International Journal of Electrical and Computer Engineering (IJECE)
Vol. 14, No. 6, December 2024, pp. 6874~6882
ISSN: 2088-8708, DOI: 10.11591/ijece.v14i6.pp6874-6882

Dipterocarpaceae trunk texture classification using two-stage convolutional neural network-based transfer learning model

Masna Wati, Novianti Puspitasari, Ummul Hairah, Joan Angelina Widians, Ade Fiqri Tjikoa
Department of Informatics, Faculty of Engineering, Mulawarman University, Samarinda, Indonesia

Article Info

Article history:

Received Nov 26, 2023
Revised Jul 30, 2024
Accepted Aug 6, 2024

Keywords:

Borneo
Convolutional neural network
Dipterocarpaceae
Image classification
Transfer learning

ABSTRACT

The importance of plant identification has been recognized by academia and industry. There have been several attempts to utilize leaves and flowers for identification. However, the trunk can also be helpful, especially for tall trees. In Borneo, the Dipterocarpaceae family are the main constituents of the tropical rainforest ecosystem. This research focuses on the classification of the dipterocarp family, which can reach a height of between 70 and 85 m. Leveraging convolutional neural network (CNN) models, this research proposes a two-stage transfer learning strategy. In the first stage, the pre-trained CNN models are refined by only modifying the classification layer while keeping the feature layer frozen. The second stage involves selecting and freezing several convolutional layers to adapt the model to classify dipterocarp stems. The dataset consists of 857 images of different dipterocarp species. Experiments show that the VGG16 model with a two-stage transfer learning strategy achieves a high accuracy of 98.246%. This study aims to accurately identify species, benefiting conservation and ecological studies by enabling fast and reliable tree species classification based on stem texture images.

Corresponding Author:

Masna Wati
Department of Informatics, Faculty of Engineering, Mulawarman University
Samarinda, East Kalimantan, Indonesia
Email: masnawati@fkti.unmul.ac.id

1. INTRODUCTION

Tropical rainforests are part of the world's ecosystem diversity and are home to a diversity of flora and fauna species. Indonesia has great biodiversity. Biodiversity is a prominent national asset to maintain its sustainability and utilization. One type of endemic plant that dominates the tropical rainforests of Kalimantan is trees from the dipterocarp family. This family is emblematic of the tropical rainforests across Southeast Asia, as well as numerous seasonally dry forests in South and Southeast Asia [1]. The Dipterocarpaceae family is a native species found in the tropical rainforests of Kalimantan. This plant family can grow to heights ranging from 70 to 85 m and holds considerable importance. Dipterocarpaceae, a subfamily within this group, comprises numerous species with a relatively broad geographical presence. These trees, known as dipterocarps, are the main sources of timber in tropical rainforests across regions including western Indonesia, Malaysia, Brunei, and the Philippines, extending eastward to Papua and Papua New Guinea [2]. This family is a plant tribe whose members are all trees that have an essential role, both from an economic and ecological perspective. From an economic perspective, most of these tribes are commercial wood producers to meet various needs. Apart from that, several types of dipterocarps also produce oil, resin and fruit with trade value. This subfamily has 13 genera and 470 species, 9 of which are found in Indonesia [3].

Differentiation of dipterocarp species has traditionally been based on various morphological characteristics, including stem texture, leaf pattern, and other visual attributes. Based on several research results in East Kalimantan, there are around eight genera with dozens of types of dipterocarp trees. These genera include *Shorea*, *Dryobalanops*, *Dipterocarpus*, *Anisoptera*, *Parashorea*, *Cotylelobium*, *Vatica*, and *Hopea* [4]. Dipterocarp species are primarily found in areas with wet climates, high humidity, altitudes of 0–800 m above sea level, and rainfall above 2,000 mm/year with a short dry season. Most types of *Dipterocarpaceae* also grow in dry, acidic, deep solum and clay soils. In acid soil conditions, the roots of dipterocarp plants associate with *ectomycorrhiza* so they can survive and develop [5]. Conservation of dipterocarp forests benefits the forestry sector by protecting biodiversity, preventing landslides, and absorbing atmospheric carbon dioxide. Meanwhile, environmental changes, pollution, temperature increases, climate change, forest fires, and ecosystem pollution can threaten biodiversity. Our research is a form of concern for the preservation of dipterocarps in Kalimantan, especially in East Kalimantan.

Identifying different types of tree trunks poses a difficulty in texture classification. Over time, researchers have relied on manually created features to address this challenge. For instance, some studies have developed their approaches using local binary patterns (LBP) as a basis [6]. Using scale-invariant feature transform (SIFT) descriptors and support vector machines (SVM) achieves an accuracy of around 70% when applied to the Austrian Federal Forest dataset. Four statistical attributes, uniformity, entropy, asymmetry, and smoothness are used in texture classification for bark images. They use decision tree algorithms for classification tasks [7]. The support vector machine (SVM) algorithm is utilized for categorizing various texture materials, as showcased in experiments conducted on the Flickr material dataset [8]. Convolutional neural networks (CNN) have recently begun to outperform traditional methods in a variety of fields, most notably pattern recognition. Convolutional neural networks use supervised machine learning techniques, and there are a lot of CNN architectures called pre-trained CNN model. CNN is the best network for image recognition tasks because it can extract special features of the input images with its large number of filters. The diversity in these architectures comes from the improvements or modifications that are applied to the original CNN architecture [9]. CNN models that are widely used and have high performance include AlexNet [10], VGG16 [11], residual network (ResNet) model [12], dense network (DenseNet) [13], and Inception [14].

Numerous CNN models have been created, demonstrating exceptional performance through the utilization of transfer learning. Transfer learning involves repurposing a model designed for one task as the foundation for another task. This method is particularly favoured in deep learning, where pre-existing models serve as the basis for computer vision and natural language processing endeavours. It addresses the significant computational resources and time typically needed to construct neural network models for such tasks. [15]. Research that has utilized the use of transfer learning in texture image recognition, such as cowhide texture detection using the VGG16 model with an accuracy performance of 95% [16]. Texture identification on leaves has also been carried out using ResNet50 with an accuracy of 98.17% [17]. Researchers also combine several pre-trained models to further increase performance. The TexFusionNet model, merges the AlexNet and VGG16 models by combining their final representation layers. They evaluated the model using various texture datasets including Brodatz, Columbia-Utrecht reflectance and texture database (CUReT), and textures under varying Illumination, pose and scale (KTH-TIPS). The results of their experiments demonstrate the superior performance of the TexFusionNet architecture in texture classification [18]. Nowadays, the advancement of deep learning technology in digital image processing allows us to automate the classification of dipterocarp stem images to support conservation initiatives and ecological research. We use a CNN-based two-stage transfer learning approach to obtain optimal results in categorizing the dipterocarp based on stem texture. The remainder of this paper is structured as follows: The second section describes the research method. Results and discussion are presented in the third section. In the final section, we will draw some conclusions and suggest future research.

2. METHOD

The complete stages of the research methodology are in Figure 1. This research methodology has five main stages: image acquisition, pre-processing, first training, second training and evaluation. The image acquisition is capturing or scanning an object to obtain a digital one. The image acquisition uses images from Dipterocarpaceae to obtain images in digital form. Digital images will be processed to classify Dipterocarpaceae. Next, two stages of training are carried out. The first stage of training is carried out for transfer learning until the model can quickly adapt to new tasks. The second stage is to adjust and refine the features it has learned in response to new tasks.

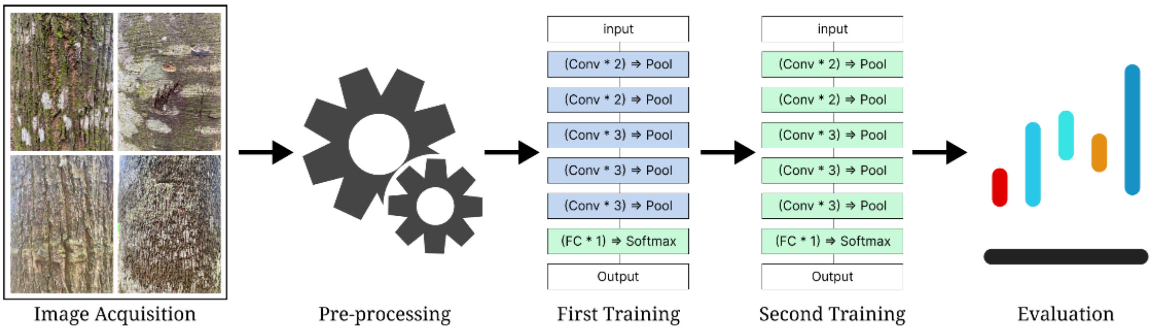


Figure 1. Research methodology

2.1. Data collection and preprocessing

We collected images from Sempaja Arboretum and BBPSILH Samarinda where foresters had identified certain trees by their wood species. These images were captured using an iPhone 11 under various weather conditions, from sunny to cloudy, and were taken from a distance of 30-60 cm away from the tree trunk. The camera was positioned with a vertical axis that differed from the image plane. An example of one of these trunk texture images can be seen in Figure 2. We collected images of 5 different *Dipterocarpaceae* species: *Dipterocarpus alatus* in Figure 2(a), *Dipterocarpus confertus* in Figure 2(b), *Dryobalanops beccarii* in Figure 2(c), *Dryobalanops lanceolata* in Figure 2(d), and *Shorea smithiana* in Figure 2(e). The number of tree datasets taken for each type is around 4 to 8 trees. This variability prevents overfitting in the future deep-learning model training. We composed the images to keep the data classes balanced. The images are divided into three separate datasets: training, validation, and testing. These datasets are allocated in a ratio of 60% for training, 20% for validation, and 20% for testing. The training and validation sets are for model training, while the testing set is for assessing model performance.


(a) (b) (c) (d) (e)

Figure 2. Trunk texture image (a) *Dipterocarpus alatus* (b) *Dipterocarpus confertus* (c) *Dryobalanops beccarii* (d) *Dryobalanops lanceolata*, and (e) *Shorea smithiana*

Table 1 displays the quantity of images allocated equally among different species. These images have been resized from their original dimensions of 3024×4032 to 750×1000. Following curation, there are a total of 857 images. Normalization of these images involves adjusting each pixel's value by subtracting the mean and dividing by the standard deviation of all pixel values. This normalization process is intended to enhance training efficiency and minimize training errors.

Table 1. Dataset distribution

Species	Training set	Validation set	Testing set
Dipterocarpus alatus	91	32	30
Dipterocarpus confertus	106	36	35
Dryobalanops beccarii	106	35	35
Dryobalanops lanceolata	102	34	35
Shorea smithiana	108	36	36
Total	513	173	171

2.2. Convolutional neural networks

Convolutional neural network (CNN) is a feedforward neural network that can extract features from data with a convolutional structure. CNN does not need to extract features manually. In addition, CNN can learn complex problems quickly due to weight sharing and more complex models, allowing for massive parallelization [19]. The standard CNN architecture comprises layers to extract the distinguishing features. The critical layers of CNN are the input layer, the convolutional layer, the pooling layer, and the fully connected layer [20]. CNN's feature extraction process involves convolution and pooling layers, while the fully connected layer does the final classification. The convolution layer is fundamental in CNN; it involves a series of mathematical operations, a special form of linear operations. The CNN architecture consists of several convolution and pooling layers repetitions, followed by one or more fully connected layers [21].

2.3. Transfer learning

Transfer learning involves reusing a model that has performed well in a particular task to solve a related task. [22]. Transfer learning seeks to enhance the effectiveness of target learners within specific domains by leveraging knowledge from related but distinct source domains. This approach reduces the reliance on extensive target-domain data for developing target learners [23]. Transfer learning holds potential to enhance the performance of target learners by tapping into external knowledge. Another perspective within transfer learning is to consider training and target data as originating from separate sub-domains but connected through a shared overarching domain [24]. Transfer learning becomes necessary when there's a shortage of target training data. This shortage could stem from limited data availability, high costs associated with collecting and labelling data, or challenges in accessing the data. As big data repositories become more common, leveraging existing datasets that are related to, though not identical with, the target domain becomes an appealing strategy for transfer learning. This approach offers several benefits, including shorter training times, enhanced neural network performance across various scenarios, and the ability to achieve good results without requiring extensive amounts of data. [25]. Typically, data is needed to train a neural model from scratch, but obtaining such data is only sometimes possible. Transfer learning becomes valuable in such cases. By training a pre-existing model, transfer learning enables the creation of practical machine learning models with relatively small training data [26].

There are a number of popular pre-trained CNN-based models available, some of them are VGG16 [11], ResNet model [12], DenseNet [13], and Inception [14]. Those models are trained on ImageNet dataset, this dataset spans 1,000 classes and contains 1,281,167 training images, 50,000 validation images and 100,000 test images [27]. These pre-trained models have achieved state-of-the-art performance on various computer vision tasks, such as image classification and object detection. Transfer learning offers a practical and efficient way to leverage pre-existing knowledge from related domains to improve the performance of machine learning models. It reduces the need for large amounts of target data and training time, making it particularly useful in scenarios where data is limited or expensive to obtain [28].

2.4. Two-stage transfer learning approach

This research employs a two-step transfer learning approach utilizing pre-trained CNN-based models such as VGG16[11], ResNet121[12], DenseNet50[13], InceptionV3[14], MobileNetV2[29], and EfficientNetV2-B0[30]. These models have been trained on the ImageNet dataset, which includes 1,000 classes. The modifications will be made to the architecture of these models, explicitly implementing a new classification layer to address dipterocarp classification. Two learning stages have been included to ensure suitability for the new classification task. In the initial stage, the previously trained feature layers of the model are frozen to prevent further updates or training. This approach aims to reconfigure the model output exclusively and train the newly added classification layer. The model quickly adapts its output to the new task while still retaining the features learned from ImageNet.

In the second stage, selectively unfreeze some or all of the convolutional layers of the model. This step aims to reintroduce these specific layers into the training process, allowing them to refine and adapt the acquired features to suit the demands of the new task. We also intentionally unfreeze certain convolutional layers to create balance. This approach allows the model to adapt to the complex task of dipterocarpus stem categorization while retaining the important features originally learned. This method empowers the model to probe deeper while leveraging its knowledge from previous training. This approach gives the model strong feature extraction capabilities from pre-trained models and effectively adapts them to dipterocarp trunk classification.

2.5. Model evaluation

The evaluation metrics we used were average accuracy and Macro F1-score. The average accuracy calculates the accuracy for each class separately and then takes the average. Average accuracy overall measures how well a classification model performs across all classes [31]. In contrast, the Macro F1-score provides a balanced measure of the model's performance to achieve good precision and recall across all classes without considering class imbalance. Macro F1-score is particularly useful when dealing with imbalanced datasets, as it ensures that the model's performance is evaluated across all classes, providing a balanced assessment of its ability to handle different types of instances [32]. The average accuracy and Macro F1-score are calculated as (1) and (2):

$$Average\ Accuracy = \frac{\sum_{i=1}^{N}(accuracy_i * support_i)}{total\ samples} \tag{1}$$

$$Macro\ F1 - Score = \frac{\sum_{i=1}^{N} F1\ Score_i}{N} \tag{2}$$

where $precision_i$ is the accuracy for the class i calculated as the number of true positive predictions for class i divided by the total number of instances predicted as class i. $Support_i$ is the number of actual instances of class i in the test set, and total samples are the total number of samples in the test set. $F1 - score_i$ is the harmonic mean of precision and recall for class i, where precision is the number of true positive predictions for class i divided by the total number of predicted instances for class i, and recall is the number of true positive predictions for class i divided by the total number of actual positive instances for class i. N represents the total number of classes in the classification problem [33].

3. RESULTS AND DISCUSSION

Our experiments utilized Python TensorFlow 2.9.2 with a NVIDIA P5000 GPU. The pre-trained model was trained with a batch size of 16 and a maximum of 50 training epochs. Early stopping was used, stopping training if the validation accuracy remained constant and unchanged for five consecutive epochs. The Adam optimizer was initially used with a learning rate of 0.001, switching to 0.0001 in the second stage to facilitate gradual changes and retain previously learned features, improving model generalization for Dipterocarpaceae stem classification. The categorical cross-entropy loss function measures the difference between the true labels and predicted probabilities. Conversely, accuracy assesses the model's performance during training, indicating the proportion of correctly predicted events. Data augmentation techniques such as random zooming, brightness adjustment, scaling, cropping, and shifting were used to augment the limited Dipterocarpaceae image dataset and improve model generalization.

A comparative analysis explores the impact of different feature layer strategies on the VGG16 model. The analysis involves freezing and thawing feature layers at different scales. Performance comparisons are performed under various scenarios, including no layers frozen, thawing all layers except the initial layer, thawing all layers completely, and thawing all layers except the last layer. Evaluation metrics assess the effectiveness and generalization ability of each strategy. Table 2 illustrates the effects of different freeze on the VGG16 model.

Table 2. VGG16 performance comparison on different strategies

Strategy	Average accuracy	Macro F1-score
No freeze the feature layer	94.118%	94.024%
Unfreeze all feature layers except the 1st block layer	97.794%	97.723%
Unfreeze all feature layers without exception	98.246%	98.183%
Unfreeze only the last block layer	95.588%	95.525%

As a comparison, we conducted the training on each pre-trained model: VGG16, InceptionV3, ResNet50, DenseNet121, MobileNetV2, and EfficientNetV2-B0. All models follow the two-stage transfer learning approach, where all convolutional layers are frozen in the first stage. In the other hand, all model feature layers are unfrozen. We unfreeze all the feature layers in the second stage for all the models based on the previous strategies experiment. Performance comparison can be seen in Table 3. In the first stage, VGG16 had the lowest accuracy, 93.907%. However, after the second stage, VGG16 became the model with the highest accuracy among six other models, namely an accuracy of 98.246% and a macro F1-score of 98.183%. ResNet50, DenseNet121, and MobileNetV2 have similar accuracy in the first stage of ~96%, with an increase in accuracy in the second stage of 1% for the ResNet50 and DenseNet121 models. However, MobileNetV2 actually experienced a reasonably high decrease in performance, a decrease in accuracy of up to 8.279%, and the EfficientNetV2-B0 model experienced a decrease in accuracy of 5.401%. These two models experienced a decrease in performance, possibly because they were small models that only had parameters of 3.5M and 7.2M, thus as the second training was carried out, it resulted in overfitting. On the other hand, in the InceptionV3 model, there is no significant change in performance from stage one to stage 2. VGG16 experienced quite a significant increase in performance due to the high parameters, there is still room for improvement during second training.

Figure 3 shows the accuracy and loss of the model for unfreezing all feature layers without exclusion. In the first stage, the model learns for 13 epochs, achieving training accuracy of ~98% and validation accuracy of ~94% can be seen in Figure 3(a) and loss at first stage can be seen in Figure 3(b). In the second stage as shown in Figures 3(c) and 3(d), there is instability in training accuracy and loss. Therefore, early stopping is appropriate to prevent overfitting/underfitting due to training instability. Figure 3(c) describes the model in the second stage reaches 100% validation accuracy at epoch 8, therefore the model stop training at epoch 13.

Table 3. Performance comparison on different models

Model	Average accuracy First stage	Average accuracy Second stage	Macro F1-score First stage	Macro F1-score Second stage	Parameters
VGG16	93.907%	98.246%	93.590%	98.183%	138.4M
InceptionV3	94.737%	94.880%	94%	94.635%	23.9M
ResNet50	96.640%	97.661%	96.544%	97.628%	25.6M
DenseNet121	96.491%	97.661%	96.605%	97.652%	8.1M
MobileNetV2	96.583%	88.304%	96.431%	86.411 %	3.5M
EfficientNetV2-B0	95.459%	90.058%	95.320%	89.581%	7.2M

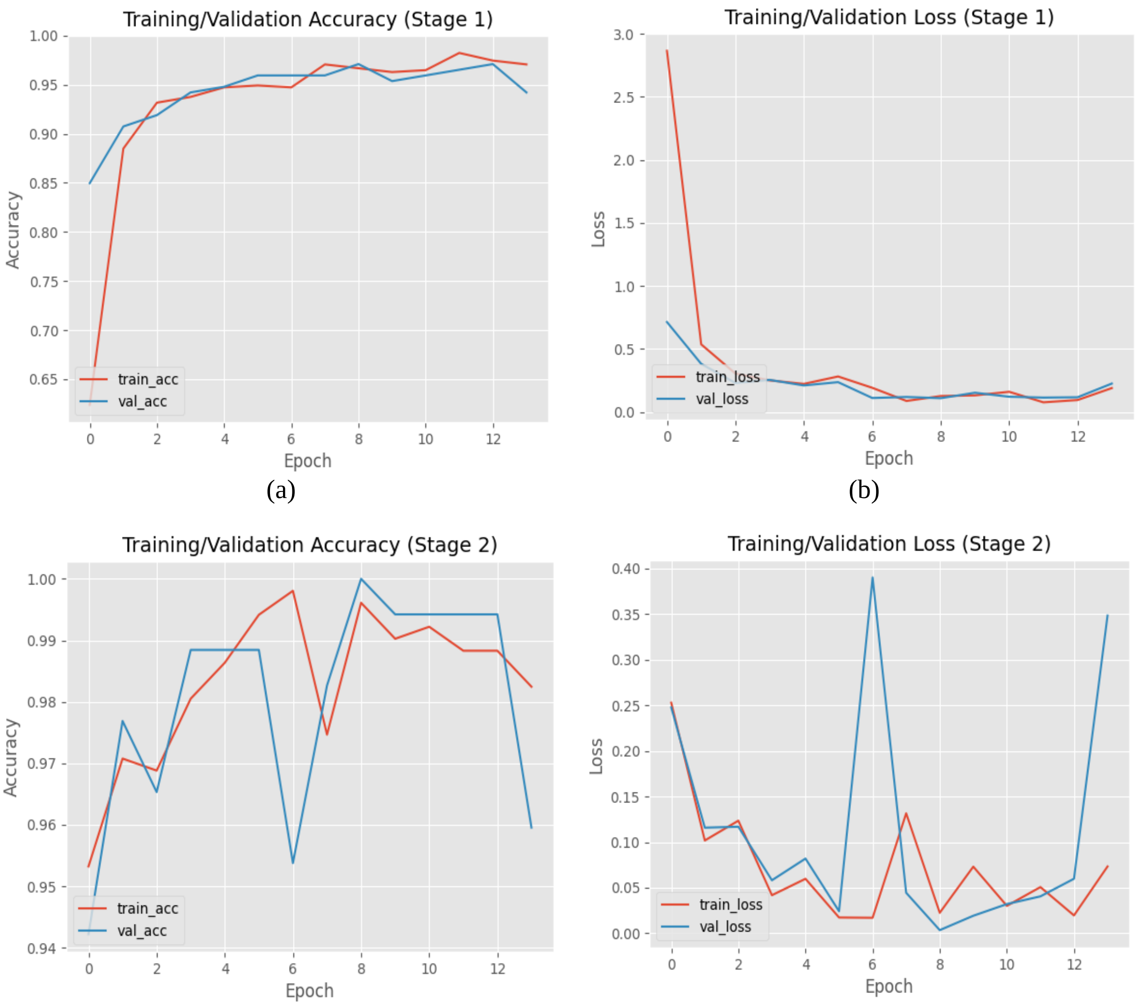


Figure 3. Accuracy and loss for two-stage transfer learning of VGG16 (a) first stage accuracy (b) loss at first stage (c) second stage accuracy (d) loss at second stage

4. CONCLUSION

Management of endemic plant classification in tropical rainforests as an effort to protect biodiversity. This study presents a two-stage transfer learning method to improve the classification of Dipterocarpaceae species based on stem texture images. This research assists conservation efforts by enabling rapid and reliable tree species identification based on trunk texture images. The results of the two-stage transfer learning VGG16 showed an increase in performance of 98.246%. However, not all models are suitable for second training; models with small parameters, such as EfficientNetV2-B0 and MobileNetV2, actually experience a decrease in performance of up to 8%. This method is suitable for models that have significant parameters where there is still a lot of room for improvement. New methods need to be developed to improve model performance with small parameters.

ACKNOWLEDGEMENTS

The express gratitude for the funding support provided in the research grant to the Directorate of Research, Technology, and Community Service (DRPM) of the Ministry of Education, Culture, Research, and Technology of the Republic of Indonesia and to the Institute for Research and Community Service (LPPM) of Mulawarman University (Contract Number 290/UN17.L1/HK/2023). The authors thank BBPSILH Samarinda and Arboretum Samarinda for supporting data collection.

REFERENCES

[1] F. Q. Brearley, L. F. Banin, and P. Saner, "The ecology of the Asian dipterocarps," *Plant Ecology & Diversity*, vol. 9, no. 5–6, pp. 429–436, 2016.

[2] M. Wati, N. Puspitasari, U. Hairah, J. A. Widians, A. Septiarini, and A. F. Tjikoa, "Borneo smart forest information system for management of dipterocarp plants in Kalimantan rainforest," in *2022 IEEE 8th Information Technology International Seminar (ITIS)*, IEEE, 2022, pp. 134–139.

[3] G. Maury-Lechon and L. Curtet, "Biogeography and evolutionary systematics of Dipterocarpaceae," *A Review of Dipterocarps: Taxonomy, Ecology and Silviculture. Center for International Forestry Research, Indonesia*, pp. 5–44, 1998.

[4] T. Cvetković, D. D. Hinsinger, D. C. Thomas, J. J. Wieringa, E. Velautham, and J. S. Strijk, "Phylogenomics and a revised tribal classification of subfamily Dipterocarpoideae (Dipterocarpaceae)," *Taxon*, vol. 71, no. 1, pp. 85–102, 2022.

[5] R. Kaewthongrach, Y. Vitasse, T. Lamjiak, and A. Chidthaisong, "Impact of severe drought during the strong 2015/2016 El Nino on the phenology and survival of secondary dry dipterocarp species in Western Thailand," *Forests*, vol. 10, no. 11, p. 967, 2019.

[6] M. Heikkilä, M. Pietikäinen, and C. Schmid, "Description of interest regions with local binary patterns," *Pattern Recognition*, vol. 42, no. 3, pp. 425–436, 2009, doi: 10.1016/j.patcog.2008.08.014.

[7] V. Remeš and M. Haindl, "Bark recognition using novel rotationally invariant multispectral textural features," *Pattern recognition letters*, vol. 125, pp. 612–617, 2019.

[8] M. Cimpoi, S. Maji, and A. Vedaldi, "Deep filter banks for texture recognition and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3828–3836.

[9] A. Y. Nawaf and W. M. Jasim, "A pre-trained model vs dedicated convolution neural networks for emotion recognition," *International Journal of Electrical and Computer Engineering*, vol. 13, no. 1, pp. 1123–1133, 2023, doi: 10.11591/ijece.v13i1.pp1123-1133.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Neural Information Processing Systems*, 2012, pp. 1106–1114.

[11] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[13] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[14] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 2818-2826, doi: 10.1109/CVPR.2016.308.

[15] J. Gupta, S. Pathak, and G. Kumar, "Deep learning (CNN) and transfer learning: a review," in *Journal of Physics: Conference Series*, IOP Publishing, 2022, p. 12029, doi: 10.1088/1742-6596/2273/1/012029.

[16] Z. Chen-McCaig, R. Hoseinnezhad, and A. Bab-Hadiashar, "Convolutional neural networks for texture recognition using transfer learning," in *2017 International Conference on Control, Automation and Information Sciences, ICCAIS 2017*, IEEE, 2017, pp. 187–192, doi: 10.1109/ICCAIS.2017.8217573.

[17] R. Zhang, Y. Zhu, Z. Ge, H. Mu, D. Qi, and H. Ni, "Transfer learning for leaf small dataset using improved ResNet50 network with mixed activation functions," *Forests*, vol. 13, no. 12, p. 2072, 2022, doi: 10.3390/f13122072.

[18] S. K. Roy, S. R. Dubey, B. Chanda, B. B. Chaudhuri, and D. K. Ghosh, "Texfusionnet: an ensemble of deep cnn feature for texture classification," in *Proceedings of 3rd International Conference on Computer Vision and Image Processing: CVIP 2018*, vol. 2, Springer, 2020, pp. 271–283.

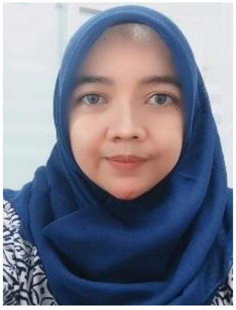
[19] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convolutional neural networks: analysis, applications, and prospects," *IEEE transactions on neural networks and learning systems*, 2021.

[20] S. Nadkarni and K. Noronha, "Breast cancer detection with multiparametric magnetic resonance imaging using deep learning," in *2023 International Conference on Smart Systems for applications in Electrical Sciences (ICSSES)*, Tumakuru, India, 2023,

pp. 1-5, doi: 10.1109/ICSSES58299.2023.10199470.
[21] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging*, vol. 9, no. 4, pp. 611–629, Jun. 2018, doi: 10.1007/s13244-018-0639-9.
[22] P. Hridayami, I. K. G. D. Putra, and K. S. Wibawa, "Fish species recognition using VGG16 deep convolutional neural network," *Journal of Computing Science and Engineering*, vol. 13, no. 3, pp. 124–130, 2019.
[23] F. Zhuang *et al.*, "A Comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, Jan. 2021, doi: 10.1109/JPROC.2020.3004555.
[24] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, pp. 1–40, 2016.
[25] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27*, Springer, 2018, pp. 270–279.
[26] C. C. Aggarwal, "Neural networks and deep learning," *Springer*, vol. 10, no. 978, p. 3, 2018.
[27] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, Dec. 2015, doi: 10.1007/s11263-015-0816-y.
[28] T. Han, C. Liu, W. Yang, and D. Jiang, "Learning transferable features in deep convolutional neural networks for diagnosing unseen machine conditions," *ISA transactions*, vol. 93, pp. 341–353, 2019.
[29] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.
[30] M. Tan and Q. Le, "Efficientnetv2: Smaller models and faster training," in *International conference on machine learning*, PMLR, 2021, pp. 10096–10106.
[31] G. Forman, "Counting positives accurately despite inaccurate classification," in *European conference on machine learning*, Springer, 2005, pp. 564–575.
[32] M. Grandini, E. Bagli, and G. Visani, "Metrics for multi-class classification: an overview," *arXiv preprint arXiv:2008.05756*, 2020.
[33] A. Tharwat, "Classification assessment methods," *Applied Computing and Informatics*, vol. 17, no. 1, pp. 168–192, Jan. 2021, doi: 10.1016/j.aci.2018.08.003.

BIOGRAPHIES OF AUTHORS

Masna Wati holds a Bachelor of Science (S.Si) in mathematics science and a master of engineering (M.T.) in informatics engineering from Hasanuddin University, Indonesia. She has lectured in the Informatics Department at Mulawarman University, Indonesia since 2014. She is a member of the Indonesia Society of Engineers and the Institute of Electrical and Electronics Engineers (IEEE). Her research areas of interest include machine learning, smart systems, and data science. She can be contacted at email: masnawati@fkti.unmul.ac.id.



Novianti Puspitasari received the B.Sc. degree in informatics engineering from the Universitas Islam Indonesia, and the M.Eng. degree in information technology from the Gadjah Mada University, Indonesia. She is currently a lecturer at the Department of Informatics, Mulawarman University, Indonesia. She is a member of the Institute of Electrical and Electronics Engineers (IEEE), Indonesian Computer, Electronics, Instrumentation Support Society (IndoCEISS), Association of Computing and Informatics Institutions Indonesia (APTIKOM) societies and the Institution of Engineers Indonesia (PII). She has authored or co-authored more than 70 publications with 5 H-index. Her research interest is in artificial intelligence, data science and analytics. She can be contacted at email: novia.ftik.unmul@gmail.com.

Ummul Hairah is member of Institute of Electrical and Electronics Engineers (IEEE), and member of Association of Computing and Informatics Institutions Indonesia (APTIKOM) societies. Currently, she is actively teaching and researching at the Department of Informatics, Mulawarman University. As a writer on several journals and conferences with more than 40 publications, she focuses her research on database, information system and artificial intelligence. She can be contacted at email: ummul.hairah@fkti.unmul.ac.id.

Joan Angelina Widians is a doctoral student in Department of Computer Science and Electronics, Gadjah Mada University, Yogyakarta, Indonesia. She received her master's degree in Department of Computer Science, Gadjah Mada University, Yogyakarta, Indonesia (2008). She received bachelor's degree in information system from STMIK Widya Cipta Dharma, East Kalimantan, Indonesia (2000). She works as a lecturer in Department of Informatics, Mulawarman University, Indonesia since 2008. She is a member of the Institute of Electrical and Electronics Engineers (IEEE). Her research interests include machine learning, text mining, and swarm intelligence. She can be contacted at email: angelwidians@unmul.ac.id.

Ade Fiqri Tjikoa is an undergraduate student in Informatics at Mulawarman University, Indonesia. During his study, he actively supports lecturers in their research and served as a laboratory assistant for one and a half years. His research interests include machine learning, deep learning, computer vision, and natural language processing. He can be contacted at email: ade.fiqri1@gmail.com.