Matrix: Jurnal Manajemen Teknologi dan Informatika
Volume 16 Issue 2 Year 2026 Pages 101-116
DOI: http://dx.doi.org/10.31940/matrix.v16i2.101-116
URL: http://ojs2.pnb.ac.id/index.php/matrix

Image classification of rerajahan ulap-ulap Bali using MobileNetV2 architecture

Pande Putu Ode Juliantara. KW [1]*, I Gede Aris Gunadi [2], I Made Gede Sunarya [3], Ni Nyoman Emang Smrti [4]

[1,2,3] *Computer Science Study Program, Universitas Pendidikan Ganesha, Indonesia*
[4] *Informatics Engineering Study Program, STMIK Bandung Bali, Indonesia*

*Corresponding Author: *odepande21@gmail.com*

Abstract: *Rerajahan Ulap-Ulap* is a visual expression in Balinese Hindu culture that functions as a sacred element in the *mlaspas* ritual of buildings, containing sacred scripts and specific ornaments. The visual complexity of these motifs presents a challenge in accurately recognizing their types, necessitating a technological approach to support the documentation of this cultural heritage. This study aims to develop an image classification model using the MobileNetV2 architecture with a transfer learning method. The research utilized a primary dataset of 810 original images across nine motif classes, expanded through augmentation to 3,888 images (432 per class) to address data limitations and prevent overfitting. Model performance was evaluated using a 5-Fold Cross Validation method across three optimization scenarios: AdamW, Nadam, and Adagrad. Experimental results demonstrate that Scenario 1 (AdamW) achieved superior performance with an average accuracy of 98.64%, followed by Nadam (98.52%) and Adagrad (71.23%). These findings indicate that the combination of MobileNetV2 and AdamW optimization provides a reliable solution for automated classification of *rerajahan*, serving as a digital instrument to support Balinese cultural preservation efforts.

Keywords: *Rerajahan Ulap-Ulap* Bali, Image Classification, MobileNetV2, Data Augmentation, Cultural Heritage.

History Article: Submitted 22 March 2026 | Revised 5 May 2026 | Accepted 12 May 2026

How to Cite: P. P. O. Juliantara. KW, I. G. A. Gunadi, I. M. G. Sunarya, and N. N. E. Smrti, *"Image classification of rerajahan ulap-ulap Bali using MobileNetV2 architecture," Matrix: Jurnal Manajemen Teknologi dan Informatika*, vol. 16, no. 2, pp. 101–116, 2026, doi: 10.31940/matrix.v16i2.101-116

Introduction

Rerajahan Ulap-Ulap is a distinctive visual expression within Balinese Hindu culture that functions as a sacred element in the *mlaspas* ritual (the purification of both temples and residential buildings). In general, *ulap-ulap* takes the form of a white cloth measuring approximately 20–30 cm in length and about 15–20 cm in width [1]. On the surface of the cloth, symbols are inscribed in the form of sacred scripts (*aksara nyasa pranawa/modre*) and specific ornaments using black ink, which carry profound religious and philosophical meanings. The creation of *ulap-ulap* is not conducted arbitrarily but is performed by individuals who have undergone a process of self-purification and possess spiritual authority, such as *Pemangku/Pinandita* or *Pandita*, thereby ensuring the preservation of its sacred legitimacy within religious practices. In addition to functioning as a symbol of spiritual protection, *rerajahan ulap-ulap* is also believed to possess metaphysical power, serving as a medium to repel negative energies and to maintain the spiritual balance of a building or sacred space. In Balinese Hindu tradition, *rerajahan* is understood as symbolic imagery or inscriptions that contain religious and magical power, used as a ritual medium and as a means of warding off misfortune in the religious practices of Balinese society [2]. Each symbol inscribed on the *ulap-ulap* cloth carries specific meanings related to Balinese Hindu cosmology, such as protection from *bhuta kala*, spatial purification, and the harmonization of relationships between humans, nature, and God, as conceptualized in the *Tri Hita Karana* philosophy [3].

In the current digital era, advancements in computer vision and deep learning have catalyzed the transition from manual expert interpretation of traditional visual imagery toward automated recognition through computational approaches [4]. However, applying such

technological interventions to *rerajahan ulap-ulap* presents unique challenges, primarily due to its profoundly sacred and esoteric (*kadyatmikan*) nature. This preservation effort is further complicated by deep-seated community beliefs that the indiscriminate study or digitization of these symbols may invoke misfortune or spiritual disturbances. These restrictive conditions have led to a scarcity of experts who possess a comprehensive understanding of the formal rules (*pakem*) and philosophical meanings embedded within these sacred scripts. Consequently, this specialized knowledge is experiencing a gradual decline and faces a significant risk of extinction [5]. In addition, there is a gap in scientific studies connecting the visual cultural heritage of rerajahan with automated image classification methods. In fact, the visual complexity and similarities between patterns often make it difficult for the general public to recognize the types accurately [6]. Developing classification models based on computer vision offers significant potential as an innovative cultural preservation tool. Beyond its ability to identify visual features of profound religious significance, this technology facilitates digital documentation and a more secure, structured dissemination of cultural knowledge for upcoming generations [7].

One of the significant developments in the field of computer vision and deep learning is the emergence of Convolutional Neural Networks (CNN) [8]. CNN is a method within artificial neural networks that utilizes multiple layers to perform the learning process. Its development concept is inspired by the mechanism of neural networks in mammals in processing visual perception. In the context of image recognition, CNN operates by learning patterns present in images through training data, which are then used as a model to produce optimal outputs in the process of image identification or classification [9]. In the classification process, CNN generally requires a relatively large amount of training data in order for the model to learn patterns effectively. However, the application of various data augmentation techniques can serve as a solution to improve model performance, particularly in enhancing its generalization capability toward previously unseen images. Through data augmentation, variations within the dataset can be artificially expanded, enabling the model to become more adaptive and less prone to overfitting [10]. In CNN, various architectures are utilized to perform identification and classification of digital images. One of the widely used architectures that demonstrates high performance is MobileNetV2 [11]. MobileNet is a CNN architecture designed to reduce the high computational resource requirements in deep learning processes, as deep learning methods generally require devices with substantial computational capabilities. This architecture optimizes the use of convolutional layers, making the computation process more efficient compared to conventional CNNs. The improved version of this architecture, MobileNetV2, was introduced in April 2017 by incorporating two main components, namely linear bottleneck and shortcut connections between bottlenecks, to enhance both efficiency and model performance [12].

Previous studies have applied the MobileNetV2 architecture based on transfer learning for the classification of traditional weapons from Central Java. The dataset consisted of five classes of traditional weapons with a total of 785 images. The testing results showed that the model achieved an accuracy of 98.64% with a loss value of 0.16 [13]. Another study focused on classifying Jumputan fabric motifs from Palembang using a dataset of 800 images across four motif classes. The model was trained using several optimizer scenarios, namely AdamW, Adagrad, Nadam, and SGD, with training parameters including 100 epochs, a batch size of 32, and a learning rate of 0.001. Based on the evaluation results, the AdamW optimizer achieved the best performance with a testing accuracy of 97%, followed by Nadam at 96%, Adagrad at 95%, and SGD at 90%. These findings indicate that the use of the MobileNetV2 architecture with a transfer learning approach can produce high classification performance in recognizing traditional fabric motifs [4]. Another study also developed a classification system for Songket Jinengdalem fabric motifs using the MobileNetV2 architecture to support cultural preservation through image recognition technology. The dataset consisted of various Balinese Songket motifs obtained directly from artisans in Jinengdalem Village. The evaluation results showed that the MobileNetV2 model achieved an average training accuracy of 99.98% [7].

Although various studies have demonstrated the successful application of the MobileNetV2 architecture in the classification of cultural image objects such as traditional fabric motifs and other visual artifacts, studies that specifically address the classification of religious symbolic images based on *rerajahan* remain very limited. Most previous research has focused on visual objects that are decorative in nature or have relatively structured patterns, such as batik, songket, and other traditional fabrics, which visually exhibit texture characteristics and repetitive patterns

that are easier for deep learning models to learn. In contrast, *rerajahan ulap-ulap* has distinct visual characteristics, as it consists of a combination of sacred *modre* scripts and complex line forms that do not always exhibit consistent repetitive patterns. In addition, to date, there are still very few studies that utilize computer vision approaches to identify or classify *rerajahan* as an object of visual cultural study. To the best of our knowledge, this is the first study to apply a CNN-based deep learning approach specifically to the automated classification of sacred *Rerajahan Ulap-Ulap* imagery in Bali. This condition identifies a critical research gap, as previous studies have predominantly focused on decorative textiles with repetitive patterns rather than esoteric religious symbols. The selection of the MobileNetV2 architecture in this research is justified by its optimal balance between computational efficiency and high classification accuracy. Unlike more complex and resource-heavy architectures such as EfficientNetV2, ResNet50V2, or Vision Transformers, MobileNetV2 utilizes depthwise separable convolutions that significantly reduce the number of parameters without sacrificing feature extraction capability. This makes it particularly suitable for future implementation on mobile devices or edge computing platforms, which is a key priority for the practical and accessible digitalization of Balinese cultural heritage. Therefore, this study is important to address the existing gap by developing a specialized classification model, thereby contributing not only to the advancement of image processing technology but also to the practical preservation of traditional Balinese cultural knowledge.

Methodology

This study adopts a systematic approach to develop an image classification model for *rerajahan ulap-ulap* using deep learning techniques, specifically leveraging a transfer learning methodology. In the transfer learning process, a large-scale source dataset is utilized to train a model to learn general features such as visual patterns, which are then repurposed to solve new, similar problems. This research utilizes the MobileNetV2 architecture, initialized with pre-trained weights from the ImageNet-1k dataset, to significantly reduce the requirement for vast amounts of labeled data while accelerating the learning process. To adapt the model to the specific visual domain of *rerajahan ulap-ulap*, the entire base model (154 layers) was frozen, and a custom classification head was integrated, consisting of a Global Average Pooling layer, a Dropout layer (0.5) to mitigate overfitting, and a Dense layer with 512 neurons.

To identify the most effective optimization strategy, the training process was conducted through three distinct optimizer scenarios: AdamW, Nadam, and Adagrad. All computational processes and experiments were performed using the Google Colab platform. This study utilizes primary data obtained directly to ensure the relevance and authenticity of the visual assets used. The selection of hyperparameters, including the 0.001 learning rate, was established based on configurations from previous studies, with specific modifications to better suit the unique visual complexities of the *rerajahan ulap-ulap* dataset [4], [7], [14]. These adjustments were further validated through empirical manual tuning to ensure stable convergence across all scenarios. By combining transfer learning with robust data augmentation techniques, the model achieves sufficient feature extraction capability despite the specialized and relatively compact nature of the dataset. To ensure statistical validity and overcome potential biases from limited sample sizes, this study implemented a 5-Fold Cross Validation procedure for each optimizer scenario to produce a more generalized, robust, and stable performance estimation. The methodology consists of several integrated stages, including data acquisition, dataset splitting, pre-processing,

and augmentation, all of which are designed to ensure rigorous data preparation and effective model training. The complete framework of this research methodology is illustrated in Figure 1.

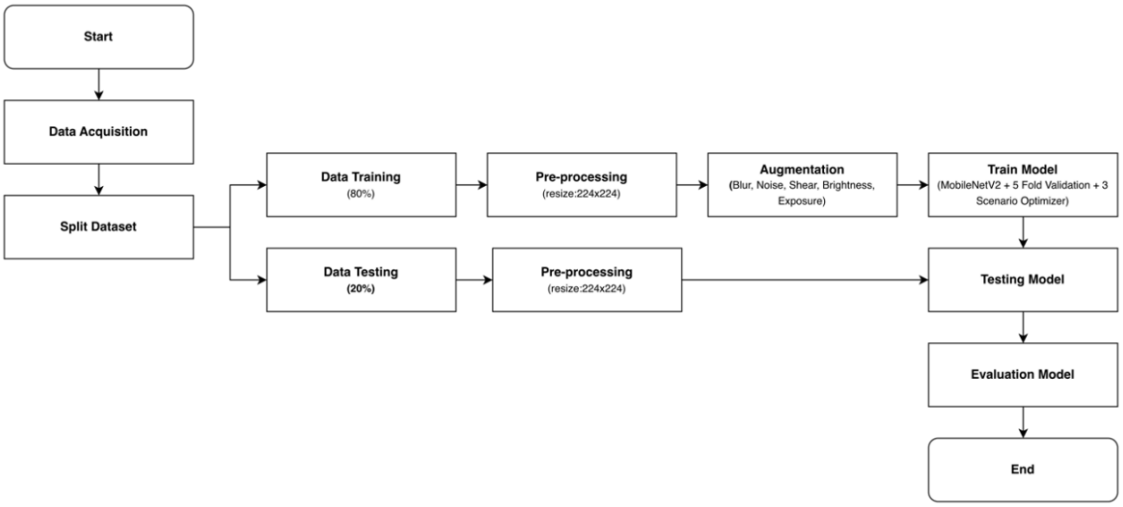


Figure 1. Research methodology

Data Acquisition

The dataset was constructed through manual drawing by five different individuals to introduce variations in form and drawing style. Each participant created *rerajahan* on a white cloth measuring 20 × 30 cm using a permanent marker to maintain consistent line clarity while preserving the authentic texture of the original medium. All drawings referred to *Makarya Ulap-Ulap Palinggih lan Wewangunan* [1]. The dataset includes nine commonly used types of *rerajahan* in traditional Balinese architecture. An example of the dataset distribution is presented in Table 1. Image acquisition was conducted using a Canon 700D DSLR camera (18 MP, 50 mm lens, f/5) at a distance of 100–150 cm under natural lighting conditions between 09:00 and 14:00 WITA to ensure clear and undistorted image quality. To enhance data diversity and model robustness, four acquisition scenarios were applied, including flat cloth, wrinkled cloth, tilted angles (15°–30°), and folded edges, simulating real-world conditions.

Figure 2. The process of drawing *Rerajahan Ulap-Ulap*

Data Augmentation

Data augmentation was applied exclusively to the training dataset to increase data variability by modifying the original images without altering their underlying labels or class identities [15]. The applied techniques include brightness and exposure adjustments to simulate lighting variations, the addition of random noise to represent environmental disturbances, blur to simulate out-of-focus conditions, and shear transformation to introduce slight geometric variations. This process effectively enhances the model's performance by expanding the diversity of the training data without the need for manual data collection.

Training Model

The model training stage is a crucial phase in which the MobileNetV2 architecture learns the visual features of *rerajahan ulap-ulap* images hierarchically through a transfer learning strategy by freezing 154 base layers and optimizing custom layers consisting of Global Average

Pooling, a 256-neuron Dense layer, and 0.6 Dropout to prevent overfitting. This experiment was conducted through three comparative optimization scenarios using the AdamW, Nadam, and Adagrad algorithms with a learning rate of 0.0001, integrated with a 5-Fold Cross Validation method to objectively evaluate the model's performance while ensuring performance stability and minimizing bias in the data samples [16]. Through this technique, a total of 3,888 image data (432 per class) were systematically divided into five balanced folds to allow each subset of data to be used as test data in rotation. In each iteration, 4 folds (3,110 data) were used as training data and 1 fold (778 data) as validation data to maintain class distribution proportions consistent with the original dataset. This process, which took place over five iterations as illustrated in Figure 3, was regulated by an Early Stopping mechanism (patience 10) to produce a final performance estimate that is robust, stable, and capable of generalizing well to unseen data.

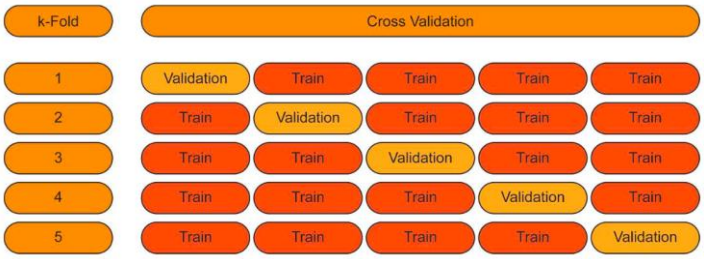


Figure 3. Illustration of 5 fold cross validation process

Further details regarding the parameter settings used during the experimental process and model training can be found in the hyperparameter configuration Table 1.

Table 1. Model training configuration

Parameter	Scenario 1	Scenario 2	Scenario 3
Pre-trained Weights	ImageNet	ImageNet	ImageNet
Input Size	224 x 224 x 3	224 x 224 x 3	224 x 224 x 3
Base Layer Status	Frozen (154 layers)	Frozen (154 layers)	Frozen (154 layers)
Custom Top Layers	GAP2D, Dense 256, Dropout 0.6, Softmax 9	GAP2D, Dense 256, Dropout 0.6, Softmax 9	GAP2D, Dense 256, Dropout 0.6, Softmax 9
Optimizer Type	AdamW	Nadam	Adagrad
Learning Rate	0.0001	0.0001	0.0001
Batch Size	32	32	32
Max Epochs	100	100	100
Early Stopping	Patience = 10	Patience = 10	Patience = 10
Validation Method	5-Fold Validation	5-Fold Validation	5-Fold Validation

Evaluation Model

Model evaluation is a systematic stage aimed at assessing the performance, quality, or effectiveness of a developed and trained algorithm or model [17]. This process is conducted after the training phase is complete, where the model is tested using a test dataset to determine the program's proficiency in performing classification or prediction on new, unseen data [18]. The model evaluation was conducted using a confusion matrix to analyze classification performance in detail. Based on this matrix, several evaluation metrics were calculated, including accuracy, precision, recall, and F1-score. Accuracy is defined as the ratio between the number of correctly predicted labels and the total number of labels, including both predicted and actual labels. The overall accuracy is obtained by averaging the accuracy across all evaluated instance [19]. The accuracy can be calculated using the formula [9]:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Precision represents the ratio between correctly predicted positive labels and the total predicted positive labels, which is then averaged across all instances [19]. The precision formula is [9]:

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

Recall is defined as the ratio between correctly predicted positive labels and the total actual positive labels, indicating the model's ability to correctly identify all relevant instances within a specific class [20]. The recall formula is expressed as [9]:

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

F1-score is a metric that combines precision and recall to provide a balanced evaluation of both. It is particularly important when there is an imbalance between precision and recall, as it uses the harmonic mean to reflect the overall classification performance of the model [20] The F1-score can be calculated using the formula [9]:

$$F1\ Score = 2\ x\ \frac{Recall\ x\ Precision}{Recall +\ Precision} \tag{4}$$

Results and Discussions

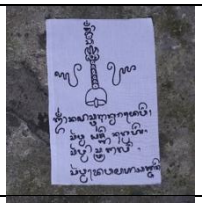
Results

Dataset

In this study, the results of model training and evaluation are presented based on predefined experimental schemes. The dataset consists of 90 image samples for each class, divided into training and testing datasets using an 80:20 ratio, resulting in 72 images per class for training and 18 images per class for testing. Data distribution was balanced across all classes to ensure fair representation and prevent bias during evaluation. All images underwent a pre-processing stage, including resizing to 224 × 224 pixels to match the input requirements of the MobileNetV2 architecture and normalization to standardize pixel values and improve training stability. Data augmentation was then applied to the training dataset, increasing the total number of training images to 432 images per class. This process enhances dataset diversity and improves the model's ability to generalize across various visual conditions. Details regarding the data distribution after the acquisition and augmentation process are presented in Table 2.

Table 2. Dataset distribution per class

Class	Image Sample	Data Train (80%)	Data Test (20%)	Total Data Train After Augmentation
angkul_angkul		72	18	432
bale_daje		72	18	432

bale_dangin		72	18	432
bale_dauh		72	18	432
bale_delod		72	18	432
kemulan		72	18	432
padmasana		72	18	432
paon		72	18	432
tugu_karang		72	18	432
	Total		**162**	**3.888**

Training

The training and validation results for this optimization scenario are comprehensively detailed in Table 3. The data reveals a high level of stability across the 5 fold cross validation process, characterized by a narrow margin of variance in performance metrics. The model achieved a robust average Test Accuracy of 97.11% and a remarkably low average Test Loss of 0.0811, while maintaining a high average Training Accuracy of 99.63%. A deeper analysis of the individual iterations shows that Fold 3 yielded the most optimal results, achieving a peak Test Accuracy of 97.86% and the minimum Test Loss of 0.0656. Furthermore, the consistency between the Training Accuracy (99.63%) and Test Accuracy (97.11%) suggests that the model possesses strong generalization capabilities. The marginal discrepancy of approximately 2.52% between the two metrics confirms that the integration of a 0.6 Dropout ratio and the Early Stopping mechanism effectively regularized the network.

Table 3. Recapitulation of loss value and accuracy of each fold in scenario 1

Fold	Train Loss	Train Acc	Test Lost	Test Acc
1	0.0217	0.9971	0.0948	0.9653
2	0.0256	0.9964	0.0909	0.9697
3	0.0247	0.9958	0.0656	0.9786
4	0.0236	0.9960	0.0772	0.9697
5	0.0256	0.9960	0.0771	0.9724
AVG	**0.02424**	**0.99626**	**0.08112**	**0.97114**

The visualization of the training progress for the Scenario 1 is illustrated in Figure 4, which displays the average accuracy and loss curves across 5 fold cross validation. The accuracy plot (left) shows a steady upward trend for both training and validation sets, reaching a state of convergence beyond the 60 epoch. Notably, the validation accuracy remains closely aligned with the training accuracy, exhibiting minimal fluctuations, which indicates stable learning behavior. Complementing the accuracy trend, the loss plot (right) demonstrates a consistent decay in both average training and validation loss. The curves maintain a parallel trajectory without significant divergence, further substantiating the effectiveness of the 0.6 Dropout ratio in preventing overfitting. As shown in the graph, the training process concluded before reaching the maximum 100 epochs due to the Early Stopping mechanism, which triggered once the validation loss ceased to improve for 10 consecutive epochs. This ensures that the model weights used in Table 3 represent the most optimal state of convergence, balancing high recognition accuracy with robust generalization.

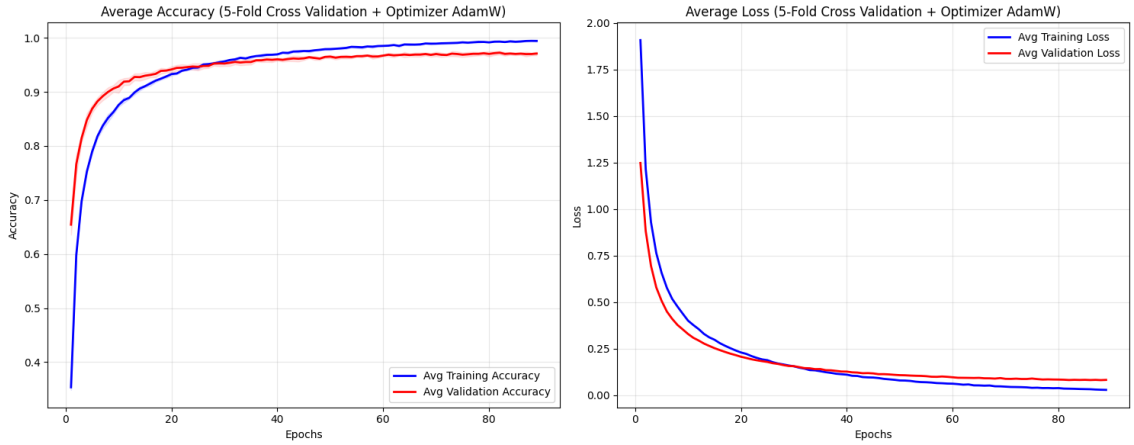


Figure 4. Average accuracy and loss curves scenario 1

The training and validation results for the Scenario 1 are comprehensively detailed in Table 4. The data reveals a high level of stability across the 5 fold cross validation process, characterized by a narrow margin of variance in performance metrics. The model achieved a robust average Test Accuracy of 97.10% and a remarkably low average Test Loss of 0.0802, while maintaining a high average Training Accuracy of 99.53%. A deeper analysis of the individual iterations shows that Fold 3 yielded the most optimal results, achieving a peak Test Accuracy of 97.68% and the minimum Test Loss of 0.0629. Furthermore, the consistency between the Training Accuracy (99.53%) and Test Accuracy (97.10%) suggests that the model possesses strong generalization capabilities. The marginal discrepancy of approximately 2.43% between the two metrics confirms that the integration of a 0.6 Dropout ratio and the Early Stopping mechanism effectively regularized the network. This balance prevented the model from memorizing the training noise, ensuring instead that it learned the fundamental structural patterns of the symbols.

Table 4. Recapitulation of loss value and accuracy of each fold in scenario 2

Fold	Train Loss	Train Acc	Test Lost	Test Acc
1	0.0321	0.9942	0.0888	0.9662
2	0.0241	0.9951	0.0937	0.9679
3	0.0234	0.9951	0.0629	0.9768
4	0.0221	0.9962	0.0742	0.9724
5	0.0257	0.9960	0.0813	0.9715
AVG	**0.02548**	**0.99532**	**0.08018**	**0.97096**

Figure 5 illustrates the training and validation progress for the Scenario 2 through average accuracy and loss plots. The accuracy curve (left) demonstrates a rapid learning phase in the early epochs, followed by a steady convergence that stabilizes after approximately the 50 epoch. The validation accuracy closely tracks the training accuracy, maintaining a consistent trend with minimal variance, which signifies the model's high stability during the cross validation process. Simultaneously, the loss curve (right) shows a significant and smooth reduction in both average training and validation loss. The absence of a widening gap between these two curves indicates that the model generalizes effectively to the validation sets. Similar to the previous scenario, the training process was automatically terminated at approximately epoch 85 by the Early Stopping mechanism, preventing unnecessary iterations once the validation loss reached its optimum.

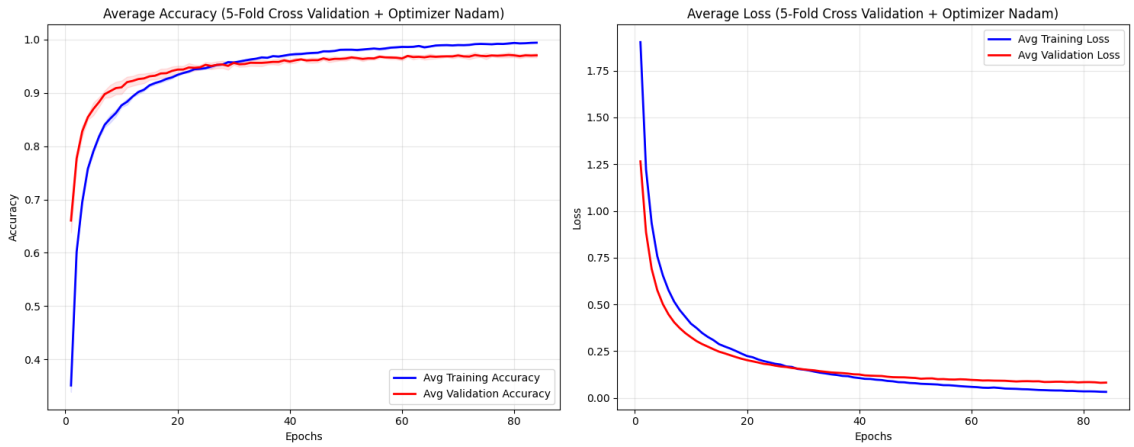


Figure 5. Average accuracy and loss curves scenario 2

The performance metrics for Scenario 3 are presented in Table 5. The data indicates a distinct learning behavior compared to previous scenarios, characterized by higher loss values and lower accuracy across all folds. The model achieved an average Training Accuracy of 59.81% and a Test Accuracy of 71.40%, with an average Test Loss of 1.09198. Despite the lower overall performance, the results remain consistent across the 5 fold cross validation process, showing a stable but slower convergence rate. Interestingly, the Test Accuracy (71.40%) consistently outperformed the Training Accuracy (59.81%) in this scenario. This phenomenon suggests that while the implementation of Scenario 3 struggled to reach a deep minimum during the training phase within the allotted epochs, the model maintained a fair level of generalization on the validation subsets. Among the five iterations, Fold 3 again recorded the highest Test Accuracy at 72.75%, while Fold 5 achieved the lowest Training Loss of 1.2217.

Table 5. Recapitulation of loss value and accuracy of each fold in scenario 3

Fold	Train Loss	Train Acc	Test Lost	Test Acc
1	1.2380	0.5917	1.0731	0.7251
2	1.2516	0.5996	1.1257	0.6963
3	1.2397	0.5998	1.0962	0.7275
4	1.2407	0.5958	1.0845	0.6990
5	1.2217	0.6034	1.0804	0.7222
AVG	**1.23834**	**0.59806**	**1.09198**	**0.71402**

Figure 6 illustrates the training and validation progress for Scenario 3 through average accuracy and loss curves. Unlike the previous scenarios, the accuracy plot (left) shows a significantly slower convergence rate, with both curves continuing to rise steadily even as they approach the 100 epoch. The loss plot (right) complements this observation by showing a gradual and continuous decline in both average training and validation loss without reaching a definitive plateau. The absence of a triggered Early Stopping mechanism as the training proceeded for the full 100 epochs indicates that Scenario 3 required more iterations to reach an optimal state compared to Scenario 1 and Scenario 2.

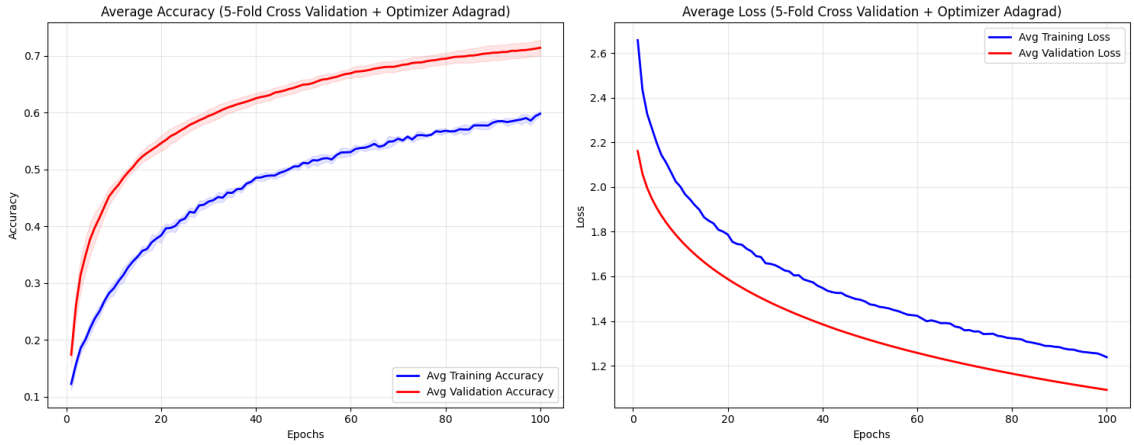


Figure 6. Average accuracy and loss curves scenario 3

Figure 7 presents a comparative analysis of the average validation loss and validation accuracy across the three optimization scenarios: Scenario 1 (AdamW), Scenario 2 (Nadam), and Scenario 3 (Adagrad). The left plot clearly demonstrates that both AdamW (blue) and Nadam (green) achieved a significantly faster and deeper convergence, with loss values dropping sharply below 0.2 within the first 40 epochs. In contrast, the Adagrad optimizer (orange) exhibited a much slower decay, maintaining a substantially higher loss throughout the 100 epoch duration, which suggests a less efficient exploration of the loss landscape for this specific dataset. The accuracy comparison in the right plot further reinforces these findings. AdamW and Nadam show nearly identical performance trajectories, both stabilizing at high accuracy levels above 97%. This indicates that both optimizers are highly effective in refining the MobileNetV2 weights for *Rerajahan Ulap-Ulap* imagery. Conversely, Adagrad reached a peak validation accuracy of only 71.40%, significantly underperforming compared to the other two candidates.

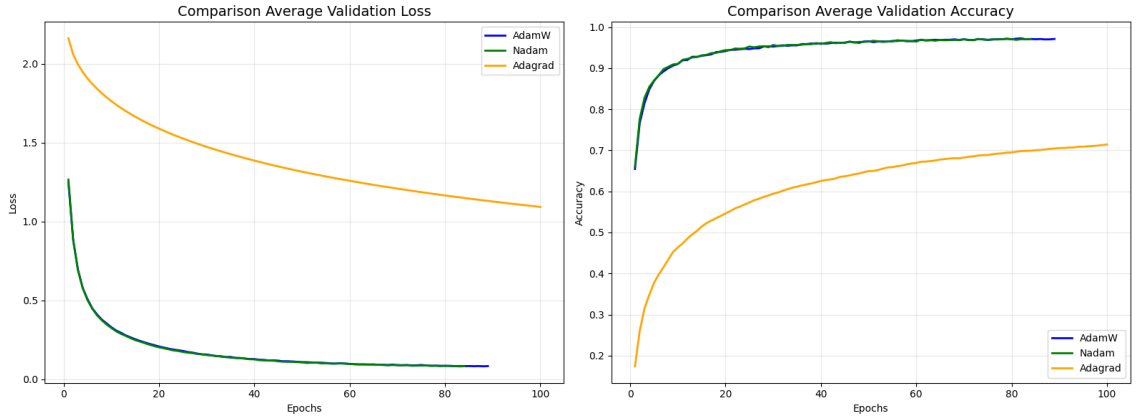


Figure 7. Comparison average accuracy and loss curves

Evaluation

The evaluation phase was conducted to measure the extent to which the trained model could recognize new, unseen data using an independent test dataset. This testing utilizes standard classification metrics to provide a comprehensive overview of the model architecture's accuracy and stability.

Based on comprehensive testing on that test dataset, the MobileNetV2 model using the Scenario 1 demonstrates impressive levels of accuracy and stability. The achieved average accuracy of 98.64% proves that this architecture is capable of recognizing the visual features of *Rerajahan Ulap-Ulap* motifs with a high degree of precision. The closely aligned values for Precision (98.75%), Recall (98.64%), and F1-Score (98.64%) indicate that the model possesses a well-balanced capability in minimizing both false positives and false negatives across all class categories.

A deeper analysis of each testing iteration reveals that Fold 2 successfully achieved the best performance compared to other folds, with an accuracy of 98.77% and the highest Precision value of 98.89%. The robustness of the model in this scenario is further strengthened by the very low Standard Deviation (STD DEV) value of 0.0025, providing statistical validation that the model is highly robust and insensitive to differences in data partitioning during the cross validation process.

Table 6. Recapitulation of evaluation model in scenario 1

Fold	Accuracy	Precision	Recall	F1-Score
1	0.9815	0.9818	0.9815	0.9815
2	0.9877	0.9889	0.9877	0.9876
3	0.9877	0.9889	0.9877	0.9876
4	0.9877	0.9889	0.9877	0.9876
5	0.9877	0.9889	0.9877	0.9876
AVG	**0.9864**	**0.9875**	**0.9864**	**0.9864**
STD DEV	0.0025	0.0028	0.0025	0.0025

A more detailed analysis of the model's prediction distribution can be seen in Figure 8, which presents the Normalized Confusion Matrix for the Scenario 1. This matrix reflects the accumulated results of the 5 fold cross validation, where the values on the main diagonal represent the percentage of successful predictions for each class relative to its actual label. Based on the data in Figure 8, the model demonstrates perfect recognition capability (100.00%) across most classes, including: *angkul-angkul*, *bale_daje*, *bale_dauh*, *kemulan*, *padmasana*, *paon*, and *tugu_karang*. This indicates that the visual features of these categories are highly distinctive, allowing them to be distinguished with exceptional accuracy by the MobileNetV2 architecture.

Nevertheless, as shown in Figure 8, a slight classification ambiguity is observed within the *bale_dangin* and *bale_delod* categories. The *bale_dangin* class achieved an accuracy rate of

88.89%, with 11.11% of the data being misclassified as *bale_delod*. Conversely, the *bale_delod* class reached 98.89% accuracy, with 1.11% of the data predicted as *bale_dangin*. This reciprocal misclassification pattern suggests a high degree of similarity in visual features or building morphology between the *bale_dangin* and *bale_delod* motifs.

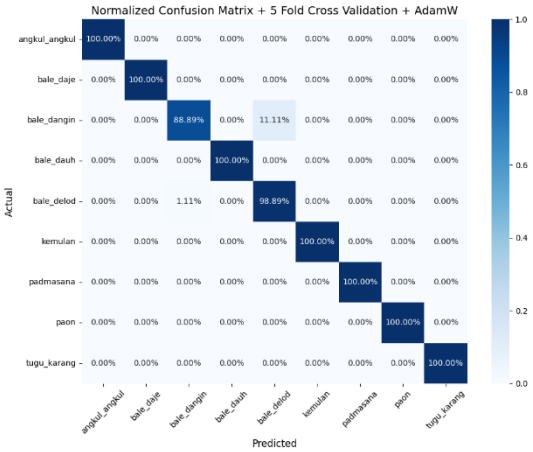


Figure 8. Confusion matrix scenario 1

Based on the comprehensive testing on the test dataset presented in Table 6, Scenario 2 demonstrates a high level of accuracy and stability. The achieved average accuracy of 98.52% proves that this architecture is highly capable of recognizing the visual features of *Rerajahan Ulap-Ulap* motifs. The highly consistent values for Precision (98.58%), Recall (98.52%), and F1-Score (98.52%) indicate that the model possesses a balanced capability in minimizing classification errors across all class categories. A deeper analysis of each testing iteration in Table 6 reveals that Fold 1 successfully achieved the best performance compared to other folds, with an accuracy of 98.77% and a Precision value of 98.89%. The robustness of the model in this scenario is further strengthened by the low Standard Deviation (STD DEV) value of 0.0030, providing statistical validation that the model is robust and insensitive to differences in data partitioning during the cross validation process.

Table 7. Recapitulation of evaluation model in scenario 2

Fold	Accuracy	Precision	Recall	F1-Score
1	0.9877	0.9889	0.9877	0.9876
2	0.9815	0.9818	0.9815	0.9815
3	0.9815	0.9818	0.9815	0.9815
4	0.9877	0.9877	0.9877	0.9877
5	0.9877	0.9889	0.9877	0.9876
AVG	**0.9852**	**0.9858**	**0.9852**	**0.9852**
STD DEV	0.0030	0.0033	0.0030	0.0030

A more detailed analysis of the model's prediction distribution in Scenario 2 can be seen in Figure 9. Based on the data, the model achieved a perfect recognition rate (100.00%) for the majority of classes, including: *angkul-angkul*, *bale_daje*, *bale_dauh*, *kemulan*, *padmasana*, *paon*, and *tugu_karang*. This demonstrates that the optimizer in Scenario 2 is highly effective in helping the MobileNetV2 architecture extract distinctive visual features for these categories.

However, as shown in Figure 9, slight ambiguities remain in the classification of *bale_dangin* and *bale_delod* motifs. The *bale_dangin* class achieved an accuracy of 90.00%, where 10.00% of the samples were incorrectly predicted as *bale_delod*. Conversely, the *bale_delod* class recorded an accuracy of 96.67%, with 3.33% of the data misclassified as *bale_dangin*. This pattern shows a shift in misclassification rates where *bale_dangin* accuracy slightly improved, while *bale_delod* accuracy experienced a slight decrease compared to the

previous scenario. This indicates that Scenario 2 optimization contributes differently to distinguishing building morphological features with high visual similarity.

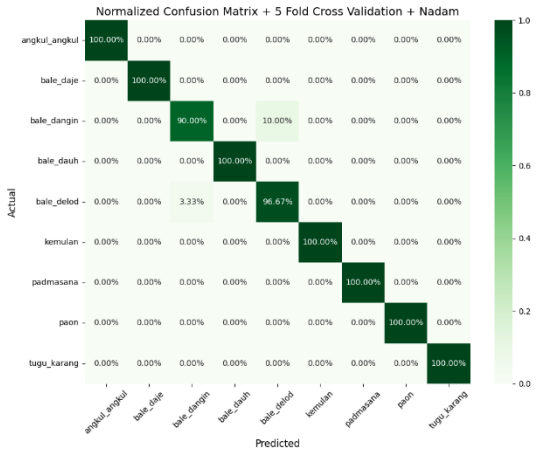


Figure 9. Confusion matrix scenario 2

Based on the comprehensive testing on the test dataset presented in Table 8, the MobileNetV2 model in Scenario 3 shows a significant decrease in performance compared to the previous scenarios. The achieved average accuracy of 71.23% indicates challenges in optimally extracting the visual features of *Rerajahan Ulap-Ulap* motifs within this scenario. The average values for Precision (75.09%), Recall (71.23%), and F1-Score (67.83%) suggest a higher misclassification rate and an imbalance in recognizing the various class categories tested. A deeper analysis of each testing iteration in Table 8 reveals that Fold 2 recorded the best performance with an accuracy of 75.93%. Unlike the previous scenarios which maintained high stability, Scenario 3 exhibits a much larger Standard Deviation (STD DEV), specifically 0.0323 for accuracy and 0.0587 for Precision. This high variance provides statistical validation that the model in this scenario is less robust, with its performance being significantly affected by differences in data partitioning during the cross validation process.

Table 8. Recapitulation of evaluation model in scenario 3

Fold	Accuracy	Precision	Recall	F1-Score
1	0.6728	0.7358	0.6728	0.6363
2	0.7593	0.7974	0.7593	0.7436
3	0.6914	0.6458	0.6914	0.6382
4	0.6975	0.7647	0.6975	0.6667
5	0.7407	0.8110	0.7407	0.7069
AVG	**0.7123**	**0.7509**	**0.7123**	**0.6783**
STD DEV	0.0323	0.0587	0.0323	0.0414

A detailed analysis of the model's prediction distribution in Scenario 3 can be seen in Figure 10, illustrating a significant performance degradation and widespread misclassification across nearly all categories of Balinese *Rerajahan* and *Ulap-Ulap* motifs. Based on the data, the model only maintained perfect performance (100.00%) for two categories, specifically the kemulan and padmasana motifs. Conversely, other categories suffered from massive predictive overlaps, with the *angkul_angkul* motif emerging as the lowest performing category only 21.11% of the data was correctly predicted, while the remaining 64.44% was incorrectly classified as the *padmasana* motif.

Extremely high ambiguity is also clearly observed in the *bale_dangin* motif, which only reached an accuracy of 34.44%, with significant misclassification spreads toward the *bale_delod* (27.78%) and *tugu_karang* (22.22%) motifs. The visualization in Figure 10 confirms that the implementation of Scenario 3 failed to help the model extract distinctive enough visual features

to differentiate the unique characteristics between motifs. The high number of values scattered outside the main diagonal indicates that the model failed to achieve optimal convergence, resulting in an average accuracy that is considerably lower and less stable than that of the previous testing scenarios.

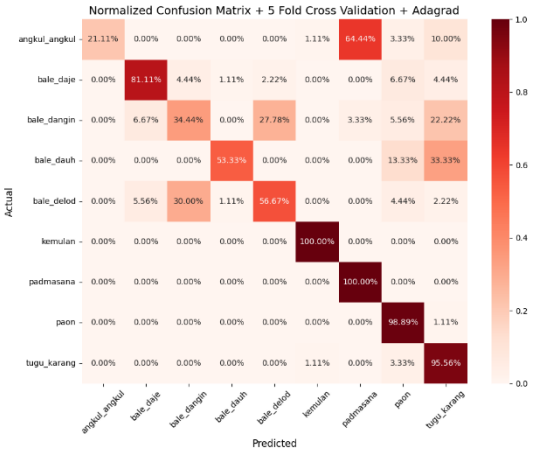


Figure 10. Confusion matrix scenario 3

Based on the comprehensive testing results presented in Table 8, it can be concluded that Scenario 1 delivers the best performance in classifying Balinese *Rerajahan Ulap-Ulap* motifs. This scenario achieved the highest average accuracy of 98.64%, followed by Precision (98.75%), Recall (98.64%), and F1-Score (98.64%). These results demonstrate that the configuration in Scenario 1 possesses the most superior and stable ability to recognize the unique visual characteristics of the dataset. Scenario 2 follows with nearly identical performance, recording an average accuracy of 98.52%. Although there is a slight decrease compared to the first scenario, the metric values remain at a very high level, proving the model's robustness in classification. On the other hand, Scenario 3 shows a significant performance decline, with an average accuracy of only 71.23% and an F1-Score of 67.83%. This indicates that the approach in Scenario 3 is less effective in capturing the complex features of *Rerajahan Ulap-Ulap* motifs compared to the other two scenarios.

Table 9. Final result

Scenario	Accuracy (avg)	Precision (avg)	Recall (avg)	F1-Score (avg)
1	98.64%	98.75%	98.64%	98.64%
2	98.52%	98.58%	98.52%	98.52%
3	71.23%	75.09%	71.23%	67.83%

Discussions

The experimental results demonstrate that the MobileNetV2 architecture, combined with a transfer learning strategy, is highly effective in recognizing the unique visual characteristics of Balinese *Rerajahan Ulap-Ulap* motifs. Based on the comparison of the three tested schemes, Scenario 1 (AdamW) delivered the most superior performance, achieving an average accuracy of 98.64%, followed closely by Scenario 2 (Nadam) with an accuracy of 98.52%. The superiority of AdamW in this study aligns with findings from previous traditional fabric motif classification studies, where the algorithm proved more adaptive in handling complex visual features. The performance stability in Scenario 1 was further reinforced by a very low standard deviation (0.0025), providing statistical validation that the model is robust and consistent despite being tested with different data partitions through the 5 fold cross validation method.

On the other hand, Scenario 3 (Adagrad) showed a significant performance decline, with an average accuracy of only 71.23%. Analysis of the training curves indicates that the approach in this scenario experienced a much slower convergence rate, where the loss values remained

high and failed to reach a definitive plateau even by the 100 epoch. This suggests that the visual features in the primary *Rerajahan Ulap-Ulap* dataset require more dynamic weight adjustments during the training process an aspect that was handled less optimally by Adagrad compared to AdamW or Nadam. The lower performance in Scenario 3 was also accompanied by a higher variance (standard deviation of 0.0323), indicating that the model was less stable in generalizing motif patterns across various data subsets.

Although the models in Scenarios 1 and 2 achieved very high accuracy, analysis through the normalized confusion matrix revealed specific challenges regarding the *bale_dangin* and *bale_delod* motifs. Reciprocal misclassification occurred, where in Scenario 1, 11.11% of *bale_dangin* data was predicted as *bale_delod*, reflecting a very strong visual morphological similarity between the two categories. The characteristics of *rerajahan ulap-ulap*, which involve complex lines and sacred *modre* scripts, indeed present a higher level of feature extraction difficulty compared to decorative objects with repetitive texture patterns. However, the integration of data augmentation techniques, the use of a 0.6 Dropout ratio, and the Early Stopping mechanism proved crucial in preventing overfitting and ensuring the model remained capable of generalizing well on the primary data used. The success in developing this model provides an innovative solution for the digitalization and preservation of Balinese cultural heritage through a computer vision technology approach.

Conclusion

Based on the experiments and analysis conducted, it can be concluded that the MobileNetV2 architecture is highly effective for classifying Balinese *Rerajahan Ulap-Ulap* motifs with a very high degree of accuracy. The test results demonstrate that the choice of optimization scenario is crucial in determining model performance, where Scenario 1 (AdamW) emerged as the best model with an average accuracy of 98.64%, Precision of 98.75%, Recall of 98.64%, and an F1-Score of 98.64%. This performance outperforms Scenario 2 (Nadam), which achieved 98.52% accuracy, and Scenario 3 (Adagrad), which only reached 71.23%. The application of data augmentation techniques and the Early Stopping mechanism successfully addressed the limitations of the primary dataset and prevented overfitting, resulting in a model with strong generalization capabilities.

The successful development of this model provides a significant contribution to the efforts of digitalizing cultural heritage and preserving traditional Balinese knowledge through a computer vision technology approach. However, given that the primary dataset used in this study is limited in size, it is highly recommended that further testing be conducted using external datasets outside the currently available set to ensure the model's reliability and robustness in real-world scenarios. Testing with independent data that includes a wider range of environmental conditions, lighting variations, and drawing styles will validate whether the model can maintain high accuracy in classifying objects more universally and reliably.

References

[1] I. W. Dharmawan and I. P. A. Suryawan, *Makarya Ulap-Ulap Palinggih lan Wewangunan* . Yogyakarta: Pustaka Pranala, 2020.

[2] I. K. D. Pendit, "Studi Pembelajaran Religi Hindu Dalam Gambar Rerajahan Ulap-Ulap," *Social Studies*, vol. 10, no. 2, pp. 1–15, Sep. 2023.

[3] I. G. W. Simrana, I. W. Mudra, I. G. Yudarta, and N. W. Ardini, "Makna Bentuk dan Aksara Rerajahan," *Jurnal Bali Membangun Bali*, vol. 4, no. 2, pp. 101–113, Aug. 2023, doi: 10.51172/jbmb.v4i2.273.

[4] M. Sahpira and Yohannes, "Transfer Learning dengan MobileNetV2 untuk Klasifikasi Motif Jumputan Palembang," *Jurnal Algoritme*, vol. 6, no. 1, pp. 1–10, Oct. 2025.

[5] I. W. Sudana, "EKSISTENSI RERAJAHAN SEBAGAI MANIFESTASI MANUNGGALNYA SENI DENGAN RELIGI," *Imaji*, vol. 7, no. 2, pp. 141–158, Nov. 2015, doi: 10.21831/imaji.v7i2.6631.

[6] N. M. R. A. Permatasari, M. W. A. Kesiman, and I. M. G. Sunarya, "Klasifikasi Jenis Samphyan Banten Upakara Adat Bali Dengan Arsitektur VGG-16 Dan InceptionResnet-V2," *SINTECH (Science and Information Technology) Journal*, vol. 8, no. 3, pp. 220–230, Dec. 2025, doi: 10.31598/sintechjournal.v8i3.2009.

[7] 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*, vol. 7, no. 2, Dec. 2025.

[8] R. M. M. Komang, J. E. D. Luh, Y. E. A. Kadek, A. S. Ketut, and V. Pariwate, "ANALISIS HYPERPARAMETER PADA KLASIFIKASI JENIS TANAMAN MENGGUNAKAN ALGORITMA RESNET50 DAN MOBILENETV2," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 6, pp. 9921–9928, Nov. 2025, doi: 10.36040/jati.v9i6.15832.

[9] M. W. A. Kesiman, I. M. G. Sunarya, and I. G. L. T. Sumantara, "Comparative Analysis of CNN Methods for Periapical Radiograph Classification," *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, vol. 13, no. 2, pp. 204–214, Jul. 2024, doi: 10.23887/janapati.v13i2.71664.

[10] N. P. D. A. S. Dewi, M. W. A. Kesiman, I. M. G. Sunarya, G. A. A. D. Indradewi, and I. G. Andika, "Klasifikasi Jenis Daun Tumbuhan Herbal Berdasarkan Lontar Usada Taru Pramana Menggunakan CNN," *Techno.Com*, vol. 23, no. 1, pp. 271–283, Feb. 2024, doi: 10.62411/tc.v23i1.9510.

[11] I. K. P. G. Hartawan and I. M. G. Sunarya, "Tinjauan Sistematis Literatur Tentang Sistem Deteksi Penyakit Berbasis Deep Learning," *Digital Transformation Technology (Digitech)*, vol. 6, no. 1, pp. 112–123, Mar. 2026.

[12] I. Mudzakir and T. Arifin, "Klasifikasi Penggunaan Masker dengan Convolutional Neural Network Menggunakan Arsitektur MobileNetv2," *EXPERT: Jurnal Manajemen Sistem Informasi dan Teknologi*, vol. 12, no. 1, p. 76, Jun. 2022, doi: 10.36448/expert.v12i1.2466.

[13] O. Saputra, D. I. Mulyana, and M. B. Yel, "Implementasi Algoritma Convolutional Neural Network (CNN) Untuk Klasifikasi Senjata Tradisional Di Jawa Tengah Dengan Metode Transfer Learning," *Jurnal SISKOM-KB (Sistem Komputer dan Kecerdasan Buatan)*, vol. 5, no. 2, pp. 45–52, Mar. 2022, doi: 10.47970/siskom-kb.v5i2.282.

[14] Y. N. FUADAH, I. D. UBAIDULLAH, N. IBRAHIM, F. F. TALININGSING, N. K. SY, and M. A. PRAMUDITHO, "Optimasi Convolutional Neural Network dan K-Fold Cross Validation pada Sistem Klasifikasi Glaukoma," *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, vol. 10, no. 3, p. 728, Jul. 2022, doi: 10.26760/elkomika.v10i3.728.

[15] M. F. A. Maulana, N. M. Anggadimas, and D. A. Sani, "Klasifikasi Citra Penyakit Daun Padi Dengan Metode CNN Menggunakan Arsitektur ResNet50V2," *CESS (Journal of Computer Engineering, System and Science)*, vol. 10, no. 2, pp. 517–529, Jul. 2025, doi: 10.24114/cess.v10i2.66960.

[16] G. W. Purnama, A. A. J. Permana, and N. K. Kertiasih, "KLASIFIKASI CITRA MEDIS MAMOGRAFI BERBASIS MULTI-VIEW CONVOLUTIONAL NEURAL NETWORKS," *Jurnal Pendidikan Teknologi dan Kejuruan*, vol. 22, no. 2, pp. 162–171, 2025.

[17] I. K. Y. Prayoga, I. M. G. Sunarya, and P. H. Suputra, "Klasifikasi Penyakit Demam Berdarah Menggunakan Algoritma Stacking Ensemble Learning," *SINTECH Journal*, vol. 8, no. 2, pp. 104–116, Aug. 2025.

[18] I. P. W. Prasetia and I Made Gede Sunarya, "Image Classification of Balinese Seasoning Base Genep Based on Deep Learning," *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, vol. 13, no. 1, Mar. 2024, doi: 10.23887/janapati.v13i1.67967.

[19] T. Bariyah, M. Arif Rasyidi, and Ngatini, "Convolutional Neural Network Untuk Metode Klasifikasi Multi-Label Pada Motif Batik Convolutional Neural Network for Multi-Label Batik Pattern Classification Method," 2021.

[20] P. P. Vedanty, M. W. A. Kesiman, and I. M. G. Sunarya, "PENGARUH DATA AUGMENTASI PADA IDENTIFIKASI PENYAKIT DAUN TANAMAN OBAT MENGGUNAKAN K-NEAREST NEIGHBOR," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 2, pp. 2094–2100, Mar. 2025, doi: 10.36040/jati.v9i2.12961.