

# Improved Javanese script recognition using custom model of convolution neural network

Ajib Susanto<sup>1</sup>, Ibnu Utomo Wahyu Mulyono<sup>1</sup>, Christy Atika Sari<sup>1</sup>, Eko Hari Rachmawanto<sup>1</sup>,  
De Rosal Ignatius Moses Setiadi<sup>1</sup>, Md Kamruzzaman Sarker<sup>2</sup>

<sup>1</sup>Department of Informatics Engineering, Dian Nuswantoro University, Semarang, Indonesia

<sup>2</sup>Department of Computing Science, University of Hartford, West Hartford, United States

## Article Info

### Article history:

Received Jul 29, 2022

Revised May 24, 2023

Accepted Jun 26, 2023

### Keywords:

Custom convolutional neural network model  
Deep learning recognition  
Handwritten recognition  
Javanese script recognition  
Optical character recognition

## ABSTRACT

Handwriting recognition in Javanese script is not widely developed with deep learning (DL). Previous DL and machine learning (ML) research is generally limited to basic characters (Carakan) only. This study proposes a deep learning model using a custom-built convolutional neural network to improve recognition accuracy performance and reduce computational costs. The main features of handwritten objects are textures, edges, lines, and shapes, so convolution layers are not designed in large numbers. This research maximizes optimization of other layers such as pooling, activation function, fully connected layer, optimizer, and parameter settings such as dropout and learning rate. There are eleven main layers used in the proposed custom convolutional neural network (CNN) model, namely four convolution layers+activation function, four pooling layers, two fully connected layers, and a softmax classifier. Based on the test results on the Javanese script handwritten image dataset with 120 classes consisting of 20 basic character classes and 100 compound character classes, the resulting accuracy is 97.29%.

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



## Corresponding Author:

Ajib Susanto

Department of Informatics Engineering, Dian Nuswantoro University

Imam Bonjol street no 207, Semarang-50131, Indonesia

Email: [ajib.susanto@dsn.dinus.ac.id](mailto:ajib.susanto@dsn.dinus.ac.id)

## 1. INTRODUCTION

Indonesia has many ethnic groups, religions, and races, so it requires a unified language, Indonesian. One of the largest ethnic groups in Indonesia, the Javanese, has the Javanese language. Indonesian uses Latin letters, while the original Javanese language uses Javanese script. This difference makes many of the younger generation of Javanese not learn much about Javanese script [1]. Digitizing and making applications for learning the Javanese language and script can be a solution for the younger generation. But the light and accurate method becomes a challenge in developing the application. To build one part of this application, a recognition method is needed for recognizing Javanese script handwriting on the smartphone screen.

Various methods of handwriting recognition have been developed, both based on Latin letters such as [2]–[5] or non-Latin letters or traditional characters like [6]–[22]. Research on Javanese script recognition in digital images has been carried out. Some of the latest research are [1], [8]–[13], [15], [16], [23]. From some of these studies, several studies use machine learning (ML) such as [1], [8], [10]–[12], [15], [16], [23], and others use deep learning (DL) [9], [13], [22], [24]. ML has relatively lighter computations than DL, but the accuracy performance of DL methods is generally better than ML methods. DL can also work better if the training dataset used is large. Almost all Javanese script recognition research is generally limited to basic characters (Carakan), consisting of 20 characters, only on research [23] which adds seven numeric characters

to the recognition process. While the Javanese script actually has far more characters because the basic character is compound with the vowel a. While in Javanese script, there are six vowels, namely *a, i, u, e, é, o*, so there are 120 Javanese script characters, and this does not include numbers and punctuation marks.

Various research on handwriting recognition of non-Latin characters has been more advanced. They use DL with more classes and more complex processes with better accuracy, such as Bangla [7], Kannada [20], Tifinagh [6], Arabic [14], [25], and Khmer [26] characters as shown in Table 1. While in research [9], [13], [22], [24], those who have used DL get an accuracy of only about 64% in [13], about 84% in [24], about 86% in [22] and the best in [9] in 94%. Based on the results, the accuracy is not even better than various ML studies proposed for Javanese script recognition. Looking at the number of datasets used, some of them are sufficient for the DL method, which is 11,500 in [9], 2,470 in [13], 16,800 in [22], and 2,000 in [24]. Reliable performance should be produced with good preprocessing and compilation of convolutional neural network (CNN) models. Several things allow for the lack of DL performance in these studies: noisy datasets without selection, the number of CNN layers that are not suitable, and the tuning parameters that are not suitable.

Table 1. State-of-the-art non-Latin handwritten recognition

Research	Script	Number of classes	Total dataset	Testing accuracy
[6]	Tifinagh	33	25,740	99.27%
[7]	Bangla	231	59,922	96.46%
[14]	Arabic number	10	61,000	99.6%
[20]	Kannada	49	8,504	99%
[25]	Arabic Words	96	10,080 original and 40,320 augmented	96.76%
[26]	Khmer	33	3,366	94.9%

Many CNN models have been designed for object recognition in images. Although generally, it can work to recognize various objects, including handwriting. But generally, these models were originally designed specifically for certain objects, so the CNN model's performance can only work optimally on certain objects. For example, in research [27] for multiple object detection, research [28] for face recognition, [29] for face mask recognition, and [30] for violence detection. This proves that the design of the CNN model for handwriting recognition, especially Javanese script, is needed to get optimal recognition performance. CNN models can be designed with a different number and arrangement of layers, affecting the performance of the recognition accuracy results and computational costs.

The layers that need to be considered when designing the CNN model are the convolutional layer (CL), pooling layer (PL), activation function (AF), fully connected layer (FC), loss function (LF), and optimizer (OP). CL is in charge of extracting features in the image. For CNN models with many CLs, CL in the lower layer layers extracts texture, edge, and line features, while at higher layers extracts abstract features. PL is used to perform downsampling and create output feature maps that are more robust against distortion and neuronal errors. AF establishes a functional relationship between input and output and introduces a nonlinear system into the neural network. FC is tasked with integrating and classifying local information and discrimination after the convolution and pooling process. LF is in charge of carrying out the final classification process. LF has an important role in recognition because different LFs optimize different recognition tasks (object classification, face recognition, and object recognition). Network training relies on the core step of gradient updates. In this case, the OP's job is to update gradients for faster, less lost, and simpler computations [31]. Based on this theory, it is believed that designing a CNN model specifically for handwriting recognition of Javanese script is necessary.

Based on the previous explanation, it has been explained that the DL method designed for Javanese script handwriting recognition still needs to be improved because it is still limited to basic characters, and the accuracy is still not satisfactory. This study aims to design a CNN model optimized for Javanese script handwriting recognition. Recognition is not limited to 20 basic characters but also 100 other characters compounded with Javanese script vowels. The number of layers in the proposed CNN model will be minimized for a cheaper computational cost but still have good accuracy performance. This paper is organized into four parts: the first part, the second part describes the proposed method and its hypotheses, the third part discusses the results and results, and the last is the conclusion.

## 2. METHOD

This research is limited to the design of the CNN model without involving the segmentation process or object detection. The handwritten image has passed the preposing stage, arranged so that an output image with a size of 150×150 24-bits (RGB) is obtained for each character. Furthermore, the DL process is carried out with the proposed model. The proposed CNN model is presented in Figure 1.

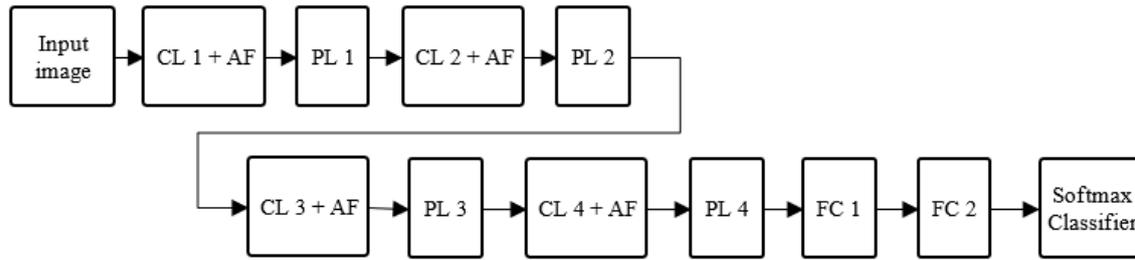


Figure 1. Proposed CNN model for Javanese script recognition

The proposed CNN model has eleven layers consisting of four convolutional layers, a pooling layer, two fully connected layers, and a softmax classifier as a loss function. The number of layers used is relatively small compared to some of the latest CNN models, such as VGGnet with 16 or 19 layers, GoogleLeNet with 22 layers, and ResNet with 152 layers [32]. The CNN model is deliberately designed with the minimum number of layers possible to reduce the computational cost. This design is determined based on the theories and hypotheses that have been described previously. Besides that, several tests have also been carried out to optimize the results. The four convolution layers perform the leaky rectified linear unit (ReLU) activation function to improve accuracy. The leaky ReLU function is a development of ReLU, which is more optimal in performance because it can improve ReLU performance, especially if a number of neurons die [33], see (1) for leaky ReLU function. The number of convolution layers is minimized because handwriting features are less complex and do not require abstract features. Every two convolution layers added a dropout of 0.2 to reduce the occurrence of overfitting. PL is placed on each CL+AF to optimize the downsampling process for output features that are more robust against distortion and neuronal errors, the type of PL used is max-pooling.

Furthermore, two FC/dense layers make integrating and classifying local information and discrimination smoother with two processing stages. Before entering the FC, it is flattened. The last layer has a softmax classifier to carry out the classification process. Meanwhile, the optimizer used in this research is ADAM, with a learning rate of 0.0007. The details of the proposed CNN model are presented in Table 2.

$$f(x) = \begin{cases} x & x > 0 \\ \alpha x & x \leq 0 \end{cases} \quad (1)$$

where  $\alpha$  is the number to be multiplied by  $x$ , providing the output even if  $x$  is negative, the neurons in the negative area are stimulated and become active.

Table 2. Proposed CNN model detailed

Layer	Output Shape	Stride	Parameter
Convolution 1	(148, 148, 32)	1×1	896
Max Pooling 1	(74, 74, 32)	2×2	0
Convolution 2	(72, 72, 64)	1×1	18496
Max Pooling 2	(36, 36, 64)	2×2	0
Convolution 3	(34, 43, 128)	1×1	73856
Max Pooling 3	(15, 15, 128)	2×2	0
Convolution 4	(15, 15, 128)	1×1	147584
Max Pooling 4	(7, 7, 128)	2×2	0
Flatten	(6272)	-	0
Dense	(512)	-	3211776
Dense	(32)	-	0
Softmax	(32)	-	0

### 3. RESULTS AND DISCUSSION

In the first stage, data was collected from handwritten images of Javanese script written using a handwritten notepad application. All images have the extension *.png*. There are 120-character classes, consisting of 20 basic and 100 compound characters. For example, the dataset in Figure 2(a) is a basic character, while in Figure 2(b) is a compound character of the character “Ba”. The basic character is always paired with the vowel ‘a’. As for the compound character, it is a basic character that is juxtaposed with *sandhangan swara*. *Sandhangan swara* has five vowels besides *a*, namely *i*, *u*, *e*, *é*, *o*, so that the character “Ba” can be “Bi”, “Bu”, “Be”, “Bo”, “Bé”, as the sample presented in Figure 2(b). This dataset was written

by 30 different people, and each person wrote four times with the HW memo android application, so there are a total of 14,400 images for 120 classes. Because this dataset is large, the augmentation process is not carried out.

This research is implemented with several libraries, modules, application programming interfaces (APIs), and frameworks in building the model. Some libraries include Google Colab with drive class, NumPy, Matplotlib, and OpenCV with *imread* class. While the module used is OS, *zipfile* with *ZipFile* class, *shutil*, *random*, *hashlib*, and *math*. The API used is Keras API with classes *ImageDataGenerator*, *Sequential*, *Conv2D*, *MaxPooling2D*, *Flatten*, *Dense*, and *Input*. The sci-kit learn framework with *classification\_report* class is also used in this research.

At the testing stage, all data is loaded and then divided into three parts: training, validation, and testing, with a composition of 70%: 15%: 15%. The training and validation process is based on the proposed method with a total of 50 epochs. The result is an accuracy of about 98% for training and 96% for validation, while for loss training, it is around 4% and validation is around 14%. More detailed results can be seen in Figure 3. As for the testing results, the resulting accuracy was 97.29%, while the loss reached 12.79%.

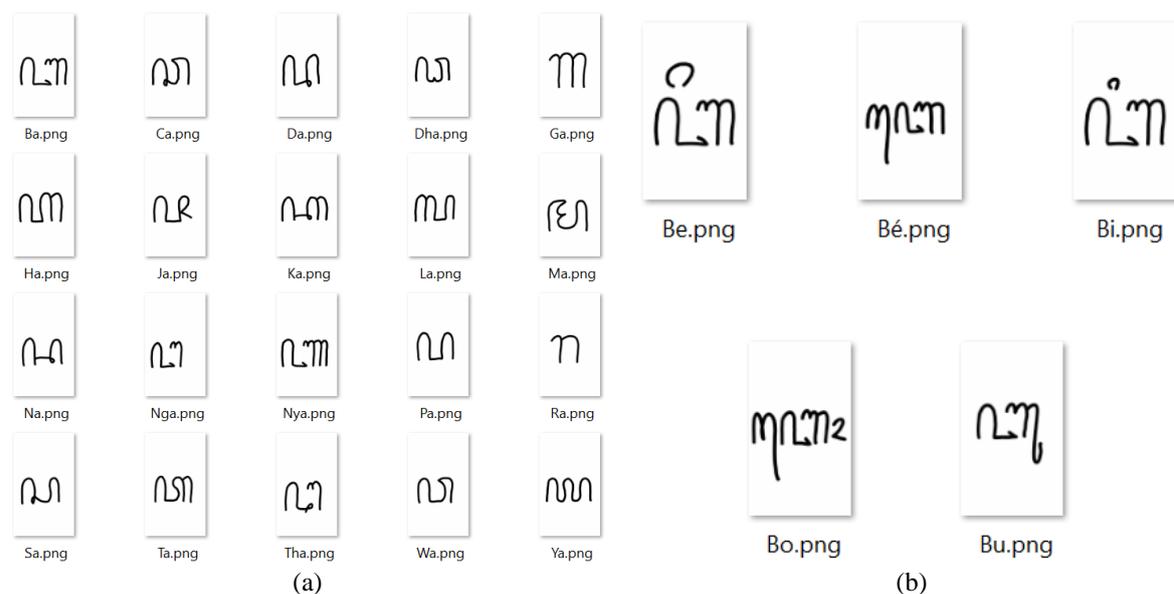


Figure 2. Sample Javanese Script dataset was used in the research (a) basic character (b) compound character of “Ba”

The results presented in Figure 3 show that the training process’s accuracy gets the best results. There is a difference of about 2% for the validation process and 1% for testing. These results can be concluded very well because the testing accuracy can reach more than 97%, as shown in Figure 3(a). However, there is a fairly large difference in the loss measurement results, whereas, in the training process, the loss is only about 4%. In comparison, it is about 14% and 12% for validation and testing, respectively, as shown in Figure 3(b). Although the resulting loss is quite large, with an accuracy above 97% for 120 classes, it can be concluded that this method is the most successful in the case of recognizing 120 Javanese script classes. Table 3 presents the evidence that the proposed CNN model is the best DL model for Javanese script recognition. However, compared to the state-of-the-art listed in Table 1, this model is perhaps not the best. Comparisons are unfair if done with different datasets, although logically, the recognition process will be more complex when you have more classes. So in this study, a comparison was also made with two other CNN models that are quite popular, namely VGG19 [34] dan ResNet50 [35]. To make the comparison easier, we use two Keras APIs, ResNet50 and VGG19.

Based on the results presented in Table 3, the proposed CNN model appears to perform better than the other two models. Another finding shows that the ResNet50 model has no better performance than VGG19, even though ResNet50 has more layers and has the most complex computations. But in the case of handwriting recognition on smartphones, it turns out there is no need for many layers. The complexity of handwritten images is unlike natural objects, which are diverse and require various abstract features. Suppose you look at the sample dataset in Figure 2. The image’s contrast is strong, so only texture, edge, line, and

shape features are logically needed. This has been proven with the CNN model we propose with a small number of layers, which successfully obtains high accuracy and low computational costs. Based on the test results, the time required for the training process per epoch is about 150 seconds, see Figure 4. This training time is also compared with two CNN models, namely ResNet50 and VGG19, where the proposed CNN model has better performance as shown in Table 4.

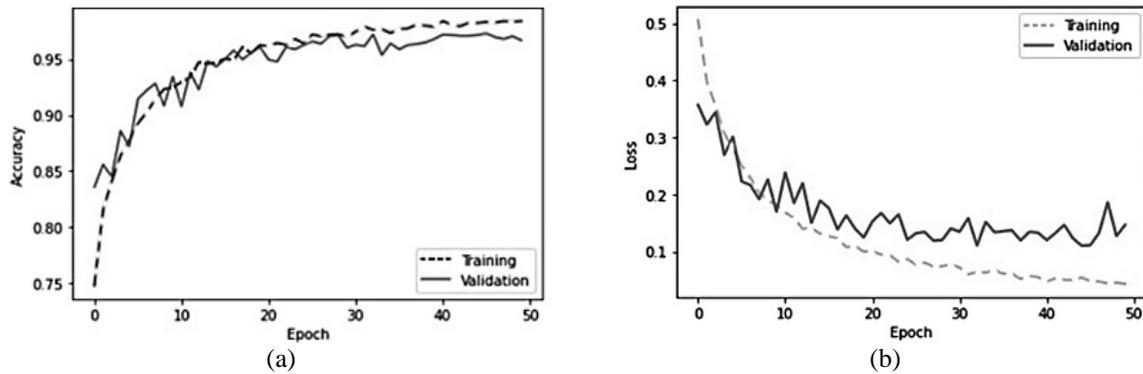


Figure 3. Recognition results of proposed CNN model (a) accuracy and (b) loss

Table 3. Comparison with another DL for Javanese script recognition

CNN models	Accuracy	Precision	Recall	F1-score
ResNet50	90.65%	91.21%	90.57%	90.88%
VGG19	94.73%	94.14%	93.82%	94.35%
Ours	97.29%	97.53%	96.83%	97.03%

```

Epoch 1/50
336/336 - 174s - loss: 0.5054 - accuracy: 0.7480 - val_loss: 0.3567 - val_accuracy: 0.8358 - 174s/epoch - 517ms/step
Epoch 2/50
336/336 - 150s - loss: 0.3972 - accuracy: 0.8153 - val_loss: 0.3222 - val_accuracy: 0.8559 - 150s/epoch - 447ms/step
Epoch 3/50
336/336 - 149s - loss: 0.3560 - accuracy: 0.8410 - val_loss: 0.3444 - val_accuracy: 0.8453 - 149s/epoch - 443ms/step
Epoch 4/50
336/336 - 149s - loss: 0.3068 - accuracy: 0.8639 - val_loss: 0.2690 - val_accuracy: 0.8860 - 149s/epoch - 444ms/step
Epoch 5/50
336/336 - 148s - loss: 0.2802 - accuracy: 0.8772 - val_loss: 0.3009 - val_accuracy: 0.8721 - 148s/epoch - 441ms/step

Epoch 46/50
336/336 - 148s - loss: 0.0479 - accuracy: 0.9831 - val_loss: 0.1109 - val_accuracy: 0.9729 - 148s/epoch - 439ms/step
Epoch 47/50
336/336 - 148s - loss: 0.0471 - accuracy: 0.9827 - val_loss: 0.1324 - val_accuracy: 0.9696 - 148s/epoch - 440ms/step
Epoch 48/50
336/336 - 148s - loss: 0.0440 - accuracy: 0.9838 - val_loss: 0.1861 - val_accuracy: 0.9679 - 148s/epoch - 441ms/step
Epoch 49/50
336/336 - 148s - loss: 0.0453 - accuracy: 0.9832 - val_loss: 0.1269 - val_accuracy: 0.9704 - 148s/epoch - 440ms/step
Epoch 50/50
336/336 - 147s - loss: 0.0430 - accuracy: 0.9839 - val_loss: 0.1467 - val_accuracy: 0.9668 - 147s/epoch - 438ms/step
    
```

Figure 4. Training computation time of proposed CNN model

Table 4. Comparison with another DL for training time consumption

CNN models	Training time (seconds)
ResNet50	8,639
VGG19	9,013
Ours	7,489

#### 4. CONCLUSION

CNN is a deep learning method that is most widely used in image recognition and classification problems. Various CNN models have been proposed for single-object recognition, multiple objects, and special objects such as faces, violence detection, and handwriting. The composition and number of layers on

CNN greatly affect performance, accuracy, and computational speed to recognize certain objects. Handwriting written on a smartphone screen is an object that has clear lines, edges, shapes, and textures, so the number of layers in the CNN model should be minimized to reduce computational costs. Recognition in Javanese script handwriting has not been widely developed as an object in various DL research and is only limited to basic characters. This study proposes a CNN model with eleven main layers, of which the number of layers is relatively less than the current CNN model. Optimizing the order and layer composition and setting tuning parameters allows the proposed CNN model to perform recognition with satisfactory accuracy, more than 97%, with a shorter training time than the ResNet50 and VGG19 models. Another contribution that should be noted is the total number of classes used in this research is 120, consisting of basic and compound characters in Javanese script. However, future research needs to be improved by increasing the accuracy and efficiency of computational costs. In addition, the number of classes can still be completed by adding numeric characters and Javanese script punctuation marks.

## ACKNOWLEDGEMENTS

The authors are grateful for the support for research funding in 2022 provided by the Ministry of Research and Technology/National Research and Innovation Agency of Indonesia with number 072/E5/PG.02.00.PT/2022, 016/LL6/PL/AK.04/202 2, 056/F9/UDN.09/VI/2022.

## REFERENCES

- [1] L. L. Zhangrila, "Accuracy level of SP algorithm for Javanese script detection on android-based application," *Procedia Computer Science*, vol. 135, pp. 416–424, 2018, doi: 10.1016/j.procs.2018.08.192.
- [2] J. O. Pinzón-Arenas and R. Jiménez-Moreno, "Comparison between handwritten word and speech record in real-time using CNN architectures," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 4, pp. 4313–4321, Aug. 2020, doi: 10.11591/ijece.v10i4.pp4313-4321.
- [3] T. A. Assegie and P. S. Nair, "Handwritten digits recognition with decision tree classification: a machine learning approach," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 5, pp. 4446–4451, Oct. 2019, doi: 10.11591/ijece.v9i5.pp4446-4451.
- [4] Z. Fouad, M. Alfonse, M. Roushdy, and A.-B. M. Salem, "Hyper-parameter optimization of convolutional neural network based on particle swarm optimization algorithm," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 6, pp. 3377–3384, Dec. 2021, doi: 10.11591/eei.v10i6.3257.
- [5] N. Saqib, K. F. Haque, V. P. Yanambaka, and A. Abdelgawad, "Convolutional-neural-network-based handwritten character recognition: an approach with massive multisource data," *Algorithms*, vol. 15, no. 4, Apr. 2022, doi: 10.3390/a15040129.
- [6] L. Niharmine, B. Outtaj, and A. Azouaoui, "Tifinagh handwritten character recognition using optimized convolutional neural network," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 4, pp. 4164–4171, Aug. 2022, doi: 10.11591/ijece.v12i4.pp4164-4171.
- [7] T. Ghosh *et al.*, "Bangla handwritten character recognition using MobileNet V1 architecture," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 9, no. 6, pp. 2547–2554, Dec. 2020, doi: 10.11591/eei.v9i6.2234.
- [8] A. Susanto, I. U. W. Mulyono, C. A. Sari, E. H. Rachmawanto, and D. R. I. M. Setiadi, "Javanese script recognition based on metric, eccentricity and local binary pattern," in *2021 International Seminar on Application for Technology of Information and Communication (iSemantic)*, Sep. 2021, pp. 118–121, doi: 10.1109/iSemantic52711.2021.9573232.
- [9] M. A. Wibowo, M. Soleh, W. Pradani, A. N. Hidayanto, and A. M. Arymurthy, "Handwritten Javanese character recognition using discriminative deep learning technique," in *2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, Nov. 2017, pp. 325–330, doi: 10.1109/ICITISEE.2017.8285521.
- [10] L. D. Krisnawati and A. W. Mahastama, "Building classifier models for on-off Javanese character recognition," in *Proceedings of the 21st International Conference on Information Integration and Web-based Applications and Services*, Dec. 2019, pp. 25–34, doi: 10.1145/3366030.3366050.
- [11] I. Prihandi, I. Ranggadara, S. Dwiasnati, Y. S. Sari, and Suhendra, "Implementation of backpropagation method for identified Javanese scripts," *Journal of Physics: Conference Series*, vol. 1477, no. 3, Mar. 2020, doi: 10.1088/1742-6596/1477/3/032020.
- [12] H. W. Herwanto, A. N. Handayani, A. P. Wibawa, K. L. Chandrika, and K. Arai, "Comparison of min-max, z-score and decimal scaling normalization for zoning feature extraction on Javanese character recognition," in *2021 7th International Conference on Electrical, Electronics and Information Engineering (ICEEIE)*, Oct. 2021, pp. 1–3, doi: 10.1109/ICEEIE52663.2021.9616665.
- [13] Rismiyati, Khadijah, and A. Nurhadiyah, "Deep learning for handwritten Javanese character recognition," in *2017 1st International Conference on Informatics and Computational Sciences (ICICoS)*, Nov. 2017, pp. 59–64, doi: 10.1109/ICICoS.2017.8276338.
- [14] R. H. Finjan, A. S. Rasheed, A. A. Hashim, and M. Murtdha, "Arabic handwritten digits recognition based on convolutional neural networks with resnet-34 model," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 21, no. 1, pp. 174–178, Jan. 2021, doi: 10.11591/ijeecs.v21.i1.pp174-178.
- [15] S. Wardoyo, K. W., A. S. Pramudyo, S. Suhendar, and S. Hidayat, "The development of Javanese language teaching materials through introduction of Java scripts using artificial neural network," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 16, no. 4, pp. 1697–1703, Aug. 2018, doi: 10.12928/telkomnika.v16i4.8465.
- [16] M. A. Rasyidi, T. Bariyah, Y. I. Riskajaya, and A. D. Septyani, "Classification of handwritten Javanese script using random forest algorithm," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 10, no. 3, pp. 1308–1315, Jun. 2021, doi: 10.11591/eei.v10i3.3036.
- [17] X. Ao, X.-Y. Zhang, and C.-L. Liu, "Cross-modal prototype learning for zero-shot handwritten character recognition," *Pattern Recognition*, vol. 131, Nov. 2022, doi: 10.1016/j.patcog.2022.108859.

- [18] M. A. Abuzaraida, M. Elmehrek, and E. Elsomadi, "Online handwriting Arabic recognition system using k-nearest neighbors classifier and DCT features," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 4, pp. 3584–3592, Aug. 2021, doi: 10.11591/ijece.v11i4.pp3584-3592.
- [19] Y. Elfakir, G. Khaissidi, M. Mrabti, D. Chenouni, and M. Boualam, "Combined cosine-linear regression model similarity with application to handwritten word spotting," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 3, pp. 2367–2374, Jun. 2020, doi: 10.11591/ijece.v10i3.pp2367-2374.
- [20] N. S. Rani, N. Manohar, M. Hariprasad, and B. R. Pushpa, "Robust recognition technique for handwritten Kannada character recognition using capsule networks," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 1, pp. 383–391, Feb. 2022, doi: 10.11591/ijece.v12i1.pp383-391.
- [21] A. Boukharouba and A. Bennia, "Novel feature extraction technique for the recognition of handwritten digits," *Applied Computing and Informatics*, vol. 13, no. 1, pp. 19–26, Jan. 2017, doi: 10.1016/j.aci.2015.05.001.
- [22] Y. Harjoseputro, Y. D. Handarkho, and H. T. R. Adie, "The Javanese letters classifier with mobile client-server architecture and convolution neural network method," *International Journal of Interactive Mobile Technologies (IJIM)*, vol. 13, no. 12, Dec. 2019, doi: 10.3991/ijim.v13i12.11492.
- [23] G. S. Budhi and R. Adipranata, "Handwritten Javanese character recognition using several artificial neural network methods," *Journal of ICT Research and Applications*, vol. 8, no. 3, pp. 195–212, Aug. 2015, doi: 10.5614/itbj.ict.res.appl.2015.8.3.2.
- [24] C. K. Dewa, A. L. Fadhillah, and A. Afiahayati, "Convolutional neural networks for handwritten Javanese character recognition," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 12, no. 1, pp. 83–94, Jan. 2018, doi: 10.22146/ijccs.31144.
- [25] A. Lamsaf, M. Ait Kerroum, S. Boulaknadel, and Y. Fakhri, "Recognition of Arabic handwritten words using convolutional neural network," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 26, no. 2, pp. 1148–1155, May 2022, doi: 10.11591/ijeecs.v26.i2.pp1148-1155.
- [26] B. Annanurov and N. Noor, "A compact deep learning model for Khmer handwritten text recognition," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 3, pp. 584–591, Sep. 2021, doi: 10.11591/ijai.v10.i3.pp584-591.
- [27] S. K. Pal, A. Pramanik, J. Maiti, and P. Mitra, "Deep learning in multi-object detection and tracking: state of the art," *Applied Intelligence*, vol. 51, no. 9, pp. 6400–6429, Sep. 2021, doi: 10.1007/s10489-021-02293-7.
- [28] X. Ke, B. Lin, and W. Guo, "LocalFace: Learning significant local features for deep face recognition," *Image and Vision Computing*, vol. 123, Jul. 2022, doi: 10.1016/j.imavis.2022.104484.
- [29] R. Mar-Cupido, V. García, G. Rivera, and J. S. Sánchez, "Deep transfer learning for the recognition of types of face masks as a core measure to prevent the transmission of COVID-19," *Applied Soft Computing*, vol. 125, Aug. 2022, doi: 10.1016/j.asoc.2022.109207.
- [30] P. Wang, P. Wang, and E. Fan, "Violence detection and face recognition based on deep learning," *Pattern Recognition Letters*, vol. 142, pp. 20–24, Feb. 2021, doi: 10.1016/j.patrec.2020.11.018.
- [31] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao, "Review of image classification algorithms based on convolutional neural networks," *Remote Sensing*, vol. 13, no. 22, Nov. 2021, doi: 10.3390/rs13224712.
- [32] L. Alzubaidi *et al.*, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-021-00444-8.
- [33] J. N. Mogan, C. P. Lee, K. S. M. Anbananthen, and K. M. Lim, "Gait-DenseNet: a hybrid convolutional neural network for gait recognition," *IAENG International Journal of Computer Science*, vol. 49, no. 2, 2022.
- [34] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, Sep. 2014.
- [35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

## BIOGRAPHIES OF AUTHORS



**Ajib Susanto**    received a bachelor's degree from the Department of Informatics Engineering, Dian Nuswantoro University, Semarang, Indonesia, in 2004 and a master's degree in the Department of Informatics Engineering, Dian Nuswantoro University, Semarang, Indonesia, in 2008. He is currently the lecturer and researcher at the Faculty of Computer Science, Dian Nuswantoro University, Semarang, Indonesia. His research interests include image processing, machine learning, and data mining. He can be contacted at email: [ajib.susanto@dsn.dinus.ac.id](mailto:ajib.susanto@dsn.dinus.ac.id).



**Ibnu Utomo Wahyu Mulyono**    received a bachelor's degree at the Department of Informatics Engineering from Dian Nuswantoro University, Semarang, Indonesia, in 2001 and a master's degree from the Department of Informatics Engineering, Dian Nuswantoro University, Semarang, Indonesia, in 2013. He is currently the lecturer and researcher at the Faculty of Computer Science, Dian Nuswantoro University, Semarang, Indonesia. His research interests include image processing, machine learning, and data mining. He can be contacted at email: [ibnu.utomo.wm@dsn.dinus.ac.id](mailto:ibnu.utomo.wm@dsn.dinus.ac.id).



**Christy Atika Sari**    received a bachelor's degree at the Department of Informatics Engineering Dian Nuswantoro University, Semarang, Indonesia, in 2010 and a dual master's degree in the Department of Informatics Engineering, Dian Nuswantoro University, Semarang, Indonesia and in Faculty of Computer Science and Information, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia, in 2012. She is currently the lecturer and researcher at the Faculty of Computer Science, Dian Nuswantoro University, Semarang, Indonesia. She has authored or co-authored more than 90 refereed journal and conference papers. She is also a reviewer of more than 10 Scopus-indexed journals indexed by Scopus. Her research interests include image processing, especially image steganography, image watermarking, cryptography, and image classification. She can be contacted at email: atika.sari@dsn.dinus.ac.id.



**Eko Hari Rachmawanto**    received a bachelor's degree at the Department of Informatics Engineering Dian Nuswantoro University, Semarang, Indonesia, in 2009 and a dual master's degree in the Department of Informatics Engineering, Dian Nuswantoro University, Semarang, Indonesia and in Faculty of Computer Science and Information, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia, in 2012. He is currently the lecturer and researcher at the Faculty of Computer Science, Dian Nuswantoro University, Semarang, Indonesia. He has authored or co-authored more than 100 refereed journal and conference papers. He is also a reviewer of more than 10 Scopus-indexed journals indexed by Scopus. His research interests include image processing, especially steganography, watermarking, cryptography, and image classification. He can be contacted at email: eko.hari@dsn.dinus.ac.id.



**De Rosal Ignatius Moses Setiadi**    received a bachelor's degree at the Department of Informatics Engineering Soegijapranata Catholic University, Semarang Indonesia, in 2010 and a master's degree in the Department of Informatics Engineering Dian Nuswantoro University, Semarang, Indonesia, in 2012. He is currently the lecturer and researcher at the Faculty of Computer Science, Dian Nuswantoro University, Semarang, Indonesia. He has authored or co-authored more than 140 refereed journal and conference papers indexed by Scopus. He is one of the academic editors in the Security and Communication Journal and Journal of Computer Networks and Communications Hindawi, and one of editorial board in the TEM (Technology, Education, Management, Informatics) Journal. He is also a reviewer of more than 50 Scopus-indexed journals. His research interests include watermarking, steganography, image encryption, cryptography, and image recognition. He can be contacted at email: moses@dsn.dinus.ac.id.



**Md Kamruzzaman Sarker**    is working as a tenure track assistant professor at the Department of Computing Sciences at the University of Hartford. He obtained his Ph.D. in computer science focusing on artificial intelligence (AI) in 2020 from Kansas State University. After his Ph.D., he also worked as a postdoc at the same university's Center for Artificial Intelligence and Data Science. He obtained his M.S. in Computer Science in 2018 from Wright State University and a B.Sc. in Computer Science and Engineering from Khulna University of Engineering and Technology. He also worked at Intel Corporation and Samsung Electronics. Furthermore, he has authored over 30 peer-reviewed papers and edited a book. He can be contacted at email: mdkamruzzamansarker@gmail.com.