Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control
Journal homepage: http://kinetik.umm.ac.id
ISSN: 2503-2267
Vol. 9, No. 3, August, Pp. 277-286

Pattern recognition of bima script handwriting using convolutional neural network method

Ghina Kamilah Ramadhani*[1], Fitri Bimantoro[1], I Gede Pasek Suta Wijaya[1]
Department of Informatics Engineering, Faculty of Engineering, Universitas Mataram, Mataram, Indonesia[1]

Article Info

Keywords:
Bima, Pattern Recognition, Convolutional Neural Network, Handwritting

Article history:
Received: February 23, 2024
Accepted: May 06, 2024
Published: August 31, 2024

Cite:
G. K. Ramdhani, F. Bimantoro, and I. G. P. S. Wijaya, "Pattern Recognition of Bima Script Handwritting using Convolutional Neural Network Method", KINETIK, vol. 9, no. 3, Aug. 2024.
https://doi.org/10.22219/kinetik.v9i3.1990

*Corresponding author.
Ghina Kamilah Ramdhani
E-mail address:
ghinakr6@gmail.com

Abstract

Bima is one of the regions in West Nusa Tenggara Province. The Bima script is a cultural heritage used as a means of communication by the Bima community in the past. The decline in the use of the Bima script threatens cultural heritage. The government has addressed this issue by providing training to teachers to teach it in schools, but this has still been insufficient due to the limited number of teachers participating in the training. Therefore, one efficient method to assist with this issue is by leveraging modern technology, particularly through machine learning for handwriting recognition. This study aims to find the best CNN model for recognizing the Bima script with diacritics to help preserve Bima's cultural heritage through handwriting recognition. The CNN model is combined with hyperparameter tuning, and then testing is conducted in four different scenarios to evaluate the performance of each model architecture and hyperparameter variation to find the best combination. The dataset used is sourced from the Kaggle platform, and augmentation is performed to increase the total number of images to 6,750, with each image containing 75 images in 90 different classes. In this study, testing is done by dividing the dataset into training and testing sets in an 80:20 ratio. The test results show high performance, achieving an accuracy of 98.00%, precision of 98.19%, recall of 98.00%, and f1-score of 98.00% in scenario 4.

1. Introduction

Bima is one of the administrative regions located in the province of West Nusa Tenggara (NTB). NTB consists of three ethnic groups, namely Sasak, Samawa, and Mbojo (Bima). Bima is the second-largest ethnic group after Sasak [1]. The Bima community has a cultural heritage, which includes a unique writing system known as the Mbojo script or Bima script [2]. The Bima script has historically been used as a medium of communication within the Bima community before the transition from the use of Malay-Arabic script to the Latin script. This script contains two patterns, namely the ancient pattern and the modern pattern, with the ancient model no longer in use except in historical documents such as the Raffles manuscript, suggesting its utilization at some point in the past [3]. Besides being a medium of communication, the Bima script also constitutes part of the cultural heritage and serves as a symbol of identity for the Bima community, which needs to be preserved properly. If this script is not preserved, it could become an endangered cultural heritage [1].

Based on a survey of 81 respondents, it was found that 48.1% of them have never learned about the Bima script, while 45.7% are unaware of the existence of the Bima script at all. This data indicates that the Bima script is facing extinction. The potential extinction can be attributed to several factors, including the lack of formal education about the Bima script in schools [4]. Therefore, the initial step taken by the Mayor of Bima was to introduce local content subjects through Mayor Regulation Number 50 of 2019 for elementary and junior high school students. In 2018, the government conducted training on teaching the Bima script to teachers. However, only 50 teachers participated in the training, which is still insufficient for this preservation effort [1]. In this context, to increase both the number and skills of teachers, more effective and innovative methods are needed. Therefore, the utilization of technological advancements in machine learning is necessary to recognize handwriting and process digital images to recognition of the Bima script. [2]. This technology can be implemented in learning applications to help increase the number of teachers and enhance their abilities in teaching the script.

In the rapidly advancing era of technology, character recognition through technology has become more feasible, especially through handwriting recognition. Handwriting recognition is the capacity of a system to identify handwritten text. Handwriting can originate from various sources, such as paper documents, images, touchscreens, or other devices [5]. The machine's process of handwriting recognition requires a training phase to enable it to perform its tasks by utilizing labeled training data from a dataset with each class. This process is carried out so that the machine can understand patterns and information contained in the training data [6]. The use of machine learning in handwriting recognition has made progress and made many significant contributions in the last few decades [7]. This progress has led to the development of many algorithms that have assisted in the field of machine learning in handwriting recognition, such as Arabic handwriting recognition using the Support Vector Machine (SVM) method [8], digit handwriting recognition using various machine learning methods like SVM, Decision Tree, Random Forest, K-Nearest Neighbor, and Artificial Neural Network (ANN) [9], and deep learning approaches like recognizing Gujarati characters using the Multi-Layer Perceptron Convolutional Neural Network (CNN) method [10].

Convolutional Neural Network (CNN) is deep learning inspired by how the human brain works. CNN is commonly used to detect and identify objects in various digital image datasets with diverse patterns and shapes, such as classifying medical images, automatically recognizing objects, facial recognition, and more [11]. The excellent performance of CNN is largely attributed to its more powerful computational capabilities, access to large datasets, and techniques for training deeper networks. CNN is claimed to be the optimal model for resolving problems related to recognition, especially in pattern recognition. As a result, it has become popular for recognizing handwritten characters across various languages and scripts [12]. Previous research utilized the CNN method for character and handwriting recognition has achieved high accuracy, including handwriting transliteration of Javanese script with an accuracy of 99.62% [13], Javanese script recognition using a 12-layer deep Convolutional Neural Network with an accuracy of 99.65% [14], recognition of Tamil characters with 95.16% accuracy [15], classification of Devanagari script numerals with 95% accuracy [16], and handwriting digit recognition with a remarkably high accuracy of 99.87% [17].

Previous studies on the Bima script were conducted by various researchers, including Fitri et al., who focused on pattern recognition of Bima characters using texture features with the K-Nearest Neighbor method, resulting an accuracy of 60.86% [1]. Arik et al. conducted research on transliterating Latin letters into Bima script using an Android application implementing hexadecimal algorithms, achieving a high accuracy of 99.36% [2]. Furthermore, Arik et al. utilized a rule-based hexadecimal approach to translate Bima script to Latin letters with an accuracy of 90.64% [3]. Mustiari et al. combined the histogram of an oriented gradient with backpropagation in recognizing handwritten Bima characters, achieving an accuracy of 97.70% [4]. Ilham et al. combined local binary pattern feature extraction with K-Nearest Neighbor in Bima script recognition, resulting in an accuracy of 86.056% [18]. Naufal et al. explored handwritten pattern recognition by combining GLCM extraction with zoning and the classification of probabilistic neural networks, achieving an accuracy of 81.35% [19]. Rizqullah et al. utilized the Artificial Neural Network method with moment invariant feature extraction in the recognition of Bima script with diactrics, resulting in an accuracy of 77.59% [20].

In this study, the recognition of handwritten Bima script with diacritics will be conducted using the CNN method. The difference between this research and the previous researches based on the reference [20] is that this research focuses on utilizing the CNN method. The same dataset is used, but in this research, augmentation is performed to increase the number of images to a total of 6,750. Additionally, this research will compare the CNN architecture from the previous researches with three architectures proposed by the author. These three architectures will be tested using hyperparameter tuning to obtain the best model architecture. The accuracy achieved by the CNN method is significantly better compared to the previous research that used the ANN method in recognizing Bima script patterns with diacritics.

Based on the existing issues, this research aims to find a CNN model to recognize patterns of handwritten Bima script with diacritics. This model is expected to be implemented in educational applications accessible across various platforms, such as iOS, Android, and others. This research is expected to make a valuable contribution by helping the government, particularly educators, enhance their understanding and skills in teaching, thus preserving and ensuring the continuity of the cultural heritage of the Bima script.

2. Research Method

In this section, the method used for the patterns recognition of handwritten of Bima script with diacritics is a type of deep learning called Convolutional Neural Network (CNN). This research also presents a combination of using Hyperparameter Tuning to obtain a good combination of model and hyperparameters.

2.1 Dataset

The dataset used in this research is a handwritten Bima script dataset with diactrics obtained from Kaggle platform [21]. This dataset consists of 90 classes with each class containing 25 images, resulting in a total of 2,250 images. This dataset consists of 90 classes, with each class containing 25 images, resulting in a total of 2,250 images. The dataset was obtained from five volunteers from the general public. Each volunteer wrote the script in 5 sets (90 characters per set), with a default stroke size of 10.0. Data collection was conducted using an Android-based application developed using the Flutter framework. This application was chosen for data collection due to its time and processing efficiency, as well as its ability to save paper and facilitate data collection. Additionally, due to the COVID-19 pandemic, data collection could be conducted online [20]. In this dataset, subsets (a) and (b) each display 45 sample images of the Bima script. Thus, the entire dataset consists of 90 sample images which can be seen in Figure 1.

(b)

Figure 1. Bima Handwritten Dataset

2.2 Preprocessing Data

Preprocessing is an initial step in handwritten recognition. In the initial stage, cropping is performed on the dataset to eliminate irrelevant sides and reduce the number of data rows [22], [23]. The next stage involves resizing to standardize the image size, with a size of 64x64 used in this dataset [12]. After all the images are of the same size, the next step is to convert the image dimensions to grayscale. The purpose of this grayscale conversion is to ensure that the images have only one-color dimension for efficient and speedy training. Furthermore, the augmentation process, which includes adding variations to the data to generate additional data within the original dataset, aims to increase the number of datasets [24]. The augmentations used in this study include zoom range, width shift, height shift, and shear range. The number of datasets after augmentation is 75 for each class, totaling 6,750 images.

The next step in this study involves dividing the dataset into two parts, namely training data and testing data. The dataset division used in this study is 80:20, where 80% of the total data will be categorized as training data, while the remaining 20% will be categorized as testing data [25].

2.3 Implementing of CNNs Model

Convolutional Neural Network (CNN) is a type of feedforward neural network which can consist of one or several convolutional layers. Similar to multilayer neural networks, CNNs can also include one or more fully connected layers. The CNN structure is specifically designed to process input in the form of 2D images [26]. A significant characteristic of CNN is the ability to perform end-to-end learning without requiring manual feature engineering, an advantage not owned by other conventional methods [27]. Although there are many variations in CNN architectures, their fundamental elements are quite similar. CNN comprises three basic types of layers: convolutional layers, pooling layers, and fully connected layers [28], as shown in Figure 2.

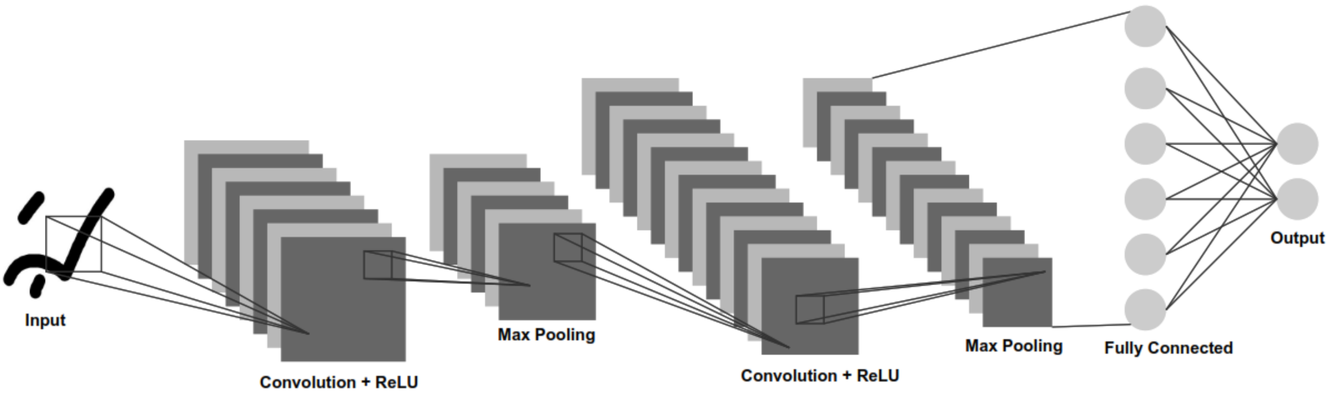

Figure 2. Example of CNN Architecture

The model structure in this research includes three basic layer components. The addition of a dropout layer is planned after the convolutional layer and pooling layer, then placed before the fully connected layer. This is aimed at avoiding overfitting and enhancing the overall performance of the model [29]. The following diagram reflects the implementation steps of the CNN model architecture on Bima script dataset starting from preprocessing, model design, to model evaluation. The workflow of CNN is illustrated in Figure 3.

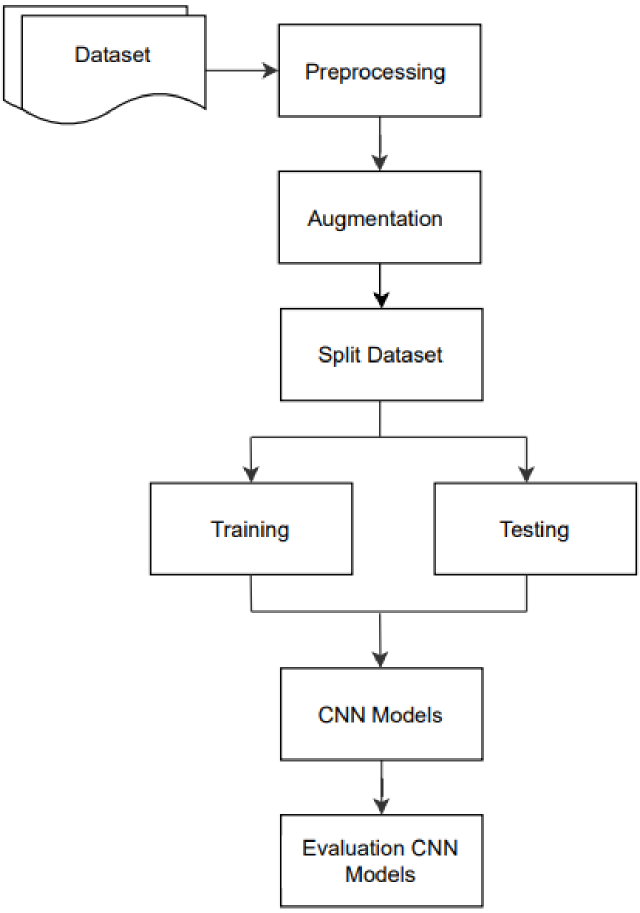

Figure 3. CNN Architecture Workflow

In this study, several CNN architectures are used to test and achieve the best accuracy in recognizing handwritten Bima script patterns. One of the architectures to be implemented is taken from previous research on recognizing Sasak script, which has successfully achieved high accuracy [30]. Model 1 will be implemented and compared with the proposed architectures in Model 2, 3, and 4 to evaluate the performance of each model architecture. Details of the CNN architecture can be seen in Table 1.

Table 1. CNN Architecture Models

Architecture	Layer & Filter
Model 1 [30]	Input (64x64) Conv2D 128 (5x5), ReLU Conv2D 128 (5x5), ReLU Conv2D 128 (5x5), ReLU Dropout: 0.5 Flatten () Dense 128, ReLU Dense 128, ReLU Dense 128, ReLU Dense 90, Softmax
Model 2	Input (64x64) Conv2D 16, ReLU Conv2D 32, ReLU Conv2D 64, ReLU Conv2D 128, ReLU Dropout Flatten Dense 256, ReLU Dense 128, ReLU Dense 90, Softmax
Model 3	Input (64x64) Conv2D 16, ReLU Conv2D 16, ReLU Conv2D 32, ReLU Conv2D 32, ReLU Conv2D 64, ReLU Conv2D 128, ReLU Dropout Flatten Dense 256, ReLU Dense 128, ReLU Dense 90, Softmax
Model 4	Input (64x64) Conv2D 32, ReLU Conv2D 32, ReLU Conv2D 64, ReLU Conv2D 64, ReLU Conv2D 128, ReLU Dropout Flatten Dense 128, ReLU Dense 64, ReLU Dense 32, ReLU Dense 90, Softmax

2.4 Hyperparameter Tuning

Hyperparameters are parameters that need to be configured before the learning process begins in a machine learning algorithm. Hyperparameters can be set manually, and their accuracy can be calculated [31]. Adjusting hyperparameters provides optimal parameters for the CNN model. Hyperparameters involve variables such as the number of output channels, kernel size, and activation function. Different combinations of hyperparameters create various possibilities for the best model architecture [32]. In this study, hyperparameter tuning is used to obtain the best performance. The details of the hyperparameter comparison can be seen in Table 2.

Table 2. Hyperparameter Values

Hyperparameter	Value
Kernel	3x3, 5x5
Dropout	0.2, 0.5

3. Results and Discussion

In this study, there are four models that will be tested with various parameters to determine the most suitable CNN model and hyperparameters for the Bima script dataset. The table below illustrates five testing scenarios designed to find the best accuracy and to evaluate the performance of each model in classifying the Bima script. Details of the scenarios can be seen in Table 3.

Table 3. Scenarios Descriptions

Scenarios	Description
Scenario 1	Implementation of CNN Model 1 from the previous research
Scenario 2	Model 2 with the best hyperparameter tuning accuracy results
Scenario 3	Model 3 with the best hyperparameter tuning accuracy results
Scenario 4	Model 4 with the best hyperparameter tuning accuracy results
Scenario 5	Analysis and selecting the best performance from scenarios

3.1 Testing Scenario 1

In scenario 1, Model 1 was adopted from a previous research journal [30]. This model was initially used to recognize Sasak script patterns with a success rate of 99.31% accuracy. In the Sasak script, the dataset used was obtained from handwritten text scans on A4 paper, which has been given columns to make the data retrieval process easier. Meanwhile, for the Bima script, the dataset is taken using a digital handwriting application and stored in image form. The Sasak script has 28 classes, while the Bima script has 90 classes.

The model configuration used to implement the Bima script includes the use of a 5x5 kernel, a dropout of 0.5, and a ReLU activation function. The model training process was carried out for 100 epochs to ensure that the model was well trained. With a total parameter of 1,916,122, it shows that the model has a high level of complexity which allows it to handle datasets with a large number of classes such as the Bima script.

The evaluation of Model 1performance shows good ability in classifying the Bima script dataset with an accuracy of 97.92%, precision of 98.00%, recall of 97.92%, and f1-score of 97.91%. This confirms that this model can also recognize Bima script handwriting patterns with a good level of accuracy.

3.2 Testing Scenario 2

In scenario 2, Model 2 was tested using hyperparameter tuning. It can be seen that using a 3x3 kernel provides higher accuracy when combined with a dropout of 0.2 compared to a dropout of 0.5. However, when using a 5x5 kernel, Model 2 is more suitable with a dropout of 0.5 and provides more optimal results. The best combination of hyperparameters in Model 2 is to use a 5x5 kernel and a dropout of 0.5 because it can produce the highest accuracy compared to other combinations of hyperparameters. The best combination produces parameters of 838,490. The experimental results can be seen in Table 4.

Table 4. Scenario 2 Performance Results

Parameters		Accuracy	Precision	Recall	F1-Score
Kernel	Dropout				
3x3	0.2	97,48%	97,71%	97,81%	97,48%
3x3	0.5	97,11%	97,46%	97,11%	97,13%
5x5	0.2	97,40%	97,64%	97,40%	97,41%
5x5	**0.5**	**97,62%**	**97,84%**	**97,62%**	**97,63%**

3.3 Testing Scenario 3

In scenario 3, Model 3 was tested using hyperparameter tuning resulting in four combinations. In Model 3, using a 5x5 kernel is more suitable than a 3x3 kernel. Meanwhile, a model with a dropout rate of 0.2 is more suitable than a dropout rate of 0.5 because it can help the model learn patterns better and avoid losing important information during the training process that might reduce the model's ability to understand the dataset well. Therefore, the best hyperparameter combination is the use of a 5x5 kernel and a dropout of 0.2 with a total parameter of 379,018. The performance result was quite good, even though this scenario produced the lowest accuracy compared to the others. The results of the experiment can be seen in Table 5.

Table 5. Scenario 3 Performance Results

Parameters		Accuracy	Precision	Recall	F1-Score
Kernel	Dropout				
3x3	0.2	95,85%	96,24%	95,85%	95,87%
3x3	0.5	95,55%	95,96%	95,55%	95,56%
5x5	**0.2**	**95,92%**	**96,41%**	**95,92%**	**95,95%**
5x5	0.5	95,85%	96,30%	95,85%	95,85%

3.4 Testing Scenario 4

In scenario 4, Model 4 was tested using hyperparameter tuning which produced four combinations. In Model 4, using a 5x5 kernel is more suitable than a 3x3 kernel because the 5x5 kernel has more space to capture complex features in the image. Meanwhile, a model with a dropout rate of 0.5 has better results than a dropout rate of 0.2 because a higher dropout rate tends to be more effective in preventing overfitting. Therefore, the best hyperparameter combination for Model 4 is the use of a 5x5 kernel and a dropout rate of 0.5, with a total of 464,090 parameters. The performance produced by the combination of model and hyperparameter tuning achieves the highest accuracy compared to other scenarios. The experiment results can be seen in Table 6.

Table 6. Scenario 4 Performance Results

Parameters		Accuracy	Precision	Recall	F1-Score
Kernel	Dropout				
3x3	0.2	96,22%	96,55%	96,22%	96,22%
3x3	0.5	96,29%	96,73%	96,29%	96,31%
5x5	0.2	96,44%	96,87%	96,44%	96,47%
5x5	**0.5**	**98,00%**	**98,19%**	**98,00%**	**98,00%**

3.5 Scenario Evaluation Results

In scenario 5, an analysis and selection of the best model from scenarios 1 to 4 was conducted. In this experiment, each model was trained for 100 epochs, using input images sized 64x64, along with applying maxpooling operations of 2x2, softmax activation, and the Adam optimizer with a learning rate of 0.0001. The analysis shows that the highest accuracy occurred in scenario 4 with a value of 98.00%, indicating that the model in this scenario was capable of recognizing handwriting patterns with very high accuracy. This was followed by scenario 1 with an accuracy of 97.92%, and scenario 2 with an accuracy of 97.62%. Meanwhile, scenario 3 showed a lower accuracy of 95.92%. The performance results can be seen in Table 7.

In scenarios 1, 2, and 4, the highest accuracy is found in the hyperparameter combination with a 5x5 kernel and a dropout of 0.5. Meanwhile, in scenario 3, the highest accuracy was found in the 5x5 kernel combination with a dropout of 0.2. This indicates that the model configuration with a 5x5 kernel and a dropout of 0.5 tends to provide better results in recognizing the dataset.

The total parameter results show that scenario 3 has the fewest parameters at 379,018, followed by scenario 4 at 464,090, then scenario 2 at 838,490, and the largest parameter is in scenario 1 at 1,916,122. Model with a large number of parameters has a greater capacity to learn complex datasets, but also runs the risk of overfitting. Meanwhile, models with fewer parameters have limited capacity to learn the dataset, but tend to generalize patterns better and are more efficient in terms of computational time compared to models with a large number of parameters.

Based on this analysis, Model 4 in scenario 4 can be considered the best model as it achieves the highest accuracy, precision, recall, and F1-Score with fewer parameters than Model 1. Although Model 1 has a larger capacity, Model 4 achieves better performance with a simpler structure. Additionally, Model 4 also has the advantage of faster training time, being able to train the model up to 50% faster than Model 1. This demonstrates efficiency in parameter usage and the model's capability to learn complex patterns from the dataset.

Additionally, the difference between using CNN and traditional machine learning methods in recognizing Bima script, such as KNN [1], [18], Rule Base Hexadecimal [3], and backpropagation [4], is in using the feature extraction process. Previous studies have shown that traditional methods require manual feature extraction from input images, while the CNN can automatically extract important features. CNN models can also be easily adapted to handle more complex problems by adding layers or increasing the dataset size. In Table 7, it can be seen that the use of the CNN method in Model 4 produces the highest accuracy compared to other traditional machine learning methods in recognizing Bima script. Although the use of the backpropagation method [4] also produces high accuracy, there is a difference in the number of dataset classes, where the dataset used is less varied with only 22 classes. Moreover, in a previous study using ANN [20] with the same dataset, the use of CNN in Model 4 was able to increase accuracy by 20%. This indicates that using the CNN method with hyperparameter tuning is significantly effective to improve accuracy in recognizing patterns of Bima characters with diacritics.

Table 7. Model Performance Results

Model	Accuracy	Precision	Recall	F1-Score
Model 1	97,92%	98,00%	97,92%	97,91%
Model 2	97,62%	97,84%	97,62%	97,63%
Model 3	95,92%	96,41%	95,92%	95,95%
Model 4	**98,00%**	**98,19%**	**98,00%**	**98,00%**
KNN [1]	60,86%	60,86%	60,86%	-
Rule Base [3]	90,64%	90,64%	90,64%	-
Backpropagation [4]	97,70%	97,72%	97,65%	-
KNN [18]	86,06%	86,06%	86,06%	-
ANN [20]	77,59%	78,44%	77,61%	77,33%

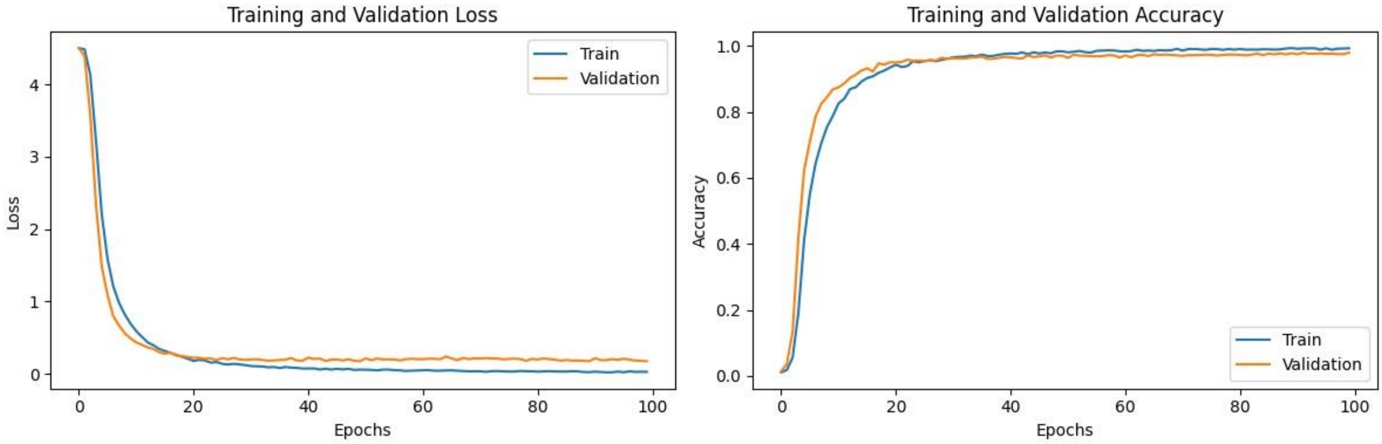

Figure 4. Graph of the Highest Accuracy Results for Model 4

The graph in Figure 4 shows the highest accuracy from scenario 4. It can be seen that the train loss consistently decreases over time, followed by the valid loss which shows a stable decline in line with the trend of the train loss in recognizing patterns within the training data. From these results, it can be concluded that the model's performance remains consistent on validation data throughout the training process. In this scenario, the model achieved the highest accuracy compared to others reaching an accuracy of 98.00%. This indicates that the model successfully recognizes patterns of Bima characters without overfitting.

4. Conclusion

Based on the research findings, several conclusions can be drawn. This study aims to find the best CNN model in recognizing handwriting patterns of Bima script with diacritics, consisting of 90 classes. The selection of the CNN method is based on its excellent ability to recognize the Bima script, and the proposed CNN model has been combined with hyperparameter tuning. The highest performance is achieved by Model 4 with an accuracy of 98.00%, precision of 98.19%, recall of 98.00%, and F1-Score of 98.00%. Additionally, this research also aims to preserve the Bima culture, especially the Bima script, which is endangered, by leveraging technology. The next step is to implement this model into educational applications, which can be used according to educational categories. Thus, this research provides a positive contribution in an academic context and has a direct impact on preserving the valuable cultural heritage of the Bima community. For future research, exploring transfer learning techniques is recommended.

References

[1] F. Bimantoro, A. Aranta, G. S. Nugraha, R. Dwiyansaputra, and A. Y. Husodo, “Pengenalan Pola Tulisan Tangan Aksara Bima menggunakan Ciri Tekstur dan KNN,” *Journal of Computer Science and Informatics Engineering (J-Cosine)*, vol. 5, no. 1, pp. 60–67, Jun. 2021. https://doi.org/10.29303/jcosine.v5i1.387.

[2] A. Aranta *et al.*, “Learning media for the transliteration of Latin letters into Bima script based on android applications,” *Journal of Education and Learning (EduLearn)*, vol. 15, no. 2, pp. 275–282, May 2021. https://doi.org/10.11591/edulearn.v15i2.19013.

[3] A. Aranta, F. Bimantoro, and I. P. T. Putrawan, “Penerapan Algoritma Rule Base dengan Pendekatan Hexadesimal pada Transliterasi Aksara Bima Menjadi Huruf Latin,” *Jurnal Teknologi Informasi, Komputer, dan Aplikasinya (JTIKA)*, vol. 2, no. 1, pp. 130–141, Mar. 2020. https://doi.org/10.29303/jtika.v2i1.96.

[4] Mustiari, F. Bimantoro, G. S. Nugraha, and A. Y. Husodo, “Bima script handwriting pattern recognition using histogram of oriented gradients and backpropagation classification method,” in *AIP Conference Proceedings*, American Institute of Physics Inc., Feb. 2023. https://doi.org/10.1063/5.0111795.

[5] N. Altwaijry and I. Al-Turaiki, “Arabic handwriting recognition system using convolutional neural network,” *Neural Comput Appl*, vol. 33, no. 7, pp. 2249–2261, Apr. 2021. https://doi.org/10.1007/s00521-020-05070-8.

[6] 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*, vol. 9, no. 5, pp. 4446–4451, Oct. 2019. https://doi.org/10.11591/ijece.v9i5.pp4446-4451.

[7] T. Q. Vinh, L. H. Duy, and N. T. Nhan, “Vietnamese handwritten character recognition using convolutional neural network,” *IAES International Journal of Artificial Intelligence*, vol. 9, no. 2, pp. 276–283, Jun. 2020. https://doi.org/10.11591/ijai.v9.i2.

[8] M. Athoillah and R. K. Putri, “Handwritten Arabic Numeral Character Recognition Using Multi Kernel Support Vector Machine,” *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, pp. 99–106, Mar. 2019. https://doi.org/10.22219/kinetik.v4i2.724.

[9] R. KARAKAYA and S. KAZAN, “Handwritten Digit Recognition Using Machine Learning,” *Sakarya University Journal of Science*, vol. 25, no. 1, pp. 65–71, Feb. 2021. https://doi.org/10.16984/saufenbilder.801684.

[10] J. Pareek, D. Singhania, R. R. Kumari, and S. Purohit, “Gujarati Handwritten Character Recognition from Text Images,” in *Procedia Computer Science*, Elsevier B.V., 2020, pp. 514–523. https://doi.org/10.1016/j.procs.2020.04.055.

[11] 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. https://doi.org/10.3390/a15040129.

[12] G. S. Nugraha, M. I. Darmawan, and R. Dwiyansaputra, “Comparison of CNN’s Architecture GoogleNet, AlexNet, VGG-16, Lenet -5, Resnet-50 in Arabic Handwriting Pattern Recognition,” *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, May 2023. https://doi.org/10.22219/kinetik.v8i2.1667.

[13] F. Ilham and N. Rochmawati, “Transliterasi Aksara Jawa Tulisan Tangan ke Tulisan Latin Menggunakan CNN,” *Journal of Informatics and Computer Science (JINACS)*, vol. 1, no. 04, pp. 200–208, Jul. 2020. https://doi.org/10.26740/jinacs.v1n04.p200-208.

[14] A. Susanto, I. U. W. Mulyono, C. A. Sari, E. H. Rachmawanto, D. R. I. M. Setiadi, and M. K. Sarker, “Handwritten Javanese script recognition method based 12-layers deep convolutional neural network and data augmentation,” *IAES International Journal of Artificial Intelligence*, vol. 12, no. 3, pp. 1448–1458, Sep. 2023. https://doi.org/10.11591/ijai.v12.i3.pp1448-1458.

[15] B. R. Kavitha and C. Srimathi, “Benchmarking on offline Handwritten Tamil Character Recognition using convolutional neural networks,” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 4, pp. 1183–1190, Apr. 2022. https://doi.org/10.1016/j.jksuci.2019.06.004.

[16] D. S. Prashanth, R. V. K. Mehta, and N. Sharma, “Classification of Handwritten Devanagari Number - An analysis of Pattern Recognition Tool using Neural Network and CNN,” in *Procedia Computer Science*, Elsevier B.V., 2020, pp. 2445–2457. https://doi.org/10.1016/j.procs.2020.03.297.

[17] S. Ahlawat, A. Choudhary, A. Nayyar, S. Singh, and B. Yoon, “Improved handwritten digit recognition using convolutional neural networks (Cnn),” *Sensors (Switzerland)*, vol. 20, no. 12, pp. 1–18, Jun. 2020. https://doi.org/10.3390/s20123344.

[18] M. I. Fidatama, F. Bimantoro, G. S. Nugraha, B. Irmawati, and R. Dwiyansaputra, “Recognition of Bima script handwriting patterns using the local binary pattern feature extraction method and K-nearest neighbour classification method,” in *AIP Conference Proceedings*, American Institute of Physics Inc., Feb. 2023. https://doi.org/10.1063/5.0111770.

[19] M. Naufal, F. Bimantoro, A. Aranta, and I. G. P. S. Wijaya, “Bima script handwriting pattern recognition with gray level co-occurrence matrix feature extraction and zoning & classification of probabilistic neural networks,” in *AIP Conference Proceedings*, American Institute of Physics Inc., Feb. 2023. https://doi.org/10.1063/5.0111802.

[20] M. N. Rizqullah, R. Dwiyansaputra, and F. Bimantoro, “Pengenalan Pola Suku Kata Aksara Bima Dengan Baris Tanda Bunyi Menggunakan Ekstraksi Ciri Moment Invariant Dengan Metode ANN,” *Jurnal Teknologi Informasi, Komputer dan Aplikasinya (JTIKA)*, vol. 6, no. 1, pp. 264–274, Mar. 2024.

[21] M. N. Rizqullah, “Bimanese Script,” 2021.

[22] E. H. Rachmawanto and P. N. Andono, “Deteksi Karakter Hiragana Menggunakan Metode Convolutional Neural Network,” *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, vol. 11, no. 3, pp. 183–191, Dec. 2022. https://doi.org/10.23887/janapati.v11i3.50144.

[23] A. Mulyanto, E. Susanti, F. Rossi, W. Wajiran, and R. I. Borman, “Penerapan Convolutional Neural Network (CNN) pada Pengenalan Aksara Lampung Berbasis Optical Character Recognition (OCR),” *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, vol. 7, no. 1, p. 52, Apr. 2021. https://doi.org/10.26418/jp.v7i1.44133.

[24] I. Maliki and A. S. Prayoga, “Implementation of Convolutional Neural Network for Sundanese Script Handwriting Recognition with Data Augmentation,” *Prayoga Journal of Engineering Science and Technology*, vol. 18, no. 2, pp. 1113–1123, 2023.

[25] R. Aryanto, M. Alfan Rosid, and S. Busono, “Penerapan Deep Learning untuk Pengenalan Tulisan Tangan Bahasa Aksara Lota Ende dengan Menggunakan Metode Convolutional Neural Networks,” *Jurnal Informasi dan Teknologi*, pp. 258–264, May 2023. https://doi.org/10.37034/jidt.v5i1.313.

[26] S. Abbas *et al.*, “Convolutional neural network based intelligent handwritten document recognition,” *Computers, Materials and Continua*, vol. 70, no. 3, pp. 4563–4581, 2022. https://doi.org/10.32604/cmc.2022.021102.

[27] D. Das, D. R. Nayak, R. Dash, B. Majhi, and Y. D. Zhang, “H-WordNet: A holistic convolutional neural network approach for handwritten word recognition,” *IET Image Process*, vol. 14, no. 9, pp. 1794–1805, Jul. 2020. https://doi.org/10.1049/iet-ipr.2019.1398.

[28] I. Khandokar, M. Hasan, F. Ernawan, S. Islam, and M. N. Kabir, “Handwritten character recognition using convolutional neural network,” in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Jun. 2021. https://doi.org/10.1088/1742-6596/1918/4/042152.

[29] A. E. Minarno, M. Hazmi Cokro Mandiri, Y. Munarko, and H. Hariyady, “Convolutional Neural Network with Hyperparameter Tuning for Brain Tumor Classification,” *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, May 2021. https://doi.org/10.22219/kinetik.v6i2.1219.

[30] R. Rismawandi, I. G. P. S. Wijaya, and G. S. Nugraha, “Implementasi Metode Convolutional Neural Network Untuk Pengenalan Huruf Aksara Sasak Pada Android,” *Jurnal Teknologi Informasi, Komputer dan Aplikasinya (JTIKA)*, vol. 4, no. 1, pp. 11–20, Mar. 2022.

[31] E. Elgeldawi, A. Sayed, A. R. Galal, and A. M. Zaki, “Hyperparameter tuning for machine learning algorithms used for arabic sentiment analysis,” *Informatics*, vol. 8, no. 4, Dec. 2021. https://doi.org/10.3390/informatics8040079.

[32] M. Munsarif, E. Noersasongko, P. N. Andono, and M. A. Soeleman, “Improving convolutional neural network based on hyperparameter optimization using variable length genetic algorithm for english digit handwritten recognition,” *International Journal of Advances in Intelligent Informatics*, vol. 9, no. 1, pp. 66–78, Mar. 2023. https://doi.org/10.26555/ijain.v9i1.881.