faculty of Engineering, Vietnam National University of Agriculture, Vietnam*
**Corresponding author Email: thanh.buidang@hust.edu.vn*

Abstract

This paper presents an improved method for a deep learning model applied to the detection of diseases in rice crops. Early detection and prevention of pests and diseases are essential to ensure effective crop productivity. The YOLOv8 deep learning model was employed to detect three common diseases in rice leaves: leaf folder, rice blast, and brown spot. To enhance the model's performance, we replaced the default CIoU loss function in YOLOv8 with WIoU, achieving an overall accuracy of 89.2%, with an improvement of 4.5% on mAP@50 and 4.4% on mAP@50-95. These results demonstrate promising potential for improving the performance and reliability of deep learning models in agricultural applications.

***Keywords**: Rice leaf diseases; Deep learning; YOLOv8; CIoU; WIoU*

1. Introduction

Each year, over 700 million tons of rice are produced, with 90% grown and consumed in Asia [1], including Vietnam. Rice is the primary staple food for over half of the global population. In Vietnam, the total value of rice exports in 2023 was around $4.4 billion, which is exported to over 150 countries worldwide, followed by the higher demand on quality day by day. However, rice crops are frequently threatened by a variety of leaf diseases such as rice folders, brown spot and blast disease which cause severe yield loss. Detection and diagnosis of these diseases is essential for timely control, helping prevent large-scale crop losses. Current practice of disease detection mostly relies on manual inspection by clinicians, which is not only time-intensive and laborious but also susceptible to human errors and inconsistencies.

The advances in computer vision and deep learning bring new frontiers to the field of automatically plant disease detection. CNN has been widely used in agricultural research for processing and classifying complex image data. One of the CNN-based methods is YOLO series where it can make predictions on bounding boxes and class probabilities directly from full images at once. For example, Detect agriculture pests and diseases with low-level features using DCF-YOLOv8 [2]; tomato detection by a lightweight YOLOv8 [3]; Rice Blast detection based on IoT and AI technology [4]. In this paper, YOLOv8, one of state-of-art models in the YOLO series, has been used in crop disease identification whereas YOLO models can achieve better speed with decent object detection sensitivity than most other approaches.

However, there remains room for improvement in handling the accuracy of rice leaf disease detection, especially in cases involving small or overlapping features. This research aims to improve the model result by modifying the original YOLOv8 model with the latest loss function Wise-IoU. [5] Wise-IoU is an advanced version of the Intersection over Union (IoU) metric used in computer vision tasks such as object detection and image segmentation. Wise-IoU loss functions own implementation details that make chance to enhance localization accuracy, convergence speed.

In this paper, we integrate three versions of loss function WIoU into YOLOv8 architecture and evaluate how well the model performs by inferencing its predictions on an image dataset of rice leaf images. The dataset contains three classes of common rice leaf diseases: leaf folder, brown spot and blast disease; collected under varying environmental conditions to improve the robustness and generalization of these models.

2. Proposed method

2.1. Overview

YOLOv8 is used to detect 03 diseases that often appear in rice leaf: folder disease, blast disease and brown spot with the dataset collected from VNUA and internet sources. Otherwise, to enhance the accuracy, we experience up-to-date loss functions WIoU, including three versions of it and compare to the default loss function CIoU in YOLOv8.

2.2. Dataset and data preparation

2.2.1. *Data collection*

The dataset consists of 3,831 images including 1,831 high-quality images collected from paddy fields at the Vietnam National University of Agriculture (VNU) and 2000 images gathered from the internet. The dataset features images of rice leaves affected by three types of diseases: rice leaf folder, leaf blast, and brown spot.

The dataset has been validated by expert Mrs. Thu Hong

Received: 22 November 2024; Accepted: 20 January 2025.

from the Vietnam National University of Agriculture. The images from VNU were collected under summer hot weather conditions with an average daily temperature of over 33°C and an average rainfall of 124mm during a period of 02 months.

2.2.2. *The final dataset*

The dataset consists of 3,831 images, labeled as follows: 0 = rice leaf folder; 1 = leaf blast; 2 = brown spot. Based on the size of the dataset and to optimize model performance, the dataset is divided into three subsets: training, validation, and test, with a ratio approximately of 80-10-10. The training set consists of 2,990 images, the validation set contains 422 images, and the test set also contains 419 images. Additionally, the images in the validation set are selected to ensure they do not overlap with the images in the training set, and the images in the test set are chosen to have no relationship with the images in both the training and validation sets.

Figure 1: Blast disease

2.3. YOLOv8

YOLOv8 is an enhanced version released in 2023 by Ultralytics, the company behind YOLOv5. It provides a unified framework for training models in object detection, segmentation, and image classification. YOLOv8 includes five architectural versions: n/s/m/l/x. The YOLOv8n model has the fewest parameters, making it the lightest and fastest but least accurate, whereas the YOLOv8x model has the most parameters, offering the highest accuracy but being the heaviest and slowest. YOLOv8 still comprises the backbone for feature extraction from input images, the neck for aggregating features from the backbone, and the head for making predictions.

2.3.1. *Backbone*

YOLOv5 [6] employs standard convolutional layers and uses the CSPDarknet53 module as its backbone. In contrast, OLOv8 incorporates the C2f module (a Cross-Stage Partial Bottleneck with dual convolutional layers) instead of the C3 module (CSPDarknet53) used in YOLOv5. The C2f module is an advancement that builds on the ELAN concept from YOLOv7, integrating C3 with ELAN to create a more comprehensive C2f module. This enables YOLOv8 to effectively capture gradient flow information while preserving a lightweight structure. The backbone's primary components are the C2f and Conv blocks. The C2f block consists of a split block, bottleneck block, Conv block, and a concatenation layer. The structure of C2f is shown in Figure 2, with ConvBS is a block composed of a Convolutional 2d, a BatchNorm and a SiLU layer. The number of bottleneck layers *n* varies according to the C2f block's position within the backbone and the architecture's depth. Specifically, the first C2f block has $n = 3 \times d$ bottleneck layers, while the second and third blocks have $n = 6 \times d$ b layers, and the final C2f block has $n = 3 \times d$ layers. Notably, the feature maps from the second and third C2f blocks are combined with the head.

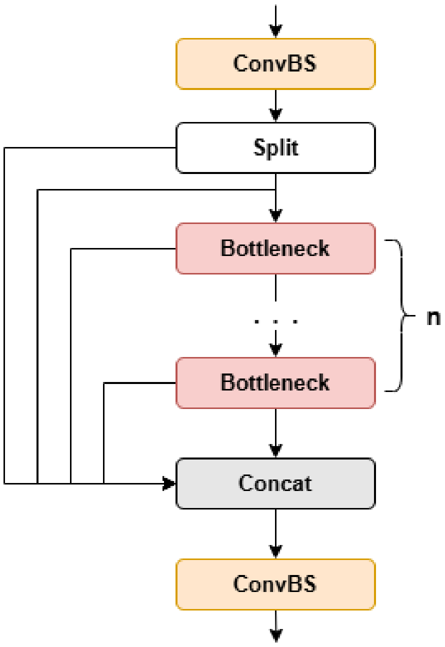

Figure 2: Architecture of the C2f module

2.3.2. *Neck*

YOLOv8 utilizes SPPF (Spatial Pyramid Pooling Fast) in the neck to combine features that enhance the ability to process spatial information at multiple levels. This process integrates features from the backbone and transfers them to the head. SPPF is an improvement over the SPP (Spatial Pyramid Pooling) technique, designed to boost performance and reduce computation time while retaining the benefits of SPP.

One major distinction between SPP and SPPF lies in the size of the max-pooling layers. While SPP utilizes varying kernel sizes, as previously mentioned, SPPF applies a uniform kernel size across all layers. Another key difference is the pooling approach - SPP employs three parallel max-pooling layers, whereas SPPF arranges them sequentially. This sequential execution reduces computational complexity and enhances performance, making SPPF considerably faster than SPP.

2.3.3. *Head*

The YOLOv8 head structure includes key components like the Conv Block, C2f Block, Upsample, and Concat. Notably, the C2f block in the head lacks shortcuts for the bottleneck block. One major improvement in YOLOv8 over previous versions is the transition from Anchor-Based to Anchor-Free methods.

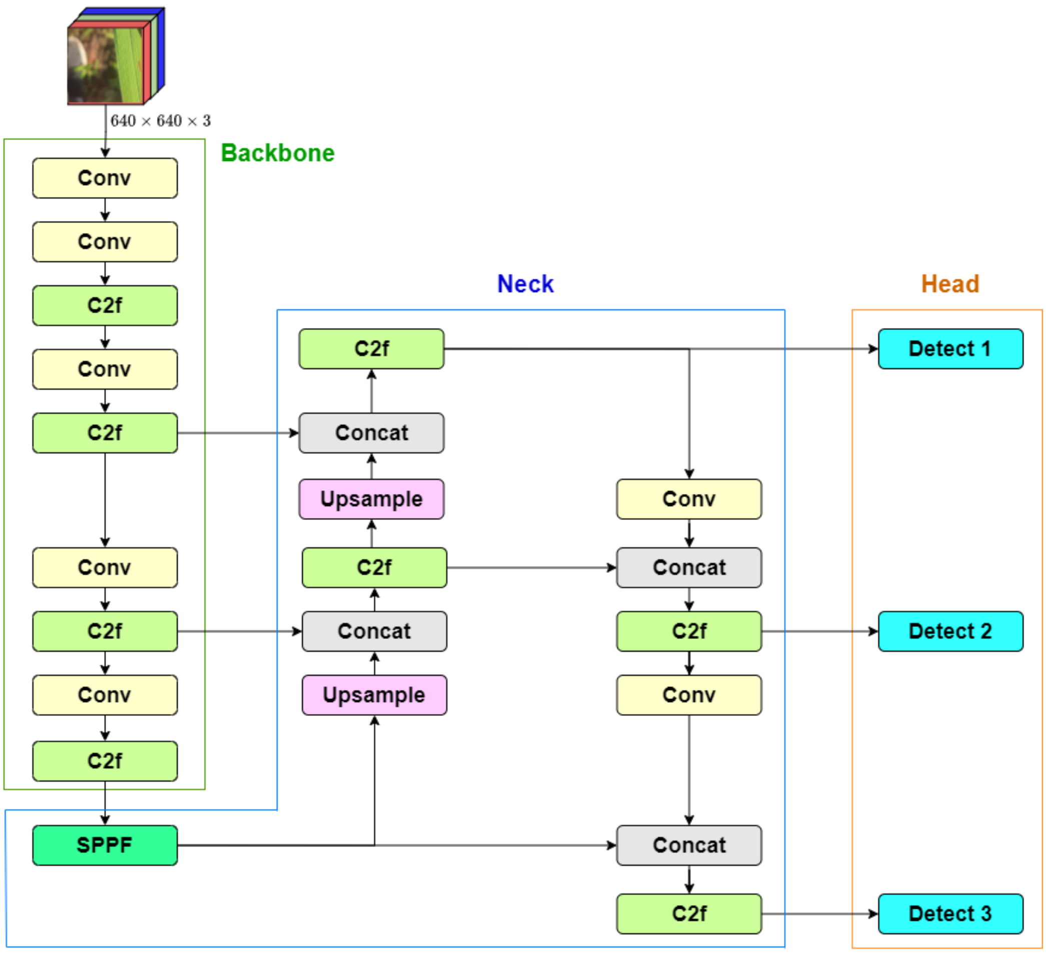

Figure 3: Architecture of YOLOv8

Another notable change is the introduction of the Decoupled Head. In earlier object detection models like Faster R-CNN and previous YOLO versions, both localization and classification were processed within the same branch of the head. However, this approach presents challenges since classification relies on discriminative features, whereas localization requires features that capture boundary details. This discrepancy, known as task conflict, can hinder performance.

To address this, YOLOv8 separates these tasks into two distinct branches, effectively resolving the issue of feature mismatches. Additionally, the authors implemented Task Alignment Loss (TAL) from the TOOD paper, a labeling strategy that enhances anchor alignment. The TAL metric used in YOLOv8 is defined by the following equation:

$$t = s^{\alpha} \times u^{\beta} \tag{1}$$

where s and u represent the classification score and IoU score, respectively. The parameters α and β control the balance between these two tasks in the anchor alignment index, while t denotes the alignment index. Based on the value of t, the loss function differentiates between positive and negative samples for training.

2.3.4. *Intersection over Union*

In object detection task, Intersection over Union (IoU) is used to evaluate accuracy by measuring the degree of overlap between the predict box (anchor box) and the target box (ground truth box).

$$IoU = \frac{|B \cap B^{gt}|}{|B| + |B^{gt}| - |B \cap B^{gt}|} \tag{2}$$

$$\mathcal{L}_{IoU} = 1 - IoU \tag{3}$$

$B = (x, y, w, h)$ and $B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ are the predict box and target box. *Eq. 2* is the IoU loss function. However, IoU has the problem that when there is no overlap between anchor box and predict box, the gradient disappears. Therefore, other parameters cannot be updated.

Therefore, penalty items are added to solve this situation. Thus, the existing IoU-based loss function family follow below general form:

$$\mathcal{L}_{IoU} = 1 - IoU + \mathcal{R}_i(B, B^{gt}) \tag{4}$$

Where $\mathcal{R}_i(B, B^{gt})$ is the penalty item.

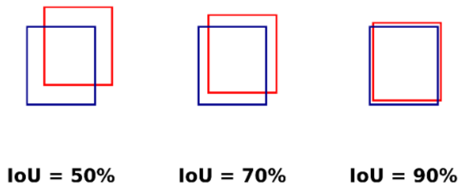

Figure 4: Visualize different IoU scores

2.4. Loss function

As the problem mentioned at the end of section *2.3.4* above, loss functions appear to improve the precision of target box localization. The loss functions guide the learning process by adjusting the model's weights to minimize the errors it makes. This research experiments two different loss functions: CIoU and novel loss functions WIoU.

2.4.1. *Complete-IoU*

Complete-IoU (CIoU) is the loss function YOLOv8 currently ultilizes. Compared to the original IoU loss function, CIoU was added distance metrics and aspect ratio, which helps overcome the shortcoming of DIoU and GIoU. Therefore, it solves the problem of gradient vanishing in **Error! Reference source not found.** as well as improves the speed. The equation of CIoU is illustrated as below:

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b,b^{gt})}{c^2} + \alpha v \tag{5}$$

- b and b^{gt} represent the central points of B and B^{gt}, $\rho(.)$ is the Euclidean distance, while c denotes the diagonal length of the smallest rectangular covering anchor box and predict box.
- α is positive trade-off parameter, and v describes aspect ratio consistency defined as below:

$$\alpha = \frac{v}{(1-IoU)+v} \tag{6}$$

$$v = \frac{4}{\pi^2}\left(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h}\right)^2 \tag{7}$$

Obviously, from *Eq. 6* it witnesses that when the predicted box matches real box, the aspect ratio is minimized (can even equal 0) and has no effect.

2.4.2. *Wise-IoU*

In fact, the dataset, especially dataset for plant disease inevitably contains many low-quality examples. Using CIoU loss function for bounding box regression can negatively affect the model's detection performance. Therefore, to address the challenges, Wise-IoU (WIoU) is introduced as a replacement for CIoU. Wise-IoU (WIoU) is a dynamic non-monotonic focusing mechanism (FM) loss function belonging to IoU-based loss family. WIoU was designed to eliminate the penalty of geometric factors when the predict box overlaps target box, then make model generalization better (wiser!). WIoU has three versions.

Wise-IoU v1:

Wise-IoU v1 incorporates two layers of attention $\mathcal{L}_{IoU}$ and $\mathcal{R}_{WIoU}$. $\mathcal{R}_{WIoU}$ is distance attention and effective to removes factor that obstructs convergence. Consequently, the novel bounding box loss $\mathcal{L}_{WIoUv1}$ helps minimize the geometric metrics' punishment as well as improve the generalization performance.

$$\mathcal{L}_{WIoUv1} = \mathcal{R}_{WIoU}\mathcal{L}_{IoU} \tag{8}$$

$$\mathcal{L}_{IoU} = 1 - IoU = 1 - \left(\frac{W_g H_g}{S}\right) \tag{9}$$

$$\mathcal{R}_{WIoUv1} = \exp\left(\frac{(x-x_{gt})^2+(y-y_{gt})^2}{W_g^2+H_g^2}\right) \tag{10}$$

where W_g *and* H_g are the size of smallest rectangular covering anchor box and target box. (x,y), (x_{gt},y_{gt}) respectively present the center coordinator of predicted bounding box and ground-truth bounding box.

Wise-IoU v2:

Monotonic FM: In monotonic FM, the gradient gain is monotonically related to the IoU value. This ensures that the model focuses on more high-quality anchor boxes and gives less emphasis to low quality ones.

Wise-IoU v2 constructs a monotone focusing coefficient in order to change the back-propagating. As a result, the model can concentrate on challenging examples, enhancing classification performance.

$$\mathcal{L}_{WIoUv2} = \left(\frac{\mathcal{L}_{IoU}^*}{\overline{\mathcal{L}_{IoU}}}\right)^{\gamma}\mathcal{L}_{WIoUv1}, \gamma > 0 \tag{11}$$

Where $\overline{\mathcal{L}_{IoU}}$ is the exponential running average with momentum, $\mathcal{L}_{IoU}^*$ is the gradient gain. We choose $\gamma = 0.5$ similar to original paper [5].

However, WIoU v2 with monotonic FM still shows limit and does not fully exploit the information from low-quality boxes.

Wise-IoU v3:

To characterize the quality of predict box, WIoU v3, which is still improve from WIoU v1 and v2, adding a non-monotonic focusing factor d. The math formula of Wise-IoU v3 and its elements are represented as in *Eq. 12, 13, 14*.

Dynamic Non-Monotonic Focusing Mechanism: This mechanism improves upon monotonic FM by dynamically adjusting the focus on each anchor box, considering not just the IoU but also its relative position in the distribution of anchor box qualities.

$$\mathcal{L}_{WIoUv3} = d\mathcal{L}_{WIoUv1} \tag{12}$$

$$d = \frac{\beta}{\delta - \alpha^{\beta-\delta}} \tag{13}$$

Where α and δ are hyper-parameters. The outlier degree β is represented as:

$$\beta = \frac{\mathcal{L}_{IoU}^*}{\overline{\mathcal{L}_{IoU}}} \in [0, +\infty) \tag{14}$$

The magnitude of outlier degree is positive and correlated with the quality of predict box. The dynamic $\overline{\mathcal{L}_{IoU}}$ makes quality classification criteria for predict box are also dynamic, allow distribution of gradient gain to reach the best fit for current situation at any given time. Therefore, WIoU v3 shows the ability to improve the localization performance.

3. Result and Discussion

3.1. Evaluation metrics

Mean Average Precision (mAP) is a popular evaluation metric used in object detection tasks to measure the accuracy of model. They help determine how well a model can detect and classify objects in an image. For result of this paper, we concern about two specific terms mAP@50 and mAP@50-95.

mAP@50 refers to Mean Average Precision at IoU threshold of 50%. Specifically, a prediction is deemed correct if the IoU between the predicted bounding box and the ground truth box is at least 0.50. The math calculation of mAP@50 is represented as below:

$$mAP@50 = \frac{1}{C}\sum_{c=1}^{C} AP_{50}(c) \tag{15}$$

where C is the total number of classes, here in this paper C=3 equals to the number of diseases. $AP_{50}(c)$ is the average precision for class c at a threshold of 0.50.

mAP@50-95 refers to the Mean Average Precision calculated over multiple IoU threshold, ranging from 0.50 to 0.95 with step size 0.05. It means unlike mAP@50 (which just evaluate model at a single IoU threshold), mAP@50-95 performs the average over 10 different thresholds.

3.2. Result of the method

In this research, 419 images in test set have 460 diseased leaves, which include 173 folder disease leaves, 141 blast leaves and 146 brown spot leaves. YOLOv8s is used, other hyper-parameters are followed orginial article [5]. We conduct experiment training YOLOv8s with orginial loss function CIoU, WIoU v1, WIoU v2, WIoU v3 respectively. Results received after training 100 epochs with batch size =16 are shown on the table below:

Table 1: Performance of each loss function

Loss function	*mAP@50*	*mAP@50-95*
CIoU	0.847	0.545
WIoU v1	0.841	0.54
WIoU v2 ($\gamma = 0.5$)	0.879 (+3.2)	0.571 (+2,6)
WIoU v3 ($\alpha = 1.9, \delta = 3$)	**0.892 (+4.5)**	**0.589 (+4.4)**

It witnesses the result achieved by applying WIoU v3 showing the best results. Summary of combined WIoUv3 model is shown as below:

Table 2: Model summary

Disease	*mAP@50*	*mAP@50-95*
Folder	0.913	0.572
Blast	0.856	0.587
Brown spot	0.907	0.608
All	0.892	0.589

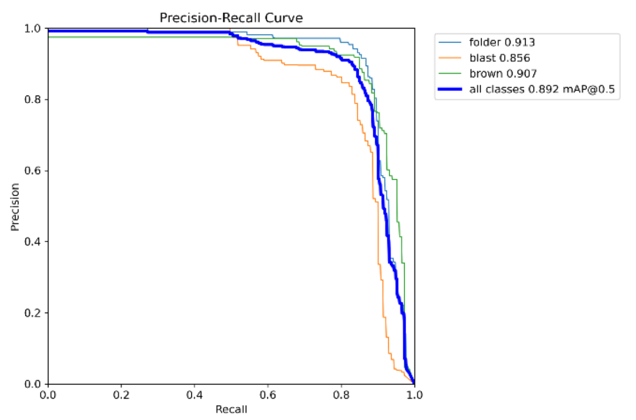

Figure 5: PR Curve of YOLOv8 with WIoU v3

Figure 5 presents the Precision-Recall (PR) curve of the trained YOLOv8 model with WIoU v3, reflecting its performance in identifying diseased regions across different confidence thresholds. We visualized predictions in Figure 6 to demonstrate the model's detection results for Rice Leaf Folder, Rice Blast and Brown Spot, respectively. Each image highlights the bounding boxes around detected diseased areas, illustrating the model's capability to accurately locate and classify the different rice diseases.

Figure 6: Visualize predictions

4. Conclusion

In this research, we implement the application of deep learning model into agriculture applications. State-of-art model in YOLO series – YOLOv8 is used with some modification in loss function. Experiment with the Wise-IoU loss function into YOLOv8 achieved 91.3% in folder disease, 85.6% in blast disease, 90.7% in brown spot and 89.2% mAP@50 overall. Meanwhile, compare to previous research, YOLOv5 in article [9] shows the results as 81.6% for blast disease and 78.5% mAP@50 for all. Therefore, this article leads to quite a prospective method about enhancing performance of the model. In the future, we want to enrich our dataset for a larger quantity of images and more diseases. Moreover, combining with the hardware system helps bring better applications into real life.

Acknowledgments

The authors sincerely appreciate the support from the Industrial Instrumentation & IoT Laboratory at the School of Electrical and Electronic Engineering, Hanoi University of Science and Technology, as well as the valuable assistance from experts at the Vietnam National University of Agriculture. This research is funded by Hanoi University of Science and Technology (HUST) under grant number T2023-PC-069.

References

[1]. M. K. Papademetriou, *Rice production in the Asia-Pacific region: Issues and Perspectives*, fao.org, [Online]. Available: https://www.fao.org/rice-production-in-asia

[2]. L. Zhang, G. Ding, C. Li, and D. Li, *DCF-Yolov8: An Improved Algorithm for Aggregating Low-Level Features to Detect Agricultural Pests and Diseases*, Agronomy, vol. 13, no. 8, article 2012, 2023. https://doi.org/10.3390/agronomy13082012.

[3]. G. Yang, J. Wang, Z. Nie, H. Yang, and S. Yu, *A Lightweight YOLOv8 Tomato Detection Algorithm Combining Feature Enhancement and Attention*, Agronomy, vol. 13, no. 7, article 1824, 2023. https://doi.org/10.3390/agronomy13071824.

[4]. W. L. Chen, Y. B. Lin, F. L. Ng, C. Y. Liu, and Y. W. Lin, *RiceTalk: Rice Blast Detection Using Internet of Things and Artificial Intelligence Technologies*, IEEE Internet of Things Journal, vol. 7, no. 2, pp. 1001–1010, Feb. 2020. https://doi.org/10.1109/JIOT.2019.2947624

[5]. Zanjia Tong, Yuhang Chen, Zewei Xu, Rong Yu, *Wise-IoU: bounding box regression loss with dynamic focusing mechanism*, arXiv preprint arXiv:2301.10051, 2023.

[6]. J. R. Terven and D. M. Cordova-Esparza, *A Comprehensive Review of YOLO: From YOLOv1 and Beyond,* Mach. Learn. Knowl. Extr., vol. 5, no. 4, pp. 1680–1716, 2023. https://doi.org/10.48550/arXiv.2304.00501.

[7]. K. Su, L. Cao, B. Zhao, N. Li, D. Wu, and X. Han, *N-IoU: better IoU-based bounding box regression loss for object detection,* Neural Computing & Applications, vol. 36, pp. 3049–3063, 2024. https://doi.org/10.1007/s00521-023-09133-4.

[8]. Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., & Ren, D. (2020, April). Distance-IoU loss: Faster and better learning for bounding box regression. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 07, pp. 12993-13000). https://doi.org/10.1609/aaai.v34i07.6999

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

[10]. T. C. Dong, M. T. Anh, G. D. Khanh, N. T. Huong, N. T. Cac, and B. D. Thanh, *Using deep learning for rice leaf diseases detection using YOLOv5*, P.ISSN 1859-4190, vol. 2, pp. 19–23, 2023.

[11]. B. D. Thanh, M. T. Anh, G. D. Khanh, T. C. Dong, and N. T. Huong, *SGDR-YOLOv8: Training Method for Rice Diseases Detection Using YOLOv8*, Advances in Computing and Data Sciences, ICACDS 2024, Communications in Computer and Information Science, vol. 2194, Springer, Cham, 2025. https://doi.org/10.1007/978-3-031-70906-7_15

[12]. Dong Cong Trinh, Anh Tuan Mac, Khanh Giap Dang, Huong Thanh Nguyen, Hoc Thai Nguyen, and Thanh Dang Bui, *Alpha-EIOU-YOLOv8: An Improved Algorithm for Rice Leaf Disease Detection*, AgriEngineering 6, no. 1: 302-317, 2024. https://doi.org/10.3390/agriengineering6010018

[13]. T. C. Dong, M. T. Anh, G. V. Khanh, N. T. Huong, and B. D. Thanh, *Rice Leaf Diseases Detection Using YOLOv8*, JST: Engineering and Technology for Sustainable Development, vol. 34, no. 2, pp. 040–047, April 2024. https://doi.org/10.51316/jst.173.etsd.2024.34.2.6

[14]. C. D. Trinh, D. T. Bui, and T. N. Huong, *Automatic detection and classification of diseases in rice plants using deep learning algorithm base on residual network architecture*, 7th International Conference and Exhibition on Control and Automation *(VCCA-2024)*, 2024.