***ICIC Express Letters***　　　 　　　　　　　 ICIC International ⓒ2020 ISSN 1881-803X

Volume **17**, Number **12**, Dec **2023** pp. **1–8**

**BATIKNET: BATIK CLASSIFICATION BASED MANAGEMENT APPLICATION FOR INEXPERIENCED USER**

Muhammad Taufik Dwi Putra1,\*, Hilmil Pradana2, Munawir1, Deden Pradeka1, Ana Rahma Yuniarti1, Jafar Sadik3, Muhammad Andhika R1

1Departement of Computer Engineering, Indonesia University of Education, Indonesia.

{tdputra, munawir, dedenpradeka, anarahmay, muh.dhkar}@upi.edu

2Departement of Informatics, Sepuluh November Institute of Technology, Indonesia. hilmi@its.ac.id

3Departement of Informatics, Datokarama State Islamic University, Indonesia. jafar.sadik001@binus.ac.id

\*Corresponding author: tdputra@upi.edu.

Received November 2023; accepted December 2023

Abstract. In this modern stylist era, *batik has significantly contributed to the Indonesian economy and is diverse and spread throughout cities. Each batik pattern has a different meaning and philosophy of life and ancestral heritage and reflects the region where the batik was made. Hence, many factors can influence the type of batik; this research focused on studying the various types and the factors to determine the correct type. Firstly, we introduce our new batik dataset containing five types of batik patterns: Kawung, Megamendung, Parang, Sekarjagad, and Truntum, where this data will have a significant effect on future studies. Additionally, our dataset is used together with EfficientNetV2 as feature extractor to categorize batik pattern according to the input images. The training process results obtained an accuracy of 98% for training, 97% for validation, and 96% for testing. For research applicable to real applications, User Interface (UI) is introduced using RESTful architecture as the API, which is useful for the user without experience in computer science.*

**Keywords:** Batik, Deep Learning, Convolutional Neural Network, EfficientNet, REST.

1. **Introduction.** Batik is a significant cultural and economic asset to Indonesia in the modern stylist era. Its diverse and widespread presence across the country reflects the rich tapestry of traditions in Indonesia. Each batik pattern carries deep-rooted meaning, embodying the philosophy of life and ancestral heritage passed down through generations. The regions where these patterns originate further enhance their significance, providing a tangible link to the cultural heritage of Indonesia [1]. On October 2, 2009, UNESCO designated batik as an Indonesian or intangible cultural heritage [2]. Nevertheless, comprehensive records of batik patterns in every region are scarce. Suspicious consideration ought to be given to Indonesian batik patterns in order to prevent the erosion of the Indonesian people's cultural heritage. [3].

The problem arises because batik has its distinctive patterns. Batik has various patterns that can come from multiple sources, such as nature and culture because batik means representing a philosophy of life and ancestral heritage and reflects the region where the batik originates. This makes us study various types of batik and the factors that influence them to determine the correct type of batik.

As innovative progress is created, Deep Learning has reached various broad aspects, such as analyzing/processing images recognizing sounds, and diagnosing diseases [4]. Convolutional Neural Network or CNN is one of method in Deep Learning designed to handle image information. CNN reduce the number of trainable parameters in an ANN, thereby assisting networks in enhancing generalization and preventing overfitting. [5].

Based on the description above and seeing the capabilities and advantages of deep Learning, the author was encouraged to create a Website-Based Indonesian Batik Motif Classification application. In this application, the user will only need to take a picture of the batik so that later, the application will classify the batik that has been uploaded and will provide some information about the batik. FastAPI was used as the framework of Application Programming Interface (API), by means of which the deep learning model is linked to the website application.

2. **Related Work.** Numerous prior studies have been related to the classification of batik patterns. Agastya et al [6]. A considerable amount of research has been dedicated to the classification of batik patterns in the past. They used VGC-16 and VGC-19 models, each with 13 convolution layers and 3 x 3 convolution. The CNN model (VCG-19) with Softmax achieved an accuracy of 89.3% in split data compared to unsplit data.

Rasyidi et al. [7] research on Batik Pattern Recognition uses Convolutional Neural Network models like AlexNet, VGG, ResNet, SqueezeNet, and DenseNet, with DenseNet achieving 94% accuracy in eight cycles.

Research by Mardani et al. [8] about Deep Learning for Javanese Batik Pattern Recognition uses a dataset of 750 images, with cross-validation results achieving an accuracy of 90.14%, proving the model's effectiveness.

Meranggi et al. [9] research on Batik Classification using Convolutional Neural Networks showed that the old dataset was enhanced by the addition of 621 data points to replace anomalous information. Resulting in the best results of 88.88% ±0.88 and 66.14% ±3.7 respectively using ResNet-18 Architecture.

Arsa et al. [10] research on Random Forest that used for batik classification, using a 50:50 and 80:20 dataset ratio, used a CNN model with 16 hidden layers, Random Forest as a classifier, and achieved an accuracy of 97.68% ± 2.71.

Research by Uswatun Khasanah et al. [11] research on batik classification using a dataset of 500 images showed successful to combine VGG16 pre-trained model and data augmentation. This method increasing accuracy from 95.83% to 98.96%.

Arif Rasyidi et al. [12] collected 120 images for batik and used transfer learning with ResNet, DenseNet, and VGG models. In order to generate a new dataset, images were divided into tiny pieces. Augmentation of the image was implemented to avoid overfitting. Densenet169 achieved 79.17% on the initial dataset which is the highest accuracy, whereas vgg13\_bn demonstrated the highest performance with 87.61% accuracy on the modified dataset.

Suhardi Aras et al. [13] The study proposed using VGG16 and Resnet50 architectures for Papuan Batik motifs classification, with fine tuning for better performance. Results showed 78.79% accuracy without data augmentation, 81.82% accuracy with various augmentation techniques, and 87.88% yield on Resnet50 architecture.

Arif Rasyidi et al. [14] The study investigates the use of convolutional neural network (CNN) in classifying batik motif images in Indonesia's Lasem, Yogyakarta, and Solo regions. There are 13 classes of batik pattern and the results show training accuracy was 56% with Kawung motif classification being better.

3. **Methodology**. This section is divided into 6 sub-sections: Dataset Preparation, CNN concept, EfficientNetV2, Performance Evaluation, Application Programming Interface, and Unified Modeling Language.

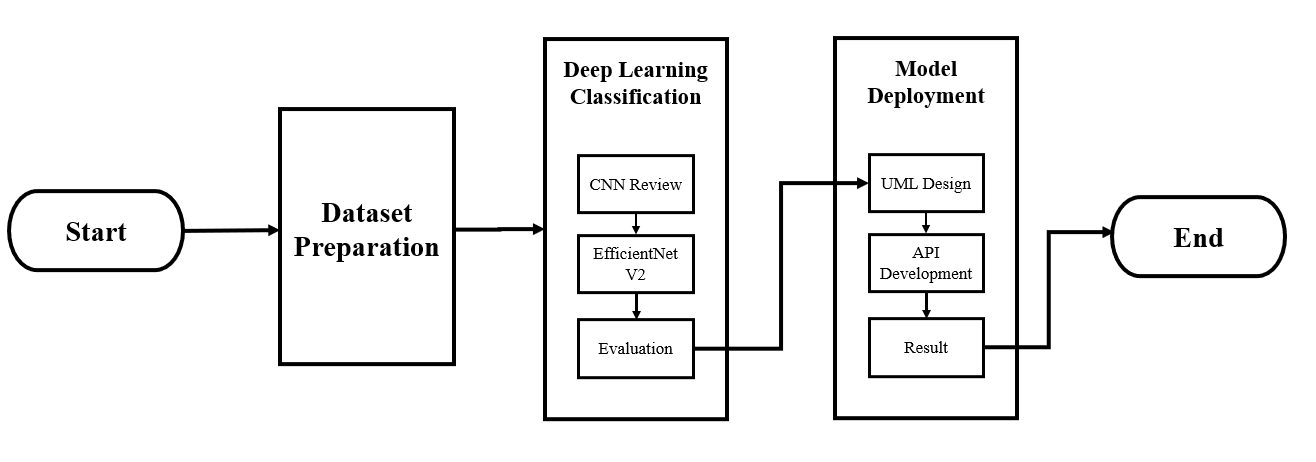


FIGURE 1. Methodology

Figure 1 is the methodology of our proposed method. First, we prepared the dataset by collecting data from various sources and do some augmentation. For batik classification, we utilized EfficientNetV2 to extract the image feature of our CNN architecture. We inputed randomly batik pattern image then predicted what kind of batik it is through the application. This overflow of this method is integrated by UI system to make it easier for user to operate our model.

3.1. **Dataset Preparation.** The dataset is a private dataset obtained through a search process on various internet sites. This dataset comprises 2451 images in total, of which five are batik pattern classes: Kawung, Megamendung, Parang, Sekarjagad, and Truntum. Figure 2 shows some examples of data from this dataset.

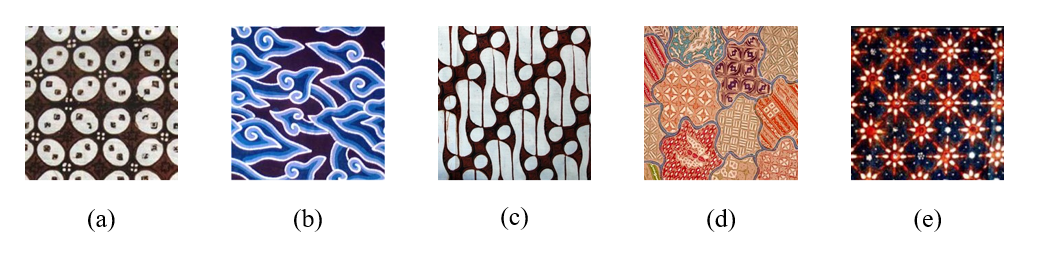


FIGURE 2. Examples of images from the dataset: (a) Kawung (b) Megamendung (c) Parang (d) Sekarjagad (e) Truntum

In this study dataset is partitioned into three distinct components: training, validation, and test data. There are 1372 images in the training, followed by 585 for validation and 491 images for test. Dimensions of 224x224 were used for each image in the dataset that will be used later. The preprocessing stage involves augmentation to vary the dataset and minimize overfitting, using a Horizontal Flip operation to flip images horizontally, ensuring a diverse and non-monotonic outcome.

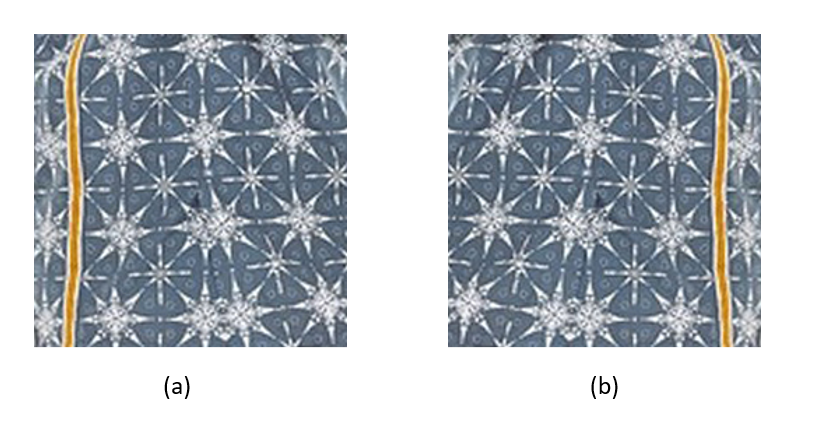


FIGURE 3. Example of Horizontal Flip: (a) Normal (b) Horizontal Flip

3.2. **CNN concept.** Convolutional Neural Networks is one of deep learning technique that is specifically designed to process visual data, including videos and images. CNN makes it possible to extract more specific image features when compared to ANN so that this method will focus more on the image while reducing parameters [15]. CNN is composed of three fundamental layers: convolution layers, pool layers, and dense layers. Convolution layers apply convolution operations to an image [16]. The convolution process will cause the dimensions of the input image to become compressed, where the compressed image combines previous information.

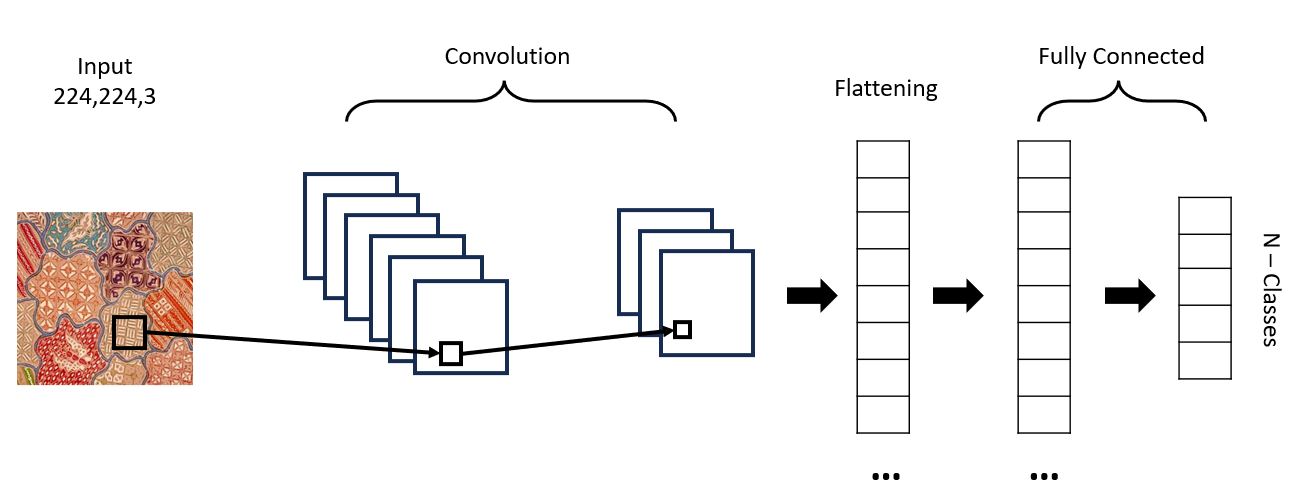
****

FIGURE 4. Convolution Process

3.3. **EfficientNetV2.** Transfer Learning is a method in deep learning to train the model that will be created but uses weights previously trained or pre-trained models in a vast dataset. The benefit of this method is that it reduces the computational burden and can handle the problem of small datasets [17]. One of example of a pre-trained model which able to be used in classification cases is EfficientNetV2. EfficientNetV2 is an iteration of the preceding model [18] that incorporates a number of modifications that accelerate training procedure and enhancing parameter effectiveness through the utilization of training-aware neural architecture search. EfficientNetV2 employs Fused-MBConv in the initial layers of its architecture and subsequently utilizes a 3x3 kernel size reduction, resulting in a lightened computation load but a quicker training process. [19].

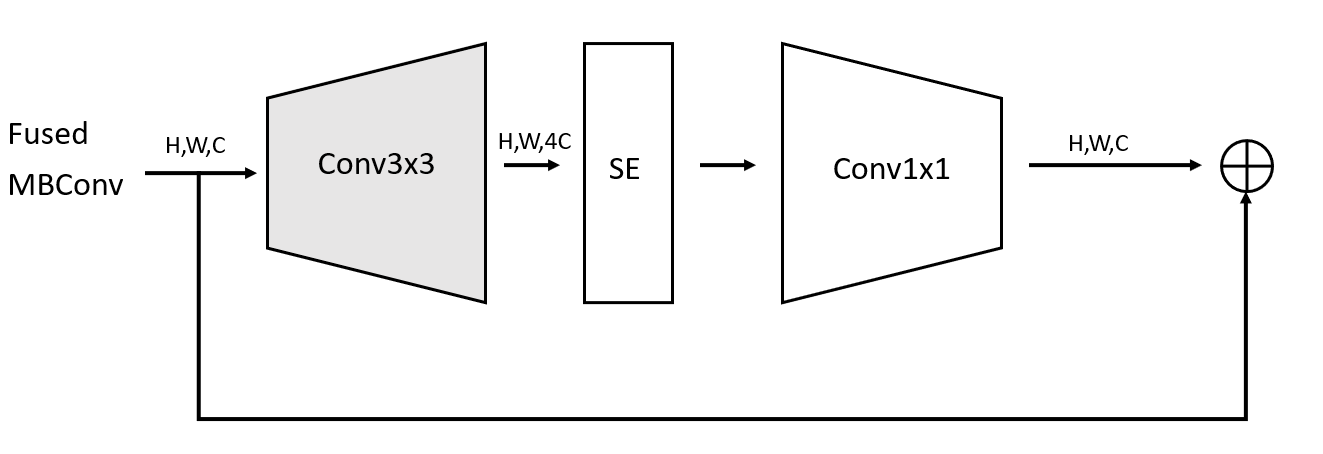
****

FIGURE 5. Fused-MBConv

As a feature extractor, EfficientNetV2 will be utilized in this research, and the completely connected layer will be modified to fit the case study of this research. The final layer employs the softmax activation function to produce probabilities of the output classes.

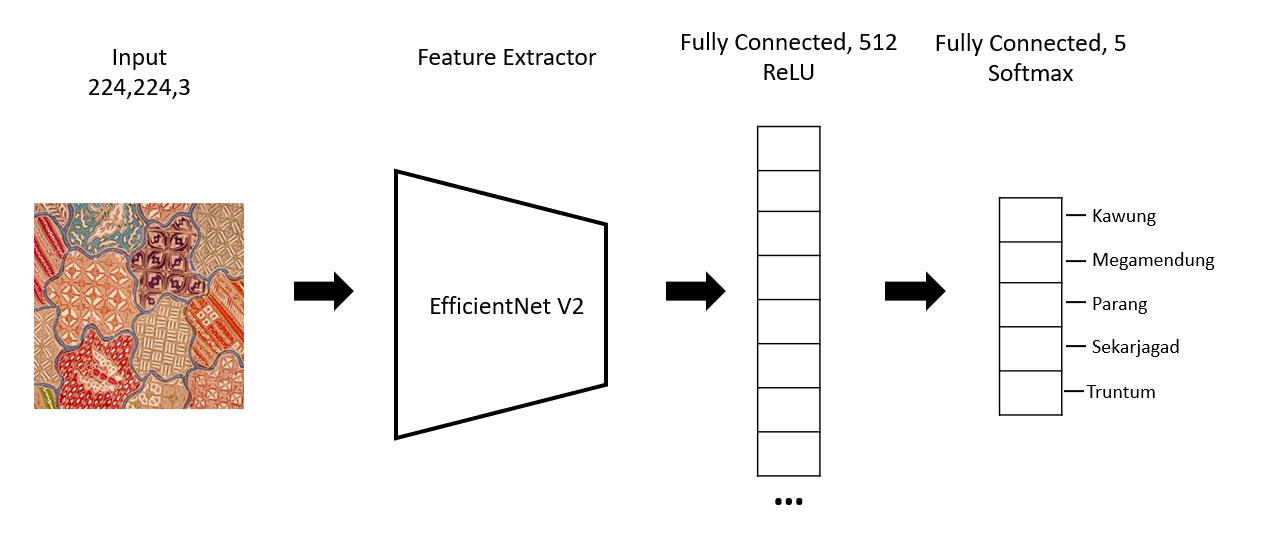


FIGURE 6. Used Architecture

3.4. **Performance Evaluation.** The confusion matrix serves as a tabular representation utilized to evaluate the effectiveness of the model that has been constructed. [20]. These computed for each formula using the matrix's four terms: True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN).

|  |  |
| --- | --- |
| Accuracy = | (1) |
| Precision = | (2) |
| Recall = | (3) |
| F1 Score= | (4) |

3.5. **Unified Modeling Language.** The analysis was carried out using UML or Unified Modeling Language. A use case diagram is a diagram that describes how it works, its scope, and how users interact with a system [21]. With this diagram, the relationship between the user and the system is visualized to understand what actions the user can take on a system being built. Figure 7 is an illustration of use case diagram.

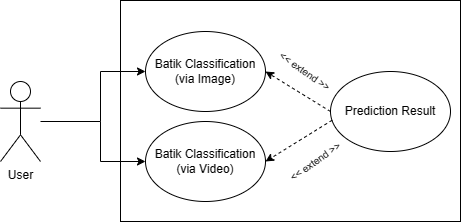


FIGURE 7. Use Case

Users can choose two classification methods to classified batik patterns. Users can use image as input or video in real-time classification.

3.6. **Application Programming Interface.** The RESTful API architecture was chosen as the method used in this research. Then, to display the prediction results from the model, it will use JSON format. RESTful is the choice based on access speed performance due to low resource consumption [22]. By utilizing the Application Programming Interface (API), one can retrieve the deep learning model from the origin server and present the obtained results in the format of JSON data. The outcomes of the predictions will be presented through a web-based application.

4. **Main Results**. The primary outcomes of this application will be structured into two components: The outcomes derived from the process of training the image classification model and integrating the model into the website.

4.1. **Image Classification.** The EfficientNetV2 architecture training phase begins with 15 epochs and requires configuration parameters like optimizer, loss, learning rate, and batch size. The Adaptive Momentum optimizer or Adam, combining Adagrad and RMSProp, is efficient and requires little memory [23]. The loss function is Categorical Cross Entropy, and the batch size is 40.

TABLE 1. Architectural configuration

|  |  |
| --- | --- |
| **Parameters** | |
| Optimizer | Adaptive Momentum |
| Loss Function | Categorical Cross Entropy |
| Training Learning Rate | 1e-3 |
| Batch Size | 40 |
| Metrics | Accuracy |

The following are Figure 8 and Table 2, the results of training that has been carried out for 15 epochs.



FIGURE 8. Training Result

TABLE 2. Detailed Training Result

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **Epoch** | **Loss** | **Acc** | **Val\_Loss** | **Val\_Acc** |
| EfficientNetV2 | 15 | 0.3792 | 0.9825 | 0.3534 | 0.9744 |

The table presents the results indicating that the highest achievable validation accuracy is 97%. Following the acquisition of the validation accuracy value at the ninth epoch, there was no further increase in validation accuracy until the fifteenth epoch.

Then, the author also tried to test the model that had been created by taking test images from the dataset that had been created and obtained a total of 491 images. Table 3 shows the outcomes of other parameters using test data.

TABLE 3. Other Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class** | **Precision** | **Recall** | **F1 Score** | **Accuracy** |
| Kawung | 0.96 | 0.94 | 0.95 | **0.96** |
| Megamendung | 0.99 | 1 | 0.99 |
| Parang | 0.96 | 0.99 | 0.98 |
| Sekarjagad | 0.92 | 0.87 | 0.89 |
| Truntum | 0.93 | 0.95 | 0.94 |
| **Average** | **0.95** | **0.95** | **0.95** |

4.2. **Website Development.** Website development is carried out in two stages, namely developing the front-end and developing back-end processes.

4.2.1. **Front-end.** Figure 9 shows the start page application. Figure 10 illustrates the classification of batik patterns using images as input. The user is asked to enter one of the batik motif images, and then the application will display the outcomes in Figure 11. Figure 12 display another prominent feature of this application, namely the classification of batik patterns using real-time video. In this section, the user is asked to point the camera at the batik motif, and then the application will display the results in Figure 12.

|  |  |
| --- | --- |
|  |  |
| FIGURE 9. Headline of the application | FIGURE 10. Classification of Batik patterns using image input |

|  |  |
| --- | --- |
|  |  |
| FIGURE 11. Results of Batik Motif Classification Predictions | FIGURE 12. Real-time classification of Batik patterns |

4.2.2. **Back-end.** After front-end development, back-end development uses the FastAPI framework with the Python programming language. Using the RESTful API architecture, there are three endpoints used, as shown in Table 4

TABLE 4. Endpoints

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **HTTP Method** | **Endpoint** | **Tasks** |
| 1 | GET | / | Enter the application and display the website page |
| 2 | POST | /predict | Predict batik pattern by inputting an image |
| 3 | GET | /video | predict batik pattern with real-time video |

The application runs on localhost with PORT 3000 and displays a headline section. It predicts batik patterns using input images, preprocessing them to RGB and resizing them to 224x224. The results are returned in JSON format. Real-time video prediction uses the same preprocessing steps.

5. **Conclusions.** The batik motif classification model demonstrated its highest level of accuracy at the ninth epoch, out of a total of fifteen epochs (0.9825% during training, 0.974% during validation, and 0.96 during test data). The transfer learning method was effective for training with few epochs but produced satisfactory results.

A website was developed to identify batik patterns in images or videos, aiming to help the public find batik cloth and information, contributing to the development of Indonesian Batik Motif Classification technology using deep learning techniques.

The research has satisfactory results but needs improvement in prediction accuracy and data variety. It's recommended to add variations in batik motif data for better results and insights into Indonesian batik patterns.

**Acknowledgment.** Contributing in part to this effort is Universitas Pendidikan Indonesia. Additionally, the authors are appreciative of the reviewers' constructive comments and recommendations, which have enhanced the presentation.

**REFERENCES**

1. Saddhono, K., Widodo, S. T., Al-Makmun, M. T., & Tozu, M. (2014). The study of philosophical meaning of Batik and Kimono motifs to foster collaborative creative industry. Asian Social Science, 10(9), 51–61. https://doi.org/10.5539/ass.v10n9p52
2. F. A. Putra et al., "Classification of Batik Authenticity Using Convolutional Neural Network Algorithm with Transfer Learning Method," 2021 Sixth International Conference on Informatics and Computing (ICIC), Jakarta, Indonesia, 2021, pp. 1-6, doi: 10.1109/ICIC54025.2021.9632937.
3. Rangkuti, A. H., Harjoko, A., & Putra, A. (2021). A Novel Reliable Approach for Image Batik Classification That Invariant with Scale and Rotation Using MU2ECS-LBP Algorithm. *Procedia Computer Science*, *179*(2019), 863–870. https://doi.org/10.1016/j.procs.2021.01.075
4. Sarker I. H. (2021). Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. SN computer science, 2(6), 420. https://doi.org/10.1007/s42979-021-00815-1
5. L. Alzubaidi et al., Review of deep learning: concepts, CNN architectures, challenges, applications, future directions, vol. 8, no. 1. Springer International Publishing, 2021. doi: 10.1186/s40537-021-00444-8.
6. I. M. A. Agastya and A. Setyanto, “Classification of Indonesian batik using deep learning techniques and data augmentation,” Proc. - 2018 3rd Int. Conf. Inf. Technol. Inf. Syst. Electr. Eng. ICITISEE 2018, pp. 27–31, 2018, doi: 10.1109/ICITISEE.2018.8720990.
7. M. A. Rasyidi and T. Bariyah, “Batik pattern recognition using convolutional neural network,” Bull. Electr. Eng. Informatics, vol. 9, no. 4, pp. 1430–1437, 2020, doi: 10.11591/eei.v9i4.2385.
8. D. A. Mardani, Pranowo, and A. J. Santoso, “Deep learning for recognition of Javanese batik patterns,” AIP Conf. Proc., vol. 2217, no. April, 2020, doi: 10.1063/5.0000686.
9. D. G. T. Meranggi, N. Yudistira, and Y. A. Sari, “Batik Classification Using Convolutional Neural Network with Data Improvements,” Int. J. Informatics Vis., vol. 6, no. 1, pp. 6–11, 2022, doi: 10.30630/joiv.6.1.716.
10. D. M. S. Arsa and A. A. N. H. Susila, “VGG16 in Batik Classification based on Random Forest,” Proc. 2019 Int. Conf. Inf. Manag. Technol. ICIMTech 2019, vol. 1, no. August, pp. 295–299, 2019, doi: 10.1109/ICIMTech.2019.8843844.
11. C. Uswatun Khasanah, E. Utami and S. Raharjo, "Implementation of Data Augmentation Using Convolutional Neural Network for Batik Classification," 2020 8th International Conference on Cyber and IT Service Management (CITSM), Pangkal, Indonesia, 2020, pp. 1-5, doi: 10.1109/CITSM50537.2020.9268890.
12. Rasyidi, M. A., Handayani, R., & Aziz, F. (2021). “Identification of batik making method from images using convolutional neural network with limited amount of data”. *Bulletin of Electrical Engineering and Informatics*, *10*(3), 1300–1307. https://doi.org/10.11591/eei.v10i3.3035
13. S. Aras, A. Setyanto and Rismayani, "Classification of Papuan Batik Motifs Using Deep Learning and Data Augmentation," 2022 4th International Conference on Cybernetics and Intelligent System (ICORIS), Prapat, Indonesia, 2022, pp. 1-5, doi: 10.1109/ICORIS56080.2022.10031320.
14. Tristanto, J., Hendryli, J., & Herwindiati, D. E. (2018). Classification of Batik Motifs Convolutional Neural Networks. The 2018 International Conference on Information Technology, Engineering, Science, and Its Applications, 1–5.
15. A. Saxena, “An Introduction to Convolutional Neural Networks,” Int. J. Res. Appl. Sci. Eng. Technol., vol. 10, no. 12, pp. 943–947, 2022, doi: 10.22214/ijraset.2022.47789.
16. S. Albawi, T. A. Mohammed, and S. Al-Zawi, “Understanding of a convolutional neural network,” Proc. 2017 Int. Conf. Eng. Technol. ICET 2017, vol. 2018-Janua, pp. 1–6, 2018, doi: 10.1109/ICEngTechnol.2017.8308186.
17. M. Iman, H. R. Arabnia, and K. Rasheed, “A Review of Deep Transfer Learning and Recent Advancements,” Technologies, vol. 11, no. 2, pp. 1–14, 2023, doi: 10.3390/technologies11020040.
18. M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” 36th Int. Conf. Mach. Learn. ICML 2019, vol. 2019-June, pp. 10691–10700, 2019.
19. M. Tan and Q. V. Le, “EfficientNetV2: Smaller Models and Faster Training,” Proc. Mach. Learn. Res., vol. 139, pp. 10096–10106, 2021.
20. J. Görtler et al., Neo: Generalizing Confusion Matrix Visualization to Hierarchical and Multi-Output Labels, vol. 1, no. 1. Association for Computing Machinery, 2022. doi: 10.1145/3491102.3501823.
21. Koç, H., Erdoğan, A. M., Barjakly, Y., & Peker, S. (2021). *UML Diagrams in Software Engineering Research: A Systematic Literature Review*. 13. https://doi.org/10.3390/proceedings2021074013
22. F. Halili and E. Ramadani, “Web Services: A Comparison of Soap and Rest Services,” Mod. Appl. Sci., vol. 12, no. 3, p. 175, 2018, doi: 10.5539/mas.v12n3p175.
23. D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., pp. 1–15, 2015.