Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control
Journal homepage: http://kinetik.umm.ac.id
ISSN: 2503-2267
Vol. 10, No. 1, February, Pp. 53-62

An ensemble learning layer for wayang recognition using CNN-based ResNet-50 and LSTM

Candra Irawan*[1], Eko Hari Rachmawanto[1], Heru Pramono Hadi[1]
Universitas Dian Nuswantoro, Semarang, Indonesia[1]

Article Info

Keywords:
Wayang, Classification, CNN, RestNet-50, LSTM

Article history:
Received: July 08, 2024
Accepted: November 04, 2024
Published: February 01, 2025

Cite:
C. Irawan, E. H. Rachmawanto, and H. P. Hadi, "An Ensemble Learning Layer for Wayang Recognition using CNN-based ResNet-50 and LSTM", *KINETIK*, vol. 10, no. 1, Feb. 2025.
https://doi.org/10.22219/kinetik.v10i1.2053

*Corresponding author.
Candra Irawan
E-mail address:
candra.irawan@dsn.dinus.ac.id

Abstract

Wayang is commonly used to tell epic stories of Mahabharata and Ramayana, as well as local legends and myths. There are various types of wayang, such as wayang kulit (made of buffalo or goat leather), wayang golek (made of wood), and wayang klithik (combination of leather and wood). Although it indicates cultural richness, such diversity also makes it difficult for the general public to identify the character of wayang they are seeing because each type has unique characteristics and details. Recognizing wayang characters is a challenging task due to their intricate designs and subtle variations. This research addresses this problem by leveraging machine learning technology, specifically CNN-based classification methods, to accurately identify wayang characters. This study proposed a novel method that integrates ResNet-50 transfer learning with LSTM, enhancing the model's ability to capture both spatial and sequential features of wayang images. The proposed model achieved an impressive accuracy of 97.92%, with precision, recall, and F1-scores all reaching 100%. Despite the extended training time of 188 minutes and 21 seconds, the results demonstrate the model's superior performance. This advancement can significantly aid in the preservation and educational dissemination of Indonesian cultural heritage. Future research can focus on optimizing the training process to reduce the time while maintaining or even improving the accuracy, potentially expanding the model's application scope and effectiveness.

1. Introduction

Wayang is a traditional performing art that is deeply rooted in Indonesian culture, especially in Java and Bali [1]. This wayang art has been around for centuries and is an important part of Indonesia's cultural heritage. Wayang is commonly used to tell epic stories of Mahabharata and Ramayana, as well as local legends and myths [2]. There are various types of wayang, such as wayang kulit (made of buffalo or goat leather), wayang golek (made of wood), and wayang klithik (combination of leather and wood) [3]. Although it indicates cultural richness, such diversity makes it difficult for the general public to identify the character of wayang they are seeing because each type has unique characteristics and details. In addition, each type of wayang has different variations in character and details, which makes the identification process even more complicated. The lack of easily accessible and structured educational resources also adds to this difficulty, so the general public often feel confused in recognizing it. The solution to this problem can be found in the application of machine learning technology based on deep learning [4], [5]. By collecting image data of different types of wayang and labeling them based on their category, a deep learning model can be trained to recognize the patterns and characteristics of each character of wayang [6], [7], [8]. After training the model, it can be used to determine the wayang character from a given image with high accuracy. The use of this deep learning-based application not only facilitates the public identification process, but also helps preserve wayang art and culture by providing a sophisticated and accessible educational tool [9], [10].

Many researchers implement CNN (convolutional neural networks)-based machine learning to classify wayang images in their research. Wibawa et al. [11] used classic Convolutional Neural Network (CNN) to classify characters in Wayang Kulit. Their study used 100 black and white wayang images downloaded one by one as the dataset. Through a training process using CNN run on Google Colab to speed up the training process, they succeeded in developing a model that is able to test the wayang images. The results of their study show that the proposed CNN model can distinguish good and evil characters in Wayang Kulit with an accuracy rate of 92%, indicating that CNN is effective in identifying Wayang Kulit characters. Banjaransari et al. [12] introduced a transfer learning approach using Convolutional Neural Network (CNN) models, namely MobileNetV2 and VGG16, which were then followed by fine-tuning. The dataset consisted of 3,000 images divided into 30 classes. This dataset was divided into training and testing data to train and evaluate the model. The evaluation results show that the MobileNetV2 model achieved precision, recall, F1-score, and accuracy of 95%, 94%, 94%, and 94.17%, respectively.

Sudiatmika et al. [13] introduced a Region-based Convolutional Neural Networks (R-CNN) model to achieve notable success in object recognition and detection. In their study, they trained Mask R-CNN to detect and recognize patterns of Indonesian Shadow Wayang. Wayang is a traditional Indonesian art form that depicts stories, including those of Mahabharata. The objective of this research was to preserve Wayang from extinction through systematic pattern recognition. This study utilized Mask R-CNN for the algorithm. Training Mask R-CNN to recognize wayang requires numerous labels, which are difficult to obtain. Hence, the authors performed data augmentation to enhance the dataset and improve the training process. Testing was conducted using various learning rates on a Cloud GPU. The results showed a training accuracy of 92.04% and a validation accuracy of 92.04% with a learning rate of 0.0001. Similarly, Wisnudhanti et al. [14] used Convolutional Neural Networks (CNN) to classify characters from Pandawa narrative in wayang performances, which include Yudhistira, Bima, Arjuna, Nakula, and Sadewa. The research utilized CNN, implemented on a Raspberry Pi 4, for the classification task. CNN was chosen due to its effectiveness in handling image classification tasks with multiple classes. The network architecture consisted of three convolutional layers, three hidden layers, and one output layer. The dataset comprised 1000 images sized at 100 × 100 pixels, with 80% allocated for training and 20% for testing. The results demonstrated a training accuracy of 97.88% and a test accuracy of 96.5%.

Based on the findings from the aforementioned research, this study proposes a novel approach in the field of wayang image classification. Leveraging Transfer Learning using ResNet-50 combined with LSTM, this study aims to achieve superior accuracy compared to the previous studies. Although Wibawa et al. [11] and Banjaransari et al. [12] employed classic CNN architectures with notable success, achieving accuracies of 92% and 94.17% respectively, this proposed approach seeks to surpass those benchmarks. Different from Sudiatmika et al. [13] who used R-CNN and achieved a training and validation accuracy of 92.04%, this study aims to integrate Transfer Learning with LSTM to capture temporal dependencies in wayang narratives, potentially enhancing accuracy beyond the previous study conducted by Wisnudhanti et al. [14] who achieved 97.88% in training and 96.5% in testing accuracy. This innovative combination of Transfer Learning and LSTM promises to advance the field of wayang image classification by effectively leveraging pretrained ResNet-50 features and sequential learning mechanisms to achieve more robust and accurate classification results.

The structure of this paper is organized as follows: Section 1 provides an introduction, including an analysis of the problem, proposed solutions, and a review of relevant literature. This section discusses the challenges faced by the general public in identifying various types of wayang characters and proposes a solution using a deep learning-based machine learning approach. Section 2 outlines the research method, explaining the dataset preparation, model architecture and the experimental setup used in this study. Section 3 presents the results and discussion, comprising the analysis of the proposed model performance through various evaluation metrics. Finally, Section 4 concludes the paper, summarizing the findings and suggesting potential future work in this area.

2. Research Method

In this study, the proposed model for classifying wayang characters used a combination of Traditional CNN and LSTM layers. The dataset was split into 80% training data and 20% testing data. During the data collection, the images were prepared and processed to be ready for the analysis. In the testing phase, an enhancement step was applied to improve image quality before the classification, ensuring more accurate results. For the training, this study employed Resnet-50-CNN layers to extract spatial features from the images, followed by LSTM layers to capture temporal dependencies. The integration of CNN and LSTM layers was designed to enhance the model's performance, as evaluated using a confusion matrix, demonstrating the effectiveness of this approach in accurately classifying the wayang characters.

The proposed model was evaluated using a confusion matrix. This metric allows for a detailed analysis of the model's performance by providing insights into the true positives, false positives, true negatives, and false negatives. By using a confusion matrix, the accuracy, precision, recall, and F1-score of the proposed model can be assessed, ensuring a comprehensive evaluation of its classification capabilities. The flow of the proposed method can be seen in Figure 1.

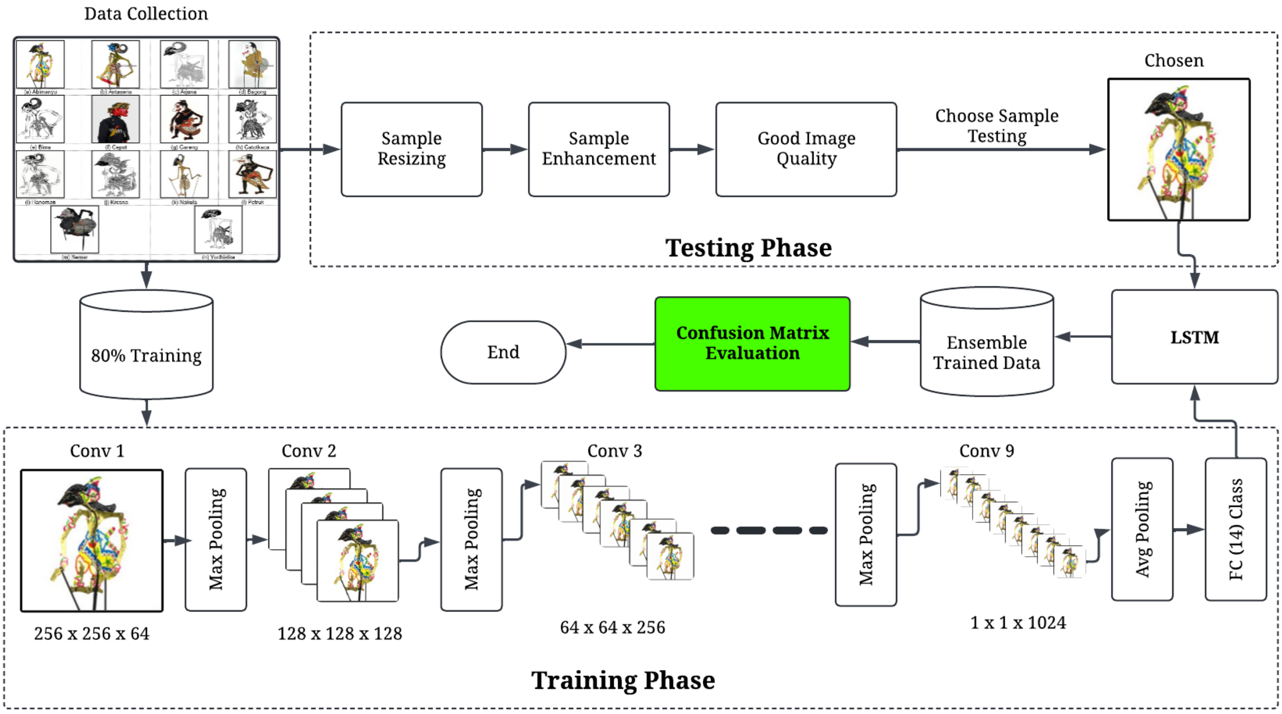

Figure 1. Proposed Method

2.1 Data Collection

In this study, a PNG dataset comprising 14 classes of wayang characters was utilized, including Abimanyu, Antasena, Arjuna, Bagong, Bima, Cepot, Gareng, Gatotkaca, Hanoman, Kresna, Nakula, Petruk, Semar, and Yudhistira. Each class contained a total of 80 sample images, which were randomly selected and converted to a size of 256 x 256 pixels to facilitate the model training process. The selection of this resolution aims to maintain a balance between sufficient visual detail and computational efficiency during the training process. This dataset was taken from Kaggle, which is known as a rich and diverse data source for various machine learning studies [15]. The use of this randomized and standardized dataset ensures that the proposed model can generalize well to the existing variations of wayang images, thereby increasing the accuracy and reliability of the classification in the context of real applications. The sample datasets depicted in Figure 2 represent images that have been preprocessed and are ready for training.

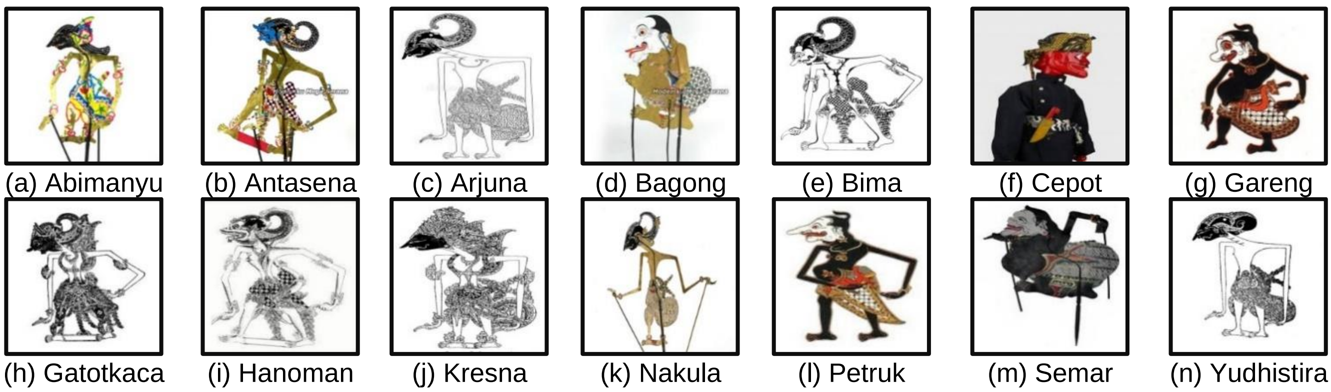

Figure 2. Sample Datasets

2.2 Convolutional Neural Networks (CNN)-based Resnet-50

The ResNet-50 architecture [16], [17], [18], [19] is a powerful CNN designed to facilitate the training of very deep networks by introducing residual learning. The architecture consists of 50 layers, including multiple convolutional layers, pooling layers, and fully connected layers, arranged into a chain of residual blocks. Each residual block contains shortcut connections that allow the network to learn the identity mapping, facilitating the training of deeper networks by alleviating the vanishing gradient problem [20], [21]. In this study, an improved version of ResNet-50 was used to classify the wayang characters [22]. The architecture starts with an initial convolutional layer, followed by a pooling layer and a sequence of nine convolutional layers alternating with pooling and residual blocks [23]. The network ends up with an average pooling layer and a fully connected layer that generates probabilities across 14 layers. The layers of CNN-based ResNet-50 can be seen in Figure 3.

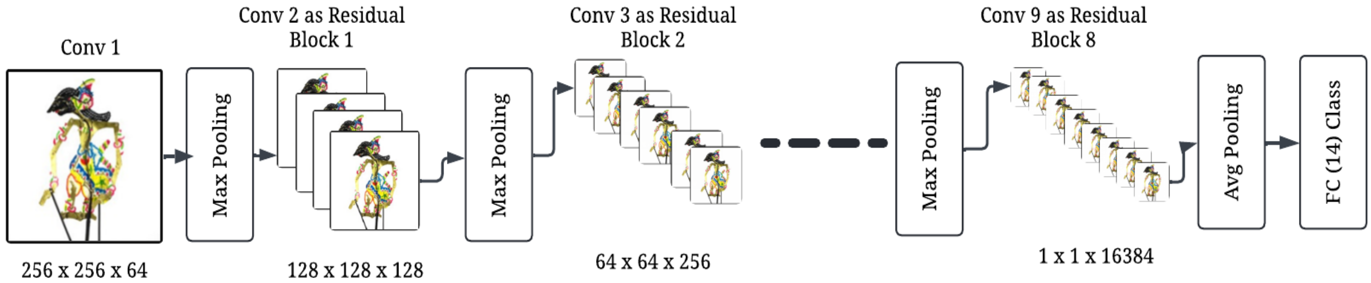

Figure 3. CNN-based ResNet-50

Based on Figure 3, the layer model begins with an input layer of shape $(256, 256, 3)$. The first convolutional layer, Conv1, applies $64\ filters\ of\ size\ (3x3)$, resulting in an output shape of $(256, 256, 64)$, followed by a max pooling layer. This is followed by Residual Block 1, which includes Conv2 with $128\ filters\ of\ size\ (3x3)$, producing an output of $(128, 128, 128)$, and another max pooling layer, with a shortcut connection added. Residual Block 2 comprises Conv3 with $256\ filters\ of\ size\ (3x3)$, resulting in an output shape of $(64, 64, 256)$, and a max pooling layer with a shortcut connection. In Residual Block 3, Conv4 uses $512\ filters\ of\ size\ (3x3)$, yielding an output of $(32, 32, 512)$, followed by a max pooling layer and a shortcut connection. Residual Block 4 contains Conv5 with $1024\ filters\ of\ size\ (3x3)$, outputting $(16, 16, 1024)$, and another max pooling layer and shortcut connection. Residual Block 5 includes Conv6 with $2048\ filters\ of\ size\ (3x3)$, resulting in $(8, 8, 2048)$, with a max pooling layer and shortcut connection. Residual Block 6 uses Conv7 with $4096\ filters\ of\ size\ (3x3)$, producing $(4, 4, 4096)$, followed by a max pooling layer and shortcut connection. Residual Block 7 applies Conv8 with $8192\ filters\ of\ size\ (3x3)$, resulting in $(2, 2, 8192)$, with a max pooling layer and shortcut connection. Residual Block 8 includes Conv9 with $16384\ filters\ of\ size\ (1x1)$, producing $(1, 1, 16384)$, and a final max pooling layer and shortcut connection. The model then used an average pooling layer before the fully connected layer (FC) with $14\ output\ classes$, using softmax activation to produce class probabilities.

2.3 Long Short-Term Memory CNN

In the proposed method, the LSTM-CNN architecture is designed to leverage the strengths of both convolutional neural networks (CNN) and LSTM networks [24], [25]. Initially, the CNN layers were utilized to extract spatial features from each image in the sequence, starting with a series of convolutional layers followed by pooling layers to down-sample the feature maps and retain important spatial hierarchies. These convolutional layers are responsible for detecting edges, textures, and other low-level features, which were then transformed into high-level abstract features as they passed through deeper layers. After CNN layers, the output feature maps were flattened and passed into LSTM layers, which capture the temporal dependencies and sequential information from the image sequence. The LSTM layers, equipped with gates to manage the flow of information, help in learning long-term dependencies across the image sequence, making the model adept at understanding the context and patterns over time. The combination of CNN for spatial feature extraction and LSTM for temporal sequence learning enhanced the model's capability to accurately classify complex sequences of images, as reflected in the improved evaluation metrics such as accuracy, precision, recall, and F1-score.

The LSTM and CNN were integrated by stacking these layers sequentially, where the spatial feature output from CNN was used as the input to LSTM to capture the temporal patterns in the image sequence. This process helps the model not only understand the visual features of each image separately, but also consider the temporal relationship between the images in the sequence. The following are the general equations for an LSTM layer, presented in Equations 1–6.

$$f_t = \sigma(Wf \cdot [ht - 1, xt] + bf) \tag{1}$$

$$i_t = \sigma(Wi \cdot [ht - 1, xt] + bi) \tag{2}$$

$$C'_t = tanh(WC \cdot [ht - 1, xt] + bC) \tag{3}$$

$$C_t = ft * Ct - 1 + it * C't \tag{4}$$

$$o_t = \sigma(Wo \cdot [ht-1, xt] + bo) \tag{5}$$

$$h_t = ot * tanh(Ct) \tag{6}$$

Where the forget gate f_t controls the flow of information from the previous cell state C_{t-1} to the current time step t. It is computed using a sigmoid activation function σ on a linear combination of the previous hidden state h_{t-1} and the current input x_t, adjusted by learnable parameters W_f and b_f. The input gate i_t determines how much of the new candidate values C'_t, generated by a $tanh$ function, are added to the cell state. Similarly, i_t is computed using a sigmoid activation function on a linear combination of h_{t-1} and x_t, adjusted by parameters W_i and b_i. The cell C_t is updated by combining C_{t-1} adjusted by f_t and C'_t adjusted by i_t. The output gate o_t then regulates the amount of information passed to the next time step's hidden state h_t, which is the cell state C_t passed through a $tanh$ function adjusted by o_t. This sequence of operations in LSTM allows the network to learn and maintain long-term dependencies in sequential data, making it effective for tasks as understanding context over time is crucial, such as in natural language processing and time series analysis.

In the integration of new layers depicted in Figure 4, the LSTM layer incorporated 32 units with a dropout rate of 0.5. This configuration aimed at enhancing the network's ability to capture temporal dependencies in sequential data while mitigating overfitting. Dropout regularization randomly set a fraction of input units to zero during training, preventing units from relying too heavily on specific inputs and thus improving the generalization capability of the model.

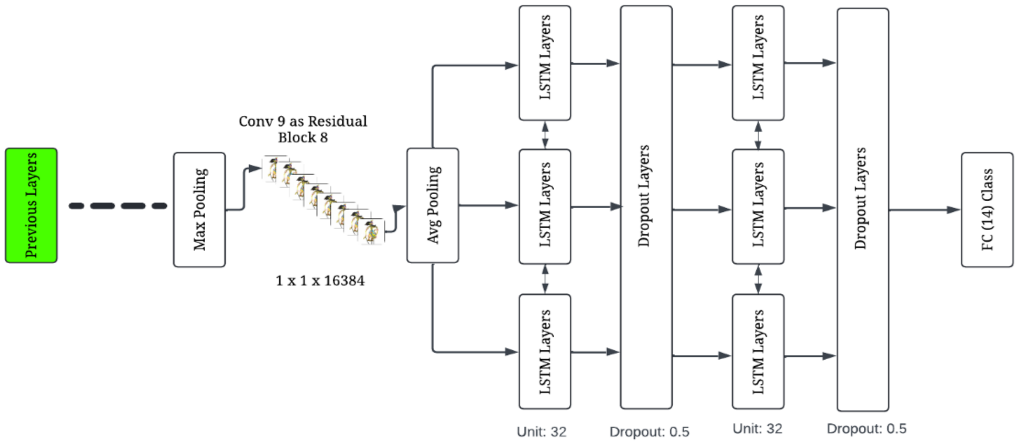

Figure 4. Integration LSTM Layers

The integration of CNN and LSTM in Figure 4 is executed through a sequential architecture where CNN layers were employed first to extract spatial features from input images, followed by LSTM layers that capture the temporal dependencies within the extracted features. Initially, CNN processed each image individually, applying convolutional and pooling layers to identify and down-sample essential spatial hierarchies, effectively detecting edges, textures, and other low-level features. Once the CNN layers produce feature maps, they are flattened and fed into the LSTM layers, which are adept at recognizing patterns across sequences of images. The LSTM layers utilized gates to control the flow of information, allowing the model to maintain long-term dependencies and context over time. This architecture not only allows the model to understand the individual visual features of each image, but also emphasizes the temporal relationships among them, which is crucial for accurately classifying complex sequences.

2.4 Confusion Matrix Evaluation

Confusion matrix [26], [27] is an effective approach to evaluate the classification performance of 14-wayang-class tasks. This matrix provides a detailed assessment of how well the model categorizes each class by comparing predicted class labels against the actual classes in the test dataset. Each cell in the matrix reflects the count of true positives (correctly predicted samples), false positives (incorrectly predicted samples), false negatives (samples incorrectly not predicted), and true negatives (correctly not predicted samples). From this matrix, key performance metrics such as accuracy, precision, recall, and F1-score can be computed [28]. This evaluation allows for a thorough understanding of the model's strengths and weaknesses in identifying each wayang character, essential for optimizing and validating the model for real-world applications. The equations based on the confusion matrix evaluation can be seen in Equations 7–10.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

$$Recall = \frac{TP}{TP + FN} \tag{9}$$

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision \times Recall} \tag{10}$$

3. Results and Discussion

In this study, computations were conducted on a system equipped with an Intel Core i3 10th generation processor, 16GB of RAM, GTX 1030 graphics card, and a 512GB SSD, running MATLAB 2020a software. During the training process, the model was trained over a maximum of 20 epochs using Adam optimizer with a mini-batch size of 32. A dropout rate of 0.5 was applied to prevent overfitting, while an initial learning rate of 0.0001 was set to regulate the adjustment of the model weights during optimization. The LSTM layers were configured with 32 units to capture and learn temporal dependencies effectively within the sequential data. From the training process, the results showed a convergence starting from epoch 7, as depicted in Figure 5 and Figure 6. The training accuracy in Figure 5 illustrates the progress towards convergence, indicating that the model's performance stabilized and improved steadily from epoch 7 onwards. This trend suggests that the training process effectively refines the model's parameters, leading to enhanced accuracy and reliability in classifying wayang characters.

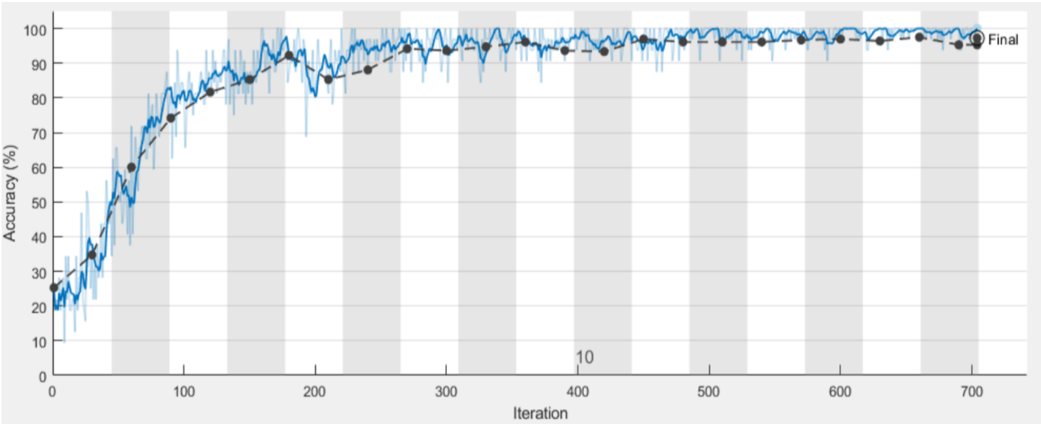

Figure 5. Training Accuracy

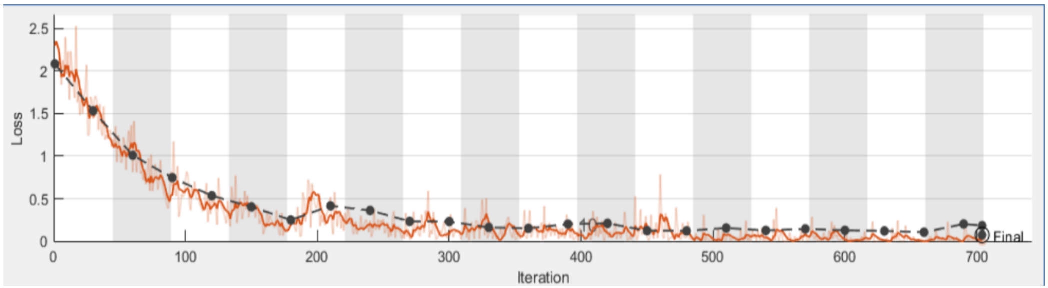

Figure 6. Training Loss

Figure 6 shows a clear reduction of loss within iterations as the model minimized errors during training, which correlated with the increase in accuracy depicted in Figure 5. As the loss decreased, the accuracy improved significantly, especially during early iterations, before both metrics stabilized around the 300th iteration. The correlation between the decreasing loss and increasing accuracy indicates that the model effectively learns to classify the data, with both figures reaching convergence, suggesting that further training yields minimal additional improvement.

The performance depicted in Figure 5 and Figure 6 correlate with the results obtained from the confusion matrix, as outlined in Table 1. These visual representations and numerical data collectively demonstrate the model's efficacy in classifying wayang characters. The training graphs highlight the convergence and improvement in accuracy over epochs, while the confusion matrix in Table 1 provides specific metrics such as precision, recall, and F1-score for each character class.

Table 1. Results of Confusion Matrix

Training Phase	Accuracy	Precision	Recall	F1-Score	Elapsed Time
1st Training without LSTM	92%	100%	100%	100%	81 minutes 36 sec
2nd Training without Transfer Learning Resnet-50	95.25%	100%	100%	100%	40 minutes 11 sec
3rd Proposed Method (Ensemble Resnet-50 with LSTM)	97.92%	100%	100%	100%	188 minutes 21 sec

The training results, as presented in Table 1, highlight the performance metrics of different training approaches. The first training phase, which did not utilize LSTM, achieved an accuracy of 92%, with precision, recall, and F1-score all at 100%, and an elapsed time of 81 minutes and 36 seconds. The second training phase, which did not employ ResNet-50 transfer learning, showed an improved accuracy of 95.25%, maintaining precision, recall, and F1-score at 100%, and reduced the training time to 40 minutes and 11 seconds. The third training phase, representing the proposed method that ensembled ResNet-50 with LSTM, achieved the highest accuracy of 97.92%, with precision, recall, and F1-score remaining at 100%, but it required longer training time,188 minutes and 21 seconds. These results illustrate the superior performance of the proposed method in terms of accuracy, despite the longer computational time.

The proposed method required significantly more computation time, 188 minutes and 21 seconds, compared to other methods such as training without LSTM (81 minutes 36 seconds) and training without transfer learning (ResNet-50), which only took 40 minutes and 11 seconds. Although the accuracy improvement from 95.25% to 97.92% seems modest, the added complexity of combining transfer learning with LSTM contributes to the increased computational demand. LSTM involves intensive temporal processing, while the ensemble using ResNet-50 requires deeper feature evaluation, leading to longer training times. Although the performance metrics, like precision, recall, and F1-score remain consistent at 100%, the longer computation time poses a challenge, especially for applications requiring time efficiency or where computational resources are limited.

In the final step of this study, a comprehensive comparison between the proposed model and those of the related research was conducted, as presented in Table 2. This comparison highlights the performance differences across various metrics such as accuracy, precision, recall, and F1-score, providing a clear evaluation of the effectiveness of the proposed model against the existing methods.

Table 2. Comparison with Related Methods

Researcher	Novel	Accuracy	Precision	Recall	F1-Score
Wibawa et al. [11]	Traditional CNN	92%	92.5%	92.25%	91.75%
Wisnudhanti et al. [14]	CNN with Transfer Learning Densenet	97.88%	-	-	-
Sudiatmika et al. [13]	Region Based Neural Networks (RNN)	92.04%	-	-	-
Banjaransari et al. [12]	CNN with Transfer Learning Mobilenet-V2	94.17%	95%	94%	94%
This Study	CNN with Ensemble Transfer Learning Resnet-50 and LSTM	97.92%	100%	100%	100%

Based on the comparison of methods presented in Table 2, the proposed method using CNN-based ResNet-50 Ensemble Transfer Learning and LSTM demonstrates superior performance. Achieving an accuracy of 97.92% and precision, recall, and F1-score of 100%, this method outperforms other methods such as Traditional CNN, Transfer Learning Densenet, RNN, and Mobilenet-V2. The notable performance can be attributed to the combination of transfer learning and LSTM, which allows the network to leverage deep feature extraction and integrate temporal information more effectively than conventional methods. In contrast, methods like Traditional CNN and RNN show lower accuracy, falling below 94%. This highlights that the integration of transfer learning and LSTM provides a significant contribution to improving the network's predictive capability**.**

4. Conclusion

This study presents a novel method by integrating ResNet-50 transfer learning and LSTM, resulting in significant improvements in wayang character classification. The proposed model achieved an accuracy of 97.92%, with precision, recall, and F1-scores all at 100%, although it required longer training time of 188 minutes and 21 seconds. The initial training phase without LSTM yielded an accuracy of 92%, while the second phase without transfer learning achieved 95.25% accuracy. Despite the increased elapsed time, the proposed ensemble method demonstrates exceptional model performance. This advancement can be highly beneficial for cultural preservation and educational purposes, helping society to better understand and appreciate traditional Indonesian wayang characters through accurate digital classification. For future research, further optimization techniques can be explored to reduce the training time while maintaining or improving the accuracy. Additionally, expanding the dataset to include more wayang characters and incorporating other advanced neural network architectures could enhance the model's robustness and applicability. Investigating real-time classification and developing user-friendly applications for interactive learning and cultural engagement can also be valuable directions for extending the impact of this research.

References

[1] A. B. Wibawa, "Analyzing the Evolution of Indonesian Wayang Puppetry and Its Fusion with Modern Theater and Performance Arts," *Studies in Art and Architecture*, vol. 3, no. 1, pp. 1–9, Mar. 2024. https://doi.org/10.56397/SAA.2024.03.01

[2] R. Siringo-ringo, A. Siagian, and N. Wahyuni, "The Resilient Tradition: Exploring the Cultural Significance of Javanese Wayang Kulit in Heritage Preservation," *Jurnal Ilmu Pendidikan dan Humaniora*, vol. 11, no. 1, pp. 69–84, Jan. 2022. https://doi.org/10.35335/jiph.v11i1.16

[3] T. Nur Fitria, "The Performance of Wayang Orang Sriwedari Surakarta: A Cultural Preservation," *Jurnal Humaya: Jurnal Hukum, Humaniora, Masyarakat, dan Budaya*, vol. 3, no. 2, pp. 123–138, Dec. 2023. https://doi.org/10.33830/humaya.v3i2.6276

[4] E. H. Rachmawanto, C. A. Sari, and F. O. Isinkaye, "A good result of brain tumor classification based on simple convolutional neural network architecture," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 22, no. 3, pp. 711–719, Jun. 2024. https://doi.org/10.12928/TELKOMNIKA.v22i3.25863

[5] A. Susanto, I. U. W. Mulyono, C. A. Sari, E. H. Rachmawanto, D. R. I. M. Setiadi, and M. K. Sarker, "Improved Javanese script recognition using custom model of convolution neural network," *International Journal of Electrical and Computer Engineering*, vol. 13, no. 6, pp. 6629–6636, Dec. 2023. https://doi.org/10.11591/ijece.v13i6.pp6629-6636

[6] D. del-Pozo-Bueno, D. Kepaptsoglou, F. Peiró, and S. Estradé, "Comparative of machine learning classification strategies for electron energy loss spectroscopy: Support vector machines and artificial neural networks," *Ultramicroscopy*, vol. 253, Nov. 2023. https://doi.org/10.1016/j.ultramic.2023.113828

[7] K. M. O. Nahar *et al.*, "Recognition of Arabic Air-Written Letters: Machine Learning, Convolutional Neural Networks, and Optical Character Recognition (OCR) Techniques," *Preprints (Basel)*, Sep. 2023. https://doi.org/10.20944/preprints202309.1806.v1

[8] A. A. Khan, A. A. Laghari, and S. A. Awan, "Machine Learning in Computer Vision: A Review," *EAI Endorsed Transactions on Scalable Information Systems*, vol. 8, no. 32, pp. 1–11, 2021. https://doi.org/10.4108/eai.21-4-2021.169418

[9] J. Gupta, S. Pathak, and G. Kumar, "Deep Learning (CNN) and Transfer Learning: A Review," in *Journal of Physics: Conference Series*, Institute of Physics, 2022. https://doi.org/10.1088/1742-6596/2273/1/012029

[10] M. Ahmad, S. Abbas, A. Fatima, G. F. Issa, T. M. Ghazal, and M. A. Khan, "Deep Transfer Learning-Based Animal Face Identification Model Empowered with Vision-Based Hybrid Approach," *Applied Sciences (Switzerland)*, vol. 13, no. 2, Jan. 2023. https://doi.org/10.3390/app13021178

[11] A. P. Wibawa, W. A. Yudha Pratama, A. N. Handayani, and A. Ghosh, "Convolutional Neural Network (CNN) to determine the character of wayang kulit," *International Journal of Visual and Performing Arts*, vol. 3, no. 1, pp. 1–8, Jun. 2021. https://doi.org/10.31763/viperarts.v3i1.373

[12] M. Banjaransari, A. Prahara, M. Banjaransari, and A. Prahara, "Image Classification of Wayang Using Transfer Learning and Fine-Tuning of CNN Models," *Buletin Ilmiah Sarjana Teknik Elektro*, vol. 5, no. 4, pp. 632–641, 2023. https://doi.org/10.12928/biste.v5i4.9977

[13] I. B. K. Sudiatmika, M. Artana, N. W. Utami, M. A. P. Putra, and E. G. A. Dewi, "Mask R-CNN for Indonesian Shadow Puppet Recognition and Classification," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Feb. 2021. https://doi.org/10.1088/1742-6596/1783/1/012032

[14] K. Wisnudhanti and F. Candra, "Image Classification of Pandawa Figures Using Convolutional Neural Network on Raspberry Pi 4," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Nov. 2020. https://doi.org/10.1088/1742-6596/1655/1/012103

[15] "14 Indonesian Wayang Types Datasets." Accessed: Jul. 04, 2024.

[16] N. R. D. Cahyo, C. A. Sari, E. H. Rachmawanto, C. Jatmoko, R. R. A. Al-Jawry, and M. A. Alkhafaji, "A Comparison of Multi Class Support Vector Machine vs Deep Convolutional Neural Network for Brain Tumor Classification," in *2023 International Seminar on Application for Technology of Information and Communication (iSemantic)*, IEEE, Sep. 2023, pp. 358–363. https://doi.org/10.1109/iSemantic59612.2023.10295336

[17] M. M. I. Al-Ghiffary, C. A. Sari, E. H. Rachmawanto, N. M. Yacoob, N. R. D. Cahyo, and R. R. Ali, "Milkfish Freshness Classification Using Convolutional Neural Networks Based on Resnet50 Architecture," *Advance Sustainable Science Engineering and Technology*, vol. 5, no. 3, p. 0230304, Oct. 2023. https://doi.org/10.26877/asset.v5i3.17017

[18] 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

[19] A. Deshpande, V. V. Estrela, and P. Patavardhan, "The DCT-CNN-ResNet50 architecture to classify brain tumors with super-resolution, convolutional neural network, and the ResNet50," *Neuroscience Informatics*, vol. 1, no. 4, p. 100013, Dec. 2021. https://doi.org/10.1016/j.neuri.2021.100013

[20] M. M. I. Al-Ghiffary, N. R. D. Cahyo, E. H. Rachmawanto, C. Irawan, and N. Hendriyanto, "Adaptive deep learning based on FaceNet convolutional neural network for facial expression recognition," *Journal of Soft Computing*, vol. 05, no. 03, pp. 271–280, 2024. https://doi.org/10.52465/joscex.v5i3.450

[21] N. R. D. Cahyo and M. M. I. Al-Ghiffary, "An Image Processing Study: Image Enhancement, Image Segmentation, and Image Classification using Milkfish Freshness Images," *IJECAR) International Journal of Engineering Computing Advanced Research*, vol. 1, no. 1, pp. 11–22, 2024.

[22] X. X. Li, D. Li, W. X. Ren, and J. S. Zhang, "Loosening Identification of Multi-Bolt Connections Based on Wavelet Transform and ResNet-50 Convolutional Neural Network," *Sensors*, vol. 22, no. 18, Sep. 2022. https://doi.org/10.3390/s22186825

[23] F. Farhan, C. A. Sari, E. H. Rachmawanto, and N. R. D. Cahyo, "Mangrove Tree Species Classification Based on Leaf, Stem, and Seed Characteristics Using Convolutional Neural Networks with K-Folds Cross Validation Optimalization," *Advance Sustainable Science Engineering and Technology*, vol. 5, no. 3, p. 02303011, Oct. 2023. https://doi.org/10.26877/asset.v5i3.17188

[24] A. K. Ozcanli and M. Baysal, "Islanding detection in microgrid using deep learning based on 1D CNN and CNN-LSTM networks," *Sustainable Energy, Grids and Networks*, vol. 32, p. 100839, 2022. https://doi.org/10.1016/j.segan.2022.100839

[25] P. Kumari and D. Toshniwal, "Long short term memory–convolutional neural network based deep hybrid approach for solar irradiance forecasting," *Appl Energy*, vol. 295, p. 117061, 2021. https://doi.org/10.1016/j.apenergy.2021.117061

[26] I. P. Kamila, C. A. Sari, E. H. Rachmawanto, and N. R. D. Cahyo, “A Good Evaluation Based on Confusion Matrix for Lung Diseases Classification using Convolutional Neural Networks,” *Advance Sustainable Science, Engineering and Technology*, vol. 6, no. 1, p. 0240102, Dec. 2023. https://doi.org/10.26877/asset.v6i1.17330

[27] S. C. Kim and Y. S. Cho, “Predictive System Implementation to Improve the Accuracy of Urine Self-Diagnosis with Smartphones: Application of a Confusion Matrix-Based Learning Model through RGB Semiquantitative Analysis,” *Sensors*, vol. 22, no. 14, Jul. 2022. https://doi.org/10.3390/s22145445

[28] A. Theissler, M. Thomas, M. Burch, and F. Gerschner, “ConfusionVis: Comparative evaluation and selection of multi-class classifiers based on confusion matrices,” *Knowl Based Syst*, vol. 247, Jul. 2022. https://doi.org/10.1016/j.knosys.2022.108651