Sistemasi: Jurnal Sistem Informasi
Volume 14, Nomor 5, 2025: 2382-2394

ISSN:2302-8149
e-ISSN:2540-9719

Optimization of CNN Activation Functions using Xception for South Sulawesi Batik Classification

[1]Aswan*, [2]Eva Yulia Puspaningrum, [3]Billy Eden William Asrul
[1,3]Faculty of Computer Science, Department of Informatic Engineering, Universitas Handayani Makassar
[2]Faculty of Computer Science, Department of Informatic Engineering, Universitas Pembangunan Nasional "Veteran" Jawa Timur
[1,3]Jl. Adiyaksa Baru No.1, Makassar, Indonesia
[2]Jl. Raya Rungkut Madya No.1, Surabaya, Indonesia
*e-mail: ajwan.pro@gmail.com

(***received***: 6 May 2025, ***revised***: 6 July 2025, ***accepted***: 10 July 2025)

Abstract

Batik motifs from South Sulawesi such as the Pinisi boat, Lontara script, Tongkonan house and Toraja combinations embody rich cultural narratives but are difficult to identify automatically. Automatic classification supports cultural preservation and education and empowers tourism and digital heritage applications. This study improves the performance of convolutional neural networks for South Sulawesi batik classification by optimizing activation functions within the Xception architecture which exploits depthwise separable convolutions for efficient and detailed feature extraction. A balanced dataset of 1400 labeled images in four classes was divided into eighty percent for training, ten percent for validation and ten percent for testing. Images were resized to 224 by 224 pixels, converted to grayscale and augmented through zoom, flip and rotation. With identical hyperparameters including a learning rate of 0.001, a batch size of 64 and training for 100 epochs using the Adam optimizer, ReLU, ELU, Leaky ReLU and Swish activation functions were compared. Evaluation metrics comprised accuracy, precision, recall, F1 score and cross entropy loss. ELU achieved the highest test accuracy of 98.57 percent, precision of 0.9864, recall of 0.9857 and F1 score of 0.9857, outperforming ReLU and Leaky ReLU with 97.86 percent accuracy and Swish with 97.14 percent accuracy. The results demonstrate that selecting an optimal activation function substantially enhances convolutional neural network classification of complex batik patterns. The findings offer practical guidance for development of resource aware batik identification systems in support of cultural digitization and education initiatives.

Keywords: convolutional neural network, activation function, xception, south sulawesi batik, classification

1 Introduction

Automatic classification of batik motifs is a challenge in the field of computer vision, especially for complex motifs with distinctive local characteristics such as those found in South Sulawesi batik. Although batik has been recognized as Intangible Cultural Heritage by UNESCO since 2009 [1], [2], [3], [4], manual motif identification still relies on individual visual understanding and is inefficient on a large scale. Furthermore, distinctive motifs like Kapal Pinisi, Lontara, Tongkonan, and Toraja have unique patterns that are similar to one another, making them difficult to recognize without the support of intelligent systems [5], [6], [7].

The limitations of the community's understanding of these motifs, in terms of their names, philosophies, and origins, also pose obstacles to cultural preservation [8], [9], [10]. Efforts to digitize and classify batik motifs using technology have not been widely carried out specifically for regional motifs such as those from South Sulawesi. This indicates an urgent need for an artificial intelligence-based approach to support cultural preservation through technology [11].

One approach that has proven effective in image classification tasks is the Convolutional Neural Network (CNN), which has the ability to extract complex visual features from images [12]. CNNs have been widely used across various fields, but their performance is highly influenced by

ISSN:2302-8149
e-ISSN:2540-9719

architectural configurations, including the selection of appropriate activation functions to enhance accuracy and efficiency.

This study aims to improve the accuracy of South Sulawesi batik motif classification by optimizing the activation function in the CNN architecture. The main focus is on a comparative evaluation between the ReLU, ELU, Leaky ReLU, and Swish activation functions implemented in the Xception architecture. Xception was chosen for its ability to extract features through depthwise separable convolution, which has proven to be efficient in classifying complex images. The evaluation was conducted using accuracy, precision, recall, and F1-score metrics. The results of this study are expected to support the development of a reliable batik motif classification system and contribute to cultural preservation through a technological approach.

2 Literature Review

In the era of computer vision and deep learning development, Convolutional Neural Networks (CNNs) have become a dominant and effective architecture for image classification tasks. CNNs have the ability to automatically and hierarchically extract features from image data through a layered convolution process, making them highly reliable in recognizing complex patterns, textures, and visual structures [12]. In the context of classifying batik patterns, which are rich in detail and have high similarity between classes, CNNs are the appropriate approach because they can learn spatial representations that cannot be handled by conventional algorithms.

One important aspect of CNN performance is the selection of activation functions. These functions are responsible for introducing non-linearity into the network, enabling CNNs to learn complex relationships in the data [13]. Some commonly used activation functions include Sigmoid, ReLU (Rectified Linear Unit), Leaky ReLU, ELU (Exponential Linear Unit), and Swish [14]. Each has different characteristics and impacts on training stability and model generalization capabilities. For example, Swish is reported to outperform ReLU in deep networks due to its smooth non-linearity, while ELU offers better gradient stability on negative inputs.

Previous studies have highlighted the importance of activation functions in optimizing CNNs. Nanni et al. (2022) conducted an in-depth exploration of 20 different activation functions, including six new functions, in the VGG16 and ResNet50 architectures for classifying 15 small to medium-sized medical datasets. The results showed that an ensemble approach that randomly replaces the ReLU function with another function can improve performance and reduce overfitting on limited data [15]. Although significant, the focus of this study was on medical classification, which is contextually very different from cultural visual motifs such as batik.

In the domain of batik motif classification, Meranggi et al. used the ResNet-18 architecture to compare the performance of three preprocessing approaches (original, patch, balanced patch) on two versions of batik datasets with five motif classes. They found that using a new dataset and the patch method could improve accuracy by up to 22.7% [16]. However, this study did not explicitly evaluate the role of activation functions on classification performance. Its primary focus was on dataset quality and augmentation methods, not on internal CNN components like activation functions.

Some other literature also discusses the comparison of activation functions, but in a less relevant context. For example, a study on animal image classification using VGG16 showed that ReLU outperforms Sigmoid in terms of accuracy and training stability [13]. However, animal images have different visual characteristics from batik motifs, which are rich in geometry, repetitive patterns, and complex textural details.

From this literature review, a clear research gap can be identified. There has been no comprehensive study that specifically evaluates and optimizes CNN activation functions in batik pattern classification, particularly for the distinctive patterns of South Sulawesi, which have unique visual characteristics such as Pinisi, Lontara, Tongkonan, and Toraja. Therefore, this study aims to bridge this gap by implementing the Xception architecture, which is known for its superiority in feature extraction through depthwise separable convolution. Using a dataset consisting of 1,400 images of four batik motifs, this study compares the performance of four activation functions ReLU, Leaky ReLU, ELU, and Swish in the context of batik motif classification based on accuracy, precision, recall, F1-score, and training time efficiency metrics [17]. The results of this study are expected to contribute to the development of a more accurate, efficient, and adaptive batik motif

 ISSN:2302-8149
e-ISSN:2540-9719

classification system, while also enriching our understanding of CNN architecture optimization in the context of cultural image recognition.

3 Research Method

Figure 1 below presents a flowchart depicting all stages of the research, starting from data collection, image pre-processing, training the CNN model with the Xception architecture, testing the activation function, to the evaluation and analysis of classification results.

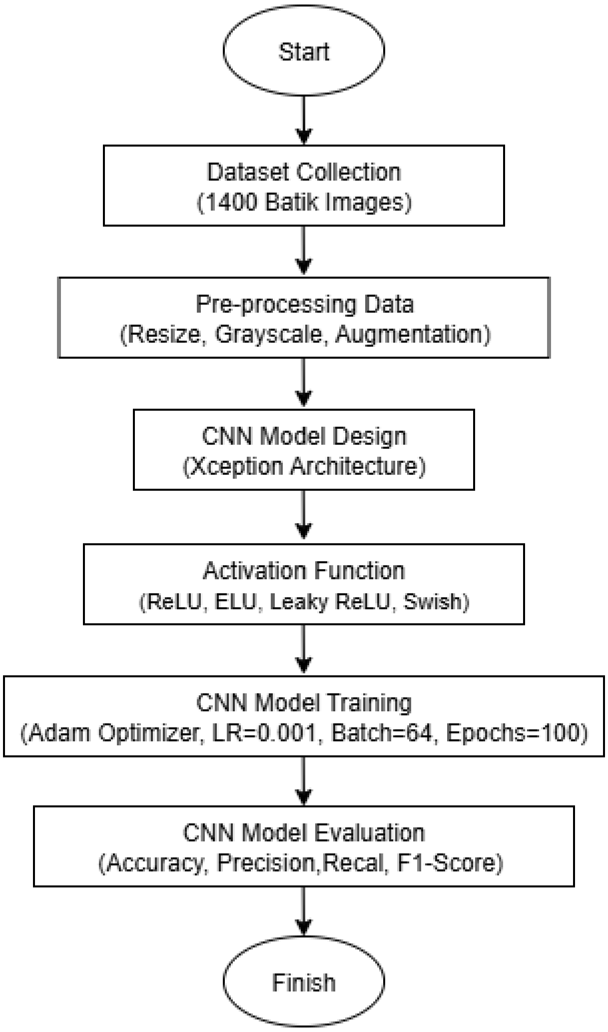


Figure 1. Research flow

3.1 Dataset Collection

The dataset in this study was taken from several sources, such as social media, Kaggle, and Google Drive. The dataset consists of four classes of batik motifs, namely lontara motifs, pinisi boat motifs, tongkonan motifs, and Toraja combination motifs, all of which originate from South Sulawesi Province. The data will then be divided into three parts, namely training data, testing data, and validation data. Each section comprises 80% training data, 10% testing data, and 10% validation data. This division is done using the stratified random splitting method. This means that each subset (training, testing, and validation) will have a proportional representation of each batik motif class, thereby preventing bias and ensuring reproducibility of results using a specific random seed. This division is also done to utilize the validation data to monitor model performance during training, prevent overfitting, and optimize hyperparameters. In comparison, if an 80% training and 20% testing split is used without validation, the model will not have separate data to monitor progress during training, increasing the risk of overfitting because the model is only trained with training data and without continuous evaluation with unseen data. By using validation data, this study can ensure that

ISSN:2302-8149
e-ISSN:2540-9719

the model is optimized more effectively. The images used have a resolution of 224 x 224 pixels. Table 1 shows the batik motif classes and the number of data points for each batik motif class. Samples from each dataset can be seen in Figure 2.

Table 1. Amount of data

Field No	Field Class	Field Total
1	Lontara	500
2	Pinisi Boat	300
3	Tongkonan	300
4	Toraja Combined	300
Total		1400
		1400

(a) (b) (c) (d)

Figure 2. Dataset (a) lontara motif, (b) pinisi boat motif, (c) tongkonan motif and (d) toraja combined motif

3.2 Pre-processing Data

The collected data underwent preprocessing before being used in the CNN. In this study, there were three preprocessing stages adjusting the pixels in each data model, as the data came from several different sources, resulting in data of varying sizes converting the data from RGB to grayscale, which aimed to facilitate data analysis and applying data augmentation [9], which is a regularization technique that aims to optimize model performance, avoid overfitting, and is easy to implement. The final step is to apply data augmentation, which is a regularization technique aimed at optimizing model performance [18], [19], preventing overfitting, and being easy to implement [20]. The augmentation techniques applied include zoom, flip, and rotation.

3.3 CNN Model Design

In building the CNN model, this study uses the Xception architecture due to its advantage in handling spatial features more efficiently through depthwise separable convolutions. The Xception architecture, which is an extension of the Inception architecture, presents a comprehensive depthwise separable convolution approach, enabling the model to generate more detailed features and process data more optimally [21]. This process is divided into two stages, namely depthwise and pointwise convolutions, which allow Xception to extract more detailed and complex visual information from images [22]. This advantage speeds up training time, improves computational efficiency, and reduces the number of parameters required. Thus, this architecture allows the model to learn important patterns more quickly and deeply. Compared to other CNN architectures, such as ResNet or VGG,

ISSN:2302-8149
e-ISSN:2540-9719

Xception offers better computational efficiency without sacrificing feature detection quality. This is important for improving training time and accuracy on datasets with image variations, such as batik patterns in this study. To evaluate its impact, activation functions such as ReLU, ELU, Leaky ReLU, and Swish were applied to the convolutional layers and also the fully connected layers. Finally, the softmax layer was used for image classification.

3.4 Activation Function

This study conducts a comprehensive analysis of CNN model performance using accuracy, precision, recall, F1-score, and loss value methods for various activation functions (ReLU, ELU, Leaky ReLU, and Swish). The ReLU activation function is one of the most widely used in contemporary CNN architectures. ReLU offers a high convergence rate and suffers from dead cell problems, where some neurons may become permanently inactive during training, leading to information loss.

Meanwhile, the ELU activation function is designed to address this issue by providing a negative value when the input is less than zero. This helps maintain gradients and better training stability. ELU is a good alternative to ReLU due to its advantages in convergence and reducing the effects of neuron death. However, ELU tends to be slower in processing compared to ReLU. Leaky ReLU, a variation of ReLU, addresses the issue of ReLU dying by introducing a small slope on its negative side. In this way, Leaky ReLU ensures that neurons do not completely die and can still update their weights during the training process, enhancing the model's ability to extract information from data.

On the other hand, the Swish activation function, created by Google, is a combination of linear and non-linear functions. Swish excels in both deeper models and convergence contexts. Although its computation is more complex due to the use of exponential operations, its advantages in generalization and accuracy make it attractive for various complex image classification tasks.

This study aims to determine the most suitable activation function for classifying South Sulawesi batik motif images, considering the training speed and generalization ability of the model. This is achieved by understanding the characteristics of each activation function.

3.5 CNN Model Training

The model training process was carried out using the Adam optimization algorithm with a learning rate of 0.001, a batch size of 64, and 100 epochs [23], [24]. This study implemented the Xception architecture, a convolutional neural network model that relies on depthwise separable convolution techniques to improve efficiency and accuracy in visual feature extraction. This architecture consists of 36 convolutional layers arranged in residual modules (entry flow, middle flow, and exit flow), and ends with global average pooling and a fully connected layer. The output layer uses the Softmax activation function for multi-class classification.

In this study, no structural modifications were made to the core Xception architecture, but experiments were conducted on the activation functions in the hidden layer. Four activation functions were tested, namely ReLU, ELU, Leaky ReLU, and Swish, by replacing the default activation function in the convolutional and fully connected layers to assess their impact on classification performance. Additionally, to reduce overfitting, the model included a dropout layer of 0.5 before the final fully connected layer.

The model was developed in a Jupyter Notebook environment using the Python programming language, with TensorFlow and Keras as the main libraries. The training and evaluation processes were assisted by additional libraries such as NumPy and Pandas for data manipulation, imutils for image processing, scikit-learn for model performance measurement (accuracy, precision, recall, F1-score), as well as Matplotlib and Seaborn for visualization. In addition, the model was integrated into a web-based interface using Flask and Flask-CORS to enable direct user interaction with the classification system.

3.6 CNN Model Evaluation

After the training is complete, the CNN model is evaluated on the test data using metrics such as accuracy, confusion matrix, precision, recall, and F1 score. These metrics help measure the model's performance in classification and analyze the differences in the results of the activation functions used. The model shown has the same hyperparameter values and kernel size. This ensures that the

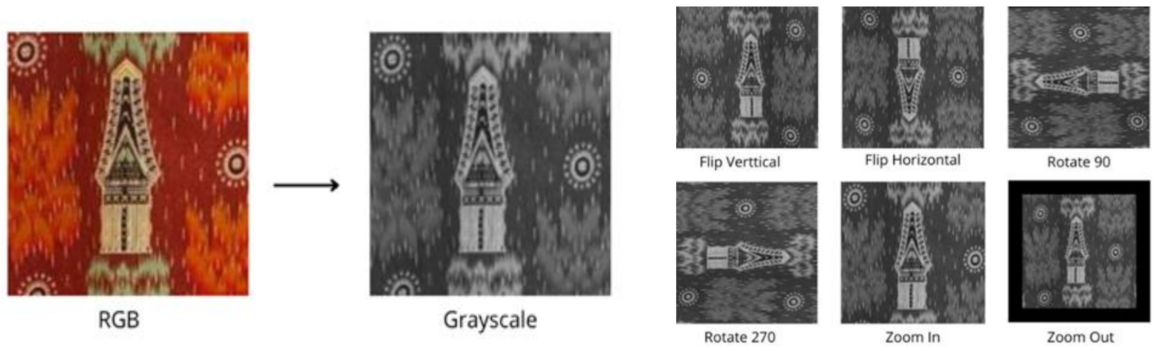
ISSN:2302-8149
e-ISSN:2540-9719

observed performance differences come from the activation function itself, not from factors like kernel size, number of filters, or other parameters.

4 Results and Analysis

4.1 Result

Based on the methodology used, the data preprocessing stage is essential to improve the quality and quantity of the dataset used. By combining the color dimensions into a single intensity channel, the transformation of RGB images into grayscale allows for more efficient analysis and without removing important information from the images. In addition, three data augmentation methods of zoom, flip, and rotate are used to increase the diversity of the learning data. These techniques proved to be effective in generating new data variations that can help the model in the learning process recognize patterns better. The use of three methods (zoom, flip, and rotate) in data augmentation techniques greatly affects the performance of the model. The zoom technique improves the model's ability to recognize objects at various size scales, allowing the model to recognize objects both near and far. Meanwhile, horizontal and vertical flip techniques change the perspective, enabling the model to recognize objects from various viewpoints. Rotation techniques that rotate the image in a certain angle help the model recognize objects in various orientations. The results of the RGB to grayscale conversion and augmentation results can be seen in Figures 3.



(a) (b)

Figure 3. (a) RGB to grayscale conversion results and (b) data augmentation results

Figure 4 shows the CNN Xception architecture used in this study for batik motif image classification. This architecture consists of three main parts: entry flow, middle flow, and exit flow. In the entry flow, there are two initial Conv2D layers with 32 and 64 filters, respectively, followed by an activation function and batch normalization. The middle flow includes eight depthwise separable convolution blocks with 728 filters, which form the core of the hierarchical feature extraction process. The exit flow then strengthens the visual features through two additional convolutional blocks up to 2048 filters. After the feature extraction process is complete, global average pooling is performed, followed by a dropout layer with a rate of 0.5, then a fully connected layer, and finally a Softmax output layer for classification into four classes of South Sulawesi batik motifs.

During the model testing phase, four activation functions ReLU, ELU, Leaky ReLU, and Swish were alternately applied to the hidden layer to evaluate the classification performance of batik images that had undergone preprocessing and augmentation. To maintain experimental fairness, all models were trained using uniform hyperparameters: a learning rate of 0.001, a batch size of 64, and 100 epochs. This standardization aims to ensure that any performance differences observed are solely due to variations in activation functions, not other parameters.

ISSN:2302-8149
e-ISSN:2540-9719

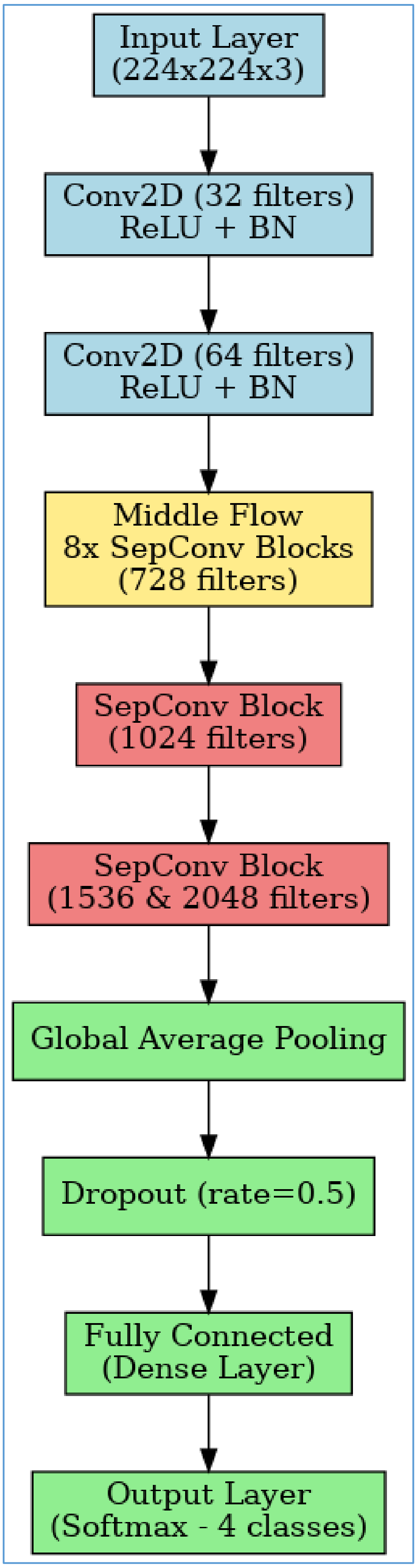


Figure 4. CNN Architecture

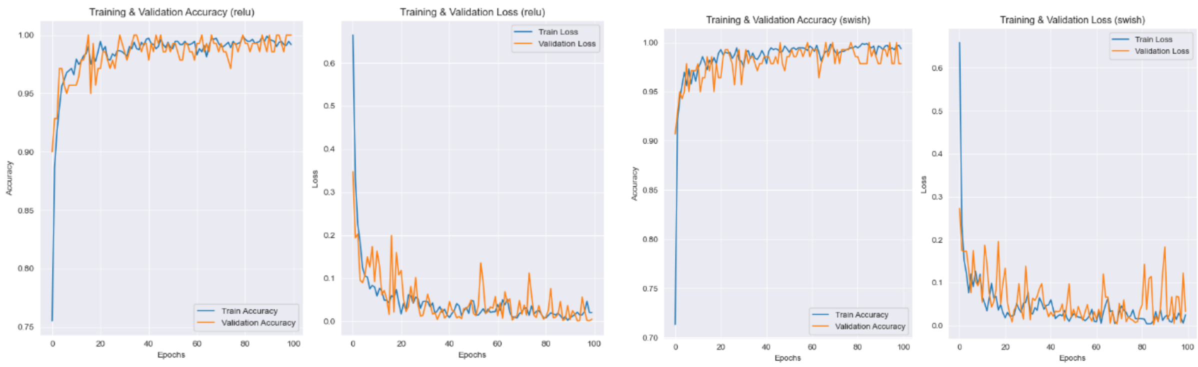
ISSN:2302-8149
e-ISSN:2540-9719

Figure 5 shows the training and validation results for models with ReLU, ELU, Leaky ReLU, and Swish activation functions over 100 epochs. The ReLU activation function is very efficient in the training process, resulting in fast convergence with almost ideal accuracy. ReLU has loss curves for training and validation that tend to be stable and parallel, indicating balanced learning and minimal overfitting. This makes it ideal for deep model architectures.

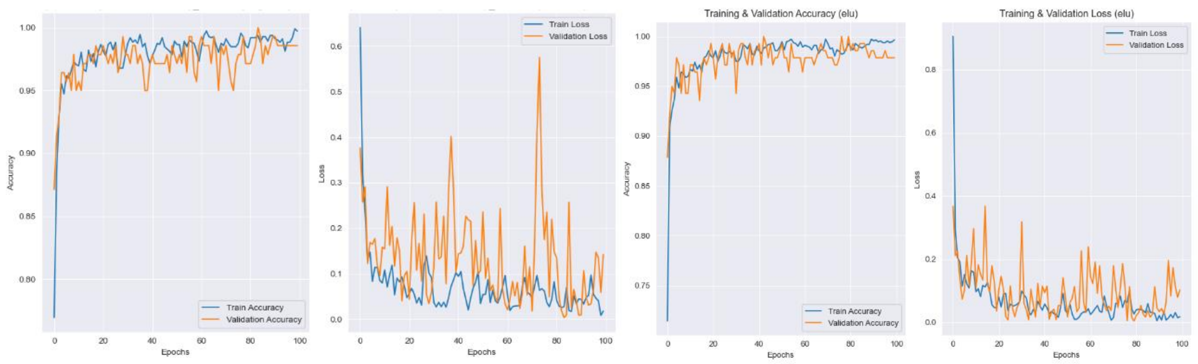
The ELU function helps the model achieve stable convergence by maintaining gradient stability, especially when the input values are negative. The accuracy curves of the ELU training and validation are consistent, with minor variations after achieving almost 100% accuracy for forty epochs. The loss curve shows a relatively steady decline despite slight variations, indicating that the ELU is able to maintain robust learning and reduce the possibility of overfitting.

By allowing gradient flow at negative input values, Leaky ReLU seems to offer an advantage in overcoming the dying problem of ReLU. Although the Leaky ReLU model has high training and validation accuracy, there are some large variations in the validation loss curves, indicating possible overfitting, although the overall performance remains good. Due to the efficient gradient flow-especially in deeper model architectures-the function is still useful for training stability.

As can be seen, Swish maintains a consistent trend of training and validation accuracy, accompanied by small fluctuations in the loss curve. Combining the linearity of ReLU with its smooth non-linear nature, this function provides smoother transitions in the output and improves learning stability. Although the loss curve shows slightly larger deviations compared to ReLU, it is still effective in maintaining a balance between training and validation. Overall, ReLU and Swish excel in maintaining model stability, while ELU shows strong convergence, and Leaky ReLU is fine with little risk of overfitting.



(a) **(b)**



(c) **(d)**

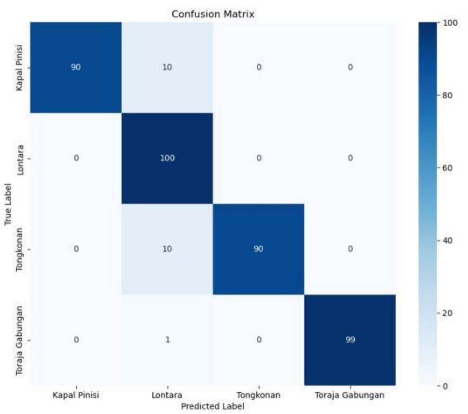
Figure 5. Activation function graph (a) ReLU, (b) ELU, (c) leaky ReLU and (d) swish

The analysis results show that the performance of the CNN model is significantly affected by the chosen activation function. Hyperparameter settings such as learning rate, batch size, and number of epochs are critical to achieving optimal performance. The right combination of activation function and hyperparameters can improve the stability and accuracy of the model while reducing the risk of overfitting. Therefore, to ensure that the model can adapt well to the data at hand and

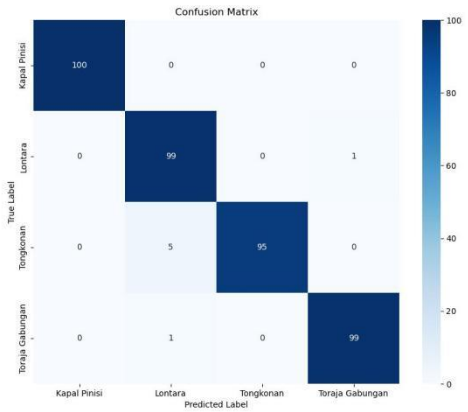
ISSN:2302-8149
e-ISSN:2540-9719

provide more consistent results, it is crucial to conduct thorough exploration and hyperparameter tuning.

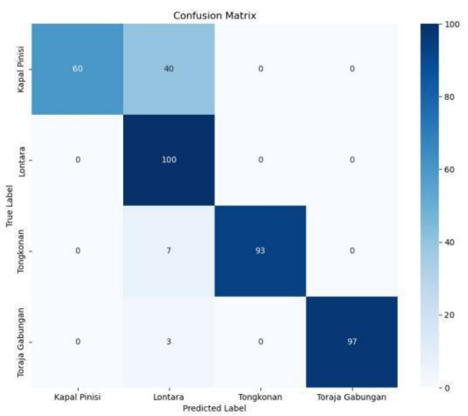
Figure 6 shows the results of the classification model evaluation through the confusion matrix for the four categories of South Sulawesi cultural heritage Pinisi Ship, Lontara, Tongkonan, and Toraja Gabungan. Based on the four experiments conducted, the model shows varying performance. In the first experiment, the model achieved perfect accuracy (100%) for the Pinisi Ship category, while the other categories achieved accuracy above 95%. However, in the second and third experiments, there was a significant decline in the Pinisi Ship classification with an accuracy of only 60-61%, where most of the misclassification occurred with the Lontara category. The Lontara category itself showed excellent performance consistency with 100% accuracy in the second and third experiments. In the fourth experiment, the model showed improvement for Pinisi Ship classification with an increase in accuracy to 90%, although there was still a 10% misclassification with the Lontara category. Overall, the results show that the model has major challenges in distinguishing the features between Pinisi and Lontara Vessels, while the classification for the Tongkonan and Combined Toraja categories is relatively stable with accuracy above 90%. This indicates the need for further optimization on the extraction of distinguishing features between Pinisi and Lontara Vessels to improve the overall performance of the model.



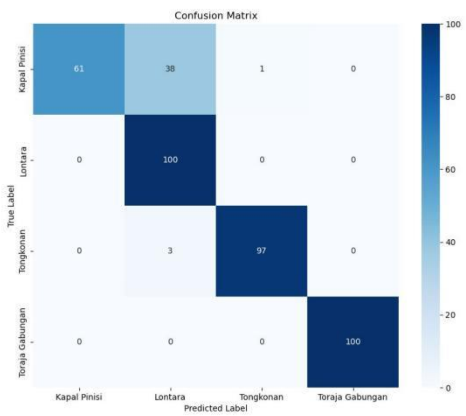
(a)



(b)



(c)



(d)

Figure 6. Confusion matrix (a) ReLU, (b) ELU, (c) leaky ReLU and (d) swish

In Figure 7, the Precision, Recall, F1 Score, and Accuracy metrics are presented; all activation functions demonstrate uniform results with values close to 1.0. This visualization provides insights into the relative performance of the activation functions. Consistently high values across these metrics indicate that the models effectively classify the data.

ISSN:2302-8149
e-ISSN:2540-9719

Furthermore, this consistency in performance suggests that the choice of activation function does not significantly impact the overall effectiveness of the model. This aspect can be further explored in Table 2, where detailed comparisons of each activation function are provided, highlighting their unique strengths and weaknesses in specific contexts, ultimately guiding future model selection and optimization strategies.

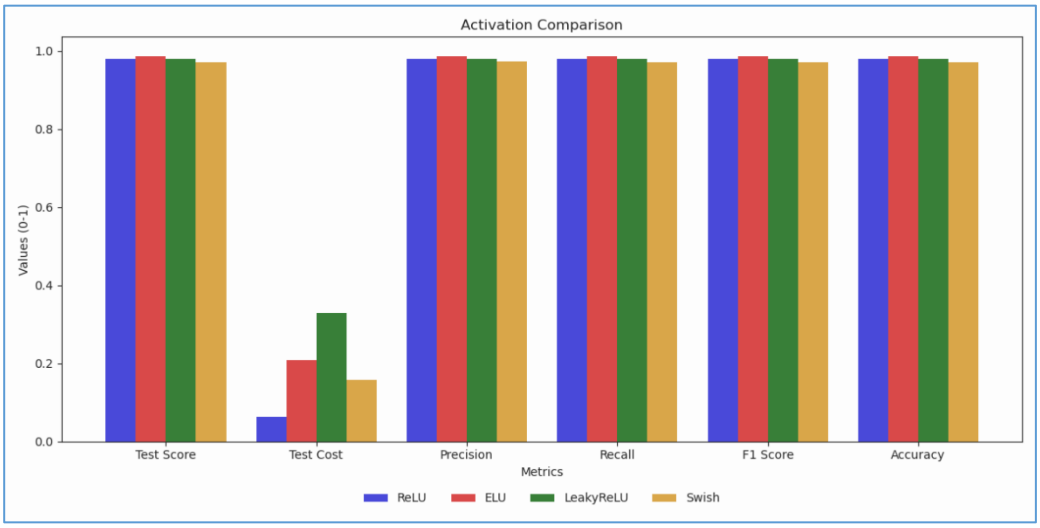


Figure 7. Comparison of ReLU, ELU, Leaky ReLU and swish activation optimization results

In Table 2, the varying performance of each activation function can be clearly observed through several important evaluation metrics, such as accuracy, precision, recall, specificity, and F1 score. All activation functions, namely ReLU, ELU, Leaky ReLU, and Swish, demonstrate excellent performance, achieving high precision, recall, and F1 scores, which collectively indicate the model's ability to classify images with minimal error. Although there are slight variations among the activation functions, overall, they consistently produce efficient results, exhibiting high accuracy and strong detection capabilities across all classes. This suggests that selecting the appropriate activation function can significantly enhance the performance of CNN models in various classification tasks, including the classification of batik motifs.

However, it should be noted that these results still need to be considered with other factors that affect model performance in batik classification using CNN, such as dataset size, data complexity, and other CNN parameter settings. Therefore, additional research is needed to refine and validate these findings by considering various important factors that affect model performance.

By understanding this relationship, we can optimize the use of CNNs in classification tasks, resulting in models that are not only consistent but also more robust in handling data complexity. Further research is needed to explore in-depth the influence of various combinations of these factors on model performance, thereby yielding more effective systems for classifying batik motifs.

Table 2. Results of activation function comparison

Activate	Test Score	Test Cost	Precision	Recall	F1 Score	Accuracy
ReLu	0.9786	0.0630	0.9791	0.9786	0.9785	97.86%
ELU	0.9857	0.2076	0.9864	0.9857	0.9857	98.57%
Leaky ReLu	0.9786	0.3295	0.9801	0.9786	0.9785	97.86%
Swish	0.9714	0.1575	0.9727	0.9714	0.9714	97.14%

4.2 Analysis

Based on the evaluation table, the difference in accuracy between the four activation functions shows that ELU provides the best performance with the highest accuracy reaching 98.57%. This indicates that ELU is capable of providing more accurate predictions compared to other activation functions. Despite having a relatively high Test Cost value (0.2076), ELU remains superior in almost all metrics, such as precision, recall, and F1-score, all of which demonstrate good prediction

ISSN:2302-8149
e-ISSN:2540-9719

consistency. Meanwhile, ReLU and Leaky ReLU have the same accuracy, which is 97.86%, but ReLU is more computationally efficient with a lower Test Cost value (0.0630 compared to 0.3295 for Leaky ReLU), making it a more efficient choice for system implementation. On the other hand, Swish shows the lowest accuracy of 97.14%, and while still reliable, it cannot match the performance of the other three functions.

In addition to activation functions, model performance is also influenced by hyperparameter settings, the number of epochs, and the characteristics of the dataset used. This study deliberately kept the training conditions consistent, using a learning rate of 0.001, a batch size of 64, and 100 epochs for all activation functions. The aim is to enable an objective and accurate comparison of performance and to ensure that the results truly reflect the differences in activation function performance, not the influence of other parameters. The use of a fixed number of epochs also ensures that each activation function has an equal opportunity to learn the data optimally.

The classification results are also greatly influenced by the diversity of features in the batik image dataset used. This dataset consists of 1,400 images with four classes of South Sulawesi batik motifs, namely Lontara, Tongkonan, Perahu Pinisi, and Toraja Gabungan, which have a high level of complexity in both pattern and texture. This complexity tests the model's ability to recognize the visual details of each motif. ELU's superior performance demonstrates its ability to capture more complex patterns—a crucial advantage in image classification tasks rich in visual details, such as batik motifs.

The implication of this research is that the selection of the appropriate activation function plays a crucial role in improving the accuracy, consistency, and efficiency of batik image classification systems. Activation functions cannot be chosen arbitrarily, as they can directly impact the model's ability to recognize the details and patterns characteristic of traditional batik. Therefore, in the development of deep learning-based batik classification systems, the selection of activation functions must be part of the strategic design process of CNN architecture. These findings can also be applied more broadly in the development of cultural education applications, cultural heritage digitization, or image-based tourism information systems, which aim to preserve and introduce the richness of batik motifs to the global community.

5 Conclusion

Based on the findings in this study, it can be concluded that the selection of activation functions greatly affects the performance of South Sulawesi batik motif image classification. The Xception architecture, as a development of Inception, has been proven to be able to extract complex visual features efficiently through the depthwise separable convolution mechanism. The advantage of this architecture in handling complex spatial information is the main reason for its selection in the construction of the CNN model in this study.

From the testing of four activation functions ReLU, ELU, Leaky ReLU, and Swish—it was found that ELU showed the best performance, with the highest accuracy reaching 98.57%. This advantage is supported by its ability to handle negative value distributions, which aids in learning complex patterns in batik motifs. ReLU and Leaky ReLU followed with an accuracy of 97.86%, while Swish had the lowest accuracy at 97.14%. The evaluation was conducted fairly by maintaining consistency in hyperparameters across all models and using a diverse dataset from four classes of batik motifs with high pattern complexity. These findings support the primary objective of the research, which is to evaluate the effectiveness of various activation functions in improving batik motif classification performance.

However, this study has several limitations. First, the experiments were only conducted on one CNN architecture, namely Xception, so the results do not represent possible performance variations in other architectures. Second, although the hyperparameters were standardized, no optimal parameter search was performed using techniques such as grid search or Bayesian optimization. Third, the number of batik classes is still limited to four motifs from South Sulawesi, which may affect the generalization of results to motifs from other regions. Additionally, image quality was controlled at a specific resolution (224x224), without testing on low-quality or damaged images commonly encountered in the real world.

 ISSN:2302-8149
e-ISSN:2540-9719

For further research development, it is recommended to explore adaptive activation mechanisms such as PReLU, RReLU, or dynamic functions like Swish and Mish, as well as attention-based activation layers or meta-learning optimized through reinforcement learning approaches. The use of hybrid architectures combining CNNs with Vision Transformers (ViT) also has the potential to enhance the ability to simultaneously capture local texture patterns and global style elements. Additionally, approaches such as Neural Architecture Search (NAS), domain-invariant regularization, and adversarial style transfer can improve model efficiency and robustness. Dataset variation should also be enhanced through synthetic data generation using GANs or diffusion models, as well as collaborative curation of underrepresented regional batik motifs, including low-quality real-world data.

Beyond technical aspects, future research can be directed toward real-world applications such as AI-based batik restoration, cultural heritage damage detection, and automatic design generation through conditional GANs, integrated with augmented reality (AR) for cultural education or e-commerce recommendation systems. Cross-disciplinary collaboration with cultural historians and batik artisans is also crucial to ensure that innovations in the classification of cultural motifs remain aligned with ethical preservation values and local wisdom.

Acknowledgement

The authors declare no conflicts of interest. This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. Support was provided in the form of research facilities by the Faculty of Computer Science at Universitas Handayani Makassar and Universitas Pembangunan Nasional "Veteran" Jawa Timur.

Reference

[1] B. J. Filia *et al.*, “*Improving Batik Pattern Classification using CNN with Advanced Augmentation and Oversampling on Imbalanced Dataset*,” in *Procedia Computer Science*, Elsevier B.V., 2023, pp. 508–517. doi: 10.1016/j.procs.2023.10.552.

[2] R. Febriani, L. Knippenberg, and N. Aarts, “*The Making of a National Icon: Narratives of Batik in Indonesia*,” *Cogent Arts Humanit*, Vol. 10, No. 1, 2023, doi: 10.1080/23311983.2023.2254042.

[3] S. Setianti, L. Gatra Hanafiani, V. Devianti, S. Wahyuni, J. W. Kusuma, and F. Khan, “*International Journal of Multidisciplinary Research and Literature Motif Tirtayasa, Kepandean and Kawangsan on Banten Batik in Mathematical Theory*,” *International Journal of Multidisciplinary Research and Literature IJOMRAL*, Vol. 1, No. 3, pp. 241–360, 2022, doi: 10.53067/ijomral.v1i3.

[4] Siti Aisyah, Rini Astuti, Fadhil M Basysyar, Odi Nurdiawan, and Irfan Ali, “*Convolutional Neural Networks for Classification Motives and the Effect of Image Dimensions*,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, Vol. 8, No. 1, pp. 181–188, Feb. 2024, doi: 10.29207/resti.v8i1.5623.

[5] A. Prayoga, Maimunah, P. Sukmasetya, Muhammad Resa Arif Yudianto, and Rofi Abul Hasani, “*Arsitektur Convolutional Neural Network* untuk Model Klasifikasi Citra Batik Yogyakarta,” *Journal of Applied Computer Science and Technology*, Vol. 4, No. 2, pp. 82–89, Nov. 2023, doi: 10.52158/jacost.v4i2.486.

[6] D. A. Anggoro, A. A. T. Marzuki, and W. Supriyanti, “*Classification of Solo Batik Patterns using Deep Learning Convolutional Neural Networks Algorithm*,” *Telkomnika (Telecommunication Computing Electronics and Control)*, Vol. 22, No. 1, pp. 232–240, Feb. 2024, doi: 10.12928/TELKOMNIKA.v22i1.24598.

[7] A. Prayoga, Maimunah, P. Sukmasetya, Muhammad Resa Arif Yudianto, and Rofi Abul Hasani, “*Arsitektur Convolutional Neural Network* untuk Model Klasifikasi Citra Batik Yogyakarta,” J. Appl. Comput. SCI. Technol., Vol. 4, No. 2, pp. 82–89, 2023, doi: 10.52158/jacost.v4i2.486

[8] S. A. Al-shami, R. Damayanti, H. Adil, F. Farhi, and A. Al mamun, “*Financial and Digital Financial Literacy Through Social Media use Towards Financial Inclusion among Batik Small*

ISSN:2302-8149
e-ISSN:2540-9719

Enterprises in Indonesia," *Heliyon*, Vol. 10, No. 15, Aug. 2024, doi: 10.1016/j.heliyon.2024.e34902.

[9] Nurilmiyanti Wardhani, B. E. W. Asrul, Antonius Riman Tampang, Sitti Zuhriyah, and Abdul Latief Arda, "*Classification of Toraja Wood Carving Motif Images using Convolutional Neural Network (CNN)*," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, Vol. 8, No. 4, pp. 486–495, Aug. 2024, doi: 10.29207/resti.v8i4.5897.

[10] D. Amalia, A. Rosdiana, N. Al Azizi, A. Wulandari, and U. Islam Nahdlatul Ulama Jepara, "Semiotika Batik Jepara sebagai Bentuk Identitas Budaya Lokal Masyarakat Jepara," *ENTITA: Jurnal Pendidikan Ilmu Pengetahuan Sosial dan Ilmu-Ilmu Sosial*, Vol. 6, No. 1, 2024, doi: 10.19105/ejpis.v5i2.12169.

[11] L. Hakim, H. R. Rahmanto, S. P. Kristanto, and D. Yusuf, "Klasifikasi Citra Motif Batik Banyuwangi menggunakan *Convolutional Neural Network*," Jurnal Teknoinfo, Vol. 17, No. 1, p. 203, 2023, doi.org/10.33365/jti.v17i1.2342

[12] D. Sinaga, C. Jatmoko, S. Suprayogi, and N. Hedriyanto, "*Multi-Layer Convolutional Neural Networks for Batik Image Classification*," *Scientific Journal of Informatics*, Vol. 11, No. 2, pp. 477–484, May 2024, doi: 10.15294/sji.v11i2.3309.

[13] M Mesran, Sitti Rachmawati Yahya, Fifto Nugroho, and Agus Perdana Windarto, "*Investigating the Impact of ReLU and Sigmoid Activation Functions on Animal Classification using CNN Models*," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, Vol. 8, No. 1, pp. 111–118, Feb. 2024, doi: 10.29207/resti.v8i1.5367.

[14] B. Ramadevi, V. R. Kasi, and K. Bingi, "*Fractional Ordering of Activation Functions for Neural Networks: A Case Study on Texas Wind Turbine*," *Eng Appl Artif Intell*, Vol. 127, p. 107308, Jan. 2024, doi: 10.1016/j.engappai.2023.107308.

[15] L. Nanni, S. Brahnam, M. Paci, and S. Ghidoni, "*Comparison of Different Convolutional Neural Network Activation Functions and Methods for Building Ensembles for Small to Midsize Medical Data Sets*," *Sensors*, Vol. 22, No. 16, Aug. 2022, doi: 10.3390/s22166129.

[16] D. G. T. Meranggi, N. Yudistira, and Y. A. Sari, "Batik *Classification using Convolutional Neural Network with Data Improvements*," International Journal on Informatics Visualization, Vol. 6, No. 1, pp. 6–11, 2022, doi: 10.30630/joiv.6.1.716

[17] M. Z. R. Azhary and A. Ritahani Ismail, "*Comparative Performance of Different Convolutional Neural Network Activation Functions on Image Classification*," *International Journal on Perceptive and Cognitive Computing*, Vol. 10, No. 2, pp. 118–122, Jul. 2024, doi: 10.31436/ijpcc.v10i2.490.

[18] A. Mumuni and F. Mumuni, "*Data Augmentation: A Comprehensive Survey of Modern Approaches*," Dec. 01, 2022, *Elsevier B.V.* doi: 10.1016/j.array.2022.100258.

[19] I. D. Susanti, S. Winarno, and J. Zeniarja, "*Yogyakarta Batik Image Classification based on Convolutional Neural Network*," *Advance Sustainable Science, Engineering and Technology*, Vol. 6, No. 1, Jan. 2024, doi: 10.26877/asset.v6i1.18002.

[20] K. Maharana, S. Mondal, and B. Nemade, "*A review: Data Pre-Processing and Data Augmentation Techniques*," *Global Transitions Proceedings*, Vol. 3, No. 1, pp. 91–99, Jun. 2022, doi: 10.1016/j.gltp.2022.04.020.

[21] Y. Brianorman and D. Utami, "*Comparative Analysis of CNN Architectures for SIBI Image Classification*," JUITA: Jurnal Informatika, Vol. 12, No. 1, pp. 61–70, May 2024, https://doi: 10.30595/juita.v12i1.20608

[22] F. Subeki and E. Eliyani, "Pengenalan Setengah Wajah menggunakan *Arsitektur Xception* pada Metode *Convolutional Neural Network*," J. Edukasi dan Penelit. Inform., Vol. 10, No. 2, p. 302, 2024, doi: 10.26418/jp.v10i2.73447

[23] P. Wang, Y. Cui, H. Tao, X. Xu, and S. Yang, "*Machining Parameter Optimization for a Batch Milling System using Multi-Task Deep Reinforcement Learning*," *J Manuf Syst*, Vol. 78, pp. 124–152, Feb. 2025, doi: 10.1016/j.jmsy.2024.11.013.

[24] S. Coulibaly, B. Kamsu-Foguem, D. Kamissoko, and D. Traore, "*Deep Convolution Neural Network Sharing for the Multi-Label Images Classification*," *Machine Learning with Applications*, Vol. 10, p. 100422, Dec. 2022, doi: 10.1016/j.mlwa.2022.100422.