LONTAR KOMPUTER VOL. 13, NO. 3 DECEMBER 2022 **p-ISSN 2088-1541**
DOI : 10.24843/LKJITI.2022.v13.i03.p06 **e-ISSN 2541-5832**
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021

HiT-LIDIA: A Framework for Rice Leaf Disease Classification using Ensemble and Hierarchical Transfer Learning

Oddy Virgantara Putra[a1], Niken Trisnaningrum[b2], Niken Sylvia Puspitasari[c3], Agung Toto Wibowo[d4], Ema Rachmawaty[d5]

[a]Department of Informatics, Universitas Darussalam Gontor
Ponorogo, Indonesia
[1]oddy@unida.gontor.ac.id (Corresponding author)

[b]Department of Agrotechnology, Universitas Darussalam Gontor
Ponorogo, Indonesia
[2]niken.trisnaningrum@unida.gontor.ac.id

[c]Department of Al-Quran and Tafsir, Universitas Darussalam Gontor
Ponorogo, Indonesia
[4]nikensylviap@unida.gontor.ac.id

[d]Department of Informatics, Telkom University
Bandung, Indonesia
[4]agungtoto@telkomuniversity.ac.id
[5]emarachmawati@telkomuniversity.ac.id

Abstract

Rice is one of the global most critical harvests, and a great many people eat it as a staple eating routine. Different rice plant diseases harm, spread, and drastically reduce crop yields. In extreme situations, they may result in no grain harvest at all, posing a severe threat to food security. In this paper, to amplify the recognition ability for rice leaf disease (RLD) classification, we proposed hierarchical transfer learning (HTL) methods incorporating ensemble models containing two-step. In the first step, an ensemble combining MobileNet and DenseNet was addressed to tackle the diseased leaf problem. Consequently, DenseNet and XceptionNet were fused to identify three RLDs. Here, we compare our models with state-of-the-art deep learning models such as ResNet, DenseNet, InceptionNet, Xception, MobileNet, and EfficientNet. Our framework at top-notch with 89 % and 91 % for accuracy. In future works, RLD segmentation is suggested to pinpoint the illness and quantify the afflicted region.

Keywords: *Rice Leaf Disease,Transfer Learning,Ensemble Learning, Hierarchical Learning, CNN.*

1. Introduction

Rice disease is one of the factors causing the decline in rice production and even crop failure. One of the causes of rice disease is fungal attack so that plants dry up and wither by Plant Pest Organisms (PPO) consisting of pests, plant diseases, and weeds. In general, the symptoms of disease in plants can be seen, although still slightly before reaching a more severe and widespread stage. Rice leaves are the easiest part to identify disease symptoms that arise in rice. This is because the leaves have a broad cross-section compared to other parts of the body of the rice plant so that changes in color and shape can be seen more clearly. Therefore, the leaves can be used as an initial step for disease detection in rice. The spots that appear on rice leaves have specific

LONTAR KOMPUTER VOL. 13, NO. 3 DECEMBER 2022 **p-ISSN 2088-1541**
DOI : 10.24843/LKJITI.2022.v13.i03.p06 **e-ISSN 2541-5832**
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021

patterns and colors. Differences in the shape and color of the spots determine the type of disease affected in rice. The spots that arise have various forms, such as round, oval, rectangular, and other shapes [1, 2].

The presence of disease attacks on rice can cause crop failure. Proper control is needed when the attack occurs, and preventive action is the most crucial to avoid harvesting failure. However, farmers often ignore this situation due to the lack of knowledge and consider these symptoms to be expected at the time of planting.

With the advancement of digital photography and computer processing in recent years, new methods for identifying plant diseases have emerged. Because of its visualization capabilities, computer vision technology is becoming an appealing tool for continuous illness monitoring. Image and machine learning (ML) research and applications have received a lot of attention, and specific classifiers are often employed to categorize plants into healthy or sick varieties [3, 4, 5]. In reality, the image obtained must span a wide range of situations in order to capture diverse symptom signs associated with plant diseases. Plant diseases can also affect any component of the plant, including the leaves, stems, seeds, and so on. Despite the restrictions, all prior research has substantially supported the CNN model's efficacy in detecting plant illnesses.

Our contribution towards this research is the improvement of RLD classifier using a hierarchical transfer learning (HTL). By using HTL, instead of classifying four class, we classify the normal and abnormal leaves first before detecting the RLD. As we know, the more targets or outputs, the longer times it takes to process. Our proposed HTL also achieved the best performance in term of accuracy.

Many works have been proposed to detect [6, 7, 8], segment [9], and classify [2, 10, 11, 12] rice leaf disease. In 2020, [13] proposed an on-field dataset containing around 6000 images. This dataset contains four distinguished RLD, Bacterial Blight, Blast, Brown Spot, and Tungro. Moreover, this work evaluated eleven pre-trained CNN methods and SVM. It was reported that CNN incorporated with SVM performs the best compared to other CNN methods.

Li [2] proposed a recognition system for rice leaf disease (RLD) using video-based on CNN. The suggested technique attempts to detect crop lesion footage in real-time for the first time. The system included a frame extraction module, an image detector, and a video synthesis module and was trained on a still-image neural network model using images from their dataset. The system can distinguish many sorts of lesion locations in a single video. Faster-RCNN is used as the architecture for the still-image detector, with a custom deep convolutional neural network (DCNN) backbone. Alidrus [14] proposed a simple CNN to classify RLD. The dataset was collected public and manually hand-picked from the local paddy field. This work contained three convolutional layers with MaxPooling paired with ReLU as an activation function. The result of this network that the accuracy gained was 77 percent. Rahman [5] proposed an RLD identification based on transfer learning (TL) and fine-tuning model. Even though this method was called Simple CNN, it adapted from some famous deep learning model, VGGNet16. The results were quite impressive for both mean accuracy and standard deviation with 94.33 percent and 0.96, respectively.

An RLD classification-based CNN and TL were proposed to differentiate Leaf Blast, Leaf Blight, and Brown Spot [15]. This method employed five convolutional blocks with a two-dimensional convolutional layer (CL) in the first two-layer followed by three-dimensional CL in the remaining layers. Each block is pooled with a maximum value. In the dense layer, it implemented a 128-neuron fully-connected layer. As the complement of CNN, this research also benefited TL based on VGG16Net. However, it was concluded that the performance of CNN was worse than VGGNet based TL with 74% and 92.46%, respectively.

Jiang [16] suggested a study the following year, collecting 40 images of leaf illnesses to improve the VGGNet model, which is based on the idea of multi-task learning. For TL, the model used ImageNET's pre-training model. The model's accuracy is 97.22 percent for rice leaf diseases, and for wheat leaf diseases, it is 98.75 percent. Comparative trials show that this strategy outperforms the single-task model, a TL reuse-model method, ResNet50, and DenseNet121 models. The experimental findings suggest that the proposed upgraded VGG16 model and multi-task TL approach can detect rice and wheat leaf illnesses simultaneously, providing a reliable method for

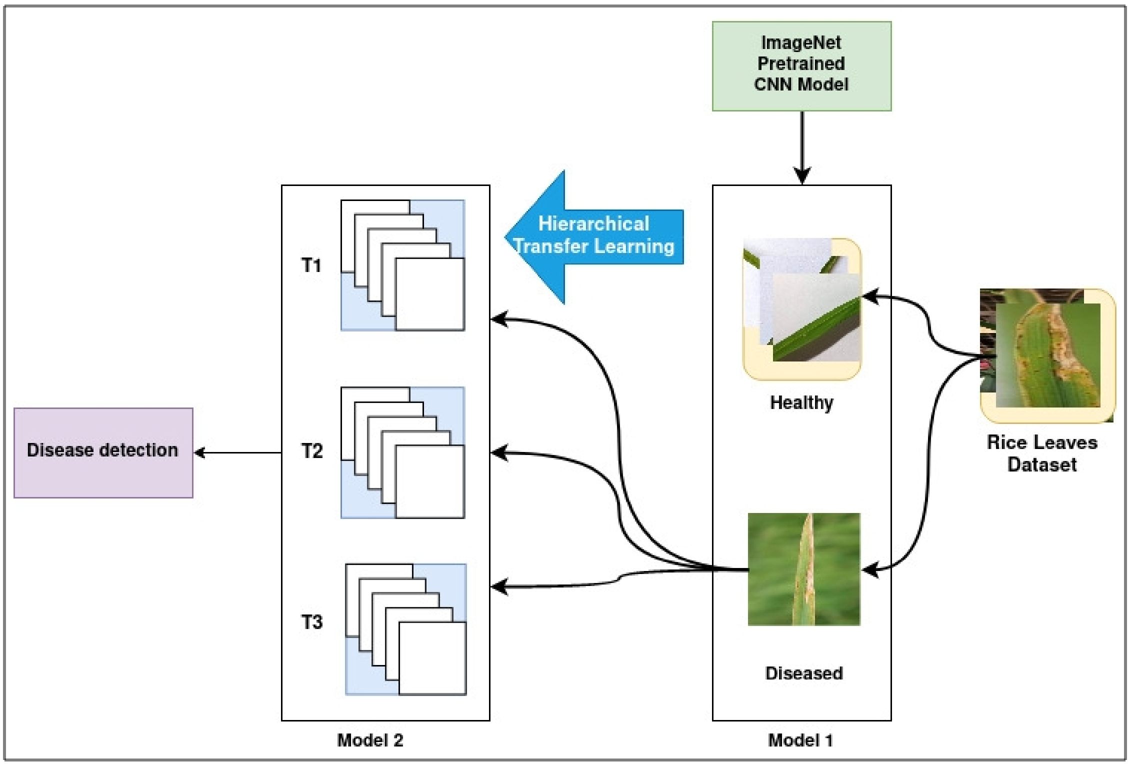

Figure 1. Our proposed framework for HTL architecture

recognizing leaf diseases in a wide range of plants.

A MobileNet based is proposed by [17] incorporating two different types of building pieces for two variations of depthwise separable convolution. To improve accuracy of the regular MobileNet, this work modified into a reduced MobileNet. A width multiplier α is introduced to reduce computational complexity within network uniformly. The network model was trained on a public PlantVillage dataset and achieved around 98% in accuracy. MobileNet is often used to gain small latency for the Internet of Things (IoT) and embedded systems. A method based on MobileNet was proposed with optimized bayesian rules for hyperparameter tuning optimization [11]. The proposed work incorporates attentive depthwise deep networks and augmented mechanism (AM). The interesting point of this work was the existence of AM part. AM can smooth and translate the efforts from inputs in a spatial domain disparity. The dataset collected contains four categories of RLD with 400 data. For the preprocessing technique, the dataset was processed to morphological segmentation before being transformed. It was shown that the method achieved accuracy at 94.65%.

2. Research Methods

In this work, we proposed a framework for RLD classification using several layers of deep learning (DL) models. This section is separated into some subsection which are data gathering, data preprocessing, first and second layer modeling.

2.1. Data Gathering

We collected some datasets from two different sources. First, we got a public dataset from kaggle.com. This dataset was collected from several sources. One of them is from [18]. The content of the dataset is three distinguished rice leaf diseases, namely Leaf Blast (LB), Hispa (Hsp), and Brown-spot (BS). Second, we took several photos from some local rice fields and combined them into the first dataset. We separated the dataset into two groups for detection and classification purposes from the last dataset. For the detection purpose, we split into healthy and diseased classes.

2.2. Data Preprocessing

This subsection preprocessed the dataset with image augmentation (IA) and image resizing. The word augment does not always mean adding a new data image. Instead of increasing data size, image augmentation is a series process of multiple transformations such as image rotation, translation, shearing, flipping, and shifting. Then, the augmentation yields a new image and replaces the existing image with arbitrary augmented batch images.

By introducing random vibrations and disturbances into the original training pattern, data IA covers various strategies for generating "new" training patterns. When using IA, the generalization function of the model improves. Our network is constantly exposed to new, slightly modified copies of the input data to learn more robust properties. However, we limit the usage of IA during testing, instead evaluate trained networks against raw test data. Training accuracy is slightly reduced in most cases, but test accuracy is improved.

2.3. Leaf Disease Detection

In recent year, numerous works based on CNN have been proposed for the task of rice leaf disease (RLD) classification [5, 19, 12] and TL [2, 20, 10]. Our previous work in 2021 is also based on CNN [14]. Here, we proposed a framework incorporating hierarchical and TL for rice leaf disease identification called HiT-LIDIA as in Figure 1. HiT-LIDIA is composed of a two-step model. In the first step (MK-I), the model classifies rice leaves into two categories, healthy (Hl) and diseased (Ds). Second (MK-II), the results from diseased leaves are separated into three distinguished classes: Hsp, LB, and BS. Our framework, not only stops here, but also improved further by combining each layer with several deep learning methods. We discuss each model as follows:

2.3.1. MK-I

Standard CNN has a weakness when the convolutional layer goes deeper. It is prone to vanishing gradient and redundant feature windows. DenseNet simplifies the layer-to-layer connectivity pattern found in previous architectures to tackle such a problem. Rather than relying on extremely deep or wide topologies for representational strength, DenseNet makes use of the network's potential by repeating features.

DenseNet [21] and MobileNet [22] have significant results for RLD context compared to other previous methods. DenseNet, contrary to popular belief, requires fewer parameters than a typical CNN because it does not require the acquisition of redundant feature images. Additionally, many layers contribute relatively minor and can be ignored in numerous ResNet versions. ResNet has considerable quantity trainable parameters with demanding weights to optimize independently. DenseNet layers, on the other hand, have a minimal number of filters, resulting in an insignificant number of new feature maps.

MobileNet is a deep learning architecture based on depthwise separable convolution (DSN), which contains two convolutions, depth-wise (DWC) and point-wise (PWC). Each input channel receives a single filter via the DWC. This calculation differs from a regular convolution, which filters all input channels. The computing cost of a conventional convolution is dependent on the extent of both input and output dimensions allotted, the input attribute map, and the convolution kernel in the spatial domain. As illustration, given an input image **M** with size d x d x c and three δ x δ kernels **k** with one channel each, we can compute **M** and **k** by using DWC. Before that, we compare the complexity of conventional CNN with DWC. We have:

$$\Theta_1 = \mathbf{k}_\delta \cdot \mathbf{k}_\delta \cdot d \cdot \hat{d} \cdot F_s \cdot F_s \tag{1}$$

Where F_s denotes the input feature map's unique dimensions, and $\mathbf{k}_\delta$ denotes the convolution

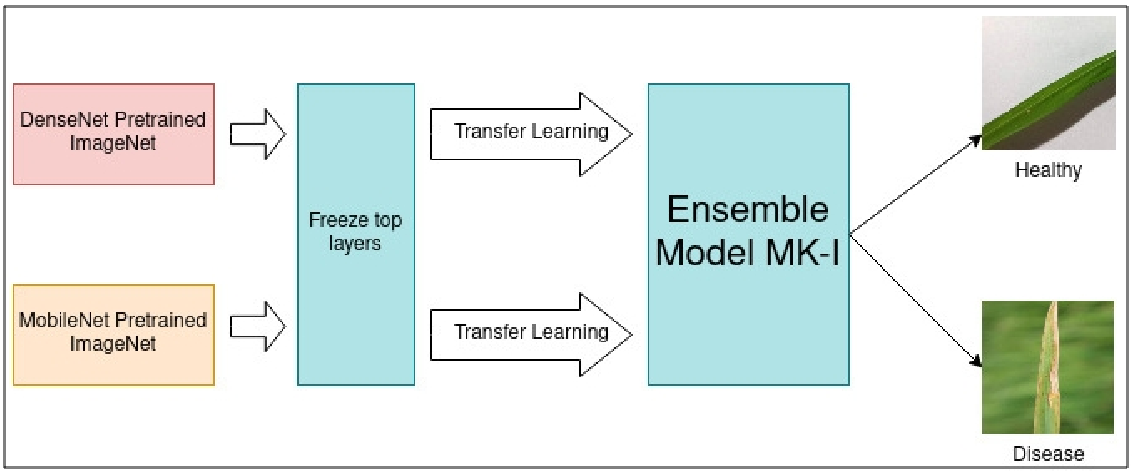

Figure 2. Architecture of MK-I.

kernel's size. The input and output channels are d and $\hat{d}$, respectively. And for the DWC cost complexity is:

$$\Theta_2 = \mathbf{k}_\delta \cdot \mathbf{k}_\delta \cdot d \cdot F_s \cdot F_s \tag{2}$$

From equations 1 and 2, we can see that the complexity of Θ_2 is reduced compared from Θ_1 because the DWC only calculate each channel and not combine them. Additionally, DWC works with PWC in order to improve the computational speed.

Two pre-trained deep learning models, DenseNet and MobileNet, are incorporated in model MK-I. A combination of two or more models is also called ensemble learning (EL) [23, 24] Initially, the top layers of DenseNet are frozen. The previous step was pre-trained with the Hl-Ds dataset (D1). Various hyperparameters are set beforehand to find the most optimized. Subsequently, the fittest model is achieved for MK-I. This whole process can be seen as in Figure 2.

2.3.2. MK-II

As we know, regular CNN (RegCNN) is computationally expensive as its layers go deep. For this reason, many CNN models have been proposed to tackle such problems. For instance, MobileNet reduced the convolution computation complexity from input images into CNN-feature maps as well as DenseNet with parameter reduction strategy. InceptionNet, as worthy of its name, is often mentioned as a breakthrough in CNN world [25]. Instead of working with a deeper layer, InceptionNet goes wider. This network utilized an auxiliary classifier (AuxC) to achieve better convergence where standard CNN failed due to vanishing gradient. However, in the next year, InceptionNet was overwhelmed by XceptionNet in ImageNet challenge [26].

Classifying Hl and Ds rice leaves are insufficient to handle a complex task. It is possible to add a new output of the Hl class into our dataset and reduce the model just to handle a single classification task. However, the distribution of both Hs and Ds is imbalanced in real life. In our case, Hs class prevails over Ds in number. Consequently, instead of putting it into a single task, we appended a new MK-II model to tackle Ds identification.

Before making the final decision, we trained and evaluated various models, including ResNet, VGGNet, DenseNet, InceptionNetV3, XceptionNet, and EfficientNet. As a result, we chose the top two levels as our ensemble model and combined those in MK-II: DenseNet and XceptionNet. These two models were chosen based on their abilities. The architecture of MK-II can be seen in Figure 3.

p-ISSN 2088-1541
e-ISSN 2541-5832

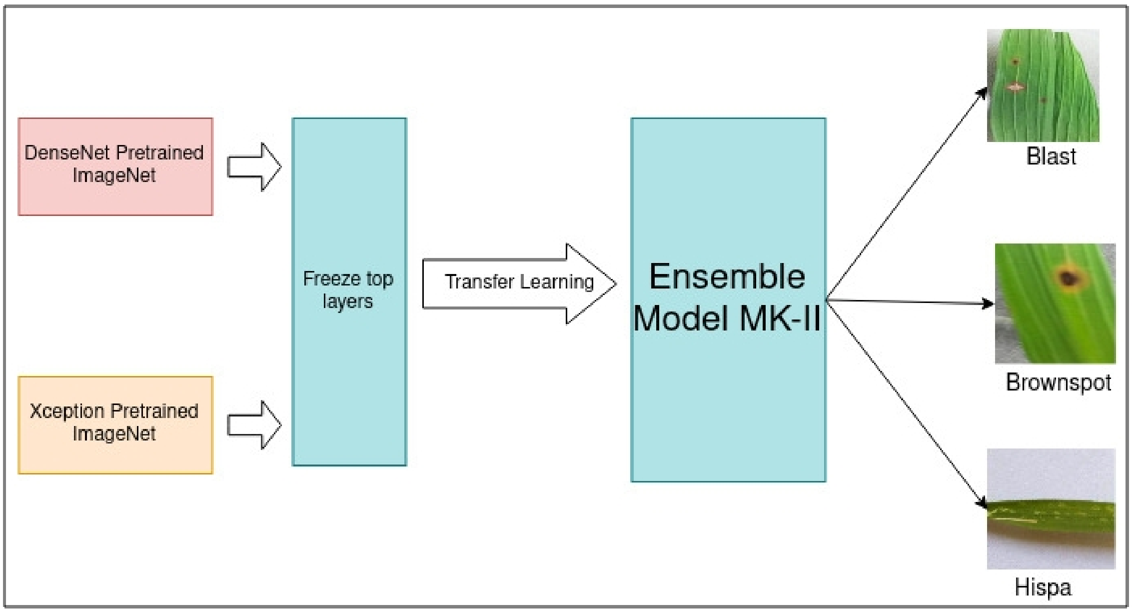

Figure 3. Architecture of MK-II.

2.4. Evalution Procedures

Considering statistical for true negative (TN), false negative (FN), true positive (TP), and false positive(FP) detections, a comprehensive experiments was conducted to evaluate the performance of all models in this paper. Various metrics such as Recall (Rec), Accuracy (Acc), F1-score (F1), and Precision (Prec) were employed to measure RLD identification as in Eq. 5, 3, 4, and 6, respectively.

$$Rec = \frac{TP}{TP + TN} \tag{3}$$

$$Prec = \frac{TP}{TP + FP} \tag{4}$$

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

$$F1 = \frac{2 * TP}{2 * TP + FN + FP} \tag{6}$$

Moreover, we added Cohen's Kappa metric to assess the settlement between two scorers as stated in Eq 7.

$$\kappa = \frac{\delta_0 - \delta_e}{1 - \delta_e} \tag{7}$$

where κ is the Cohen's Kappa value, δ_0 is the sum of all accuracy of the model, δ_e is the settlement score between ground truth and predicted value for related class.

3. Results and Analysis

In this part, both MK-I and MK-II were tested on public rice leaf disease datasets received from Kaggle. ResNet, DenseNet, EfficientNet, InceptionNet, XceptionNet, and VGGNet were all examined to determine how well they compared to our models. This test was performed on a PC with

LONTAR KOMPUTER VOL. 13, NO. 3 DECEMBER 2022 p-ISSN 2088-1541
DOI : 10.24843/LKJITI.2022.v13.i03.p06 e-ISSN 2541-5832
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021

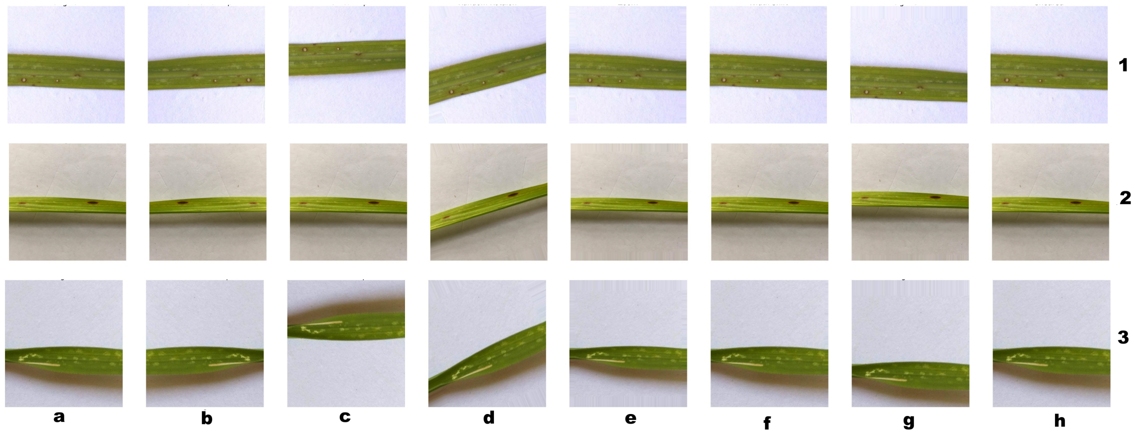

Figure 4. The sample images from augmentation techniques. (1 - 3) are the rice disease where (1) is the Brown spot, (2) is Blast, (3) is Hispa. (a) is the original image, (b-h) are the augmentation results where (b) is horizontal flipping, (c) is vertical flipping, (d) is a 30-degree of freedom rotation, (e) is zooming, (f) is width shifting, (g) is height shifting, and (g) is shearing

16GB of RAM, an NVIDIA GTX 1050Ti 4GB GPU, and an Intel Core i7 8th-gen processor running Ubuntu 20.04 x64.

3.1. Dataset Gathering

We collected dataset from public sources https://www.kaggle.com/shayanriyaz/riceleafs (D1) and https://www.kaggle.com/tedisetiady/leaf-rice-disease-indonesia (D2). The D1 contains four directories BrownSpot, Healthy, Hispa, and LeafBlast. In order to train MK-I, D1 directories were restructured into two categories, Healthy and Diseased. BS, Hl, and Hsp were accumulated into the Diseased directory. We called this dataset Dt1. BS, Hl, and Hsp directories (Dt2) are used to train MK-II. In addition to the test dataset, we handpicked some RLD from local rice fields with a smartphone camera. The dataset sample are shown in 4.

3.2. Data Preprocessing

The dataset is partitioned into training and testing with portions 80% and 20%. As aforementioned, we implemented an image augmentation method to enrich the dataset because our images are taken in a particular condition with vivid background and foreground. Many augmentations are deployed here, such as horizontal and vertical flipping, random rotation, zooming, shearing, and shifting width and height to generate synthetic images. Some hyperparameters were included as input to the synthetic image generator, such as 30 degrees of freedom for rotation, zooming range at 0.2, shifting range at 0.1 and 0.2 for width and height, respectively, and shearing at 0.2.

All of the images were resized to 224 x 224 pixels with three RGB channels because it is well-known that the 224-sized image is the most suitable setup for SOTAs in here [27]. As the training parameters, we set the number of epochs to 20 and batch size to 32. A stochastic gradient descent (SGD) based algorithm, named ADAM, is adapted as the back-propagation optimization method. Learning Rate is set to 1e-04, β_1 to 0.9, β_2 to 0.999, and ϵ to $1e-08$.

3.3. Quantitative Assessment on Ds1

In here, we conducted assessment test on previous SOTAs compared with MK-I. We trained each model with some hyper-parameters. Prior to training, we set iteration to 20 and batch size 32. As the cost function, we implemented the binary cross-entropy.

Table 1 illustrates the performance comparison of several SOTAs models and our MK-I model. Overall, the majority of SOTA's achieved significantly high validated accuracy with scores greater

LONTAR KOMPUTER VOL. 13, NO. 3 DECEMBER 2022
DOI : 10.24843/LKJITI.2022.v13.i03.p06
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021
p-ISSN 2088-1541
e-ISSN 2541-5832

Table 1. Assessment results from SOTAs and MK-I.

Models	**Acc (%)**	**Loss**	κ	Prec	**Rec**	**F1-score**
MK-I	**0.89**	0.3432	**0.783**	**0.895**	**0.89**	**0.895**
EfficientNet	0.5	0.7046	0.029	0.52	0.515	0.49
InceptionNetV3	0.88	0.3406	0.767	0.88	0.885	0.88
ResNet50	0.86	0.4061	0.710	0.855	0.86	0.855
VggNet19	0.57	0.6409	0.143	0.57	0.575	0.57
Xception	0.87	0.3451	0.739	0.87	0.87	0.87
DenseNet201	0.88	**0.2964**	0.766	0.885	0.88	0.885
MobileNet	0.84	0.4404	0.687	0.845	0.85	0.84

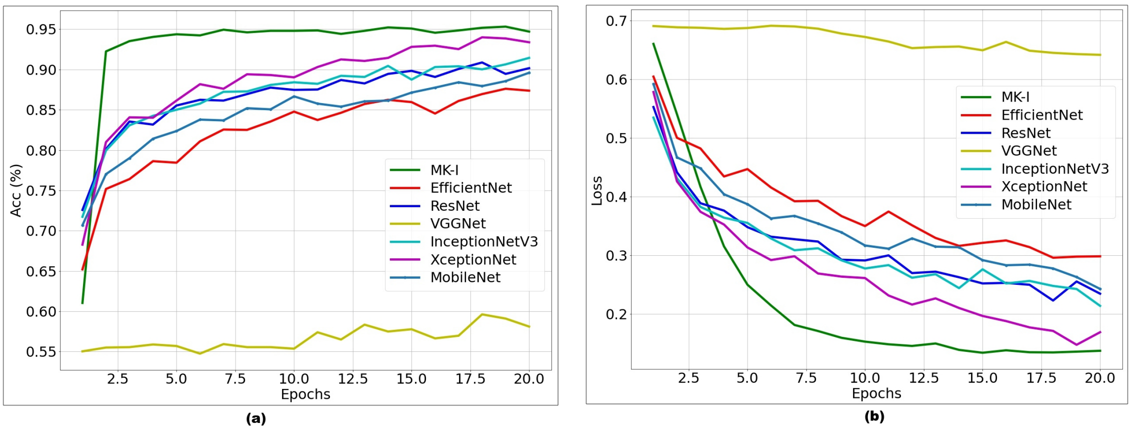

Figure 5. Training results from SOTAs and MK-I using Ds1. (a) is the training accuracy, (b) is the loss

than 80%. Meanwhile, the proportion of the remaining counterparts is relatively lower than 60%. MK-I stood at overall SOTAs with the percentage of accuracy at roughly 89%. It can be clearly seen that MK-I was significantly more accurate than EfficientNet and VggNet19. Despite the fact that MK-I has a tiny improvement over the rest, it is still top-notch. Unfortunately, our model is at second runner-up in terms of Loss. DenseNet and InceptionNetV3 conquer MK-I.

MK-I outperforms other models for Cohen's Kappa, Precision, Recall, and F1-score in the following metrics. MK-I hit κ with a score of 0.783, followed by DenseNet201 and InceptionNetV3 at the second and third positions, respectively. EfficientNet is at the bottom tiers with VggNet19, and the disparity is quite significant compared to others. Furthermore, MK-I is also unrivaled for Precision, Recall, and F1-score, with a score nearly hitting 0.9.

The line chart in Figure 5 exhibits the trend of six different types of SOTAs and MK-I by training performance for both accuracy and loss. Overall, it can be seen that the training accuracy is exponentially increasing while loss is declining. To begin with, it is clear that MK-I was undeniably the most accurate and converged earlier at epoch 3. MK-I reached a smooth curve while the others fluctuated and increased steadily. In loss performance, MK-I falls earlier at the first quarter of epochs while the remainders roughly downward at a slow pace to the end of the epoch. At the end of the epoch, MK-I is outstanding with 95% accuracy and almost reach 0.1 for loss. Second, EfficientNet, ResNet, InceptionNetV3, XceptionNet, and MobileNet have remained similar since the beginning of training and slightly increased and decreased for accuracy and loss over time during iteration. Finally, VggNet did not make significant gains in both accuracy and loss from the start. At epoch 18, the accuracy test only climbed from 55% to 60%, and then gradually fell for loss. Nonetheless, towards the end of the iteration, it began to demonstrate remarkable progress.

LONTAR KOMPUTER VOL. 13, NO. 3 DECEMBER 2022 p-ISSN 2088-1541
DOI : 10.24843/LKJITI.2022.v13.i03.p06 e-ISSN 2541-5832
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021

Table 2. Assessment results from SOTAs and MK-II using dataset Ds2

Model	**Acc (%)**	**Val Loss**	κ	**Precision**	**Recall**	**F1-score**
MK-II	**0.9100**	**0.2371**	**0.8566**	**0.9067**	**0.9133**	**0.9100**
EfficientNet	0.4500	1.1231	0.1902	0.5333	0.4700	0.4400
InceptionNetV3	0.8700	0.3088	0.7996	0.8700	0.8700	0.8667
ResNet50	0.4861	2.2417	0.1237	0.4733	0.4300	0.3367
VggNet19	0.8160	0.5861	0.6208	0.7500	0.7800	0.7500
Xception	0.8829	0.9570	0.8218	0.8800	0.8967	0.8867
DenseNet201	0.8100	0.6239	0.7048	0.8067	0.8167	0.8000
MobileNet	0.3200	2.0657	0.0350	0.2800	0.3633	0.2133

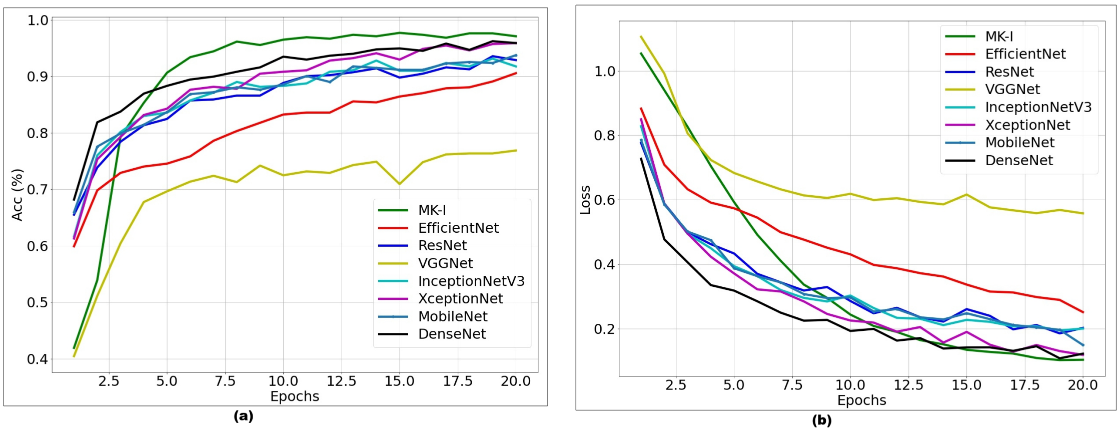

Figure 6. Training results from SOTAs and MK-II using Ds2. (a) is the training accuracy, (b) is the loss

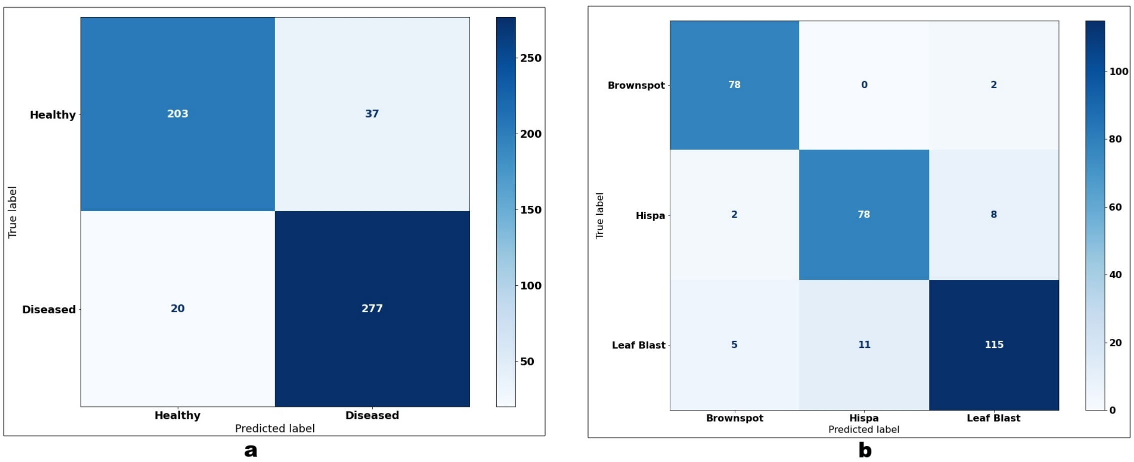

Figure 7. Confusion Matrix results from our models. (a) is MK-I, (b) is MK-II

3.4. Quantitative Assessment on Ds2

We compared preceding SOTAs to the MK-II in our assessment. With some hyper-parameters, we trained each model. We adopted categorical cross-entropy as the cost function and configured iteration to 20 and batch size to 32 before training.

The performance comparison of different SOTAs models and our MK-II model is shown in table 2. In general, the majority of SOTAs attained considerably high validated accuracy, with scores over 85%. Meanwhile, the remaining equivalents make up a smaller percentage of the total. MK-II took first place in overall SOTAs, with an accuracy rate of over 91 % percent. It is apparent that MK-II outperformed InceptionNetV3 and Xception in terms of accuracy. Despite the fact that MK-II is only little better than the rest, it is still excellent.

Cohen's Kappa, Precision, Recall, and F1-score are among the metrics where MK-II beats other models. With $kappa$ score of 0.8566, MK-II at top-notch while XceptionNet and InceptionNetV3 came in second and third, respectively. With ResNet50, EfficientNet is in the bottom tiers, and the difference is noticeable when compared to other networks. In addition, the MK-II is unparalleled in terms of Precision, Recall, and F1-score, with scores over 0.9.

The trend of six varying sorts of SOTAs and MK-II by training performance for both accuracy and loss can be seen in Figure 6 train acc loss. Overall, the training accuracy steadily improves while the loss is significantly reduced. To begin, it is indisputable that MK-II was the most accurate and converged sooner at epoch 3. While others fluctuated and climbed consistently, MK-II reached a smooth curve. In terms of loss performance, MK-II drops faster in the first quarter of epochs, then gradually declines to the end of the epoch. MK-II performs admirably at the end of the period, with 95 percent accuracy and less than 0.1 percent loss rate.

The confusion matrix results was presented in Figure 7. Our models can easily distinguish most of test dataset. MK-I is capable of separate between healthy and diseased leaf accurately with correct identification more than 200 images. Only one of ten is misclassified of both classes. Model MK-I also successfully recognized three RLDs. For instance, only 9 out of 103 for BS class, Hsp 11 from 115 images, and LB under 10%.

4. Conclusion

In both theory and practice, CNN is a useful pattern identification approach. Two models are used in this study to identify and categorize LB, Hsp, and BS images. The leaf lesions of three rice illnesses were retrieved using an artificial computation approach and the MK-I ensemble model to assess whether they were healthy or sick before disease categorization. After that, a TL-based ensemble model known as MK-II was employed to identify and distinguish three different types of rice leaf lesions. The experimental findings suggest that MK-I outperforms the compared SOTA algorithms with an accuracy score of 89 percent, a loss score of 0.3432, and a Cohen's Kappa of 0.783. MK-II beats SOTA techniques, with loss and Cohen's Kappa values of 0.2371 and 0.8566, respectively. Our models are able to recognize RLDs with a 91% accuracy rate. To summarize, using a hierarchical mix of TL to identify rice leaf diseases and give technical help for future crop fields can be beneficial.

Considering CNN's significant advances in machine learning, certain research obstacles are confirmed. CNN, for instance, has dozens, if not hundreds, of levels. Finally, the optimal number of layers and neurons is determined by each layer's number of layers and neurons. Instead of engaging on a vast number of tests, it does not have an option. The second issue is that large-scale datasets have been necessary for an efficient deep learning platform. A better quality of images dataset is required to increase RLD accuracy. We intend to utilize a range of deep networks and training algorithms in the future to improve recognition performance.

References

[1] R. Deng, M. Tao, H. Xing, X. Yang, C. Liu, K. Liao, and L. Qi, "Automatic diagnosis of rice diseases using deep learning," *Frontiers in Plant Science*, vol. 12, 2021.

[2] D. Li, R. Wang, C. Xie, L. Liu, J. Zhang, R. Li, F. Wang, M. Zhou, and W. Liu, "A recognition method for rice plant diseases and pests video detection based on deep convolutional neural network," *Sensors (Switzerland)*, vol. 20, 2020.

[3] Y. Lu, S. Yi, N. Zeng, Y. Liu, and Y. Zhang, "Identification of rice diseases using deep convolutional neural networks," *Neurocomputing*, vol. 267, pp. 378–384, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2017.06.023

[4] F. Jiang, Y. Lu, Y. Chen, D. Cai, and G. Li, "Image recognition of four rice leaf diseases based on deep learning and support vector machine," *Computers and Electronics in Agriculture*, vol. 179, p. 105824, 2020. [Online]. Available: https://doi.org/10.1016/j.compag.2020.105824

[5] C. R. Rahman, P. S. Arko, M. E. Ali, M. A. I. Khan, S. H. Apon, F. Nowrin, and A. Wasif, "Identification and recognition of rice diseases and pests using convolutional neural networks," *Biosystems Engineering*, vol. 194, pp. 112–120, 2020. [Online]. Available: https://doi.org/10.1016/j.biosystemseng.2020.03.020

[6] G. An, M. Akiba, K. Omodaka, T. Nakazawa, and H. Yokota, "Hierarchical deep learning models using transfer learning for disease detection and classification based on small number of medical images," *Scientific Reportss*, vol. 11, p. 4250, 3 2021. [Online]. Available: https://doi.org/10.1038/s41598-021-83503-7

[7] R. Gandhi, S. Nimbalkar, N. Yelamanchili, and S. Ponkshe, "Plant disease detection using cnns and gans as an augmentative approach," 2018, pp. 1–5.

[8] M. Sardogan, A. Tuncer, and Y. Ozen, "Plant leaf disease detection and classification based on cnn with lvq algorithm," *UBMK 2018 - 3rd International Conference on Computer Science and Engineering*, pp. 382–385, 2018.

[9] G. Zhou, W. Zhang, A. Chen, M. He, and X. Ma, "Rapid detection of rice disease based on fcm-km and faster r-cnn fusion," *IEEE Access*, vol. 7, pp. 143 190–143 206, 2019.

[10] E. C. Tetila, B. B. Machado, G. Astolfi, N. A. de Souza Belete, W. P. Amorim, A. R. Roel, and H. Pistori, "Detection and classification of soybean pests using deep learning with uav images," *Computers and Electronics in Agriculture*, vol. 179, 2020.

[11] Y. Wang, H. Wang, and Z. Peng, "Rice diseases detection and classification using attention based neural network and bayesian optimization," *Expert Systems with Applications*, vol. 178, p. 114770, 2021. [Online]. Available: https://doi.org/10.1016/j.eswa.2021.114770

[12] P. Temniranrat, K. Kiratiratanapruk, A. Kitvimonrat, W. Sinthupinyo, and S. Patarapuwadol, "A system for automatic rice disease detection from rice paddy images serviced via a chatbot," *Computers and Electronics in Agriculture*, vol. 185, p. 106156, 2021. [Online]. Available: https://doi.org/10.1016/j.compag.2021.106156

[13] P. K. Sethy, N. K. Barpanda, A. K. Rath, and S. K. Behera, "Deep feature based rice leaf disease identification using support vector machine," *Computers and Electronics in Agriculture*, vol. 175, p. 105527, 2020. [Online]. Available: https://doi.org/10.1016/j.compag.2020.105527

[14] S. A. Alidrus, A. Musthafa, and O. V. Putra, "Deteksi penyakit pada daun tanaman padi menggunakan metode convolutional neural network," 2021, pp. 103–109.

[15] S. Ghosal and K. Sarkar, "Rice leaf diseases classification using cnn with transfer learning," 2020, pp. 230–236.

[16] Z. Jiang, Z. Dong, W. Jiang, and Y. Yang, "Recognition of rice leaf diseases and wheat leaf diseases based on multi-task deep transfer learning," *Computers and Electronics in Agriculture*, vol. 186, p. 106184, 2021. [Online]. Available: https://doi.org/10.1016/j.compag.2021.106184

[17] K. KC, Z. Yin, M. Wu, and Z. Wu, "Depthwise separable convolution architectures for plant disease classification," *Computers and Electronics in Agriculture*, vol. 165, p. 104948, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0168169918318696

[18] H. B. Prajapati, J. P. Shah, and V. K. Dabhi, "Detection and classification of rice plant diseases," *Intelligent Decision Technologies*, vol. 11, pp. 357–373, 2017.

[19] S. T. Cynthia, K. M. S. Hossain, M. N. Hasan, M. Asaduzzaman, and A. K. Das, "Automated detection of plant diseases using image processing and faster r-cnn algorithm," *2019 International Conference on Sustainable Technologies for Industry 4.0, STI 2019*, vol. 0, pp. 24–25, 2019.

[20] N. Duong-Trung, L.-D. Quach, M.-H. Nguyen, and C.-N. Nguyen, "Classification of grain discoloration via transfer learning and convolutional neural networks." Association for Computing Machinery, 2019, p. 27–32. [Online]. Available: https://doi.org/10.1145/3310986.3310997

[21] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," 2017, pp. 4700–4708.

[22] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *CoRR*, vol. abs/1704.0, 2017. [Online]. Available: http://arxiv.org/abs/1704.04861

[23] H. B. F. David, A. Suruliandi, and S. P. Raja, "Stacked framework for ensemble of heterogeneous classification algorithms," *Journal of Circuits, Systems and Computers*, vol. 30, p. 2150269, 2021. [Online]. Available: https://doi.org/10.1142/S0218126621502698

[24] V. Tayanov, A. Krzyżak, and C. Y. Suen, "Ensemble learning using matrices of classifier interactions and decision profiles on riemannian and grassmann manifolds," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 35, p. 2160011, 2021. [Online]. Available: https://doi.org/10.1142/S0218001421600119

[25] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *CoRR*, vol. abs/1409.4, 2014. [Online]. Available: http://arxiv.org/abs/1409.4842

[26] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," 2017, pp. 1800–1807.

[27] J. Chen, D. Zhang, A. Zeb, and Y. A. Nanehkaran, "Identification of rice plant diseases using lightweight attention networks," *Expert Systems with Applications*, vol. 169, p. 114514, 2021. [Online]. Available: https://doi.org/10.1016/j.eswa.2020.114514