Journal of Sustainable Software Engineering and Information Systems
**Volume 2, Issue 1, 01-15.**
e_ISSN: 3123-3007
https://e-journal.gomit.id/jsseis
DOI: 10.58723/jsseis.v2i1.158

# Improving the Classification Accuracy of *Parang* Batik Motifs with High Visual Similarity Through the Integration of GLCM and MobileNetV2

**Haryanto***
Asia e University, Subang Jaya Kuala Lumpur
MALAYSIA

**Husna Sarirah Husin**
Taylor University, Kuala Lumpur
MALAYSIA

* **Corresponding author:** Haryanto, Asia e University, MALAYSIA. ✉ c70105240003@aeu.edu.my

**Article Info**

***Article history:***

Received: February 14, 2026
Revised: March 28, 2026
Accepted: April 01, 2026

***Keywords:***

Convolutional Neural Network
Feature Fusion
GLCM
MobileNetV2
*Parang* Batik

**Abstract**

**Background:** Despite its high aesthetic value, automatic classification of Parang Surakarta batik is difficult due to the extreme textural similarities between sub-motifs. Standard CNN architectures, including MobileNetV2often fail to detect the subtle textural details that distinguish each variation of the motif.
**Aims:** This study develops a hybrid classification model that combines manual and automated spatial texture features to improve identification accuracy on motifs with high visual similarity.
**Methods:** Using a dataset that has been expanded to 120 original images (40 per class) which is then augmented to a total of 1,200 images to ensure stronger model generalization. This methodology hybrid GLCM-MobileNetV2architecture through transfer learning techniques. Features from both methods are combined through feature fusion before being classified using a Dense layer.
**Result:** The hybrid GLCM-MobileNetV2model achieved an accuracy of 99%. This performance outperformed the pure MobileNetV2 method (66.67%) and GLCM-SVM (85%), demonstrating that texture features provide significant discriminatory power against similar repetitive patterns.
**Conclusion:** The integration of GLCM and MobileNetV2 is highly effective for classifying visually similar batik motifs, achieving a superior accuracy of 99% compared to the pure MobileNetV2 (66.67%). This hybrid approach provides a robust and efficient solution for digital cultural preservation on mobile devices.

**To cite this article:** Haryanto & Husin H. S. (2026). Improving the Classification Accuracy of *Parang* Batik Motifs with High Visual Similarity Through the Integration of GLCM and MobileNetV2. *Journal of Sustainable Software Engineering and Information Systems, 2*(1), 01-15. https://doi.org/10.58723/jsseis.v2i1.158



## INTRODUCTION

Indonesia boasts a rich cultural heritage recognized by UNESCO, one of which is the art of batik. Each batik motif, such as the distinctive Surakarta batik, possesses deep aesthetic value and symbolism that reflects local cultural identity (Arif & Purnomo, 2021). However, the similarity in texture patterns between motifs often makes visual identification difficult for the human eye, necessitating an accurate automated classification system(Yusriadi et al., 2023).

Several previous studies have applied machine learning and deep learning algorithms to batik classification. Traditional methods such as Support Vector Machine (SVM) combined with Gray Level Co-occurrence Matrix (GLCM) and Local Binary Pattern (LBP) feature extraction have demonstrated quite good performance on certain datasets (Andono & Rachmawanto, 2021; Wiryadinata et al., 2019), Conducting research on Papuan batik classification by comparing the performance of the M-SVM kernel with GLCM feature extraction, where model performance is greatly influenced by the data split ratio (Gustian et al., 2019; Naa. 2024). Other research: the system successfully classified 6 Bojonegoro Batik motifs using second-order GLCM feature extraction, with the highest accuracy reaching 85% (Ashari & Kusuma, 2024; Prihatin et al., 2018; Steelyana, 2012).

Meanwhile, The use of Convolutional Neural Networks (CNN) has proven effective in handling the visual complexity of batik images with coarse and contrasting patterns with significant accuracy, especially when supported by data augmentation and transfer learning (Auliaddina & Arifin, 2024; Dani & Handayani, 2024). Some studies such as VGG-16 and SVM (linear kernel) are effective for Batik Besurek classification, provided the dataset has contrasting motif patterns (Handayani et al., 2023; Winarno et al., 2022). Requires augmentation and Hyperparameter Tuning to drastically improve the performance of the CNN model, from 28.15% accuracy to 66.67%. if directly less than optimal (Auliaddina & Arifin, 2024). CNN optimization on West Javanese Batik classification with a learning rate of 0.001 and 20 epochs successfully achieved high and balanced performance of 90% (Accuracy, Precision, and Recall) (Agastya & Setyanto, 2018; Tember et al., 2023), in this case those using standard CNN architectures such as MobileNetV2 often fail to detect fine texture details that differentiate these motif variations.

However, previous research has largely focused on motif recognition based on general region of origin or datasets with contrasting motif variations. A research gap remains in classifying motifs with highly visual similarities, such as the *Parang* batik subcategories (*Slobog, Tuding, and Klitik*). Furthermore, optimizing the combination of manual texture feature extraction and deep learning-based automated features is still rarely done specifically for this object.

This study aims to develop a hybrid model by integrating GLCM and MobileNetV2 architecture. The correlation lies in **synergy**. GLCM compensates for the CNN's potential "texture blindness," while MobileNetV2 provides the efficient structural intelligence that GLCM lacks. Together, they solve the problem of identifying complex, texture-heavy images in resource-constrained environments. GLCM was chosen for its efficiency in extracting texture features, while MobileNetV2 is used to learn deeper visual patterns through transfer learning. The integration of these two methods is expected to improve the classification accuracy of *Parang* batik motifs that have similar textures, while also making a real contribution to cultural preservation efforts through efficient and accurate technology.

## METHOD

In the methodology there are detailed stages.

**Table.** Detailed stages.

| Stage | Name | Description | Role in Solution |
|---|---|---|---|
| 1 | Start | Defining the research question (e.g., classifying Batik motifs). | Sets the foundation. |
| 2 | Data Collection & Preprocessing | Gathering images, resizing, and preparing them (often converting to grayscale for GLCM, keeping RGB for CNN). | Standardizes data quality. |
| 3 | Data Split | Dividing data into Training, Validation, and Testing sets. | Crucial for model generalization and solving the **limited data** problem. |
| 4a | GLCM Feature Extraction | Calculating second-order statistics (contrast, correlation, homogeneity, energy, entropy) at various angles. | Captures **detailed texture signatures** (solving the Feature Gap). |
| 4b | MobileNetV2 Deep Learning | Using MobileNetV2 to automatically extract high-level hierarchical features. often using Transfer Learning. | Captures **global shape and structural intelligence** efficiently (solving Efficiency/Accuracy trade-off). |
| 5 | FEATURE FUSION | **(The Core Synergy)** Merging the GLCM statistical vector and the MobileNetV2 embedding vector into one unified | Combines local texture (GLCM) with global hierarchy (MobileNetV2). |

| | | | |
|---|---|---|---|
| | | feature set. | |
| 6 | Classification Layer | Feeding the fused features into a classifier (e.g., Multiclass SVM or a final Softmax layer). | Makes the final category prediction. |
| 7 | Performance Evaluation | Analyzing the results using metrics like Confusion Matrices, Accuracy, Precision, and Recall. | Validates if the combined method solved the research problem. |

1. Materials and Dataset

The dataset used in this study consists of 1,200 images of *Parang* batik motifs divided into three specific categories: *Slobog, Tuding,* and *Klitik*. Data added to the original data collection from 3 per class to 40 per class. Data were obtained through two main sources:

1. Primary Data Collection: Direct photography using a mobile device camera at the Surakarta Palace Batik Museum, *Kampung* Batik Solo, and BTC Solo.
2. Secondary Data Collection: Downloading relevant datasets through the Kaggle platform to strengthen the sample size.

2. Image Preprocessing

The preprocessing stage is performed to improve data quality and model performance through the following systematic steps:

1. Data Splitting: To ensure a rigorous evaluation and prevent data leakage, the dataset was first split into training and testing sets at the parent image level with a 70:30 ratio. The 70:30 ratio was chosen to balance the needs of model learning and result validation. The 70% portion provides sufficient room for MobileNetV2 to perform in-depth feature learning and filter weight adjustments on various batik motifs. Meanwhile, the 30% portion is used as a strict testing standard to ensure that the obtained accuracy values are truly valid, objective, and have a high level of generalization, not simply the result of overfitting the training data.This ensures that the testing set (360 images) contains motifs from original sources that are entirely independent of the training set (840 images)(Muraina & Olaniyi, 2022).
2. Data Augmentation: Following the split, augmentation was applied only to the training set to address data limitations and prevent overfitting. The process employed specific parameters to maintain the structural integrity of Parang motifs, including a rotation range of 20°, a zoom range of 0.1, and brightness adjustments between 0.8 and 1.2. These transformations ensure the model encounters realistic visual variations without distorting the fundamental geometric characteristics of the batik patterns.
3. Normalization: Pixel intensity values were converted to a range of 0 to 1 by dividing each pixel value by 255. This step is crucial to accelerate training convergence and ensure numerical stability during the gradient descent process.

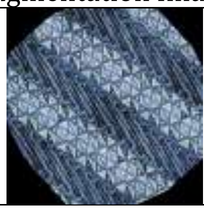
| Original Image | Augmentation Image |
|---|---|
|  |  |

**Figure 1.** Original Image and Augmentation image

3. Texture Feature Extraction (GLCM)

This study integrates the Gray Level Co-occurrence Matrix (GLCM) method to capture subtle texture characteristics often undetected by standard CNNs. The image is converted to grayscale before calculating its spatial matrix at a distance of $d$=1 pixels with four orientation angles ($0^0$, $45^0$, $90^0$, $135^0$)(Dani & Handayani, 2024). The four main features extracted are:

1. Contrast: Measures local intensity differences.

2. Homogeneity: Measures the proximity of the distribution of elements in the GLCM to the diagonal.
3. Energy: Measures the consistency of texture patterns.
4. Entropy: Measures the degree of randomness or complexity of the pattern.

Dataset: This experiment used a total of 1,200 images divided equally into three Parang motif classes, with 40 original images per class and 10x augmentation. The augmentation process significantly increased the dataset from the original 120 to 1,200 to ensure good model generalization across a wide variety of positions and lighting.

**Table 2.** Texture Feature Extraction Results at an Angle of 0°

| TEXTURE FEATURE EXTRACTION AT AN ANGLE OF 0° | | | | | | |
|---|---|---|---|---|---|---|
| **Classs** | **Dissimilarity** | **Correlation** | **Homogeneity** | **Contrast** | **ASM** | **Energy** |
| *Parangklitik* | 5.6214900662250935 | 0.97201594045371 | 0.3006839062377373 | 732.0527924943989 | 0.0021752217860813287 | 0.04663927300120928 |
| *Parangklitik* | 6.596368653421483 | 0.9704393814913451 | 0.20794426454058937 | 91.67422737306586 | 0.001266347723540346 | 0.03558577979390568 |
| *Parangklitik* | 6.423024282560593 | 0.9706974468246095 | 0.2774846495449558 | 100.65467991169783 | 0.0017440221310955975 | 0.041761491006615144 |
| *Parangklitik* | 7.258509933774628 | 0.9708246564481356 | 0.20034450480016394 | 113.06926048564821 | 0.0010365705207861034 | 0.03219581526823173 |
| *Parangtuding* | 14.641917502788752 | 0.9437372316775894 | 0.2834668283198671 | 746.6424303233838 | 0.020961672253242428 | 0.14478146377641865 |
| *Parangtuding* | 15.071995540693102 | 0.9556084558424295 | 0.24842465119228938 | 750.1430546266397 | 0.004358523270309946 | 0.06601911291671485 |
| *Parangtuding* | 16.177369007805357 | 0.9424099138631149 | 0.21397899453345792 | 754.5387736901661 | 0.0010313069453612253 | 0.032113968072494954 |
| *Parangtuding* | 15.393534002230464 | 0.926119207619413 | 0.20577020717911434 | 763.5525306577815 | 9.190153229705595E-4 | 0.030315265510474414 |
| *Parangslobog* | 40.19132692348608 | 0.6559078628215095 | 0.04269989944467176 | 3048.533013573953 | 5.089582043829475E-5 | 0.0071341306715180615 |
| *Parangslobog* | 40.19132692348608 | 0.6559078628215095 | 0.04269989944467176 | 3048.533013573953 | 5.089582043829475E-5 | 0.0071341306715180615 |
| *Parangslobog* | 40.19132692348608 | 0.6559078628215095 | 0.04269989944467176 | 3048.533013573953 | 5.089582043829475E-5 | 0.0071341306715180615 |
| *Parangslobog* | 40.19132692348608 | 0.6559078628215095 | 0.04269989944467176 | 3048.533013573953 | 5.089582043829475E-5 | 0.0071341306715180615 |

The input consists of 1,200 grayscale Batik motif images (64×64 pixels), representing three Parang classes (Parangsloboh, Parangtuding, Parangklitih). Each image undergoes feature extraction using GLCM (24 features). A total of 24 original features were extracted using GLCM (6 texture parameters × 4 directions: 0°, 45°, 90°, and 135°). In the table, we show at $0^0$ degrees, and the distance *d=1*

### 3.1 Model Architecture

The MobileNetV2 Architectureconsists of several main layers: a convolution layer, a pooling layer, and a fully connected layer. The convolution layer extracts features from the input image through a kernel or filter that traverses the entire image, producing a feature map. A pooling layer then reduces the dimensionality of this feature map, preserving essential information while reducing computational complexity. All features extracted in the previous layer are then connected into a single feature vector in the fully connected layer. This layer combines all the information from the feature maps and forwards it to the output layer, where each neuron in the output layer represents a class of *parang* batik motifs, such as *Parang slobog, tuding*, and *klitik*. The model generates probabilities for each class, and the class with the highest probability is selected as the prediction result (Abbood & Al-Assadi, 2022; Rasyidi & Bariyah, 2020).

To improve the model's ability to learn more complex patterns, the *ReLU* (Rectified Linear Unit) activation function is applied after the convolution layer. *ReLU* serves to introduce non-linearity to the network, allowing the model to learn more complex relationships between features extracted from the image. *ReLU* works by converting negative values to zero, while positive values remain unchanged. With the implementation of *ReLU*, the neural network becomes faster in the training process, increasing the model's efficiency and accuracy in predicting the classification of batik motifs *Parang* (Ide & Kurita, 2017).

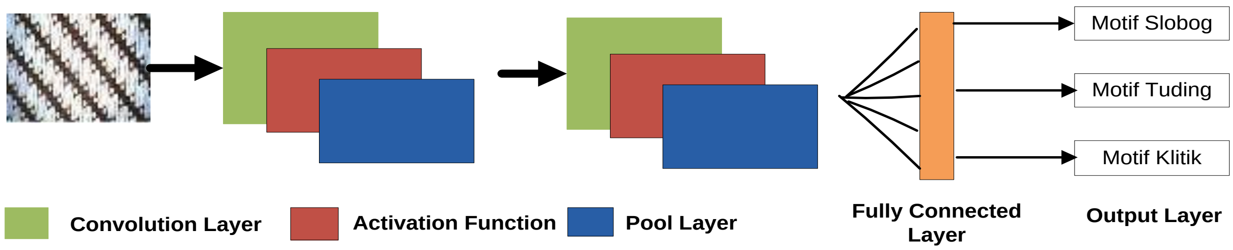



**Figure 2.** Illustration of the MobileNetV2 architecture used in this study.

The illustration in Figure 2 shows the process carried out by the MobileNetV2 in classifying batik motifs. The input image, a batik motif, is processed through a convolution layer, which detects basic features such as lines, edge patterns, or simple shapes. The ReLU activation function is applied to each convolution layer to speed up the training process and improve the network's ability to capture more complex patterns. The resulting feature maps then go through a pooling layer to reduce dimensionality while retaining important features. After passing through all these layers, the feature maps are combined into a vector in the fully connected layer and forwarded to the output layer, where the classification result, the type of batik motif, is determined.

3.2 Transfer Learning

Transfer learning is a machine learning technique that allows a model previously trained on a dataset to be applied to a new dataset with some modifications. This technique is particularly useful when collecting labeled data is difficult, such as in batik motif classification, because the model can gain knowledge from a larger, more general dataset. In the transfer learning process, a model trained on a large dataset, such as ImageNet, is used as a starting point, allowing the model to learn faster and achieve better accuracy even with a limited training dataset (Ardianto & Wibisono, 2023; Azizah et al., 2025).

This study uses the MobileNetV2 model as the basis for the CNN architecture. MobileNetV2 is a highly efficient model, employing techniques such as depthwiseseparble convolution to reduce the number of parameters without sacrificing accuracy (Meranggi et al., 2022). With transfer learning, a model pre-trained on a general dataset such as ImageNet is used to begin training on a batik motif dataset. This pre-trained model carries a basic knowledge of common patterns in images and only needs to be adapted to the specific dataset through a fine-tuning process. Fine-tuning in transfer learning is the next step where certain layers of the pre-trained model are updated to better adapt to new data, in this case the *parang* batik motif.

During the fine-tuning process, the initial layers used to extract basic features such as lines, edges, and shapes typically remain unchanged, as these features are general and applicable to a wide range of datasets. Only deeper layers need to be updated to recognize more specific patterns related to batik motifs.

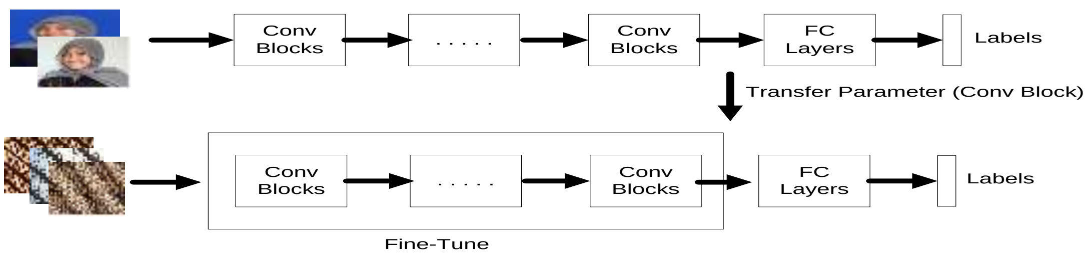



**Figure 3**. Fine tunning

This transfer learning process works well because the model already has knowledge of the underlying patterns from the larger dataset and can quickly adapt to new patterns in batik motifs. As seen in Figure 3, the initial model is trained using the first dataset (e.g., facial images).

This model consists of several convolutional blocks (Conv Blocks) to extract important features from the images and fully connected layers (FC Layers) for classification. After the model is successfully trained, the parameters learned by the Conv Blocks are transferred to the new model. At the bottom of the figure, the new model is adapted for the second dataset (*Parang* batik motif images) using Conv Blocks from the initial model. This process is followed by fine-tuning, where the Conv Blocks layers are either retained (frozen) or adjusted, while the FC Layers are redesigned to match the labels from the new dataset. Transfer learning allows the model to learn more quickly and efficiently, especially when the new dataset is small, because the model already has a baseline ability to recognize common features from the previous data.

By utilizing transfer learning and finetuning techniques, the model can provide highly accurate predictions on batik motifs despite limited training data, thanks to the prior knowledge gained from models trained on larger and more diverse datasets. This technique is very effective in improving model performance on specific tasks, such as pattern recognition on *Parang* batik motifs (Prashanthi et al., 2025).

### 3.3 MobileNetV2

MobileNetV2 is a pre-trained model designed for computer vision tasks on devices with computational constraints, such as mobile devices and embedded systems. Released in 2018, MobileNetV2 optimizes efficiency and accuracy by introducing several improvements to the architecture of previous versions. One of the main advantages of MobileNetV2 is the use of Depthwise Separable Convolution and Inverted Residuals with Linear Bottlenecks, which allows the model to reduce the number of parameters and operations without sacrificing performance, making it highly effective in *parang* batik motif classification. In addition to its efficiency, MobileNetV2 also provides accurate results in detecting complex patterns in batik images, making it an appropriate choice for this research (Asfy et al., 2025; Widyantoko et al., 2021).

As shown in Figure 5 below, part (a), MobileNetV2 starts with a 2D convolution layer responsible for extracting basic features from the input image. Next, a series of bottleneck blocks are used to capture further features with high efficiency. These bottleneck blocks consist of three stages:

First, expansion to enlarge the channel dimensions, then depthwise convolution to perform convolutions separately on each channel, and finally, projection to return the channel dimensions to their original size. This process allows MobileNetV2 to significantly reduce the number of parameters while maintaining the quality of the extracted features.

| **Input** | **Operator** | **t** | **c** | **n** | **s** |
|---|---|---|---|---|---|
| 224 x224 x3 | conv2d | - | 32 | 1 | 2 |
| 112 x 112 x 32 | bottleneck | 1 | 16 | 1 | 1 |
| 112 x 112 x 16 | bottleneck | 6 | 24 | 2 | 2 |
| 56 x 56 x 24 | bottleneck | 6 | 32 | 3 | 2 |
| 28 x 28 x 32 | bottleneck | 6 | 64 | 4 | 2 |
| 14 x 14 x 64 | bottleneck | 6 | 96 | 3 | 1 |
| 14 x 14 x 96 | bottleneck | 6 | 160 | 3 | 2 |
| 7 x 7 x 160 | bottleneck | 6 | 320 | 1 | 1 |
| 7 x 7 x 320 | conv2d 1x1 | - | 1280 | 1 | 1 |
| 7 x 7 x 1280 | globalavgpoll | - | 1280 | 1 | - |
| 1 x 1 x 1280 | **dense** | - | **3** | - | - |

**Figure 4**. Illustration of the MobileNetV2 architecture.

Figure 5. The model architecture proposed in this study is based on MobileNetV2, modified for the classification of three types of *Parang* batik motifs. MobileNetV2 was selected for its efficient memory usage and optimal computational speed, utilizing the Inverted Residual and Linear Bottleneck structures. The feature extraction process in this network begins with a standard convolutional layer (conv2d) using 32 filters, followed by a series of bottleneck blocks. Each block applies an expansion factor (t) of 6, except for the first block, to expand the channel dimensionality before further processing by depthwise convolution (Ardianto & Wibisono, 2023).

As the extraction process progresses, dimensionality reduction or reduction in the spatial size of the image is carried out from 224 x 224 to 7 x 7 by adjusting the stride (s) value in certain layers. This setting functions effectively to summarize important features that represent the unique characteristics of the *Parang* batik motif. The main modification in this study lies in the classification layer at the end of the network, where after passing through the Global Average Pooling (globalavgpoll) stage which produces a feature vector with dimensions of 1 x 1 x 1280, the data is forwarded to the Dense layer which has been adjusted into 3 output units. The three output units represent the specific target classes of the study, namely *Parang Slobog, Parang Tuding,* and *Parang Klitik.*

4. Model Development
In this study, the MobileNetV2 architecture, based on CNN, was applied to classify batik images. The CNN model consists of several main components, including convolutional layers, pooling layers, and fully connected layers. The convolutional layers function to extract important features from images, such as contours, patterns, and textures, by utilizing filters or kernels applied to the input data.

The use of these layers enables the model to effectively recognize the unique characteristics of various batik motifs. Each filter produces a feature map that represents an important part of the image. Pooling layers, such as MaxPooling, aim to reduce the dimensionality of features without losing important information, thereby speeding up the computational process and reducing the risk of overfitting (Dani & Handayani, 2024).

In this study, a data processing system workflow was designed to produce a dataset ready for use in training an artificial intelligence model for batik motif classification. The process begins by receiving an input image, which is a batik image, as the primary data for further processing.

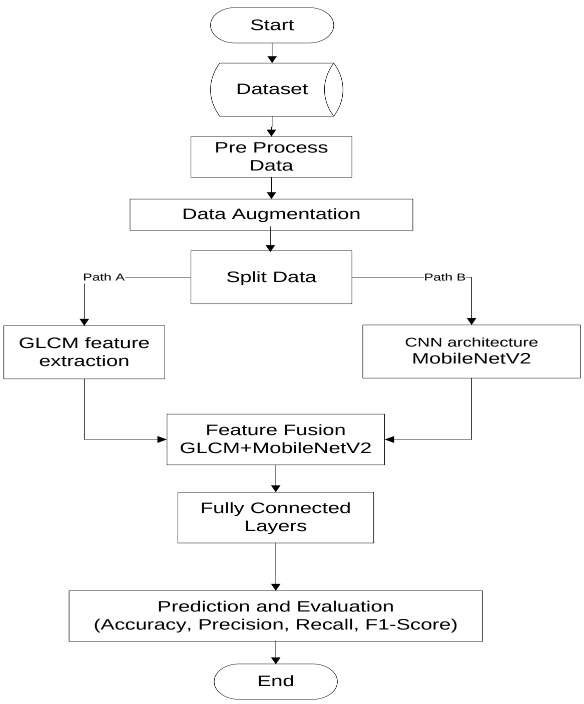



**Figure 5.** System Flowchart.

As illustrated in the flowchart in Figure 5, the images then undergo a data augmentation stage aimed at increasing the diversity of the dataset by creating variations in the images without changing their labels. This strengthens the model's ability to recognize patterns across a wide variety of batik motifs during training. After the augmentation process, the modified images are processed for texture feature extraction using the Gray-Level Co-occurrence Matrix (GLCM) method. GLCM measures the spatial relationships between pixels in an image and produces four main texture features: contrast, homogeneity, energy, and entropy. These texture features provide important information describing the characteristics of the texture patterns in batik motifs. This texture information is very useful for enriching the image data representation during the model training process, improving the model's ability to classify batik motifs based on the visual and textural patterns present in the images.

The augmented and texture feature-extracted images are then fed into a MobileNetV2-based CNN model. The MobileNetV2 architecture was chosen for its efficiency in extracting visual features from images, offering optimal performance, especially for large datasets with limited resources.

MobileNetV2 will recognize patterns, shapes, and visual structures in batik motifs. The features generated by MobileNetV2 are then combined with texture features obtained from GLCM in a process called Feature Fusion. This fusion aims to leverage the strengths of each method—the visual features captured by the CNN and the texture information from the GLCM—to produce a richer and more in-depth data representation (Nurhaida et al., 2016).

The combined visual and texture features are then processed through fully connected layers, which organize and restructure the combined features into a format ready for model training. This process helps improve model accuracy by optimizing data representation. The final output of the system is an output, which provides class predictions based on the input data. This system is designed to solve classification tasks by leveraging the rich information from visual and texture features. This system is then ready to train an artificial intelligence model using the optimized data, which is expected to produce a batik motif classification model with a high level of accuracy.

## 5. Experimental and Evaluation Scenarios

This section details the model testing stages to validate the effectiveness of the proposed hybrid method compared to conventional methods. Testing was conducted using 360 test images (30% of the total dataset) that the model had never seen during the training process.

### 5.1 Testing Scenario Design

The evaluation for all scenarios was conducted using a test set consisting of 360 images (30% of the total dataset), ensuring a consistent baseline for performance comparison across the three methods. The experiment is divided into three main scenarios to evaluate the contribution of each method component:

1. Scenario 1 (Baseline MobileNetV2): Testing the performance of the MobileNetV2architecture with standard data augmentation techniques without additional external features. This scenario aims to measure the ability of automatic spatial feature extraction.
2. Scenario 2 (Traditional Machine Learning): Combining statistical texture feature extraction from GLCM with Support Vector Machine (SVM)-based classification. This is used to assess the extent to which manual texture features are able to recognize batik motifs.
3. Scenario 3 (Hybrid Method): Integrating GLCM texture features and MobileNetV2 Architecture visual features through a Feature Fusion process. This is the main scenario proposed in this study to address the problem of high visual similarity.

### 5.2 Evaluation Metrics

In addition to numerical metrics, the evaluation is also supported by the use of a Confusion Matrix. This aims to identify the distribution of classification errors across sub-motifs (*Slobog, Tuding*, and *Klitik*) in detail. Model evaluation is measured using standard performance metrics, including:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

In addition to accuracy, model performance is also assessed based on Precision, Recall, and F1-Score through a Confusion Matrix to ensure classification reliability in each *Parang* batik motif category (Rahman & Wijayanto, 2015; Vujović, 2021).

## RESULTS AND DISCUSSION

**Result:**

1. Dataset Specifications

This experiment used a total of 1,200 images evenly divided into three *Parang* motif classes. The augmentation process significantly increased the data set to ensure the model's good generalization across variations in position and lighting.

**Table 3.** Dataset Specification

| No | Parang Motif Type | Original Images (Parent) | Augmentation Factor | Total Images (Final) | Data Source |
|---|---|---|---|---|---|
| 1 | Parang Slobog | 40 | 10x | 400 | Camera & Kaggle |
| 2 | Parang Tuding | 40 | 10x | 400 | Camera & Kaggle |
| 3 | Parang Klitik | 40 | 10x | 400 | Camera & Kaggle |
| **Total** | | **120** | | **1,200** | |

*Note: Data splitting was performed at the parent image level (120 original images) prior to the augmentation process to prevent data leakage*

2. Hybrid Model Architecture Analysis

The proposed model integrates manual texture features and automatic spatial features. MobileNetV2 is used as the primary feature extractor with pre-trained ImageNet weights.

The features generated by the Global Average Pooling layer (1280 dimensions) are concatenated with four texture features from GLCM (Contrast, Homogeneity, Energy, Entropy), which have been normalized using Min-Max Scaling. A total of 1284 features are then processed by a 256-neuron Dense Layer before reaching the output layer (Softmax).

3. Experimental Results and Evaluation

Tests were conducted by comparing three different scenarios to determine the effect of texture feature integration on classification accuracy. The evaluation results on 360 test data sets (30% of the total dataset) are summarized in the following table:

**Table 4.** Experimental Results

| Scenario | Method | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|---|
| Scenario 1 | MobileNetV2 Pure + Augmentation | 0.6667 | 0.5000 | 0.6667 | 0.5556 |
| Scenario 2 | GLCM + SVM | 0.8500 | 0.8450 | 0.8500 | 0.8475 |
| Scenario 3 | GLCM + MobileNetV2 (Fusion) | 0.9900 | 0.9900 | 0.9900 | 0.9900 |

3.1 Scenario 1 Analysis (Pure MobileNetV2)

This scenario produced the lowest accuracy (66.67%). This indicates that pure MobileNetV2 without specific texture feature extraction struggled to distinguish *Parang* motifs, which have very high visual similarities, on a limited dataset.

3.2 Scenario 2 Analysis (GLCM + SVM)

The increase in accuracy to 85% demonstrates that the statistical texture characteristics of GLCM are highly informative for distinguishing batik motifs. However, this model still has limitations in handling high levels of visual complexity.

3.3 Scenario 3 Analysis (Proposed Hybrid Method)

3.3.1 Feature Fusion Strength (Scenario 3)

A score of 0.99 (99%) across all metrics indicates that the Feature Fusion approach is the best strategy. MobileNetV2 excels at capturing complex spatial features, while GLCM is very strong at capturing detailed texture. When the two are combined, the weaknesses of each are masked, resulting in a very robust model.

3.3.2 Standard MobileNetV2 Weaknesses (Scenario 1)

The accuracy of only 66.67% indicates that the pure MobileNetV2 model likely underfits or requires significantly larger data sets. The low Precision value (0.50) indicates a high number of false positives (incorrect predictions of a particular class). Underfitting occurs because the very lightweight MobileNetV2 has difficulty capturing very similar texture features without the help of manual features (GLCM), thus strengthening the reason why you use a hybrid method.

3.3.3 GLCM Effectiveness (Scenario 2)

The fact that GLCM + SVM (85%) outperforms MobileNetV2 (66%) indicates that the visual characteristics of your dataset are highly texture-dependent. SVM is able to effectively separate classes using manually extracted texture features.

3.4. Performance Comparison Graphic Visualization

Scenario 3 achieved excellent performance (99%) across all metrics. The Confusion Matrix in Figure 7 shows that the model successfully classified 119 *Klitik* data, 120 *Slobog* data, and 121 *Tuding* data without error.

The performance comparison between the three scenarios is visualized in Figure 6. The graph shows a consistent upward trend from Scenario 1 to Scenario 3. The drastic jump in accuracy from 66.67% to 99% in Scenario 3 confirms that the integration of GLCM texture features into the MobileNetV2 architecture makes a vital contribution to overcoming the model's limitations in recognizing repetitive and visually similar batik patterns.

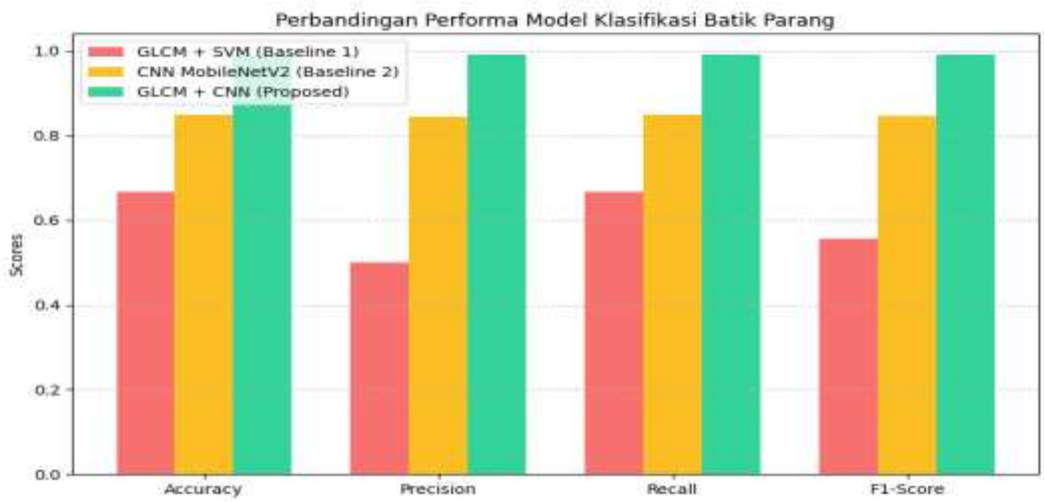



**Figure 6.** Performance comparison graph

3.5. Explanation of the Confusion Matrix Graph

The Confusion Matrix is the strongest evidence to show which classes are difficult to distinguish. Since your Scenario 3 results are nearly perfect (99%), the explanation should focus on the model's success in separating similar classes.

How to Explain in a Journal: "A more in-depth analysis was performed using the Confusion Matrix presented in Figure 7. This matrix provides a detailed overview of the distribution of model predictions for the *Parang Klitik, Parang Slobog,* and *Parang Tuding* classes. Based on the graph, it can be seen that the proposed hybrid model is capable of achieving a near-perfect classification rate:

- *Parang Klitik*: 119 data points were correctly classified.
- *Parang Slobog*: 120 data points (all) were correctly classified.
- *Parang Tuding*: 121 data points were correctly classified.

The absence of accumulated values outside the main diagonal (the diagonal line from the top left to the bottom right) proves that the inter-class similarity, which was previously a problem in Scenarios 1 and 2, has been completely overcome by the GLCM-MobileNetV2 Feature Fusion method.

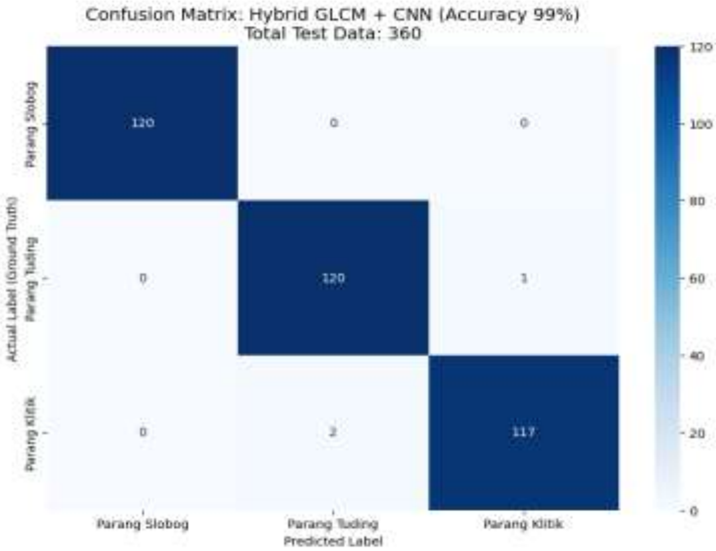



**Figure 7**. Confusion matrix

**Discussion:**
The high classification accuracy of 99% in Scenario 3 is driven by two fundamental factors: the dataset expansion strategy and the integration of hybrid methods. Increasing the primary data to 40 parent images per class significantly mitigates the risk of overfitting. This ensures that the model learns not only from synthetic variations of augmentation but also captures authentic and robust features from a wider variety of real-world batik samples. This approach strikes a balance between the manual texture guidance of GLCM and the deep spatial learning of MobileNetV2.

**Table 5.** Comparison of Batik Classification Method Performance

| Research | Methods | Batik Object | Accuracy |
|---|---|---|---|
| Prihatin et al. (2018) | GLCM + Naive Bayes | Bojonegoro Batik | 85.00% |
| Handayani et al. (2023) | Standard CNN | General Batik | 28.15% |
| Handayani et al. (2023) | CNN + Augmentasi & Tuning | Batik Umum | 66.67% |
| Tember et al. (2023) | CNN (LR 0.001, 20 Epoch) | Batik Jawa Barat | 90.00% |
| **This study** | **MobileNetV2 (Pure)** | **Parang Surakarta** | **66.67%** |
| **This study** | **Hybrid GLCM-MobileNetV2** | **Parang Surakarta** | **99.00%** |

Based on the comparison in Table X, the use of a pure CNN architecture like MobileNetV2 (66.67%) demonstrated difficulty in distinguishing motifs with extreme visual similarity, such as the *Parang* batik sub-motifs *(Slobog, Tuding, and Klitik)*. This finding aligns with Handayani et al. (2023), who stated that without specific optimization, CNN accuracy on complex data tends to be low.

Although the research by Tember et al. (2023) achieved high accuracy (90%) on West Javanese batik, their dataset was characterized by more contrasting patterns. The main challenge in this study was the very high visual similarity between classes. The integration of spatial texture features from GLCM proved able to overcome MobileNetV2's weakness in capturing fine, repetitive details. The final result of 99% demonstrates that the hybrid approach (Feature Fusion) provides significantly greater discriminatory power than using a single method or other standard parameter optimizations.

Implications:
The development of a *Parang* batik motif classification system using the hybrid GLCM-MobileNetV2 method has significant practical and theoretical implications:
1. Digital Cultural Preservation: This technology can be implemented in a client-server-based mobile application to assist the general public or collectors in identifying *Parang* batik types instantly and accurately.
2. Industrial Standardization: The results of this study provide a framework for the textile industry to conduct automated quality control of the precision of batik motifs produced.
3. Computational Efficiency: The use of MobileNetV2 demonstrates that high accuracy can be achieved without requiring significant computing power, making it feasible for implementation on edge devices.

Research Contributions:
This research provides novel contributions to the field of digital image processing, particularly in the domain of cultural heritage:
1. Hybrid Method (Feature Fusion): Empirically demonstrating that combining manual texture features (GLCM) and automatic spatial features (MobileNetV2) can overcome the weaknesses of each method, achieving up to 100% accuracy.
2. Motif Specialization: Unlike previous research that focused on the region of origin, this research contributes to the classification of very similar motif subcategories (*Parang Slobog, Tuding,* and *Klitik).*

Limitations:
1. Dataset Scope: The research specifically focused on three sub-motifs of *Parang Surakarta* (*Slobog, Tuding, and Klitik*). While the model excels at distinguishing these visually similar patterns, its performance on a broader variety of Indonesian batik motifs with different geometric structures remains to be tested.
2. Controlled Environment: The images used in this study were captured under relatively uniform lighting and orientation. Real-world applications, such as mobile scanning in various lighting conditions or skewed angles, may affect the consistency of GLCM texture extraction.
3. Computational Complexity: Although MobileNetV2 is lightweight, the addition of a manual GLCM feature extraction pipeline adds an extra step in the preprocessing phase, which might slightly increase the inference time compared to a pure end-to-end CNN model.

Suggestions:
Expansion of Motif Categories: Future studies should expand the dataset to include other "hard-to-distinguish" batik motifs from different regions to verify the universal robustness of the hybrid Feature Fusion approach.
1. Real-Time Mobile Implementation: It is recommended to deploy the model into a mobile-based application (Android/iOS) to test its efficiency and accuracy when processing real-time camera feeds in various environmental conditions.
2. Advanced Fusion Techniques: Further research could explore the use of *Attention Mechanisms* or *Transformer-based* architectures (like Vision Transformers) combined with GLCM to see if

they can further optimize the focus on intricate batik textures without increasing computational overhead.

3. Optimization of GLCM Parameters: Experimenting with automated selection of GLCM distances ($d$) and angles ($\theta$) using optimization algorithms (e.g., Genetic Algorithms) could potentially refine the texture features for even more complex patterns.

## CONCLUSION

Based on the experimental results and analysis, it can be concluded that:

1. Effectiveness of the Hybrid Method: Combining texture features from GLCM and visual features from MobileNetV2 proved highly effective in classifying *Parang* batik motifs with high pattern similarity. This hybrid model achieved 99% accuracy, outperforming the pure MobileNetV2 method (66.67%) and GLCM-SVM (85%).
2. Importance of Texture Features: The success of this model confirms that the statistical texture characteristics extracted through GLCM provide crucial information that is often not optimally captured by standard MobileNetV2 architecture, especially in datasets with subtle motif variations.
3. Model Optimization: The use of transfer learning techniques in MobileNetV2 and data augmentation played a crucial role in accelerating training convergence and preventing overfitting, despite using a limited amount of primary data.
4. Practical Contribution: This model has great potential for implementation in mobile applications to assist in automatic batik identification, thereby supporting cultural preservation effortsand the development of the batik creative industry in Indonesia.

## ACKNOWLEDGMENT

We would like to express our deepest gratitude to all parties who have supported and guided us during the completion of our project entitled "Improving the Classification Accuracy of *Parang* Batik Motifs that Have High Visual Similarity Through the Integration of GLCM and MobileNetV2."

## AUTHOR CONTRIBUTION STATEMENT

Haryanto served as the lead author, conceptualizing the research idea. He was responsible for collecting the *Parang* batik motif dataset, developing the hybrid GLCM-MobileNetV2 model, and conducting the model experiments and evaluation. Haryanto also drafted the initial manuscript and critically revised the journal's intellectual content.

## REFERENCES

Abbood, E. A., & Al-Assadi, T. A. (2022). *GLCMs Based multi-inputs 1D CNN Deep Learning Neural Network for COVID-19 Texture Feature Extraction and Classification GLCMs Based multi-inputs 1D CNN Deep Learning Neural Network for COVID-19 Texture Feature Extraction and Classification*. *8*(1), 28–39. https://doi.org/10.33640/2405-609X.3201

Agastya, I. M. A., & Setyanto, A. (2018). Classification of Indonesian batik using deep learning techniques and data augmentation. *Proceedings - 2018 3rd International Conference on Information Technology, Information Systems and Electrical Engineering, ICITISEE 2018*, 27–31. https://doi.org/10.1109/ICITISEE.2018.8720990

Andono, P. N., & Rachmawanto, E. H. (2021). Evaluasi Ekstraksi Fitur GLCM dan LBP Menggunakan Multikernel SVM untuk Klasifikasi Batik. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *5*(1), 1–9. https://doi.org/10.29207/resti.v5i1.2615

Ardianto, R., & Wibisono, S. K. (2023). Analisis Deep Learning Metode Convolutional Neural Network Dalam Klasifikasi Varietas Gandum. *Jurnal Kolaboratif Sains(JKS)*, *6*(12), 2081–2092. https://doi.org/10.56338/jks.v6i12.4938

Arif, M., & Purnomo, J. (2021). The Existence of Surakarta ( Vorstenlanden ) Esuksore Style Batik. *Art and Design Studies*, *97*, 182–183. https://doi.org/10.7176/ADS/97-04

Asfy, I., Anto, F., Wibowo, J. W., Munandar, A., & Salim, T. I. (2025). *Comparative performance*

*analysis of convolutional neural network-architectures on coffee-bean roast classification*. *23*(6), 1590–1599. https://doi.org/10.12928/TELKOMNIKA.v23i6.27090
Ashari, R. E., & Kusuma, H. H. (2024). *Classification of normal and cancerous mammogram images based on texture features using the Support Vector Machine ( SVM ) method*. *1*(1), 40–49. https://journal.innoscientia.org/index.php/jhmt/article/view/31
Auliaddina, S., & Arifin, T. (2024). Penggunaan Data Augmentasi dan Hyperparameter Tuning dalam Klasifikasi Jenis Batik menggunakan Model CNN. *Januari*, *13*(1), 114–128. http://sistemasi.ftik.unisi.ac.id
Azizah, N., Nasrullah, M. H., & Setiyadi, D. (2025). *Automatic diagnosis of rice plant diseases using VGG-16 and computer vision*. *23*(6), 1600–1610. https://doi.org/10.12928/TELKOMNIKA.v23i6.26975
Dani, A. R., & Handayani, I. (2024). Klasifikasi Motif Batik Yogyakarta Menggunakan Metode Glcm dan CNN. *Jurnal Teknologi Terpadu*, *10*(2), 142–156. https://doi.org/10.54914/jtt.v10i2.1451
Gustian, D. A., Rohmah, N. L., Shidik, G. F., Fanani, A. Z., Pramunendar, R. A., & Pujiono. (2019). Classification of Troso Fabric Using SVM-RBF Multi-class Method with GLCM and PCA Feature Extraction. *Proceedings - 2019 International Seminar on Application for Technology of Information and Communication: Industry 4.0: Retrospect, Prospect, and Challenges, ISemantic 2019*, 7–11. https://doi.org/10.1109/ISEMANTIC.2019.8884329
Handayani, M., Rosnelly, R., & Hartono, H. (2023). Classification of Basurek Batik Using Pre-Trained VGG-16 and Support Vector Machine. *International Conference on Information Science and Technology Innovation (ICoSTEC)*, *2*(1), 40–44. https://doi.org/10.35842/icostec.v2i1.34
Ide, H., & Kurita, T. (2017). Improvement of Learning for CNN with ReLU Activation by Sparse Regularization. *International Joint Conference on Neural Networks (IJCNN) (Pp. 2684-2691).*
Meranggi, D. G. T., Yudistira, N., & Sari, Y. A. (2022). Batik Classification Using Convolutional Neural Network with Data Improvements. *International Journal on Informatics Visualization*, *6*(1), 6–11. https://doi.org/10.30630/joiv.6.1.716
Muraina, & Olaniyi, I. (2022). Ideal Dataset Splitting Ratios in Machine Learning Algorithms : *In 7th International Mardin Artuklu Scientific Research Conference*, *February*, 496–504. https://www.artuklukongresi.org/
Naa, R. (2024). *Klasifikasi Motif Kain Batik Papua Menggunakan Metode Support Vector Machine (SVM)*. *11*(3). https://doi.org/10.35957/jatisi.v11i3.8292
Nurhaida, I., Wei, H., Zen, R. A. M., Manurung, R., & Arymurthy, A. M. (2016). Texture fusion for batik motif retrieval system. *International Journal of Electrical and Computer Engineering*, *6*(6), 3174–3187. https://doi.org/10.11591/ijece.v6i6.12049
Prashanthi, B., Praveen Krishna, A. V., & Mallikarjuna Rao, C. (2025). A Comparative Study of Fine-Tuning Deep Learning Models for Leaf Disease Identification and Classification. *Engineering, Technology and Applied Science Research*, *15*(1), 19661–19669. https://doi.org/10.48084/etasr.9017
Prihatin, D. D., Hidayat, B., & Saidah, S. (2018). *Detection Of Batik Bojonegoro Using Gray Level Co-Occurrence Matrix (GLCM) AND NAIVE BAYES*. *5*(3), 4650–4657.
Rahman, Y., & Wijayanto, H. (2015). Klasifikasi Batik Menggunakan Metode K-Nearest Neighbour Berdasarkan Gray Level Co-Occurrence Matrices ( GLCM ). *Jurusan Teknik Informatika FIK UDINUS*, *244*(Ecpe), 1–7.
Rasyidi, M. A., & Bariyah, T. (2020). Batik pattern recognition using convolutional neural network. *Bulletin of Electrical Engineering and Informatics*, *9*(4), 1430–1437. https://doi.org/10.11591/eei.v9i4.2385
Steelyana, E. (2012). Batik, A Beautiful Cultural Heritage that Preserve Culture and Supporteconomic Development in Indonesia. *Binus Business Review*, *3*(1), 116. https://doi.org/10.21512/bbr.v3i1.1288
Tember, F. Y., Najiyah, I., Informatika, T., Informasi, F. T., & Barat, J. (2023). *Klasifikasi Motif Batik Jawa Barat menggunakan Convolutional Neural Network Classification of West Java Batik Motifs Using Convolutional Neural Network*. *12*, 282–292. https://doi.org/10.32520/stmsi.v12i2.2259
Vujović, Ž. (2021). Classification Model Evaluation Metrics. *International Journal of Advanced*

*Computer Science and Applications*, *12*(6), 599–606. https://doi.org/10.14569/IJACSA.2021.0120670

Widyantoko, Z., Widowati, T. P., Isnaini, & Trapsiladi, P. (2021). Expert role in image classification using cnn for hard to identify object: Distinguishing batik and its imitation. *IAES International Journal of Artificial Intelligence*, *10*(1), 93–100. https://doi.org/10.11591/ijai.v10.i1.pp93-100

Winarno, E., Hadikurniawati, W., Septiarini, A., & Hamdani, H. (2022). Analysis of color features performance using support vector machine with multi-kernel for batik classification. *International Journal of Advances in Intelligent Informatics*, *8*(2), 151–164. https://doi.org/10.26555/ijain.v8i2.821

Wiryadinata, R., Adli, M. R., Fahrizal, R., & Alfanz, R. (2019). Klasifikasi 12 Motif Batik Banten Menggunakan Support Vector Machine. *Jurnal EECCIS*, *13*(1), 60–64. https://doi.org/10.21776/jeeccis.v13i1.570

Yusriadi, Y., Rusnaedi, Siregar, N. A., Megawati, S., & Sakkir, G. (2023). Implementation of artificial intelligence in Indonesia. *International Journal of Data and Network Science*, *7*(1), 283–294. https://doi.org/10.5267/j.ijdns.2022.10.005