Classification of Tasikmalaya batik motifs using convolutional neural networks

Teuku Mufizar[1], Aso Sudiarjo[1], Evi Dewi Sri Mulyani[1], Agus Ahmad Wakih[2], Muhammad Akbar Kasyfurrahman[1], Luthfi Adilal Mahbub[1]
[1]Department of Informatics Engineering, Faculty of Engineering, Perjuangan Tasikmalaya University, Tasikmalaya, Indonesia
[2]Departement of Primary Education, Faculty of Educational Sciences, Perjuangan Tasikmalaya University, Tasikmalaya, Indonesia

Article Info

Article history:

Received Sep 29, 2024
Revised Jun 9, 2025
Accepted Jul 9, 2025

Keywords:

Batik motifs
Classification
Convolutional neural networks
Computer vision
Image processing

ABSTRACT

This paper presents a study on the classification of traditional Tasikmalaya batik motifs using convolutional neural networks (CNN). The experiments revealed that the high complexity of batik motifs significantly impacted model performance, as the handling of each class influenced the overall results. Initial experiments with the original dataset demonstrated suboptimal performance, characterized by accuracy and validation curves indicating overfitting, with only 75% accuracy achieved at a learning rate of 0.001, a batch size of 32, and 50 epochs. To enhance performance, we implemented data segmentation, data augmentation, optimized the choice of the best optimizer, utilized an optimal architecture, and conducted hyperparameter tuning. The best-performing model was trained on data subjected to specific preprocessing for each class, using the Adam optimizer with hyperparameter tuning set to a learning rate of 0.001, a batch size of 32, and 50 epochs. In the hyperparameter tuning experiment with the visual geometry group network (VGGNet) architecture, it was shown that there is an improvement in the prediction of the *kumeli* class, achieving an accuracy of 100%.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

Corresponding Author:

Evi Dewi Sri Mulyani
Department of Informatics Engineering, Faculty of Engineering, Perjuangan Tasikmalaya University
PETA St, No 177, Tasikmalaya, West Java, Indonesia
Email: eviajadech@gmail.com

1. INTRODUCTION

Batik, a traditional Indonesian textile art, is renowned for its intricate designs and cultural significance, especially in the Tasikmalaya region, which is characterized by unique motifs that reflect local heritage and artistry. As globalization challenges the preservation of such cultural identities, leveraging technology becomes essential to safeguard and promote these art forms. Recent advancements in machine learning, particularly convolutional neural networks (CNNs), offer promising solutions for automating the classification of complex batik motifs, thereby enhancing the accessibility and appreciation of this cultural treasure.

Deep learning may be a kind of an artificial intelligence that has recently achieved fantastic success in classification and segmentation tasks [1]–[3], attracting a lot of attention in the treatment of numerous disorders [4]. CNN has been used in previous studies and has become popular in image classification tasks due to its ability to automatically learn hierarchical feature representations from raw data, significantly improving accuracy and efficiency [5], but in some cases research requires several techniques to improve its accuracy, including augmentation and hyperparameter techniques. In this study, based on the complexity of the image, the CNN algorithm is able to classify batik motifs. The resulting model will be optimized with augmentation and hyperparameter techniques. CNN is very effective for image processing tasks [6], including those involving color images [7], color processing including the extraction of information about the spectral properties of the object's surface and look for the best similarity of a set of descriptions which have been known to do an introduction [8]. CNNs as the classification algorithm, it specifies the sample characteristics of the acquired images, and the software developed to generate datasets of different characteristics [9].

The image classification process requires the completeness of the features of the image which form an informative image pattern so that information from the image can be displayed [10], [11]. This study aims to investigate the effectiveness of CNN architectures in classifying Tasikmalaya batik motifs. By exploring various model configurations and optimization techniques, we seek to identify the most effective approach for achieving high classification accuracy. Through this research, we hope to contribute to the preservation of cultural heritage while providing a practical tool for artisans and stakeholders in the batik industry.

2. METHOD

For the classification of Tasikmalaya batik motifs, image data has been collected from various batik artisans and shops in Tasikmalaya, including areas such as Singaparna in Tasikmalaya Regency, known for its Sukapura hand-drawn batik, as well as the city of Tasikmalaya itself. There are four distinctive Tasikmalaya batik motifs that will serve as classes in the classification: *payung*, *kumeli*, *kujang*, and *merak ngibing*. In total, there are 163 records, which are presented in the Table 1.

Figure 1 shows the research stages in classifying typical Tasikmalaya batik motifs. The diagram illustrates a systematic workflow for classifying traditional Tasikmalaya batik motifs, beginning with identifying the research problem, reviewing relevant literature, and collecting motif data from four classes. The process continues with preprocessing, including resizing images, segmenting motifs using methods like canny edge detection and thresholding, and augmenting data through techniques such as random cropping, rotation, flipping, affine transformation, and padding. Once prepared, the image data is used to train and validate a CNN, with performance evaluated using a confusion matrix. The model is then optimized by selecting the best optimizer (such as Adam or stochastic gradient descent (SGD)), refining the CNN architecture, and tuning hyperparameters like learning rate, batch size, and epoch count. Finally, model performances are compared, conclusions are drawn about the most effective approach, and the classification system for Tasikmalaya batik motifs is completed. In Table 1 is the dataset generated from the image digitization process based on 4 classes of batik motifs.

Table 1. Dataset processing result

Class	Record
Payung	24
Kumeli	50
Kujang	23
Merak Ngibing	66

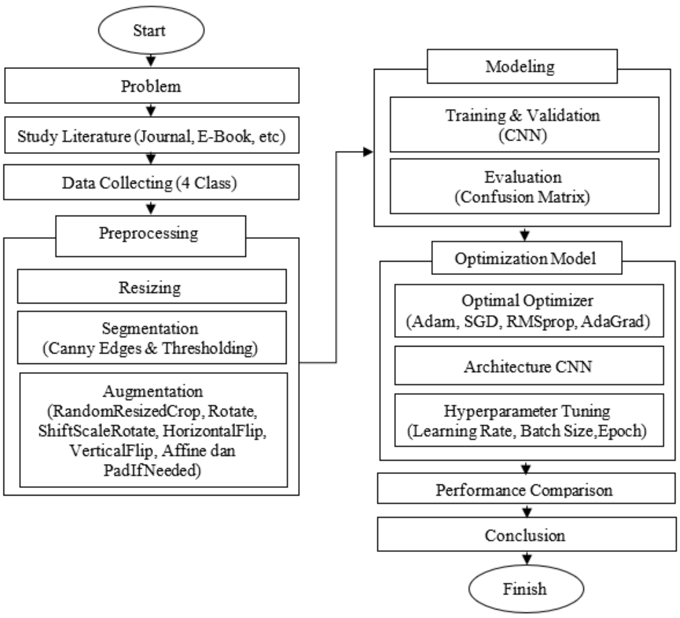


Figure 1. Experimental methods

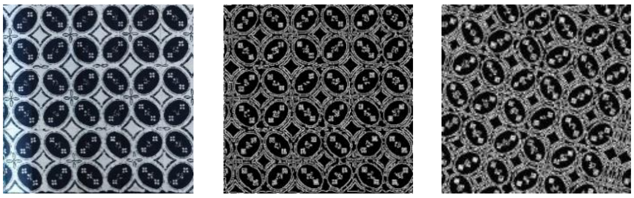
3. RESULTS AND DISCUSSION
3.1. Preprocessing
3.1.1. Resizing

The purpose of resizing images to 150×150 pixels during the preprocessing stage is to standardize the input size, ensuring that all images have uniform dimensions for consistent processing by the model. This uniformity is crucial, as most deep learning architectures require fixed input shapes. Resizing also reduces computational load and speeds up training by minimizing memory usage, allowing the model to iterate through the dataset more efficiently [12], [13]. Additionally, maintaining a consistent image size enhances model performance by enabling it to focus on the essential features of batik motifs without being distracted by variations in size and shape, ultimately contributing to more accurate classification outcomes.

3.1.2. Segmentation and augmentation

In the data preprocessing phase of our batik motif classification project, we implemented image data segmentation as a crucial step. This process involved organizing the image dataset into distinct categories, facilitating effective training and evaluation of the machine learning model. The purpose of using data segmentation is to enhance feature extraction, improve classification accuracy, and facilitate better data representation [14], [15]. The techniques used for this batik data are thresholding and canny edge detection.

Deep learning models require large data sets to recognize images accurately, data augmentation techniques can be applied to expand the dataset by modifying existing images to increase data diversity [16], [17]. The next stage is to perform data augmentation. The meaningful data augmentation can accomplish the highest accuracy with a lower error rate on all datasets by using transfer learning models [18], [19]. This process is crucial for recognizing rich and complex patterns, such as batik motifs, with the aims of increasing data variability, reducing overfitting, and improving model robustness. The intentions behind this are to strengthen learning, enhance classification accuracy, and optimize the use of limited datasets. The data augmentation techniques applied include RandomResizedCrop(p=0.1), Rotate(limit=10, p=0.5), ShiftScaleRotate(shift_limit=0.1, scale_limit=0.2, rotate_limit=30, p=0.7), HorizontalFlip(p=0.5), VerticalFlip(p=0.5), Affine(shear=20, p=0.5), and PadIfNeeded(min_height=300, min_width=300, border_mode=0, value=0, p=0.1). Figure 2 shows the results of data segmentation and data augmentation for original image in Figure 2(a), after segmentation in Figure 2(b), and after segmentation and augmentation in Figure 2(c).



(a) (b) (c)

Figure 2. Results of data segmentation and data augmentation of (a) original image, (b) after segmentation, and (c) after segmentation and augmentation

Batik motifs exhibit a diverse range of patterns, making the classification process complex. Previous studies have discussed how image size, image quality, and pattern characteristics affect the classification of batik [20]. This finding is also evident in this research, which concludes that achieving good performance requires specific treatments for each batik motif to ensure optimal model performance. The treatment of each motif during the preprocessing stage is illustrated in the Table 2.

Table 2. Dataset processing result

Class	Original	Segmentation	Augmentation
Payung	√	√	√
Kumeli	√	√	√
Kujang	-	√	
Merak Ngibing	√	-	-

Figure 3 illustrates the outcomes of applying different combinations of data segmentation and data augmentation techniques. Figure 3(a) shows the performance results using segmentation and augmentation. Figure 3(b) shows the performance results using augmentation without segmentation. Figure 3(c) shows the performance results using segmentation and without augmentation. Figure 3(d) shows performance results without segmentation and without augmentation.

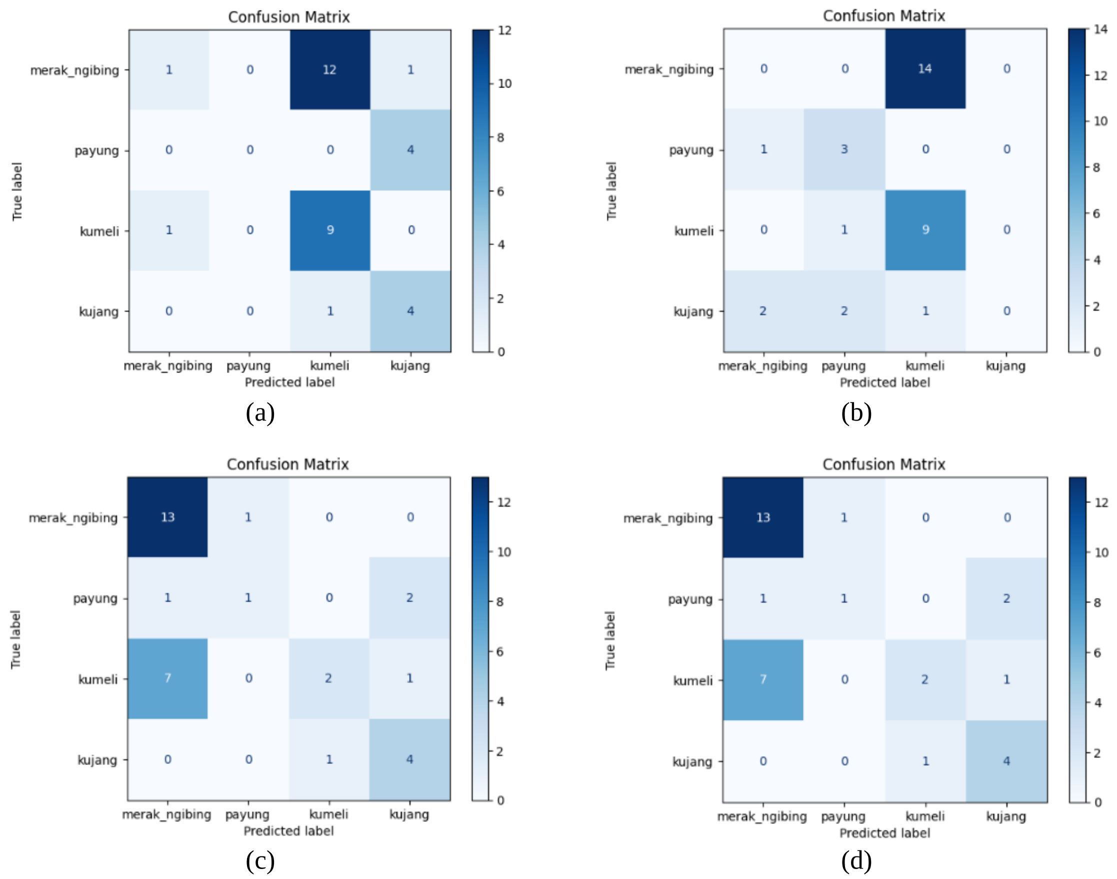


Figure 3. Performance results of data segmentation and data augmentation of (a) with segmentation and with augmentation, (b) without segmentation and with augmentation, (c) with segmentation and without augmentation and (d) without segmentation and without augmentation

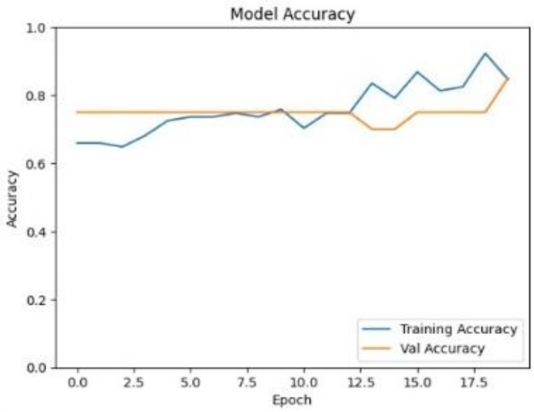
3.2. Modeling
3.2.1. Training and validation

The data training process involves splitting the dataset into two parts: training data and testing data, with a ratio of 80% for training and 20% for testing. In the modeling phase, the first training process uses the original dataset without segmentation and augmentation, employing a CNN (optimizer: Adam, learning rate: 0.001, batch size: 32, and epochs: 20). The architecture consists of three convolutional layers (32, 64, 128), three max pooling layers (2, 2), and one dense layer (128).

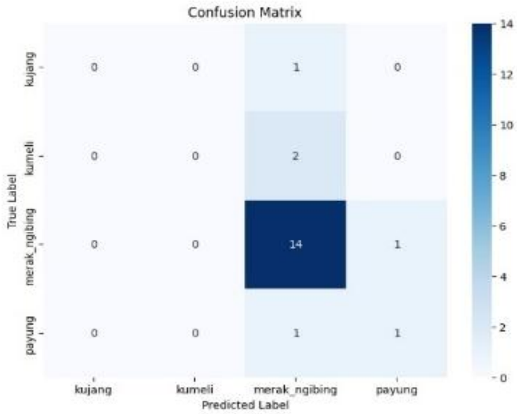
The model achieved an accuracy of 75%. However, the accuracy graph shows that training accuracy increases while validation accuracy decreases as the epochs progress, indicating that the model is likely experiencing overfitting. To improve the model's performance, it is necessary to make some adjustments. Among deep learning types, CNN are the most common types of deep learning models utilized for medical image diagnosis and analysis. However, CNN suffers from high computation cost to be implemented and may require to adapt huge number of parameters [21], to enhance the model's performance, data will be processed using segmentation and augmentation techniques. Secondly, the selection of a CNN optimizer will be conducted, and thirdly, hyperparameter tuning will be performed to further improve the model's performance [22].

3.3. Optimazation model
3.3.1. Optimizer comparison

From the experiments conducted with four optimizers (Adam, SGD, root mean square propagation (RMSprop), and adaptive gradient algorithm (AdaGrad)) using a learning rate of 0.001, a batch size of 32, and 50 epochs, the image above shows that all four optimizers perform effectively. The graphs of model accuracy and validation accuracy indicate that both training and validation accuracies increase as the epochs progress. Figure 4 presents the performance results obtained using the original dataset, specifically illustrating the model accuracy in Figure 4(a) and the confusion matrix in Figure 4(b).



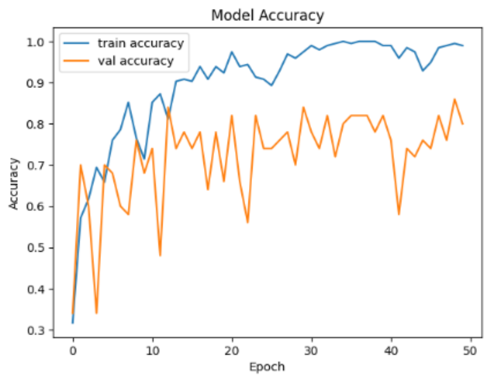
(a)



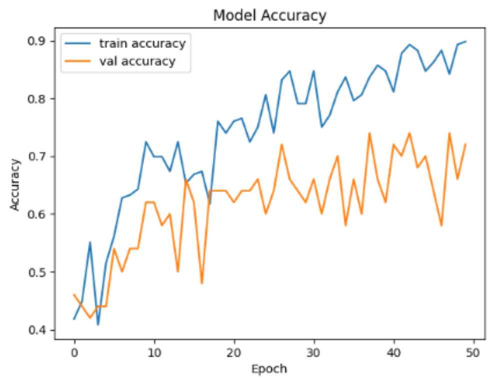
(b)

Figure 4. Performance result with original data of (a) model accuracy (b) confusion matrix

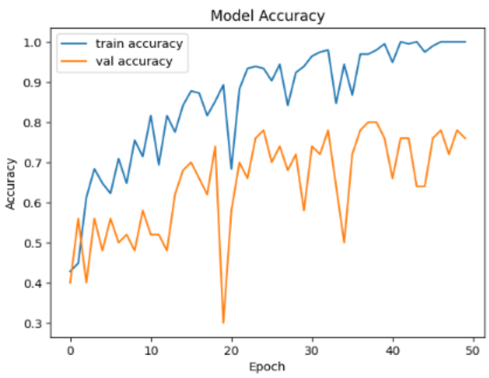
Figure 5 shows the result of model accuracy using 4 optimizers, namely Adam optimizer in Figure 5(a), SGD optimizer in Figure 5(b), RMSprop optimizer in Figure 5(c), and AdaGrad optimizer in Figure 5(d). Next, we tested the model's performance using the four optimizers. From the confusion matrix shown, we found that the highest performing model used the Adam optimizer, which achieved 80% accuracy. Figure 6 is the result of the confusion matrix using 4 optimizers, namely Adam optimizer in Figure 6(a), SGD optimizer in Figure 6(b), RMSprop optimizer in Figure 6(c), and AdaGrad optimizer in Figure 6(d).



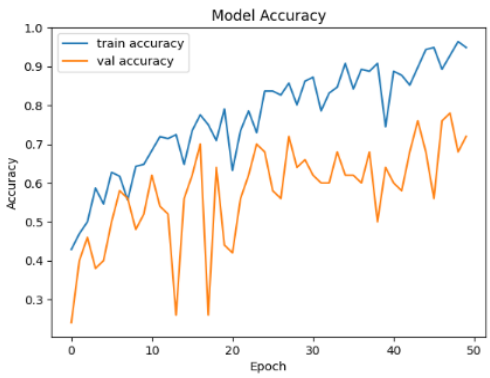
(a)



(b)



(c)



(d)

Figure 5. Model accuracy with 4 optimizers of (a) Adam, (b) SGD, (c) RMSprop, and (d) AdaGrad

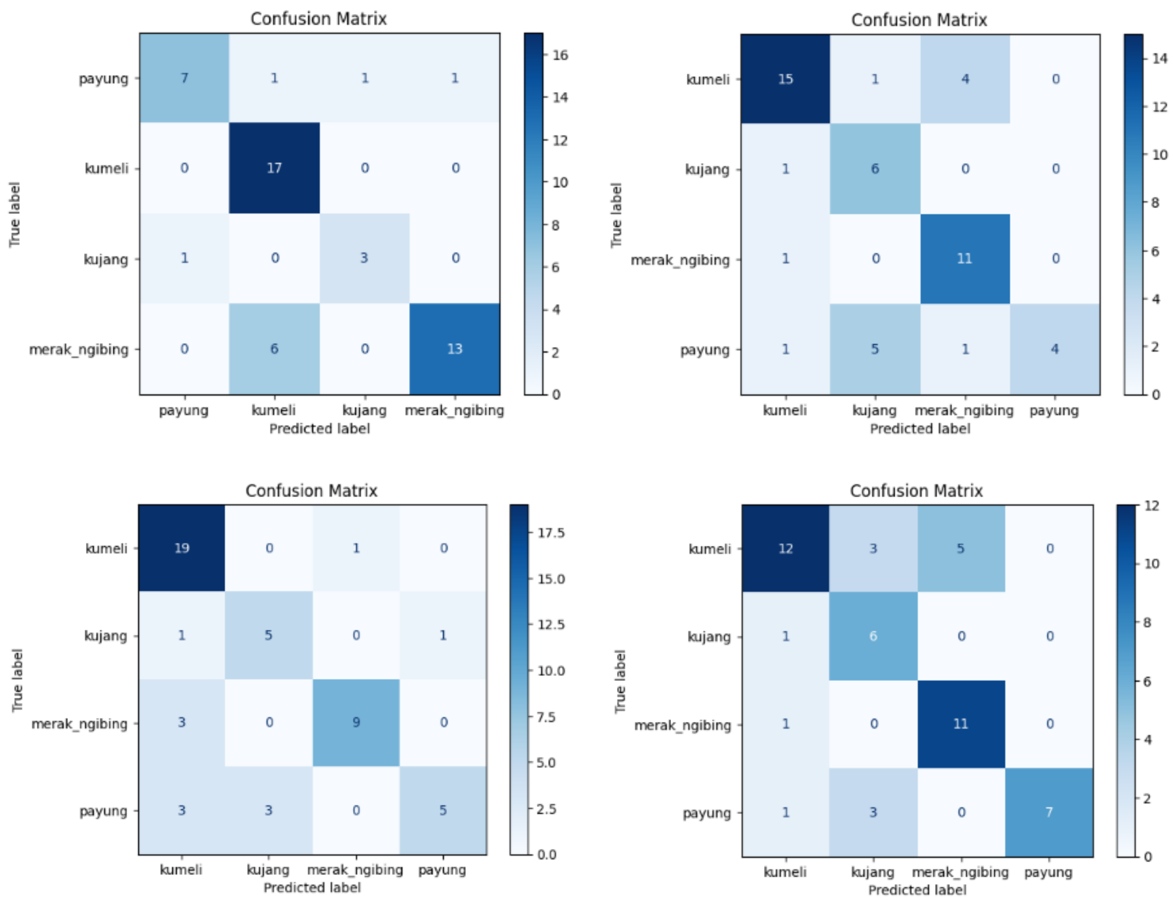


Figure 6. Confusion matrix with 4 optimizers of (a) Adam, (b) SGD, (c) RMSprop, and (d) AdaGrad

3.3.2. Architecture comparison

CNNs have gained remarkable success on many images' classification tasks in recent years. However, the performance of CNNs highly relies upon their architectures [23]. The next step to improve the model's performance is to select the best CNN architecture, In the CNN architecture, an average-pooling layer and a max-pooling layer are connected in parallel in order to boost classification performance [24], for these batik motifs. In this experiment, three architectures will be tested, detailed as in Table 3.

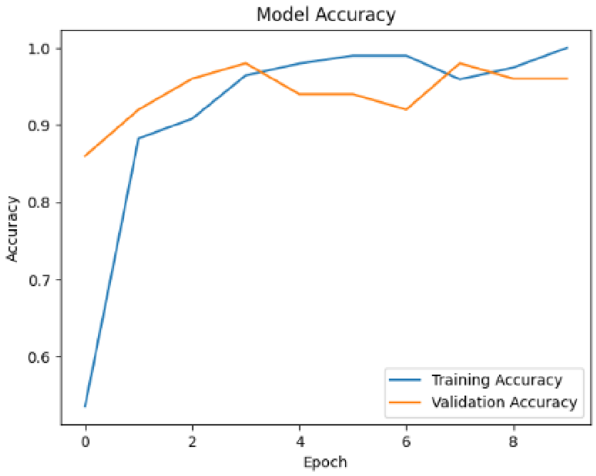
Table 3. Three architectures model CNN

Model	Optomizer	Learning rate	Batch size	Epoch
VGGNet	Adam	0.001	32	10
ResNet	Adam	0.001	32	10
GoogLeNet	Adam	0.001	32	10

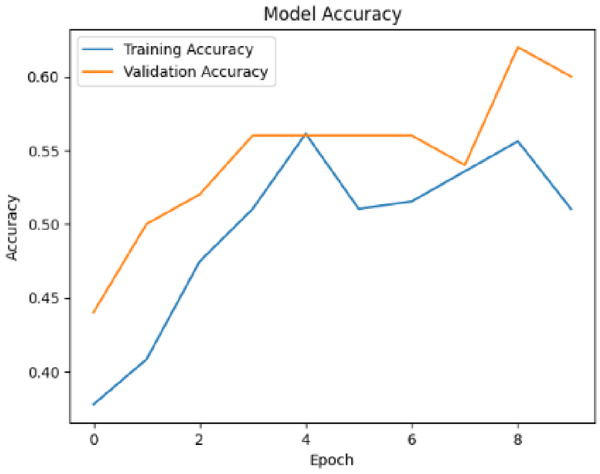
Based on the experiments with three CNN architecture (visual geometry group network (VGGNet), residual network (ResNet), and GoogLeNet), we found the graph shows that the accuracy and validation accuracy of each model increase as the epochs progress, indicating that the models demonstrate good performance. However, hyperparameter tuning is needed to controlling the learning process, preventing overfitting/underfitting and improving accuracy and generalization. Without proper hyperparameter tuning, a model may fail to achieve optimal performance, even if the algorithm itself is advanced.

3.3.3. Tunning hyperparameter

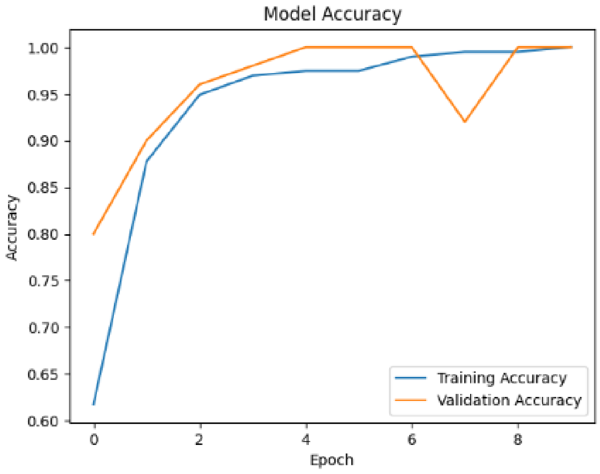
Hyperparameter tuning is the process of optimizing the settings of hyperparameters, which are parameters not learned by the model during training. Hyperparameter tuning is essential in training such models and significantly impacts their final performance and training speed [25]. Figure 7 shows the result of model accuracy using the Adam optimizer: VGGNet optimizer in Figure 7(a), ResNet optimizer in Figure 7(b), and GoogleNet optimizer in Figure 7(c). Figure 8 shows the confusion matrix of each architecture using the Adam optimizer: VGGNet architecture in Figure 8(a), ResNet architecture in Figure 8(b), and GoogLeNet architecture in Figure 8(c).



(a)

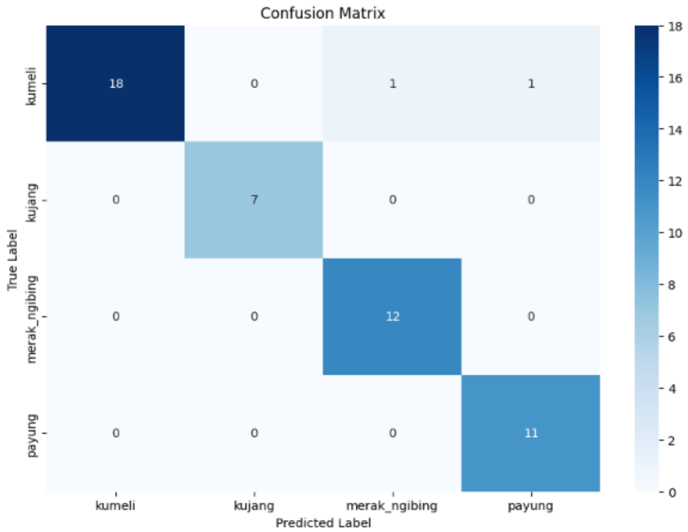


(b)

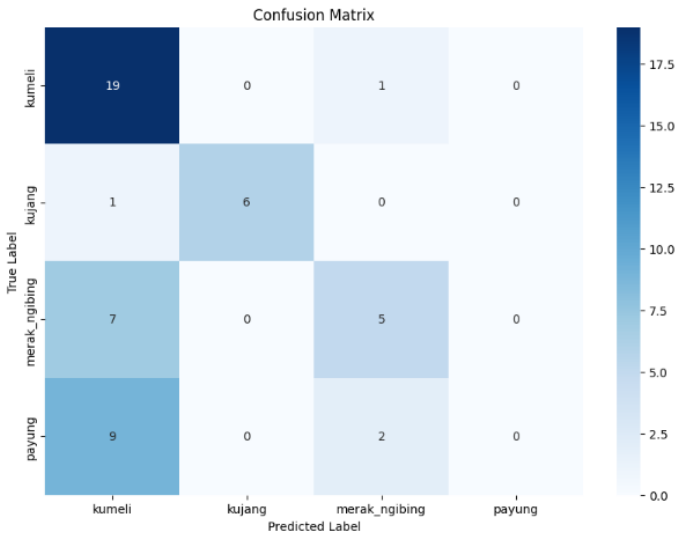


(c)

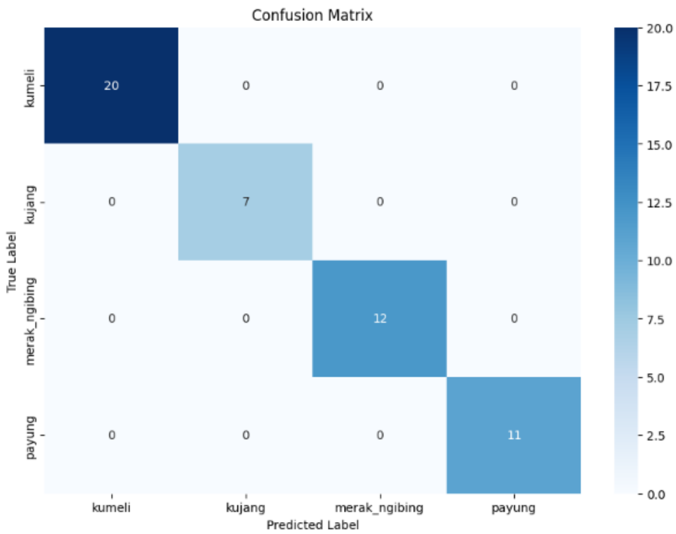
Figure 7. Model accuracy results using Adam optimizer of (a) VGGNet, (b) ResNet, and (c) GoogLeNet



(a)



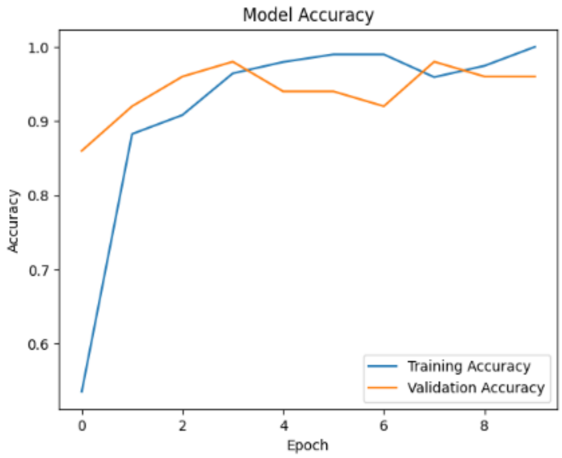
(b)



(c)

Figure 8. Confusion matrix using Adam optimizer of (a) VGGNet, (b) ResNet, and (c) GoogLeNet

Figure 9 shows the results of hyperparameter tuning on the VGGNet architecture by increasing the number of epochs to 50 for 1st experiment in Figure 9(a) and 2nd experiment in Figure 9(b). It can be observed that the accuracy graph indicates improvement, with both training accuracy and validation accuracy increasing closely together as the number of epochs increase. This demonstrates that the model performs well.

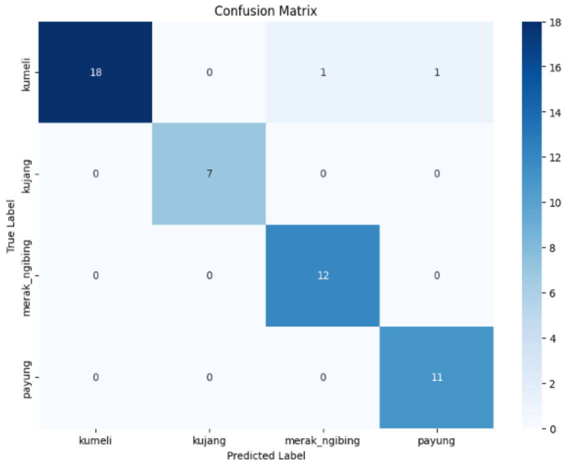


(a)

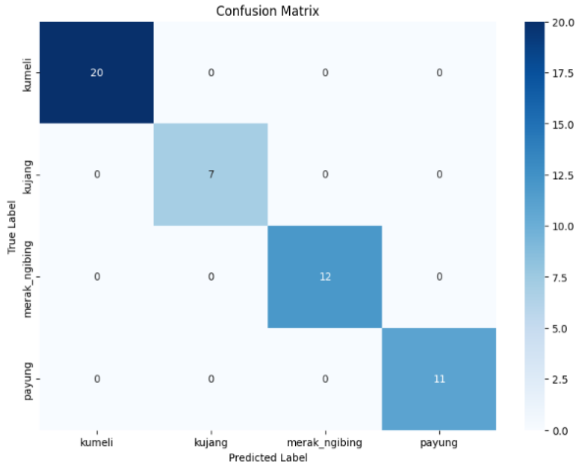
(b)

Figure 9. Model accuracy results tunning hyperparameter (a) 1st experiment and (b) 2nd experiment

Figures 10 shows the results of the confusion matrix from the hyperparameter tuning experiment on the VGGNet architecture. The number of epochs was increased to 50 for 1st experiment in Figure 10(a) and 2nd experiment in Figure 10(b). The results indicate that the model performs well, with an improvement in the prediction of the *kumeli* class.



(a)



(b)

Figure 10. Confusion matrix results tunning hyperparameter (a) 1st experiment and (b) 2nd experiment

3.4. Performance comparison

After conducting experiments to improve model performance, the comparison results can be determined. The best-performing model was achieved using the Adam optimizer with hyperparameters set to a learning rate of 0.001, a batch size of 32, and 50 epochs, resulting in an accuracy of 80%. The performance results for the Adam optimizer can be seen in Table 4. Table 5 shows the accuracy values for each optimizer.

Table 4. The performance of optimizer Adam

Class	Precision	Recall	F1-score	Support
Payung	0.88	0.70	0.78	10
Kumeli	0.71	1.00	0.83	17
Kujang	0.75	0.75	0.75	4
Merak ngibing	0.93	0.68	0.79	19

Table 5. Accuracy optimizer

Optimizer	Accuracy (%)
Adam	80
SGD	72
RMSprop	76
AdaGrad	72

Table 6 shows the accuracy values from experiments using three architectures: VGGNet, ResNet, and GoogLeNet. The highest accuracy was achieved GoogLeNet with the Adam optimizer, reaching 100%. However, hyperparameter tuning is needed to further improve performance, because the accuracy graph shows fluctuations between training accuracy and validation accuracy. This leads to the model being less effective and inconsistent.

Table 6. Accuracy of models using Adam optimizer

Model	Learning rate	Batch size	Epoch	Accuracy (%)
VGGNet	0.001	32	10	96
ResNet	0.001	32	10	60
GoogLeNet	0.001	32	10	100

Table 7 displays the accuracy values from the experiments with the three architectures. It can be concluded that the GoogLeNet architecture achieved the highest accuracy. However, hyperparameter tuning is needed to further improve performance. Table 7 shows the accuracy values from three types of architectures tested using the Adam optimizer, which is considered the best optimizer and GoogLeNet which is considered the best architecture as the accuracy remained at 100%, The proposed method in this study tends to have a much higher accuracy proportion than other architectures. According to our study, lower accuracy does not necessarily indicate poor model performance in classification. The proposed optimization technique can potentially improve accuracy with the available dataset. This study tested the performance of a comprehensive CNN model with the optimized model. However, more thorough research may be needed to validate its accuracy, especially in relation to the limitations of the dataset.

Table 7. Accuracy of tunning parameter GoogLeNet architecture

Model	Learning rate	Batch size	Epoch	Accuracy (%)
1st experiment	0.001	32	50	100
2nd experiment	0.001	32	50	100

4. CONCLUSION

Our findings provide conclusive evidence that experiments conducted on batik motif classification using CNN reveal that the high complexity of batik motifs significantly affects model performance, as the handling of each class affects the overall results. Initial experiments with the original dataset showed suboptimal model performance, characterized by accuracy and validation curves indicating overfitting, achieving only 75% accuracy with a learning rate of 0.001, a batch size of 32, and 50 epochs. However, by employing data segmentation, data augmentation, selecting the best optimizer, utilizing an optimal architecture, and tuning hyperparameters, model performance improved significantly. The best model was obtained by training on data that underwent specific preprocessing for each class, using the Adam optimizer with hyperparameter tuning set to a learning rate of 0.001, a batch size of 32, and 50 epochs. In the hyperparameter tuning experiment with the VGGNet architecture, it was shown that there is an improvement in the prediction of the *kumeli* class, achieving an accuracy of 100%. Our study shows that optimization techniques on CNN model performance can be better and more accurate than before. Future studies can develop other features in batik and explore feasible methods to produce the best model performance.

ACKNOWLEDGEMENTS

The authors would like to their appreciation to Universitas Perjuangan Tasikmalaya for the continuous support and facilitation throughout the research process.

FUNDING INFORMATION

This research was funded by the Directorate of Research, Technology, and Community Service (DRTPM), Ministry of Education, Culture, Research, and Technology of the Republic of Indonesia, through the 2024 Fundamental Research Grant Scheme (Regular Scheme) under contract numbers 106/E5/PG.02.00.PL/2024, 047/SP2H/RT-MONO/LL4/2024, and 020/KP/LP2M-UP/06/2024.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Teuku Mufizar	✓	✓		✓	✓	✓	✓	✓	✓	✓			✓	✓
Aso Sudiarjo	✓	✓			✓	✓			✓	✓	✓			
Evi Dewi Sri Mulyani	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓		✓
Agus Ahmad Wakih	✓	✓			✓	✓		✓	✓	✓				
Muhammad Akbar Kasyfurrahman	✓	✓			✓	✓		✓	✓	✓				
Luthfi Adilal Mahbub	✓	✓			✓	✓		✓	✓	✓				

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

INFORMED CONSENT

We have obtained informed consent from all individuals included in this study.

ETHICAL APPROVAL

The research related to animal use has been complied with all the relevant national regulations and institutional policies for the care and use of animals.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, [EDSM], upon reasonable request.

REFERENCES

[1] S. Kumar and D. Kumar, “Human brain tumor classification and segmentation using CNN,” *Multimedia Tools and Applications*, vol. 82, no. 5, pp. 7599–7620, 2023, doi: 10.1007/s11042-022-13713-2.

[2] P. I. R. Jenifer and S. Kannan, “Deep learning with optimal hierarchical spiking neural network for medical image classification,” *Computer Systems Science and Engineering*, vol. 44, no. 2, pp. 1081–1097, 2023, doi: 10.32604/csse.2023.026128.

[3] J. W. Tan, S. W. Chang, S. Abdul-Kareem, H. J. Yap, and K. T. Yong, “Deep learning for plant species classification using leaf vein morphometric,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 17, no. 1, pp. 82–90, 2020, doi: 10.1109/TCBB.2018.2848653.

[4] V. Anand, S. Gupta, D. Koundal, and K. Singh, “Fusion of U-Net and CNN model for segmentation and classification of skin lesion from dermoscopy images,” *Expert Systems with Applications*, vol. 213, 2023, doi: 10.1016/j.eswa.2022.119230.

[5] H. Gao, B. Cheng, J. Wang, K. Li, J. Zhao, and D. Li, "Object classification using CNN-based fusion of vision and LIDAR in autonomous vehicle environment," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 4224–4230, 2018, doi: 10.1109/TII.2018.2822828.

[6] M. Polsinelli, L. Cinque, and G. Placidi, "A light CNN for detecting COVID-19 from CT scans of the chest," *Pattern Recognition Letters*, vol. 140, pp. 95–100, 2020, doi: 10.1016/j.patrec.2020.10.001.

[7] R. Karthik, M. Hariharan, S. Anand, P. Mathikshara, A. Johnson, and R. Menaka, "Attention embedded residual CNN for disease detection in tomato leaves," *Applied Soft Computing Journal*, vol. 86, pp. 1-26, Jan. 2020, doi: 10.1016/j.asoc.2019.105933.

[8] E. D. S. Mulyani and J. P. Susanto, "Classification of maturity level of fuji apple fruit with fuzzy logic method," *2017 5th International Conference on Cyber and IT Service Management (CITSM),* Denpasar, Indonesia, 2017, pp. 1-4, doi: 10.1109/CITSM.2017.8089294.

[9] L. D. Medus, M. Saban, J. V. Francés-Víllora, M. Bataller-Mompeán, and A. Rosado-Muñoz, "Hyperspectral image classification using CNN: application to industrial food packaging," *Food Control*, vol. 125, Jul. 2021, doi: 10.1016/j.foodcont.2021.107962.

[10] Z. N. K. Swati *et al.*, "Brain tumor classification for MR images using transfer learning and fine-tuning," *Computerized Medical Imaging and Graphics*, vol. 75, pp. 34–46, 2019, doi: 10.1016/j.compmedimag.2019.05.001.

[11] M. Zhang, W. Li, Q. Du, L. Gao, and B. Zhang, "Feature extraction for classification of hyperspectral and LiDAR data using patch-to-patch CNN," *IEEE Transactions on Cybernetics*, vol. 50, no. 1, pp. 100–111, Jan. 2020, doi: 10.1109/TCYB.2018.2864670.

[12] R. Qin, L. Huangfu, D. Hood, J. Ma, and S. Huang, "Kernel inversed pyramidal resizing network for efficient pavement distress recognition," in *Neural Information Processing (ICONIP 2022)*, Singapore: Springer, Apr. 2023, pp. 302–312, doi: 10.1007/978-981-99-1645-0_25.

[13] H. Alhichri, "CNN ensemble approach to detect COVID-19 from computed tomography chest images," *Computers, Materials & Continua*, vol. 67, no. 3, pp. 3581–3599, 2021, doi: 10.32604/cmc.2021.015399.

[14] F. Gu, H. Zhu, H. Wang, Y. Zhang, F. Zuo, and S. Ablameyko, "Monitoring of egg growing in video by the improved DeepLabv3+network model," *Pattern Recognition and Image Analysis*, vol. 34, no. 2, pp. 288–298, 2024, doi: 10.1134/S1054661824700081.

[15] J. Li *et al.*, "Next-ViT: next generation vision transformer for efficient deployment in realistic industrial scenarios," *arXiv-Computer Science*, vol. 1, pp. 1–14, 2022.

[16] D. Tellez *et al.*, "Quantifying the effects of data augmentation and stain color normalization in convolutional neural networks for computational pathology," *Medical Image Analysis*, vol. 58, 2019, doi: 10.1016/j.media.2019.101544.

[17] M. Sajjad, S. Khan, K. Muhammad, W. Wu, A. Ullah, and S. W. Baik, "Multi-grade brain tumor classification using deep CNN with extensive data augmentation," *Journal of Computational Science*, vol. 30, pp. 174–182, 2019, doi: 10.1016/j.jocs.2018.12.003.

[18] Z. Mushtaq, S. F. Su, and Q. V. Tran, "Spectral images based environmental sound classification using CNN with meaningful data augmentation," *Applied Acoustics*, vol. 172, pp. 1–15, 2021, doi: 10.1016/j.apacoust.2020.107581.

[19] A. Waheed, M. Goyal, D. Gupta, A. Khanna, F. Al-Turjman, and P. R. Pinheiro, "CovidGAN: data augmentation using auxiliary classifier GAN for improved Covid-19 detection," *IEEE Access*, vol. 8, pp. 91916–91923, 2020, doi: 10.1109/ACCESS.2020.2994762.

[20] N. D. Girsang, "Literature study of convolutional neural network algorithm for batik classification," *Brilliance: Research of Artificial Intelligence*, vol. 1, no. 1, pp. 1–7, 2021, doi: 10.47709/brilliance.v1i1.1069.

[21] D. Albashish, R. Al-Sayyed, A. Abdullah, M. H. Ryalat, and N. Ahmad Almansour, "Deep CNN model based on VGG16 for breast cancer classification," *2021 International Conference on Information Technology (ICIT)*, Amman, Jordan, 2021, pp. 805-810, doi: 10.1109/ICIT52682.2021.9491631.

[22] M. Ula, Muhathir, and I. Sahputra, "Optimization of multilayer perceptron hyperparameter in classifying pneumonia disease through x-ray images with speeded-up robust features extraction method," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 10, pp. 203–210, 2022, doi: 10.14569/IJACSA.2022.0131025.

[23] Y. Sun, B. Xue, M. Zhang, G. G. Yen, and J. Lv, "Automatically designing CNN architectures using the genetic algorithm for image classification," *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 3840–3854, Sep. 2020, doi: 10.1109/TCYB.2020.2983860.

[24] F. Demir, A. M. Ismael, and A. Sengur, "Classification of lung sounds with CNN model using parallel pooling structure," *IEEE Access*, vol. 8, pp. 105376–105383, 2020, doi: 10.1109/ACCESS.2020.3000111.

[25] W. Wang, Y. Pei, S.-H. Wang, J. Manuel Gorrz, and Y.-D. Zhang, "PSTCNN: Explainable COVID-19 diagnosis using PSO-guided self-tuning CNN," *Biocell*, vol. 47, no. 2, pp. 373–384, 2023, doi: 10.32604/biocell.2023.025905.

BIOGRAPHIES OF AUTHORS

Teuku Mufizar master's degree graduate in Information System from Indonesian Computer University (UNIKOM), Bandung, in 2012. He is a lecturer in the Department of Informatics Engineering, Faculty of Engineering, Perjuangan Tasikmalaya University. His research interests are in information system, data science, machine learning, computer vision, and expert system. He can be contacted at email: fizargama@gmail.com or teukumufizar@unper.ac.id.

Aso Sudiarjo master's degree graduate in Computer Science from Budi Luhur University, Jakarta, in 2015. He is a lecturer in the Department of Informatics Engineering, Faculty of Engineering, Perjuangan Tasikmalaya University. His research interests are in information system, decision support system, and data science. He can be contacted at email: asosudiarjo@unper.ac.id.

Evi Dewi Sri Mulyani master's degree graduate in Computer Engineering from Dian Nuswantoro University, Semarang, in 2012. She is a lecturer in the Department of Informatics Engineering, Faculty of Engineering, Perjuangan Tasikmalaya University. Her research interests are in machine learning, data mining, computer vision, expert system, and data science. She can be contacted at email: eviajadech@gmail.com or evidewi@unper.ac.id.

Agus Ahmad Wakih is an Assistant Expert Lecturer at the Elementary School Teacher Education Program (PGSD), Perjuangan Tasikmalaya University. He holds both undergraduate and master's degrees in arts and social sciences and is actively involved in educational research, particularly in fields related to primary education, traditional arts, and cultural literacy. His scholarly portfolio includes works on mathematics learning difficulties, the integration of traditional games into physical education, art education, and the use of technology in cultural heritage, often focusing on the Tasikmalaya region and early childhood or elementary students. His research contributions have been published in various academic journals, and he has also taken part in community service and internal research grant projects at Perjuangan Tasikmalaya University. He can be contacted at email: aweagus67@gmail.com.

Muhammad Akbar Kasyfurrahman currently pursuing an undergraduate degree in the Department of Informatics Engineering, Faculty of Engineering, Perjuangan Tasikmalaya University. His areas of expertise include HTML5, CSS3, Python, MySQL, Bootstrap, and Tailwind CSS. He can be contacted at email: 2103010031@unper.ac.id.

Luthfi Adilal Mahbub received Bachelor of Informatics from Universitas Perjuangan with expertise in Android app development using Kotlin and Flutter, as well as website development. In addition, he also mastered UI/UX design to create intuitive, functional, and visually appealing interfaces. He can be contacted at email: 2103010067@unper.ac.id.