Vol. 8, No. 2, December 2024, page. 111-118 ISSN 2598-3245 (Print), ISSN 2598-3288 (Online) DOI: http://doi.org/10.31961/eltikom.v8i2.1165 Available online at http://eltikom.poliban.ac.id

APPLICATION OF FASTER R-CNN DEEP LEARNING METHOD FOR RICE PLANT DISEASE DETECTION

Halim Pujiono¹, Anik Vega Vitianingsih^{1*}, Slamet Kacung¹, Anastasia Lidya Maukar², Seftin Fitri Ana Wati³

¹⁾ Department of Informatics, Universitas Dr. Soetomo, Surabaya, Indonesia
 ²⁾ Department of Industrial Engineering, Universitas Presiden, Bekasi, Indonesia
 ³⁾ Department of Information System, UPN Veteran Jawa Timur, Surabaya, Indonesia
 e-mail: hpujiono1@gmail.com, vega@unitomo.ac.id, slamet@unitomo.ac.id, almaukar@president.ac.id, seftin.fitri.si@upnjatim.ac.id

Received: 31 May 2024 - Revised: 15 September 2024 - Accepted: 23 September 2024

ABSTRACT

Plant diseases, particularly in staple crops like rice, significantly affect the stability of rice production in Indonesia. Crop failure caused by rice plant diseases present a critical challenge for farmers. Early diagnosis is crucial for preventing and managing rice diseases, as it facilitates more effective preventive measures, reduces yield losses, and boosts overall agricultural production. This study aims to apply the Faster Region Convolutional Neural Network (Faster R-CNN), a deep learning approach, to detect rice plant diseases. The Grid Search method was employed as a hyperparameter tuning technique to identify the optimal parameter combination for enhancing algorithm performance. Experimental results demonstrate the model's performance, achieving an accuracy rate of 88%, recall and precision of 100%, and an F1 Score of 93%. These findings indicate that the Faster R-CNN method effectively recognizes and classifies rice plant diseases with a high degree of accuracy.

Keywords: deep learning, faster R-CNN, grid search, hyperparameter tuning, rice plant disease detection.

I. INTRODUCTION

Rec plays a pivotal role as a staple crop in Indonesia, serving as the primary source of carbohydrates and fulfilling the dietary needs of the population[1]. According to the Statistics Indonesia (*Badan Pusat Statistik*/BPS), rice production in 2022 increased by 0.59%, reaching 31.54 million tons, an improvement of 184.50 thousand tons compared to the 31.36 million tons produced in 2021 [2]. However, the rice farming sector in Indonesia continues to face challenges, particularly crop failures caused by rice plant diseases. Common diseases affecting rice plants include blast disease caused by fungi, bacterial leaf blight caused by bacteria, and tungro caused by viruses, among others [3]–[5]. Symptoms of diseased rice plants can be observed through changes in leaf color, the appearance of spots on leaves, leaf drying, and other visible indicators [6]. Each disease requires prompt treatment; delays or errors in managing rice plant diseases can reduce both the quality and quantity of rice harvests [7].

A lack of knowledge about rice plant diseases greatly affects decision-making in agricultural management [8]. Advances in artificial intelligence (AI), particularly deep learning—a subset of machine learning that employs neural networks with multiple layers for complex data processing—offer promising solutions. Deep learning has been applied successfully in various fields, including disease diagnosis [9]. By analyzing patterns or unique characteristics in images of diseased rice plants, it is possible to identify these diseases [10]. Consequently, the application of Artificial Intelligence for diagnosing and classifying rice plant diseases is becoming increasingly significant, aiming to simplify and improve the diagnostic process for such agricultural challenges.

Prior studies have investigated the application of deep learning and machine learning for identifying diseases in rice plants. Several studies have explored the use of deep learning, including the Convolutional Neural Network (CNN) technique. These studies employed specific parameter settings, such as a



batch size of 32, 100 epochs, the Adam optimizer, and a learning rate of 0.0001, resulting in a test accuracy of 62.21% [11]. Another study utilized the Faster R-CNN method with parameters including a batch size of 60, the SGD optimizer, a learning rate of 0.01, and a weight decay of 0.0001, achieving a test accuracy rate of 85.4% [12]. Similarly, machine learning approaches, such as the learning vector quantization method, with parameters of alpha 0.02, error tolerance 0.001, alpha reduction 0.5, and 5 epochs, produced an accuracy rate of 75% and an error rate of 25% [13]. Based on these findings, it can be concluded that research on rice plant disease detection using the Faster R-CNN method combined with hyperparameter tuning via Grid Search is still limited.

This research aims to advance rice plant disease detection by applying deep learning, specifically the Faster R-CNN method, to develop a disease detection application. The method is chosen based on prior research utilizing the Pascal 2007 dataset, which applied hyperparameters such as 13 epochs, a batch size of 64, momentum of 0.9, and a decay rate of 0.005. The results achieved a Mean Average Precision (MAP) of 70.4, outperforming R-CNN, Fast R-CNN, and YOLO methods [14]. In this study, the Faster R-CNN method is combined with the Grid Search technique for hyperparameter tuning to enhance the accuracy and precision in identifying specific rice plant diseases.

This study contributes to the field of computer vision, particularly in detection and classification, by improving algorithm model performance. This is reflected in metrics such as accuracy, precision, recall, and F1 score. Furthermore, this research is expected to significantly aid the development of rice plant disease detection, providing practical benefits to farmers by enabling the early identification of diseases in rice plants.

II. RESEARCH METHOD

This section discusses key aspects of the research, including the rice plant disease dataset, the data preprocessing workflow, an explanation of the Faster R-CNN method for building rice plant disease detection models, the use of Grid Search for hyperparameter tuning, and model evaluation through a confusion matrix. The research process is depicted in the flowchart in Figure 1. The research flow in



Figure 2. Sample image of blast disease



Figure 3. Sample image of blight disease



Figure 4. Sample image of tungro disease

Figure 1 comprises several stages: data input, data preprocessing, data splitting, hyperparameter tuning, Faster R-CNN model construction, and model evaluation. Each stage is detailed as follows:

A. Dataset

This study utilizes a public dataset containing 80 images per class of rice plant diseases [15]. Additionally, data were collected from agricultural areas in Trowulan District, Mojokerto Regency, East Java, providing 50 images of diseased rice plants for each category. Each class includes 130 images representing three disease categories: Blast (Figure 2), Blight (Figure 3), and Tungro (Figure 4). In total, the dataset comprises 390 images of rice plant diseases.

B. Preprocessing

The preprocessing stage customizes the data to meet the research requirements [16]. During this phase, the image extensions are adjusted, and annotations are applied to each image. The annotation process identifies the disease's location and labels each bounding box. The bounding box data are organized in a table with eight columns: file name, width, height, class, x-min, y-min, x-max, and y-max. The "class" column specifies the bounding box category, while the x-min, y-min, x-max, and y-max columns denote the bounding box's precise position within the image. Figure 5 illustrates the distribution of image annotations. The dataset is divided into training and testing sets, with 80% allocated for training and 20% for testing. Figure 6 provides a visual representation of the data preprocessing workflow.

C. Faster R-CNN

Faster R-CNN is an image object detection algorithm developed by Microsoft Research, with the Region Proposal Network (RPN) as its core component. The RPN generates proposals or identifies regions likely to contain objects [17]. In this research, Faster R-CNN is applied by first extracting features



Figure 6. Data Preprocessing Workflow

from rice plant images through a convolutional layer to create a feature map. This feature map is then passed to the RPN stage, which produces proposals indicating the locations in the image suspected of exhibiting characteristics of rice plant diseases. These proposals proceed to the ROI pooling stage for adjustment, ensuring compatibility with the classifier stage for rice plant disease detection.

In this method, a loss function is utilized to calculate the RPN loss and the classifier loss, helping assess the model's optimal performance during training [17]. The loss function is defined in (1)

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}} \sum_{i} L_{cls}(p_i,p_i^*) + \lambda \frac{1}{N_{reg}} \sum_{i} p_i^* L_{cls}(t_i,t_i^*)$$
(1)

In (1), N_{cls} represents the number of elements in the mini-batch used for the classification loss calculation, while N_{reg} denotes the number of elements contributing to the regression loss. The variable p_i corresponds to the predicted probability, and p_i^* is the ground truth classification label. Meanwhile, t_i represents the model's regression prediction, and t_i^* refers to the ground truth regression values. The parameter λ acts as a weight that regulates the contribution of the regression loss to the total loss. This formulation ensures a balance between classification and regression tasks, enhancing the model's overall performance.

D. Grid Search

A hyperparameter tuning method that tests all combinations of candidate parameters to achieve the highest classification accuracy [18]. In this study, the loss function was used as the primary criterion for determining the best model after tuning.

Table 1 summarizes the parameters used in the grid search process for hyperparameter tuning. These parameters include two optimizers (Adam and SGD), three momentum values (0.3, 0.6, and 0.9), and three weight decay values (0.01, 0.001, and 0.0001). Each parameter combination was tested to identify the configuration resulting in the lowest total loss, corresponding to the model's optimal performance.

E. Model Evaluation

This study employs a confusion matrix to evaluate the model's performance. The confusion matrix is a tool for assessing a model's effectiveness in predicting the categories or classifications of data. It does so by counting the number of correct and incorrect predictions made on test data [19]. Metrics derived from the confusion matrix include precision, recall, accuracy, and the F1 score. The following sections explain these metrics and their interpretations.

Accuracy measures the model's ability to correctly classify both predicted and true classes [20]. The formula for accuracy is presented in (2), where TP represents true positives (correct predictions for the positive class), TN represents true negatives (correct predictions for the negative class), FP represents false positives (incorrect predictions for the positive class), and FN represents false negatives (incorrect predictions for the negative class). Precision indicates the model's ability to correctly classify instances of the predicted positive class [20]. The precision calculation uses (3). It is calculated using (3), where TP denotes true positives and FP denotes false positives. Recall measures the model's ability to correctly classify instances of the actual positive class [20]. It is calculated using (4), where TP refers to true positives and FN refers to false negatives. The F1 Score evaluates the balance between precision and recall. A higher F1 Score signifies better performance in predicting a specific class [20]. The F1 Score is calculated using (5), where recall is obtained from (4) and precision from (3).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$
(2)

$$Precision = \frac{TP}{TP + FP}$$
(3)

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$F1 Score = 2 * \frac{(Recall * Precision)}{(Recall + Precision)}$$
(5)

III. RESULT AND DISCUSSION

A. Training and Tuning

After completing the data preprocessing stage, the training process was conducted using various hyperparameter combinations obtained through the grid search procedure. Each tuning configuration included optimizer, momentum, and weight decay parameters. The best model was determined based on the combination of Region Proposal Network (RPN) loss and classifier loss values that yielded the smallest total loss. It is worth noting that the Adam optimizer does not include a momentum parameter, so tuning for this optimizer was performed only on the weight decay parameter.

Table 2 presents the training results for each hyperparameter combination, providing an overview of the impact of different configurations on the total loss. The Adam optimizer with weight decay of 0.0001 achieved the smallest total loss of 1.368, ranking first. Conversely, the Stochastic Gradient Descent (SGD) configuration with a momentum of 0.9 and weight decay of 0.01 produced the largest total loss

| TABLE 2 Hyperparameter Tuning Results | | | | |
|--|----------|--------------|------------|------|
| Optimizer | Momentum | Weight Decay | Total Loss | Rank |
| ADAM | - | 0.0001 | 1.368 | 1 |
| SGD | 0.9 | 0.0001 | 1.374 | 2 |
| SGD | 0.3 | 0.0001 | 1.595 | 3 |
| ADAM | - | 0.001 | 1.671 | 4 |
| SGD | 0.6 | 0.0001 | 1.685 | 5 |
| SGD | 0.6 | 0.001 | 2.27 | 6 |
| SGD | 0.3 | 0.001 | 2.567 | 7 |
| ADAM | - | 0.01 | 2.663 | 8 |
| SGD | 0.3 | 0.01 | 2.746 | 9 |
| SGD | 0.6 | 0.01 | 2.892 | 10 |
| SGD | 0.9 | 0.001 | 3.51 | 11 |
| SGD | 0.9 | 0.01 | 4.082 | 12 |



of 4.082, ranking last. Based on these results, the detection model was configured with the following parameters: a learning rate of 0.0001, Adam optimizer, weight decay of 0.0001, 100 epochs, and an epoch length of 1000.

The results of the loss function for the Adam optimizer with a weight decay of 0.0001 are detailed in Figure 7, showing the loss function's progression over 100 epochs. demonstrates the decrease in loss values over time, reflecting the training process's effectiveness.

B. Model Evaluation

At this stage, the model, after undergoing tuning and training, was evaluated to assess its performance in classifying disease types in images that were not part of the training dataset. The evaluation employed a confusion matrix. As shown in Figure 8, the confusion matrix indicates that the number of correctly predicted instances for the blast class is 17, while 9 instances were misclassified. For the leaf blight and tungro classes, the predictions matched the actual values perfectly, with 26 instances correctly classified for each class.

Using the confusion matrix depicted in Figure 8, metrics such as accuracy, precision, recall, and F1 score were calculated. For the blast category, the precision, recall, and F1 score were 1.00, 0.65, and 0.79, respectively. In comparison, the blight category achieved precision, recall, and F1 score values of 0.84, 1.00, and 0.91. Similarly, the tungro category recorded precision, recall, and F1 score values of 0.87, 1.00, and 0.93. Table 3 provides a detailed breakdown of these metrics.

C. Discussion

The Grid Search method was effective in optimizing the model for disease classification, with the Adam optimizer and a weight decay of 0.0001 producing the best results, as evidenced by the lowest total loss of 1.368.

The evaluation using the confusion matrix highlights the model's strengths, achieving an overall accuracy of 0.88 and balanced performance across different classes. Specifically, the model demonstrated high precision for the blast class (1.00) but a lower recall of 0.65, indicating that while the model correctly identified blast cases, it missed some instances. This limitation may stem from an excessive number of annotations, causing the model to overemphasize specific patterns within the blast class and struggle to generalize effectively. As a result, the model exhibited signs of overfitting.

In contrast, the blight and tungro classes achieved perfect recall scores of 1.00, indicating no missed instances, with precision scores of 0.84 and 0.87, respectively. These results yielded high F1 scores of 0.91 and 0.93, demonstrating the model's effectiveness in distinguishing between disease classes. This performance is particularly notable for its accuracy in identifying blight and tungro instances.

IV. CONCLUSION

The experimental results demonstrate that the developed model is effective in identifying and categorizing diseases in rice plant images. The model configuration includes hyperparameters determined through a grid search process, featuring a learning rate of 0.0001, the Adam optimizer, a weight decay value of 0.0001, 100 total epochs, and an epoch length of 1000. Performance evaluation using a confusion matrix revealed an accuracy of 88%. The highest precision was achieved in the Blast class (100%), while the highest recall was observed in the Blight and Tungro classes (100% each). Additionally, the Tungro class recorded the highest F1 score (93%), showcasing the model's robust capability to address the challenge of disease identification and categorization in rice plants.

Future research could focus on three main areas. First, improving the model's performance by expanding the dataset through the collection of more images depicting rice plant diseases. A larger dataset is expected to enhance the model's generalization and overall accuracy. Second, optimizing the model further by exploring a wider range of values and parameters during hyperparameter tuning. Advanced techniques, such as Bayesian optimization, could be employed to identify the most effective combination of hyperparameters. Lastly, expanding the range of diseases the model can identify and categorize would equip it with broader functionality, enabling it to address a wider array of rice plant disease detection tasks.

REFERENCES

- D. Hafizah, D. B. Hakim, H. Harianto, and R. Nurmalina, "Analisis Elastisitas Pendapatan Rumah Tangga di Indonesia," J. Ilmu Pertan. Indones., vol. 26, no. 3, pp. 428–435, 2021, doi: 10.18343/jipi.26.3.428.
- [2] BPS, Produksi Padi Di Indonesia 2022. Badan Pusat Statistik, 2022. [Online]. Available: https://www.bps.go.id/publication/2023/08/03/a78164ccd3ad09bdc88e70a2/luas-panen-dan-produksi-padi-di-indonesia-2022.html
- [3] A. Fiddin, M. Sutrawati, H. Bustamam, D. W. Ganefianti, and S. Sipriyadi, "Penyakit Tungro Pada Tanaman Padi (Oryza Sativa) Di Kecamatan Taba Penanjung : Insidensi Penyakit Dan Deteksi Virus Secara Molekuler," J. Ilmu-Ilmu Pertan. Indones., vol. 23, no. 1, pp. 37–45, Jun. 2021, doi: 10.31186/jipi.23.1.37-45.
- [4] S. Sopialena, S. Suyadi, S. Sofian, D. Tantiani, and A. N. Fauzi, "Efektivitas Cendawan Endofit Sebagai Pengendali Penyakit Blast Pada Tanaman Padi (Oryza sativa)," Agrifor, vol. 19, no. 2, p. 355, 2020, doi: 10.31293/af.v19i2.4813.
- [5] N. Sari *et al.*, "Pengabdian kepada Masyarakat: Pengenalan Penyakit Tanaman Padi dan Teknik Pengendaliannya di Desa Bentok Darat, Bati-bati, Kalimantan Selatan," *Lumbung Inov. J. Pengabdi. Kpd. Masy.*, vol. 8, no. 2, pp. 232–243, 2023, doi: 10.36312/linov.v8i1.1130.
 [6] L. Budiarti *et al.*, "Inventarisasi Serangga Hama dan Penyakit pada Berbagai Galur Tanaman Padi," *J-Plantasimbiosa*, vol. 4, no. 1, pp.
- [6] L. Budiarti *et al.*, "Inventarisasi Serangga Hama dan Penyakit pada Berbagai Galur Tanaman Padi," *J-Plantasimbiosa*, vol. 4, no. 1, pp. 36–49, Jun. 2022, doi: 10.25181/jplantasimbiosa.v4i1.2561.
- [7] Y. Linda, Marsuki Iswandi, and Idrus Salam, "Analisis Risiko Usahatani Padi Sawah di Desa Iwoimea Jaya Kecamatan Aere Kabupaten Kolaka Timur," *Perbal J. Pertan. Berkelanjutan*, vol. 11, no. 2, pp. 219–229, Jul. 2023, doi: 10.30605/perbal.v11i2.2645.
- [8] Y. Yusuf Hidayat, M. Marli Batubara, and R. Kurniawan, "Peran Penyuluh Pertanian Lapangan Dalam Mendampingi Kelompok Tani Padi Di Kecamatan Lalan Kabupaten Musi Banyuasin," Soc. J. Ilmu-Ilmu Agribisnis, vol. 6, no. 1, p. 30, Jun. 2017, doi: 10.32502/jsct.v6i1.824.
- [9] S. Xie, Z. Yu, and Z. Lv, "Multi-disease prediction based on deep learning: A survey," C. Comput. Model. Eng. Sci., vol. 127, no. 3, 2021, doi: 10.32604/CMES.2021.016728.
- [10] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.
- [11] H. Noprisson, "Fine-Tuning Model Transfer Learning VGG16 Untuk Klasifikasi Citra Penyakit Tanaman Padi," JSAI (Journal Sci. Appl. Informatics), vol. 5, no. 3, pp. 244–249, 2022, doi: 10.36085/jsai.v5i3.3609.
- [12] A. K and A. S. Singh, "Detection of Paddy Crops Diseases and Early Diagnosis Using Faster Regional Convolutional Neural Networks," in 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), IEEE, Mar. 2021, pp. 898–902. doi: 10.1109/ICACITE51222.2021.9404759.
- [13] E. Murniyasih and L. Suryani, "Penerapan Metode Learning Vector Quantization Untuk Identifikasi Penyakit Padi Berdasarkan Bentuk Bercak Daun," *Electro Luceat*, vol. 6, no. 1, pp. 28–35, Jul. 2020, doi: 10.32531/jelekn.v6i1.190.

- [14] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.
 [15] T. Setiady, "Leaf Rice Disease," *Https://Www.Kaggle.Com/Datasets/Tedisetiady/Leaf-Rice-Disease-Indonesia/*. 2021.
- [16] T. Sedady, Ecar Rice Disease, *Hups// www.laggle.com/Datasets/reasting/Jeag-Rice-Disease-Indonesia*, 2011.
 [16] S. Saifullah, M. Zarlis, Z. Zakaria, and R. W. Sembiring, "Analisa Terhadap Perbandingan Algoritma Decision Tree Dengan Algoritma Random Tree Untuk Pre-Processing Data," *J-SAKTI (Jurnal Sains Komput. dan Inform.*, vol. 1, no. 2, p. 180, 2017, doi: 10.30645/j-sakti.v1i2.41.
- [17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *Total Perform. Scorec.*, pp. 159–183, Jun. 2015, doi: 10.48550/arXiv.1506.01497.
- [18] G. Erdogan Erten, S. Bozkurt Keser, and M. Yavuz, "Grid Search Optimised Artificial Neural Network for Open Stope Stability Prediction," Int. J. Mining, Reclam. Environ., vol. 35, no. 8, pp. 600–617, Sep. 2021, doi: 10.1080/17480930.2021.1899404.
- [19] D. Chicco, N. Tötsch, and G. Jurman, "The matthews correlation coefficient (Mcc) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation," *BioData Min.*, vol. 14, pp. 1–22, 2021, doi: 10.1186/s13040-021-00244-z.
- [20] M. M. Baharuddin, H. Azis, and T. Hasanuddin, "Analisis Performa Metode K-Nearest Neighbor Untuk Identifikasi Jenis Kaca," Ilk. J. Ilm., vol. 11, no. 3, pp. 269–274, 2019, doi: 10.33096/ilkom.v11i3.489.269-274.