International Journal of Informatics and Computation (IJICOM)
Vol. 8, No. 1, 2026
ISSN: 2685-8711, E-ISSN: 2714-5263 DOI: 10.35842/ijicom

Classification of Monkey Characters Using CNN-VGG16 and ResNet50 Architectures

Wayan Restu Cahyana[1], Luh Joni Erawati Dewi[2], Putu Hendra Suputra[3]

Abstract
Wayang kulit is one of Indonesia's cultural heritages that possesses significant artistic and philosophical value. In the Balinese Ramayana puppet tradition, monkey characters exhibit highly similar visual characteristics, making manual identification difficult and requiring specialized expertise. To address this challenge, this study proposes an image classification approach based on Convolutional Neural Networks (CNN) using transfer learning with the VGG16 and ResNet50 architectures. We utilize a dataset consisting of 270 images divided into 15 classes, including 14 monkey puppet character classes and 1 non-monkey puppet class. This study conducts multiple experimental scenarios involving different dataset partitioning strategies and hyperparameter configurations to analyze model performance comprehensively. Furthermore, we apply data augmentation techniques to the training dataset to improve model generalization and reduce overfitting. Based on the obtained results, VGG16 achieves the best performance with a testing accuracy of 98.67%, while ResNet50 achieves 98.33%. We also obtain stable training and validation performance from both architectures, indicating strong capability in learning visual patterns from limited datasets. However, several classification errors still occur in classes with high visual similarity, demonstrating the challenges of fine-grained image classification. This study demonstrates that transfer learning-based CNN architectures can effectively classify Balinese Ramayana monkey puppet characters. It also contributes to the development of intelligent systems for cultural heritage preservation under limited dataset conditions.

Keywords:
Shadow Puppets, Monkey Characters, CNN, VGG16, ResNet50

1. Introduction

Wayang is one of Indonesia's diverse cultural traditions, passed down from generation to generation. Wayang is an artistic performance in the form of leather sculptures made to resemble puppets, acting as characters in stories performed by a puppeteer [1]. Wayang has distinctive characteristics across regions, such as the Balinese shadow puppets, which feature flat, stemmed figures [2].The figures have symbolic and philosophical meanings that reflect the spirit of the culture in the arts, resulting in each Balinese shadow puppet story having a unique form [3]. The uniqueness of the shadow puppet form has led to the creation of various shadow puppet types, each with visual similarities in attributes such as shapes, colors, and patterns. These similarities create challenges, especially for the younger generation, in recognizing the art of shadow puppetry [1]. This problem is further complicated by the monkey characters in the Balinese Ramayana, which exhibit a high degree of visual similarity, making identification difficult.

The distinguishing characteristics of the monkey puppet characters can be identified by studying each character's pattern. Image-based management is the process of improving

Corresponding Author: Wayan Restu Cahyana (restu.cahyana@student.undiksha.ac.id)
1 Wayan Restu Cahyana, Universitas Pendidikan Ganesha, restu.cahyana@student.undiksha.ac.id
2 Luh Joni Erawati Dewi, Universitas Pendidikan Ganesha, joni.erawati@undiksha.ac.id
3 Putu Hendra Suputra, Universitas Pendidikan Ganesha, hendra.suputra@undiksha.ac.id

image quality so that both humans and machines can easily identify images [4]. Advances in information technology, particularly in computing, have impacted every field of work. One such development is the creation of the Convolutional Neural Network (CNN). CNN is a deep learning algorithm used in image-based classification and is highly effective, with multiple architectures [5]. The Convolutional Neural Network (CNN) method has proven effective in recognizing visual patterns in images and is widely used in various image classification cases [6]. Research by [7] reported 80%-85% accuracy in classifying wayang puppet images using a CNN with the LeNet architecture.

Although the CNN method performs well in image processing, efficiency is a crucial consideration because CNNs use a large number of parameters, requiring substantial memory. This problem can be overcome using transfer learning [8]. Research conducted by [9] showed that applying transfer learning to peach plant disease classification using the EfficientNetB2 architecture achieved an accuracy of 96.6%. However, conventional CNN architectures still have limitations in extracting complex visual features. Therefore, more advanced and optimal CNN architectures, such as ResNet-50 and VGG-16, have been developed.

This study tested the ResNet-50 and VGG-16 architectures for classifying monkey characters in the Balinese Ramayana puppet show. ResNet-50 and VGG-16 were chosen for their classification advantages and implemented using transfer learning. ResNet-50 has the advantage of using residual learning, resulting in more stable and efficient models. VGG-16 also has the advantage of improving image segmentation performance through transfer learning, where the model is trained on a large dataset such as ImageNet [10].

Based on the problems described, this study has several key contributions as follows. First, this study compiles a dataset of monkey characters from the Balinese Ramayana puppet show, which exhibits a high degree of visual similarity, making it suitable for challenging fine-grained classification problems. Similarity in visual characteristics between classes is known to cause difficulties in classification, as models tend to struggle to distinguish features that are very similar across classes [6]. Therefore, a model capable of extracting features more discriminatively is needed to improve classification accuracy.

Second, this study conducted a comparative analysis of the VGG16 and ResNet50 architectures on a limited dataset to examine the performance characteristics of each model. Third, this study analyzed the effect of variations in dataset partitioning and hyperparameter configuration on model stability and accuracy. The selection of the VGG16 and ResNet50 architectures was based on differences in network depth characteristics: VGG16 has a simpler structure, while ResNet50 has a deeper architecture with residual connections. This difference is important to analyze, especially on limited datasets, to determine which model is more effective in addressing classification problems involving similarity.

2. Related Works

Various studies across fields such as the environment, agriculture, and fisheries have applied Convolutional Neural Network (CNN) methods to image classification problems. One study [5] analyzed the performance of the ResNet-50 and VGG-16 architectures for plant type classification. The results showed that hyperparameter settings, such as image size and the number of epochs, significantly influenced model performance. In this study, ResNet-50 achieved 63.83% accuracy, while MobileNetV2 achieved 87.23%. This indicates that training parameter optimization is a critical factor in improving model performance. A comparative study of deep learning architectures for semantic segmentation of remote sensing images using ResNet-50 and Attention U-Net [11] showed that architectural characteristics significantly influence model performance. This is

evidenced by the accuracy achieved by both models, which exceeded 90%, although Attention U-Net showed more stable results compared to ResNet-50.

A similar study in species classification, conducted by [12] on tuna identification using ResNet-50, achieved a global accuracy of 91% on a dataset of 500 images. The study also confirmed that data augmentation and hyperparameter adjustment play a crucial role in improving classification accuracy, especially for data with visual similarities between classes. Furthermore, the evaluation approach is a crucial aspect in multi-class research, as demonstrated by [13], who stated that model evaluation requires more than accuracy alone; other metrics such as precision, recall, and F1-score should also be considered to obtain a more comprehensive picture of performance. Another study conducted by [14] compared the performance of the VGG-16 and MobileNetV2 architectures in classifying mangrove leaf diseases. The results showed that the parameter complexity of VGG-16 does not always guarantee better performance than that of a lighter architecture. MobileNetV2 in that study demonstrated better generalization, achieving 0.96 accuracy in the early stopping scenario.

Although in some studies ResNet-50 and VGG-16 performed worse than the other architectures discussed, this does not necessarily indicate that they are suboptimal. Model performance is strongly influenced by dataset characteristics, problem complexity, and the training configuration used. ResNet-50 and VGG-16 were chosen in this study because both CNN architectures have proven capable of extracting deep visual features. ResNet-50 excels at addressing the vanishing gradient problem through residual connections, enabling a deeper, more stable network. Meanwhile, VGG-16 has a simple yet effective architecture that gradually extracts features through small convolutional layers.

Furthermore, this study aims to conduct a comparative analysis of the performance of the two architectures for classifying images of monkey characters from the Balinese Ramayana puppet show, which exhibit high visual similarity. Therefore, the selection of ResNet-50 and VGG-16 was based on their feature representation capabilities and relevance to the problem characteristics. Based on the research discussed, it can be concluded that ResNet-50 and VGG-16 have been widely used for various image classification problems and have demonstrated strong performance. However, no research has specifically compared the two architectures using transfer learning for the classification of monkey characters from the Balinese Ramayana puppet show, which exhibit high visual similarity and are trained on a limited local dataset. Therefore, this study focuses on a comparative analysis of ResNet-50 and VGG-16, using a controlled training and evaluation process to obtain a model with the most stable performance.

3. Proposed Method

This study aims not only to compare the performance of two Convolutional Neural Network (CNN) architectures, namely ResNet-50 and VGG-16, but also to analyze how the distinct architectural characteristics of each model influence their ability to address fine-grained classification problems involving high visual similarity within a limited dataset. We investigate the classification of monkey characters from the Balinese Ramayana puppet show, which presents unique challenges due to similarities in shape, color, and ornamentation among classes that complicate the extraction of discriminative visual features. In this study, VGG-16, which employs a simple sequential deep-layer architecture, is explored for its capability in hierarchical feature learning, while ResNet-50, which incorporates residual connection mechanisms, is evaluated for its ability to preserve information across layers and improve training stability in deeper neural networks. Furthermore, we examine the extent to which the residual learning mechanism in ResNet-50 provides advantages over conventional architectures such as VGG-16 under conditions of limited training data and highly similar visual patterns. To ensure a fair and objective comparison, this study applies several control variables by equalizing hyperparameters

across both models, including the number of epochs, batch size, learning rate, optimizer configuration, and dataset partitioning strategy, thereby allowing the observed performance differences to be more accurately attributed to the intrinsic architectural characteristics of each CNN model rather than external experimental factors.

3.1. Data Collection

The data collection stage involved observing several local datasets in the research area and datasets available online. The dataset used in this study was collected directly from Balinese shadow puppet artisans in Nagasepaha Village, Sukasada District, Buleleng Regency, Bali Province. The dataset comprises 15 classes: 14 monkey puppets and 1 non-monkey puppet. The fourteen monkey characters include Anala Gini, Anggada, Anila, Asti Muka, Gawaksa, Hanuman, Indra Janur, Jembawan, Kesari, Menda, Mong Muka, Sampati, Sarpa Muka, and Sugriwa.

This study began by conducting a comprehensive literature review of journals, scientific articles, and previous studies related to puppet character recognition, image processing, transfer learning, and deep learning architectures such as ResNet-50 and VGG-16 to establish the theoretical foundation supporting the research implementation. Subsequently, we collected image data for each monkey puppet character class, and divided it into training, validation, and testing datasets. We then designed classification models for each architecture using transfer learning approaches. We conducted model training under several experimental scenarios to obtain optimal performance for each model. After the training phase was completed, this study evaluated the trained models using unseen test data to measure the mode capability. The total dataset consisted of 270 images, with 18 images per class. The distribution of the total dataset across classes is shown in Fig. 1.

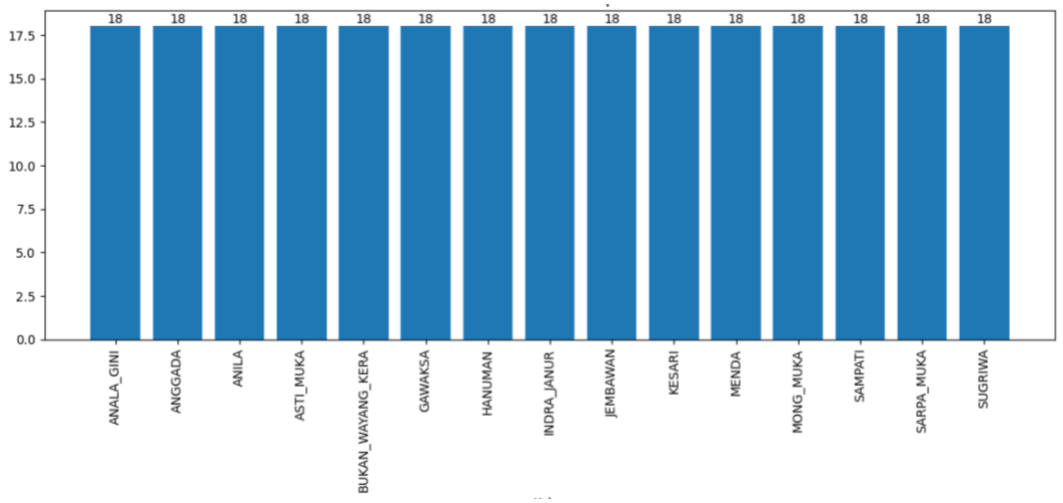

Fig. 1. Distribution of Total Dataset per Class

This study uses several dataset-sharing ratio scenarios, as in previous research [8]: 50:50, 60:40, 70:30, 80:20, and 90:10. These ratios represent the proportion of training data to the total dataset, with the remaining data divided equally between validation and testing. The divided data is then organized into a directory structure by training, validation, and test categories to support the structured model training and evaluation process. The distribution of the training, validation, and test data is shown in Fig. 2.

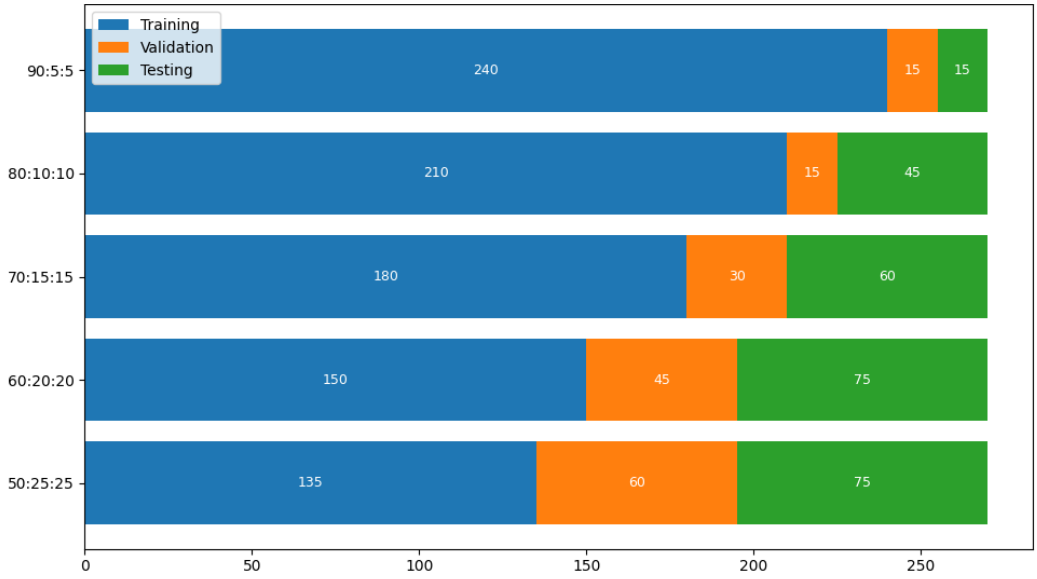

Fig. 2. Distribution of Training, Validation, and Testing Data

3.2. Data Augmentation

Data augmentation was applied to increase the diversity of the training dataset, enabling the model to learn more robust features and reducing the risk of overfitting by simulating various real-world conditions. In this study, augmentation was performed only on the training data, while the validation and test datasets were kept unchanged to maintain objective model evaluation. The augmentation techniques included horizontal flipping, image rotations of 30°, 45°, 90°, −30°, −45°, and −90° to increase orientation variation, zoom-in and zoom-out transformations of up to 20% of the original image size to simulate different object scales, and brightness adjustments through brightness enhancement and reduction to represent varying lighting conditions.

3.3. Resnet 50

The Residual Network (ResNet) is a CNN architecture proposed in 2015 by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. ResNet-50 comes in various forms depending on the number of layers. ResNet-50 is a Residual Network that addresses the problem using residual blocks and has 48 convolution layers, 1 MaxPool, and 1 AveragePool layer. This approach enables improved learning and model representation, and can increase prediction accuracy [1][15]. ResNet-50 is also known as an architecture that sits between shallow and very deep networks, allowing for a moderate number of parameters without sacrificing performance. This makes ResNet-50 effective for medium-scale datasets, reducing the risk of overfitting while maintaining generalization. Furthermore, the use of a bottleneck structure and shortcut connections allows for stable gradient flow during backpropagation, resulting in more efficient training and better convergence. With a deep network architecture, ResNet-50 can extract features in a more complex, hierarchical manner, thereby improving classification performance [16][17].

In addition to its architectural advantages, ResNet-50 is commonly used as a pre-trained model trained on large-scale datasets such as ImageNet, thus providing the initial ability to extract general visual features. In this study, the ResNet-50 model was fine-tuned by replacing the fully connected layer at the end to match the number of classes in the wayang image classification task. This process allows the model to adapt to the more specific visual characteristics of the dataset used. During training, weight updates are performed using a backpropagation algorithm with an optimizer to minimize prediction errors gradually. The output of the model is a probability distribution for each class

generated through a softmax activation function, which is then used to determine the class with the highest probability. Thus, ResNet-50 not only leverages the general features of the pre-training process but also adapts to the specific characteristics of the dataset [18]. The ResNet-50 architecture is shown in Fig. 3.

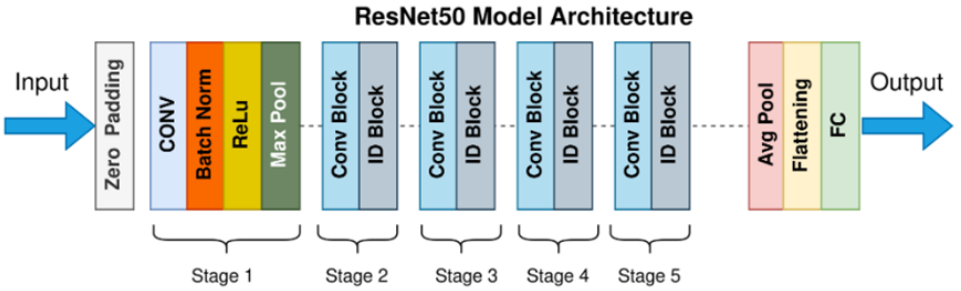

Fig. 3. Resnet-50 Architecture

Unlike conventional CNN architectures that directly learn the mapping function $H(x)$, ResNet formulates the learning process in terms of a residual function $F(x)$, such that the relationship between input and output can be expressed as:

$$y = F(x) + x \tag{1}$$

where x represents the input to a residual block, $F(x)$ is the residual function learned by several convolutional layers, and y is the output of the block. With this approach, the network only needs to learn the difference between the input and the desired output, making the learning process more stable and efficient.

The residual function $F(x)$ in ResNet-50 generally consists of a sequence of operations such as convolution, normalization, and activation functions. Mathematically, this function can be represented as:

$$F(x) = W_2 \cdot \sigma(W_1 x) \tag{2}$$

where W_1 and W_2 are the weight parameters of the convolutional layers, and σ denotes a non-linear activation function such as the Rectified Linear Unit (ReLU). Thus, the final output of the residual block becomes:

$$y = W_2 \cdot \sigma(W_1 x) + x \tag{3}$$

In the ResNet-50 architecture, a bottleneck structure is employed, consisting of three consecutive convolutional layers: a 1×1 convolution for dimensionality reduction, a 3×3 convolution for feature extraction, and a 1×1 convolution for dimensionality restoration. This structure enables the model to be deep while remaining computationally efficient. Mathematically, the residual function in the bottleneck structure can be expressed as:

$$F(x) = W_3 \cdot \sigma(W_2 \cdot \sigma(W_1 x)) \tag{4}$$

In some cases, the input and output dimensions differ, requiring an additional linear projection in the shortcut path. This can be expressed as:

$$y = F(x) + W_s x \tag{5}$$

where W_s is the weight matrix used to match the input dimensions. The presence of this shortcut connection allows gradients to flow more effectively during backpropagation, thereby mitigating the vanishing gradient problem. As a result, ResNet-50 can construct very deep networks and extract more complex hierarchical features, thereby improving performance in image classification tasks.

3.4. VGG 16

VGG-16 is a neural network architecture introduced by Simonyan and Zisserman in 2014. This architecture has 16 layers (13 convolutional layers, 2 fully connected layers, and 1 classifier layer) consisting of 3x3 convolutional layers and pooling layers. VGG-16 has the main advantage of its simplicity, which allows high-resolution image processing and good accuracy in complex image classification [19] [20]. In addition, VGG-16 is widely used as a pre-trained model for various image classification tasks via transfer learning.

This model extracts features hierarchically via five convolutional blocks, followed by a pooling layer, resulting in increasingly complex feature representations at each stage [19]. In the process, an input image of size 224x224x3 will go through a series of convolutional and pooling layers to produce increasingly complex hierarchical feature representations. The feature extraction results are then flattened and passed to three fully connected layers, each of which combines high-level features and performs classification. The final layer uses the Softmax activation function to generate probabilities for each class [21]. The VGG-16 architecture is shown in Fig. 4.

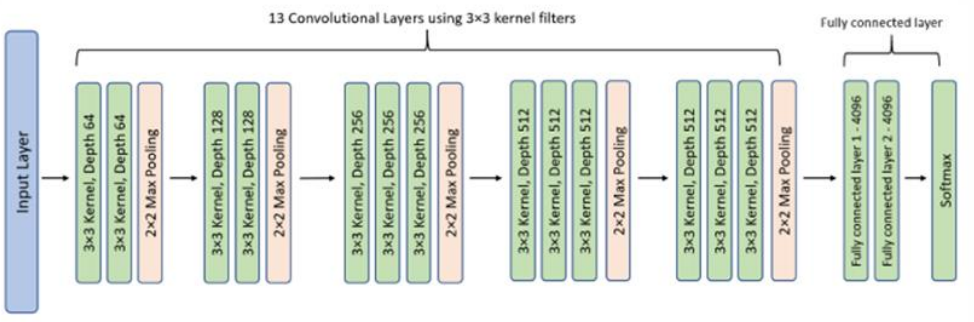

Fig. 4. VGG-16 Architecture

In the process, the input image of size 224×224×3 passes through a series of convolutional and pooling layers to produce increasingly complex hierarchical feature representations. The extracted features are then flattened and passed to three fully connected layers, each of which combines high-level features and performs classification. The final layer uses a softmax activation function to generate the probability for each class [22]. The convolution operation can be mathematically expressed as:

$$S(i,j) = (X * W)(i,j) = \sum\sum X(i+m, j+n) \cdot W(m,n) \quad (6)$$

where X represents the input image, W is the kernel/filter, and $S(i,j)$ is the convolution result at a specific position. This process is applied repeatedly at each layer to produce increasingly complex feature representations [23]. After the convolution process, the Rectified Linear Unit (ReLU) activation function is applied to introduce non-linearity into the model and to mitigate the vanishing gradient problem, which is formulated as:

$$f(x) = \max(0, x) \quad (7)$$

Subsequently, a pooling operation is performed to reduce feature dimensions while retaining important information, thereby improving computational efficiency and reducing the risk of overfitting. The extracted features are then flattened and passed to the fully connected layer to integrate high-level information [23]. Mathematically, the feature aggregation process at this stage can be represented as:

$$F(x) = \frac{1}{N} \sum_{i=1}^{N} x_i \quad (8)$$

In the final stage, the softmax activation function is used to produce the probability distribution over each output class, which is formulated as:

$$P(j \mid x) = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \quad (9)$$

With this structure, VGG-16 can extract features in a deep, hierarchical manner, yielding discriminative representations. This makes it effective for various image classification tasks, including datasets with complex visual characteristics such as Balinese Ramayana wayang character images [20].

3.5. Transfer Learning

Transfer learning is a method that uses a previously trained network as a starting point for learning subsequent tasks [24]. This method aims to streamline the training process and make it more effective [25]. Transfer learning is performed by freezing several layers at the beginning and only training the last layer used for classification. The frozen layer extracts feature common to all images, such as patterns, angles, and gradients. Meanwhile, the unfrozen layer is used to extract other features in the new dataset [26]. The transfer learning process is shown in Fig. 5.

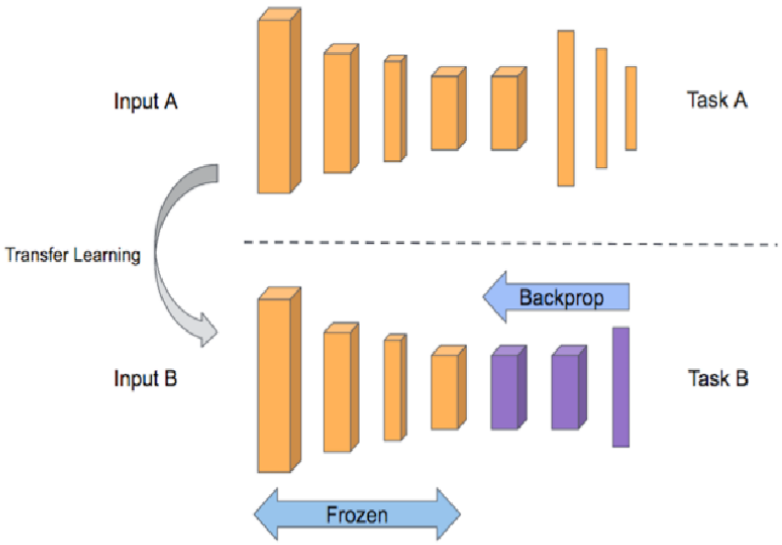

Fig. 5. Transfer Learning Process

During the model training phase, we adjusted model parameters based on predetermined hyperparameters, including the number of epochs and batch size, using the training dataset while monitoring performance on the validation dataset through accuracy and loss values to ensure effective learning and reduce overfitting. After training was completed, this study conducted model testing using unseen test data to evaluate the model's ability to recognize previously learned patterns and classify new data across all wayang character classes. Furthermore, model evaluation was performed using a confusion matrix to compare predicted labels with actual class labels and identify correctly and incorrectly classified instances for each class. In addition, a classification report was used to summarize model performance through precision, recall, and F1-score metrics, [27].

In this study, the we conduct comparisons of data-sharing ratios, model architectures, and training parameter adjustments. The dataset sharing ratios used were 50:25:25, 60:20:20, 70:15:15, 80:10:10, and 90:5:5, with the sequence training:validation:test. These scenarios aimed to examine the effect of the proportions of training, validation, and testing data on model classification performance. The architectures used in this study were ResNet50 and VGG16. The two architectures were compared to determine which performed better at classifying monkey characters from the Balinese Ramayana Wayang using a transfer learning approach. In this study, the parameter scenarios used consisted of epoch 16 with a batch size of 14, epoch 32 with a batch size of 32, and epoch 64 with a batch size of 128. Through a combination of variations in the data-sharing ratio, model architecture, and training parameters, an analysis was conducted to determine the best configuration for achieving optimal classification performance.

4. Experimental Setup

This study used a dataset of images of monkey characters from the Balinese Wayang Ramayana puppet show, consisting of 15 classes, with an initial total of 256 images before ratio division. All images were saved as JPGs and resized to 1080x1080 pixels. The dataset was divided into several ratio scenarios: 50:25:25, 60:20:20, 70:15:15, 80:10:10, and 90:5:5, in the order training:validation:test. The data was divided stratified to maintain a balanced distribution of each class within each subset.

Data augmentation was applied only to the training data to increase data variation and reduce the risk of overfitting. The augmentation techniques used included horizontal flipping, rotation by 30°, 45°, 90°, -30°, -45°, and -90°, zooming in and out by 20%, and brightness adjustment. After the augmentation process, the amount of training data varied depending on the scenario used, while the validation and test data remained unchanged. The experiment was run using Python with the PyTorch framework on a device with an Intel Core i5 processor and an NVIDIA RTX 3050 GPU. The training process was carried out according to a predetermined number of epochs.

This study compared two architectures, ResNet50 and VGG16, using a transfer learning approach. The model was initialized with pre-trained weights, and some of the initial layers were frozen to retain previously learned basic features. The final layer was modified to match the 15 classes in this study. The training parameters used consisted of several combinations: 16 epochs with a batch size of 14, 32 epochs with a batch size of 32, and 64 epochs with a batch size of 128. The optimizer used was Adam with the CrossEntropyLoss loss function for multi-class classification.

This study compares the performance of two CNN architectures, ResNet-50 and VGG-16, while analyzing how their architectural characteristics affect fine-grained classification performance on a limited dataset with high visual similarity. We explore the classification of monkey characters from the Balinese Ramayana puppet show, where similarities in shape, color, and ornamentation across classes create challenges in extracting discriminative features. VGG-16 is evaluated for its hierarchical feature learning capability through a sequential deep-layer architecture, whereas ResNet-50 is analyzed for its residual connection mechanism that helps preserve information and improve training stability in deeper networks. Furthermore, this study investigates whether the residual learning approach in ResNet-50 provides advantages over conventional architectures such as VGG-16 under constrained data conditions. To ensure an objective comparison, we apply equal hyperparameter settings, including epochs, batch size, learning rate, optimizer, and dataset partitioning, so that performance differences can be attributed primarily to the architectural characteristics of each model.

5. Result and Analysis

Based on testing 30 scenarios, it was found that most configurations achieved very high accuracy, with some achieving perfect accuracy on the test data. This indicates that the model has a strong ability to learn patterns from the dataset. However, these high accuracy values do not necessarily reflect optimal model performance, as results may be influenced by data distribution, particularly in scenarios with a relatively small test data proportion. Therefore, further analysis is needed to ensure that the selected model has strong generalization to previously unseen data.

In the ResNet-50 architecture, scenario 14 achieved high performance with training accuracy of 99.95%, validation accuracy of 100%, and test accuracy of 98.33%. The relatively small difference among these three metrics indicates that the model maintains consistent performance between the training and test data, suggesting good generalization ability. Furthermore, the F1-score of 98.31% also indicates that the model has a good balance between precision and recall in the classification process. ResNet-50 results are shown in the Table. 1.

Table 1. ResNet-50 Results in Scenario 14

Metric	Grade
Training Accuracy	99.95%
Validation Accuracy	100.00%
Testing Accuracy	98.33%
Training Loss	0.02%
Validation Loss	0.05%
Testing Loss	0.69%
F1-score	98.31%

The classification report results in Table 2 show that most classes, such as Anala Gini, Anggada, Anila, Asti Muka, Hanuman, Indra Janur, Jembawan, Kesari, Menda, Mong Muka, Sampati, Sarpa Muka, and Sugriwa, have precision, recall, and F1-score values of 1.00. This indicates that the model can recognize and classify images in these classes very well. However, the Gawaksa class has a precision of 0.83, and the Bukan Wayang Kera class has a recall of 0.80, indicating that some samples still exhibit prediction errors. The classification report for ResNet-50 is shown in Table 2.

Table 2. Clasification Report Resnet-50

Class	Precision	Recall	F1-Score
Anala Gini	1.00	1.00	1.00
Anggada	1.00	1.00	1.00
Anila	0.80	1.00	0.89
Asti Muka	1.00	1.00	1.00
Bukan Wayang Kera	1.00	1.00	1.00
Gawaksa	1.00	1.00	1.00
Hanuman	1.00	1.00	1.00
Indra Janur	1.00	1.00	1.00
Jembawan	1.00	1.00	1.00
Kesari	1.00	1.00	1.00
Menda	1.00	1.00	1.00
Mong Muka	1.00	1.00	1.00
Sampati	1.00	0.75	0.86
Sarpa Muka	1.00	1.00	1.00
Sugriwa	1.00	1.00	1.00

Based on the training and validation accuracy graphs in Fig 10, the model showed a rapid increase in performance in the initial epochs, then reached a stable state near the maximum value. This pattern indicates that the model quickly captures important features of the wayang image and converges without significant fluctuations. This is reinforced by the training and validation loss graphs, which show a consistent downward trend until they reach very small values, indicating that the optimization process is running well and the model has successfully minimized prediction errors. The accuracy and loss graphs for ResNet-50 are shown in Fig. 6.

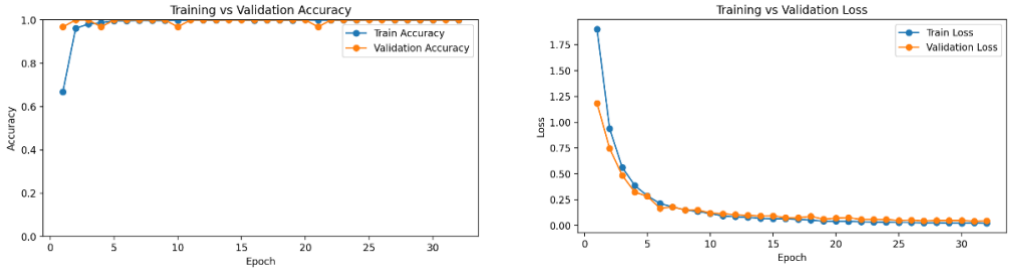

Fig. 6. Accuracy and Loss Graph of ResNet-50

Despite the model's overall high performance, the confusion matrix in Fig. 7 shows that classification errors persist across several classes. One of the most prominent errors occurs in the Sampati class, where a single data item was misclassified as the Anila class. This error indicates that the model still struggles to distinguish between classes with similar visual characteristics.

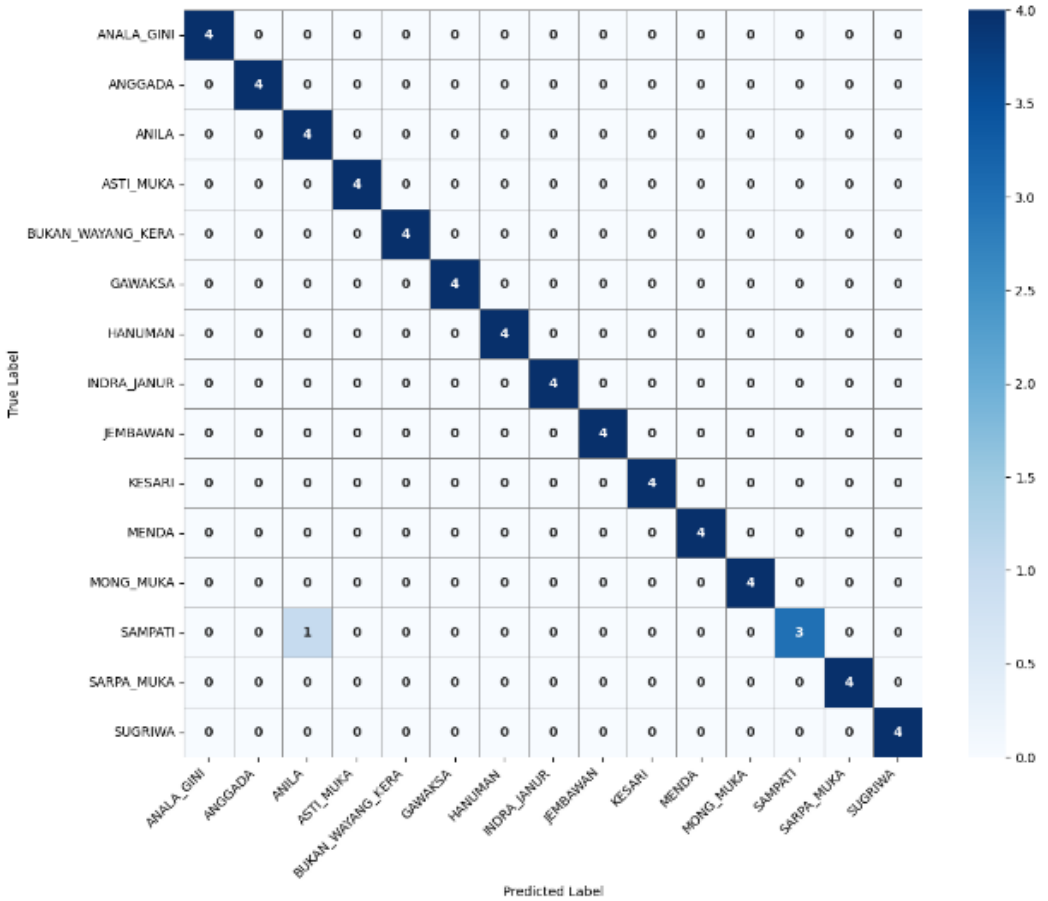

Fig. 7. Confusion Matrix ResNet-50

Upon further analysis, the misclassification between Sampati and Anila is likely due to the visual similarities between the two characters, such as facial features, body posture, and the ornaments used in wayang puppets. This similarity can make the model's extracted features less discriminatory, leading to incorrect predictions in some cases. This suggests that while the model is capable of recognizing global patterns well, it still struggles to distinguish local details when classes are highly similar.

The visualization of the prediction results in Fig. 8 further supports this finding, as the image labeled Sampati is actually predicted as Anila. The differences between the two classes are not visually striking, leading to ambiguity in classification. Therefore, this error does not necessarily indicate a weakness of the model, but rather reflects the inherent challenges posed by datasets with similar characteristics across classes.

Fig. 8. The Sampati Monkey Puppet (Left) Is Predicted To Be The Anila Monkey Puppet (Right)

Overall, the ResNet-50 model in scenario 14 demonstrated excellent performance, with high accuracy and a stable training process. However, the analysis also indicated that the model still has limitations in distinguishing classes with high visual similarity. Therefore, improving dataset quality or adding more data variations could be solutions to improve model performance in the future.

In the VGG-16 architecture, scenario 10 achieved excellent performance, with training accuracy of 99.62%, validation accuracy of 100%, and test accuracy of 98.67%. The relatively small difference among these three values indicates that the model maintains consistent performance between the training data and previously unseen data. This indicates that the model does not experience significant overfitting and has good generalization capabilities. VGG-16 results are shown in the Table. 3.

Table 3. VGG-16 Results in Scenario 10

Metric	Grade
Training Accuracy	99.62%
Validation Accuracy	100.00%
Testing Accuracy	98.67%
Training Loss	0.12%
Validation Loss	0.00%
Testing Loss	0.16%
F1-score	98.65%

The classification report results show that most classes, such as Anala Gini, Anggada, Anila, Asti Muka, Hanuman, Indra Janur, Jembawan, Kesari, Menda, Mong Muka, Sampati, Sarpa Muka, and Sugriwa, have precision, recall, and F1-score values of 1.00. This indicates that the model can recognize and classify images in these classes very well. However, the Gawaksa class has a precision of 0.83, and the Bukan Wayang Kera class has a recall of 0.80, which indicates that there are still prediction errors in some samples. The classification report for VGG-16 is shown in Table 4.

Table 4. Classification Report VGG-16

Class	Precision	Recall	F1-Score
Anala Gini	1.00	1.00	1.00
Anggada	1.00	1.00	1.00
Anila	1.00	1.00	1.00
Asti Muka	1.00	1.00	1.00
Bukan Wayang Kera	1.00	0.80	0.89
Gawaksa	0.83	1.00	0.91
Hanuman	1.00	1.00	1.00
Indra Janur	1.00	1.00	1.00
Jembawan	1.00	1.00	1.00
Kesari	1.00	1.00	1.00
Menda	1.00	1.00	1.00
Mong Muka	1.00	1.00	1.00
Sampati	1.00	1.00	1.00
Sarpa Muka	1.00	1.00	1.00
Sugriwa	1.00	1.00	1.00

Based on the training and validation accuracy graphs in Fig. 13, the model showed a fairly rapid increase in performance during the initial epochs, then reached a stable state near the maximum value. The validation accuracy curve, which consistently maintains a high value with minimal fluctuations, indicates that the model effectively captures data patterns. This also indicates that the training process is running optimally, with no indication of excessive underfitting or overfitting. The training and validation loss graphs in Fig. 9 show a consistent downward trend as the number of epochs increases. At the beginning of training, the loss value is relatively high, but gradually decreases to near zero. This pattern indicates that the model has effectively minimized prediction errors. Furthermore, the lack of a significant spike in validation loss indicates that the model did not overfit during training.

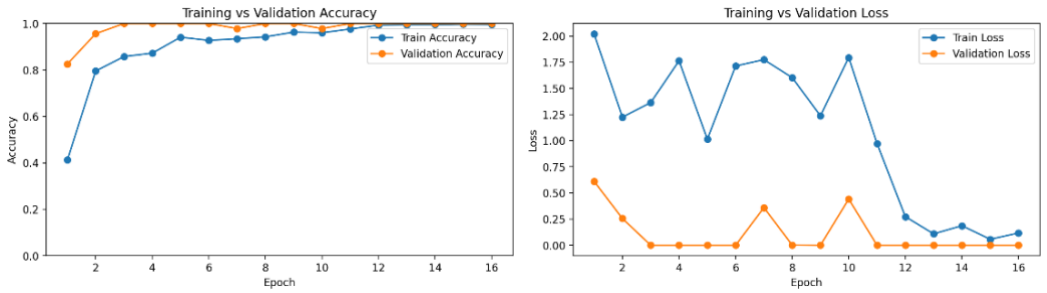

Fig. 9. VGG-16 Accuracy and Loss Graph

Despite the model's overall high performance, the confusion matrix results in Fig 10 indicate that several classification errors still occurred. One such error occurred in the Non-Wayang Kera class, which was predicted as Gawaksa. This error indicates that the model still struggles to distinguish between classes with visual characteristics that share some degree of similarity.

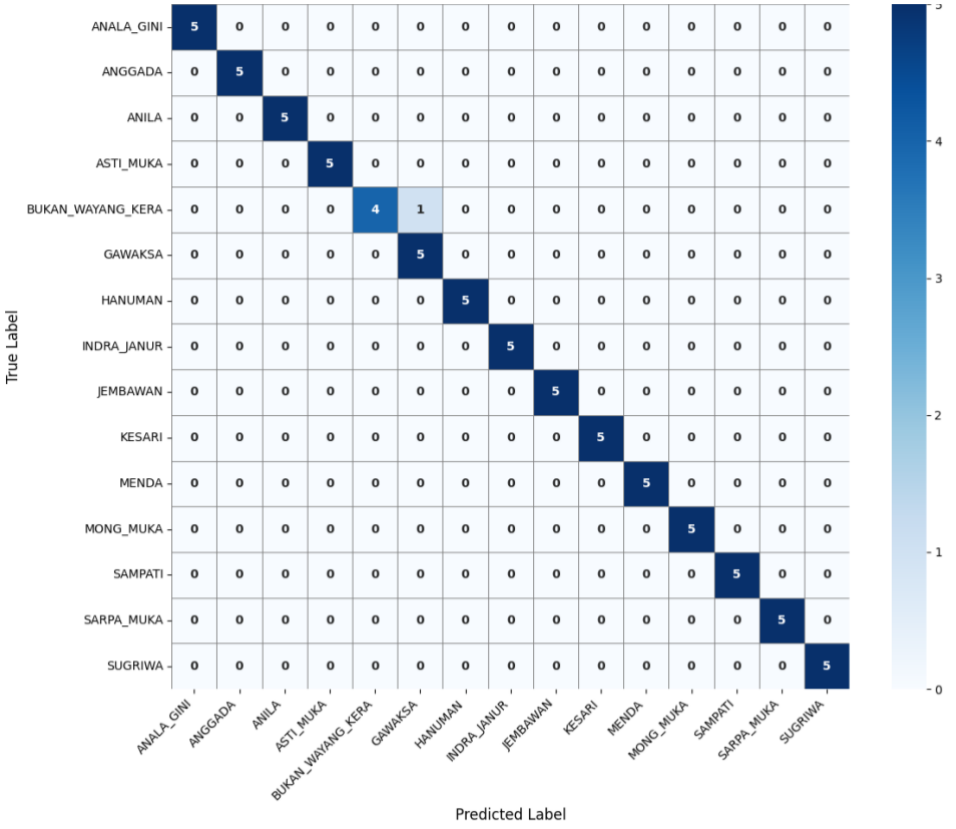

Fig. 10. Confusion Matrix VGG-16

Upon further analysis, the similarity between the Bukan Wayang Kera and Gawaksa classes likely lies in their body structure, color composition, and ornamental details in the wayang images. Both classes exhibit complex visual patterns and share similar decorative elements, rendering the features extracted by the model less discriminatory. Furthermore, CNN architectures such as VGG-16 tend to be more powerful at capturing global patterns than fine local details, making it difficult to distinguish small differences between classes with high similarity optimally. Variations in lighting, shooting angles, and object positions can also minimize visual differences between classes, increasing the likelihood of misclassification. As a result, the model tends to associate images from the Bukan Wayang Kera class with the Gawaksa class under certain conditions.

These findings indicate that, despite the model's high overall performance, its ability to discriminate between classes with high visual similarity remains a challenge, especially in fine-grained classification. also supports these findings, as some predicted images do not match the actual labels. This error indicates that while the model is very good at recognizing global patterns, it still struggles to distinguish local details between classes with high visual similarity. Therefore, the VGG-16 model's performance in this scenario is very good, but it still has room for improvement, especially in discriminating between similar classes. The visualization of the prediction results in Fig. 11.

Fig.11. Not A Monkey Puppet (Left) Predicted to Be a Gawaksa Monkey Puppet (Right)

Based on the test results for the best scenarios for each architecture, namely scenario 10 for VGG-16 and scenario 14 for ResNet-50, both models demonstrated very high performance with accuracy values above 98%. However, upon further analysis, there are differences in performance characteristics, including stability, generalization ability, and classification error patterns. In terms of stability, both models exhibited good convergence during training, as evidenced by consistently increasing training and validation accuracy curves that then stabilized at a maximum. However, VGG-16 tended to exhibit a smoother pattern and minimal fluctuations during training, particularly in the loss graph, which decreased consistently without significant spikes. In contrast, ResNet-50 still showed slight fluctuations in some epochs, though it still achieved good convergence. This indicates that VGG-16 has slightly better training stability than ResNet-50 in the scenario used.

In terms of generalization ability, both models showed relatively small differences between training, validation, and test accuracies, indicating they did not experience significant overfitting. However, VGG-16 showed more consistent performance across these three metrics, while ResNet-50, despite achieving very high training and validation accuracy, showed a slight decrease in testing accuracy. This indicates that VGG-16 has slightly better generalization ability, maintaining performance on unseen data.

In terms of classification errors, both models showed very low error rates, but with different patterns. In VGG-16, errors occurred in the Bukan Wayang Kera class, which was predicted as Gawaksa, while in ResNet-50, errors occurred in the Sampati class, which was predicted as Anila. These two errors share similar characteristics, occurring in pairs of classes that are highly visually similar. However, compared with ResNet-50, the errors in ResNet-50 have a greater impact on recall values for certain classes, indicating that the model tends to fail to recognize some samples in those classes.

Overall, both architectures performed very well in shadow puppet image classification. However, VGG-16 tended to outperform in terms of training stability and performance consistency, while ResNet-50 demonstrated an excellent ability to achieve high accuracy with little variation during training. These findings suggest that architecture selection depends not only on accuracy values but also on the model's performance in handling data variation and classification complexity.

6. Conclusion

This study comparatively evaluated the performance of the ResNet-50 and VGG-16 architectures for the classification of monkey characters in the Balinese Ramayana puppet dataset through 30 experimental scenarios. Based on the obtained results, both architectures demonstrated excellent classification capability, with several scenarios achieving accuracy values above 98%, indicating that the models successfully learned important visual patterns from the dataset. Among the evaluated configurations, scenario 14 for ResNet-50 and scenario 10 for VGG-16 produced the best performance for their respective architectures. ResNet-50 achieved training, validation, and testing accuracies of 99.95%, 100%, and 98.33%, respectively, with an F1-score of 98.31%, while VGG-16 achieved 99.62% training accuracy, 100% validation accuracy, and 98.67% testing accuracy with an F1-score of 98.65%. The relatively small differences between training, validation, and testing performance indicate that both models possess strong generalization ability and do not experience significant overfitting. Furthermore, the accuracy and loss curves showed stable convergence during training, demonstrating that the optimization process was effective in minimizing prediction errors and capturing relevant features from wayang puppet images.

This study also found that most classes achieved precision, recall, and F1-score values close to or equal to 1.00, indicating that both architectures were highly effective in recognizing the majority of puppet character classes. However, several classification errors still occurred in visually similar classes, such as the misclassification of Sampati as Anila in ResNet-50 and the prediction of Bukan Wayang Kera as Gawaksa in VGG-16. These findings indicate that although CNN-based transfer learning models are capable of extracting strong global visual representations, distinguishing fine-grained local details between highly similar classes remains a challenge. The confusion matrix analysis and prediction visualizations further confirmed that similarities in facial structure, ornamentation, color composition, and object posture contributed significantly to classification ambiguity.

Therefore, this study demonstrates that both ResNet-50 and VGG-16 can produce highly accurate and reliable classification models for Balinese Ramayana monkey puppet recognition using transfer learning approaches. Nevertheless, VGG-16 showed slightly better stability and consistency during training and testing, whereas ResNet-50 demonstrated excellent capability in achieving high accuracy through its residual learning mechanism. These findings suggest that architecture selection should not rely solely on accuracy values, but also consider training stability, generalization capability, and robustness in handling visually similar classes. Furthermore, this study indicates that improving dataset quality, increasing image variation, and incorporating additional fine-grained feature extraction techniques may further enhance classification performance in future research.

References

[1] H. K. Maulana, “Application of CNN-EfficientNetB2 Architecture with Transfer Learning for Wayang Kulit Character Image Classification,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 1, Jan. 2025. doi: https://doi.org/10.23960/jitet.v13i1.5626.

[2] W. Sugita, “The Function of Balinese Wayang Kulit Performing Arts in the Bhima Swarga Story for Yadnya Ceremonies,” *Jurnal Penelitian Agama Hindu*, 2022. [Online]. Available: https://www.academia.edu/download/103930295/830.pdf. [Accessed: Mar. 1, 2026].

[3] S. Subiyantoro, D. Fahrudin, and S. B. Amirulloh, “Character Education Values of Pancasila Student Profiles in the Puppet Figure Wayang Arjuna: A Javanese Cultural Perspective,” *ISVS E-Journal*, vol. 10, no. 6, pp. 106–118.

[4] N. Hilmi, E. Y. Puspaningrum, and H. E. Wahanani, “Implementation of the K-Nearest Neighbor (KNN) Algorithm for Citrus Plant Disease Identification Based on Leaf Images,”

Router: Jurnal Teknik Informatika dan Terapan, vol. 2, no. 2, pp. 107–117, Jun. 2024. doi: https://doi.org/10.62951/router.v2i2.78.

[5] K. R. M. Manikam, L. J. E. Dewi, K. Y. E. Aryanto, K. A. Seputra, and P. Varnakovida, "Hyperparameter Analysis in Plant Type Classification Using ResNet50 and MobileNetV2 Algorithms," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 6, pp. 9921–9928, Nov. 2025. doi: https://doi.org/10.36040/jati.v9i6.15832.

[6] K. Y. Mahendra, L. J. E. Dewi, I. K. Purnamawan, and F. B. Pasaribu, "Songket Motif Classification Using MobileNet Architecture for Cultural Heritage Preservation," *International Journal of Informatics and Computation (IJICOM)*, vol. 7, no. 2, Dec. 2025. doi: https://doi.org/10.35842/ijicom.

[7] Muhathir, N. Khairina, I. Barus, M. Ula, and I. Sahputra, "Preserving Cultural Heritage Through AI: Developing LeNet Architecture for Wayang Image Classification," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 9, Jan. 2023. doi: https://doi.org/10.14569/ijacsa.2023.0140919.

[8] Herlangga, "Application of Transfer Learning EfficientNetB3 for West Sumatran Traditional Weapon Recognition Using Convolutional Neural Network (CNN)," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 2, Apr. 2024. doi: https://doi.org/10.23960/jitet.v12i2.4256.

[9] H. Farman, J. Ahmad, B. Jan, Y. Shahzad, M. Abdullah, and A. Ullah, "EfficientNet-Based Robust Recognition of Peach Plant Diseases in Field Images," *Computers, Materials & Continua*, vol. 71, no. 1, pp. 2073–2089, 2022. doi: https://doi.org/10.32604/cmc.2022.018961.

[10] P. H. Setadewa, K. Y. Ernanda Aryanto, and L. J. Erawati Dewi, "Urban Building Segmentation in High-Resolution Satellite Imagery: CNN, U-Net (VGG16), and DeepLabV3+ (ResNet-50)," *STORAGE Jurnal Ilmiah Teknik dan Ilmu Komputer*, vol. 4, no. 4, pp. 337–347, Nov. 2025. doi: https://doi.org/10.55123/storage.v4i4.6552.

[11] G. A. R. Wijaya, K. Y. E. Aryanto, and N. P. N. P. Dewi, "Comparative Analysis of U-Net Attention and ResNet-50 for River Semantic Segmentation in Remote Sensing Images," *STORAGE Jurnal Ilmiah Teknik dan Ilmu Komputer*, vol. 4, no. 4, pp. 393–400, Nov. 2025. doi: https://doi.org/10.55123/storage.v4i4.6637.

[12] D. A. Pusparani, M. W. A. Kesiman, and K. Y. E. Aryanto, "Identification of Little Tuna Species Using Convolutional Neural Networks (CNN) Method and ResNet-50 Architecture," *Indonesian Journal of Artificial Intelligence and Data Mining*, vol. 8, no. 1, p. 86, Dec. 2024. doi: https://doi.org/10.24014/ijaidm.v8i1.31620.

[13] K. N. Ananda, N. P. N. Puspa Dewi, N. W. Marti, and L. J. E. Dewi, "Multilabel Classification of Elementary School Students' Learning Styles Using Machine Learning Algorithms," *Journal of Applied Computer Science and Technology*, vol. 5, no. 2, pp. 144–154, Dec. 2024. doi: https://doi.org/10.52158/jacost.v5i2.940.

[14] K. P. Yudhantara, N. K. Kertiasih, and I. N. S. Wahyu Wijaya, "Comparison of Mangrove Leaf Disease Classification Models Using VGG16 and MobileNetV2 Architectures," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 14, no. 1, Jan. 2026. doi: https://doi.org/10.23960/jitet.v14i1.8816.

[15] B. Anthony and Y. Yohannes, "Kinship Verification Using ResNet50 Architecture," *MDP Student Conference*, vol. 2, no. 1, pp. 265–273, Apr. 2023. doi: https://doi.org/10.35957/mdp-sc.v2i1.4320.

[16] Aboghanem, M. Abd Elfattah, H. M. Amer, and A. Tawkol Khalil, "A Hybrid ResNet50-Vision Transformer Model with an Attention Mechanism for Aerial Image Classification," *Scientific Reports*, vol. 16, no. 1, Feb. 2026. doi: https://doi.org/10.1038/s41598-026-36492-4.

[17] Wan, B. Li, K. Wang, X. Teng, T. Wang, and B. Mao, "An Improved ResNet50 for Environment Image Classification," *Procedia Computer Science*, vol. 242, no. 3, pp. 1000–1007, Jan. 2024. doi: https://doi.org/10.1016/j.procs.2024.08.246.

[18] M. A. Sultan *et al*., "Brain Tumor Detection and Classification Using Fine-Tuned CNN with ResNet50 and EfficientNet," *International Journal of Informatics and Computation*, vol. 6, no. 1, p. 22, Aug. 2024. doi: https://doi.org/10.35842/ijicom.v6i1.80.

[19] N. Rumui, A. Mualo, J. Rahayaan, L. Batjo, and M. Mokansi, "Comparative Analysis of CNN Deep Learning Models with VGG16 in Flower Type Classification," *Informatik: Jurnal Ilmu Komputer*, vol. 21, no. 1, pp. 35–44, Apr. 2025. doi: https://doi.org/10.52958/iftk.v21i1.11105.

[20] Rismiyati and A. Luthfiarta, “VGG16 Transfer Learning Architecture for Salak Fruit Quality Classification,” *Jurnal Informatika dan Teknologi Informasi*, vol. 18, no. 1, pp. 37–48. doi: https://doi.org/10.31515/telematika.v18i1.4025.
[21] N. Deb and T. Rahman, “An Efficient VGG16-Based Deep Learning Model for Automated Potato Pest Detection,” *Smart Agricultural Technology*, vol. 12, no. 4, p. 101409, Sep. 2025. doi: https://doi.org/10.1016/j.atech.2025.101409.
[22] M. Hussain, T. Thaher, M. B. Almourad, and M. Mafarja, “Optimizing VGG16 Deep Learning Model with Enhanced Hunger Games Search for Logo Classification,” *Scientific Reports*, vol. 14, no. 1, Dec. 2024. doi: https://doi.org/10.1038/s41598-024-82022-5.
[23] S. Winiarti, S. Sunardi, and A. Fadhil, “Deep Learning for Malay Architectural Identification: A CNN Approach to Heritage Recognition and Preservation,” *International Journal of Informatics and Computation*, vol. 7, no. 1, pp. 154–167, May 2025. doi: https://doi.org/10.35842/ijicom.v7i1.116.
[24] Solihin, D. I. Mulyana, and M. B. Yel, “Classification of Papuan Traditional Musical Instruments Using Transfer Learning and Data Augmentation Methods,” *Jurnal Sistem Komputer dan Kecerdasan Buatan*, vol. 5, no. 2, pp. 36–44, Mar. 2022. doi: https://doi.org/10.47970/siskom-kb.v5i2.279.
[25] E. Anggiratih, S. Siswanti, S. K. Octaviani, and A. Sari, “Rice Plant Disease Classification Using EfficientNet B3 Deep Learning Model with Transfer Learning,” *Jurnal Ilmiah SINUS*, vol. 19, no. 1, p. 75, Jan. 2021. doi: https://doi.org/10.30646/sinus.v19i1.526.
[26] E. Putra, M. F. Naufal, and V. R. Prasetyo, “Spice Type Classification Using Convolutional Neural Networks and Transfer Learning,” *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, vol. 9, no. 1, pp. 12–12, Apr. 2023. doi: https://doi.org/10.26418/jp.v9i1.58186.
[27] N. W. Y. Wiani, I. M. A. Wirawan, and K. Y. E. Aryanto, “Hand Gesture Classification Based on sEMG Signals Using Deep Learning,” *Jurnal Edukasi dan Penelitian Informatika*, vol. 11, no. 1.