









Comparative Evaluation of CycleGAN Variants for Neural Style Transfer: Translating Floral Images into Madurese Batik Motifs

Achmad Jauhari^{1*}, Arik Kurniawati¹, Firlir Irhamni², Wahyu Cahyanto Bayu Agung¹, Aeri Rachmad²,
Yuli Panca Asmara³

¹ Department of Informatics Engineering, Universitas Trunojoyo Madura, Bangkalan 69162, Indonesia

² Department of Information Systems, Universitas Trunojoyo Madura, Bangkalan 69162, Indonesia

³ Department of Mechanical Engineering, INTI International University, Nilai 71800, Malaysia

Corresponding Author Email: jauhari@trunojoyo.ac.id

Copyright: ©2026 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/mmep.130118>

ABSTRACT

Received: 9 October 2025

Revised: 19 December 2025

Accepted: 27 December 2025

Available online: 28 February 2026

Keywords:

neural style transfer, Cycle-Consistent Generative Adversarial Network, edge-aware self-attention, adaptive instance normalization, cultural heritage digitization, Madurese batik motifs

This study presents a systematic evaluation of three Cycle-Consistent Generative Adversarial Network (CycleGAN)-based neural style transfer frameworks: CycleGAN, Enhanced Spatial Attention CycleGAN (ESA-CycleGAN), and Adaptive Instance Normalization CycleGAN (AdaIN-CycleGAN) for translating floral images into Madurese batik motifs. A dataset comprising 360 images per domain was used for training, while five unseen floral images were reserved for testing. Although all models share the cycle-consistent adversarial learning paradigm, they differ in their mechanisms for encoding and transferring stylistic information. The baseline CycleGAN employs a ResNet-based generator architecture. ESA-CycleGAN integrates edge extraction and self-attention modules to enhance structural awareness. AdaIN-CycleGAN incorporates adaptive instance normalization to explicitly modulate style statistics during translation. Model performance was quantitatively evaluated using peak signal-to-noise ratio to assess structural preservation and learned perceptual image patch similarity to measure perceptual alignment. Experimental results indicate that ESA-CycleGAN achieves the most stable and consistent performance across test samples. The baseline CycleGAN exhibits higher variability and occasional blurring of fine contours, whereas AdaIN-CycleGAN enables flexible style modulation but may introduce structural distortions in floral forms. Qualitative visual assessment corroborates the quantitative findings, with ESA-CycleGAN producing clearer boundaries and more coherent stylization. Overall, ESA-CycleGAN demonstrates superior reliability for the translation of floral imagery into Madurese batik motifs. Future work will expand the evaluation dataset, incorporate user-based perceptual studies, and explore additional generative architectures to further support AI-assisted cultural heritage preservation.

1. INTRODUCTION

Batik has long been a part of people's daily life on Madura Island, Indonesia [1]. It is known for its vibrant colors, bold patterns, and the cultural stories associated with each motif [2, 3]. How do you make batik? The process of making batik is heavily dependent on the expertise and skill of local batik artisans. Said artisans must learned their craft through years of experience preparing dyes, arranging motifs that reflect the local identity, and other processes [4-6]. Most artisans still choose the traditional dyeing method, even if it has issues. The two most common issues are uneven color and failure to keep up with demand because the whole workflow depends on individual expertise [5, 7]. In recent years, many batik artists have reported that younger generations are no longer interested in pursuing this kind of career due to the time it consumes and the extensive apprenticeships required [8, 9].

Floral elements appear in many Indonesian batik traditions

and often serve as the starting point for the development of ornamental motifs [10, 11]. For instance, in Madurese batik, natural forms are simplified or exaggerated until they become patterns that local artisans recognize and use. Translating a flower's structure into a motif is not always straightforward; artisans must make many small artistic decisions, which often lead to variations in line thickness, continuity, or color transitions because everything is done by hand [5, 12]. As of recently, some artists and researchers have begun experimenting with neural style-transfer techniques to assist parts of this process. But actually, maintaining the distinct visual identity of traditional works is still difficult.

Neural style transfer (NST) is often used in newer digital batik projects to separate image content from visual appearance, allowing stylistic features like texture and color to be applied to different images. With this separation, textures, color schemes, and pattern characteristics typical of batik can be transferred onto many different types of images [13-16]. In

practice, most NST models rely on Visual Geometry Group (VGG) networks, and their results can vary considerably depending on which layers are used and how the content–style weights are set [17, 18]. Several studies have also pointed out that NST is quite sensitive to its hyperparameter choices and often produces distortions in batik motifs, especially in areas that contain fine lines, floral contours, or repetitive patterns [18]. Autoencoder-based improvements and layer modifications only partially solve these problems because NST is still unable to maintain optimal shape consistency and texture stability [16, 19]. Also, using NST in other places, like fashion design, ethnic motifs, and fractal transformation, tends to produce mixed results. Recurring visual artifacts and gradual loss of motif characteristic as design complexity increases have been reported to happened on previous studies [20-23].

Because of that limitation, recent research has shifted to Generative Adversarial Network (GAN)-based generative models to produce more stable batik styles. Previous studies show that GAN-based models perform well in tasks such as image reconstruction, feature transformation, and visual enhancement [24]. But in practice, GAN models still have some issues; they are prone to unstable training behavior and are sensitive to variations in the underlying data distribution. To address these issues within style-transfer tasks, CycleGAN introduces a cycle-consistency mechanism with two-way translation ($A \rightarrow B$ and $B \rightarrow A$), which helps the model maintain content structure because the generated image must be recoverable to its original form [25], for example, Madurese Batik to Balinese Batik [26]. Meanwhile, approaches such as SegCycle-SPADE use segmentation and a conditional GAN to extract and reconstruct traditional craft motifs [27], but do not combine floral content with Madurese batik style.

The cycle consistency feature of CycleGAN became the foundation for the development of advanced models. While the ESA-CycleGAN boosts contour preservation through edge extraction and self-attention mechanisms [28], the AdaIN-CycleGAN introduces flexibility with adaptive instance normalization that dynamically adjusts content and style statistics [29]. CycleGAN and its variants are commonly used for translating floral images into Madura batik patterns. The reason why is that the model learns two-way mappings between domains, and that helps preserve structural information that is hard to maintain with conventional NST.

In this context, the main difficulty lies in teaching the model to convert the shapes found in flowers (commonly used as references in Indonesian batik) into motifs that still resemble the traditional Madura design. The result images must have or retain the basic form of the flowers while having the typical Madurese batik color and visual.

In this experiment, we analyze three models: CycleGAN, Enhanced Spatial Attention CycleGAN (ESA-CycleGAN), and Adaptive Instance Normalization CycleGAN (AdaIN-CycleGAN). Because each model handles style transfer differently, all three are examined under the same floral inputs for a direct comparison. We focus on how much each model preserves the main shape of flowers after transformation into Madurese batik motifs and see if the outputs have any noticeable visual quality differences among them. Structural preservation is measured using peak signal-to-noise ratio (PSNR), but perceptual differences from the reference images are measured with learned perceptual image patch similarity (LPIPS).

2. METHODOLOGY

This study explores three CycleGAN versions: CycleGAN, ESA-CycleGAN, and AdaIN-CycleGAN. Each model translates floral images into patterns that resemble Madurese batik. The overall workflow of the experiment follows a simple sequence of steps. First, the dataset is assembled, and then the images are processed and normalized. Next, each model applies its own transformation method, and the outputs are evaluated using PSNR and LPIPS. Although the same images are used for all three models, each architecture handles style information differently, leading to distinct visual outcomes. Figure 1 provides an overview of this process.

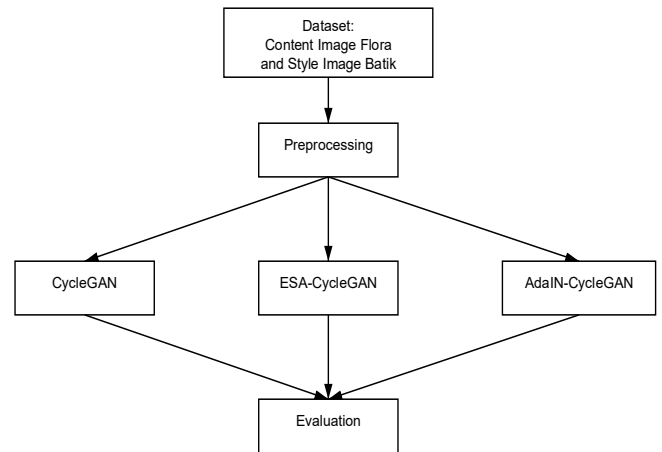


Figure 1. Proposed method for neural style transfer (NST) from flora images to Madurese batik

2.1 Dataset description and preprocessing

The dataset contains two unpaired image domains (Figure 2): floral photographs in Domain A and Madurese batik motifs in Domain B. We prepared 360 training images for each domain, along with five additional images for testing. One example from Domain A is a photograph of an orchid with petals showing a gradual color transition from pink to purple. In this image, the variation in the red and green channels is quite noticeable; the texture of the petals stays distinguishable, too. For Domain B, we use a Madurese batik motif with floral elements. The dominant colors are white, black, and maroon, combined with ‘isen’ pattern and bold outlines that are characteristic of beach-inspired Madurese batik. The corresponding histogram shows strong color contrasts, especially in the blue, red, and green channels.

We adjusted all images to a size of 256 by 256 pixels. Then converted them into Red–Green–Blue (RGB) format and scaled them to a value range of -1 to 1. No data augmentation was applied, so every image used in the experiment was exactly as collected. This preprocessing was done to keep the input format uniform across the models, which helps when comparing how each one translates the floral images into Madurese batik motifs.

2.2 Model-specific transformation modules

Each architecture model introduces different modules for handling style and content features.

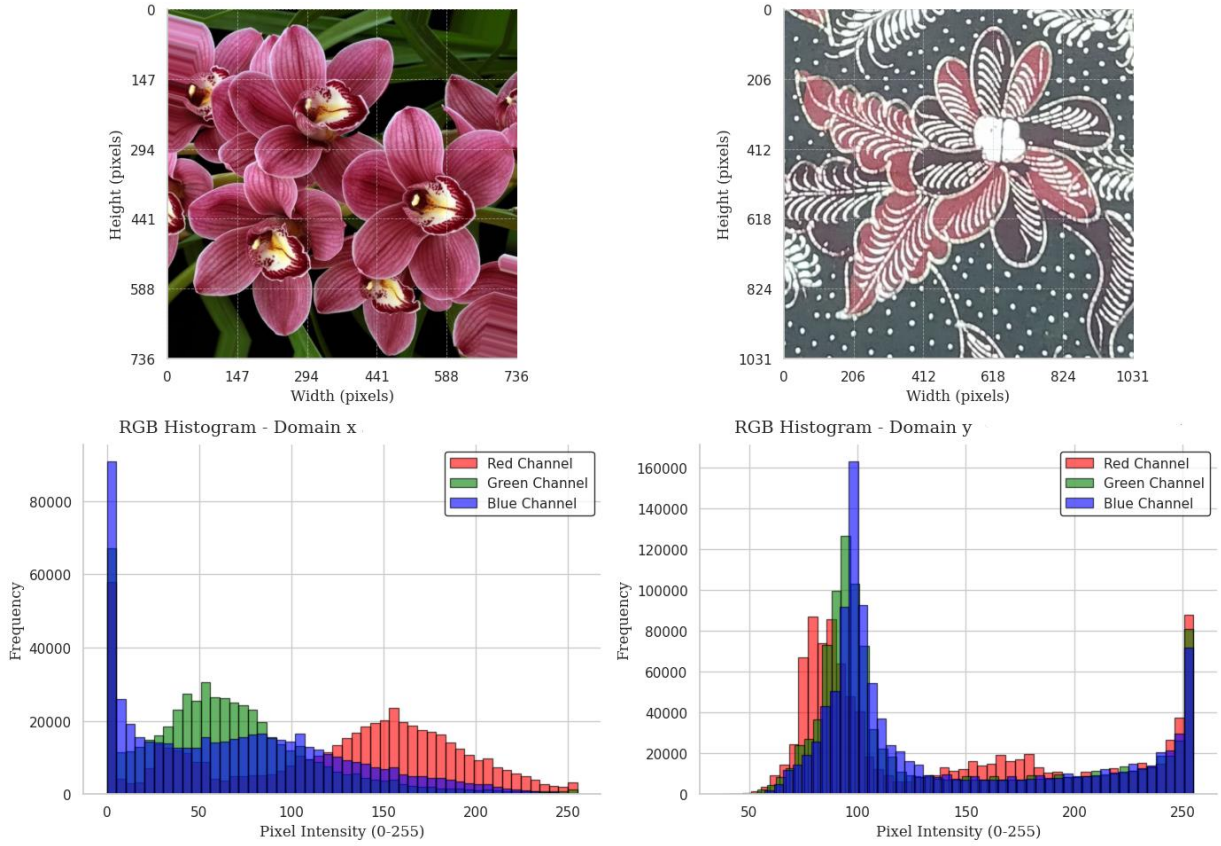


Figure 2. Sample images and Red–Green–Blue (RGB) histograms from the flora (Domain X) and batik (Domain Y) datasets

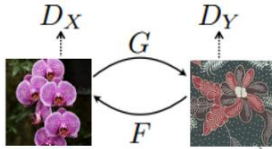


Figure 3. Cycle-Consistent Generative Adversarial Network (CycleGAN) flora and batik Madurese

2.2.1 Cycle-Consistent Generative Adversarial Network implementation

CycleGAN is a Generative Adversarial Network class designed for image-to-image translation tasks with unpaired data. Unlike conventional methods, such as Pix2Pix, which require paired input and output images, CycleGAN maps two domains separately by applying the cycle consistency theory. Its architecture consists of two generators, G and F , and two discriminators, D_X and D_Y . Generator G translates images in domain X (e.g., flower images) to domain Y (e.g., Madura batik design images), and generator F translates images in domain Y to domain X .

In both domains, both discriminators distinguish between original and generated images. Through the collaboration of the generators and discriminators, CycleGAN produces realistic target images with the same organizational structure as the original images, without using paired data. For instance, a flower image can be transformed into a Madura batik pattern, as shown in Figure 3, using data from both domains. To generate realistic images while preserving the original content structure, CycleGAN applies three loss functions: adversarial, cycle-consistency, and total [25].

1. Adversarial loss: It ensures that the discriminator cannot differentiate between the generated and original images from the generator. It is used to make the transformation results generated in the target domain

look realistic. The adversarial loss function is shown in Eq. (1).

$$L_{GAN}(G, D_Y, X, Y) = \mathbb{E}_{y \sim P_{data}(y)} [\log D_Y] + \mathbb{E}_{x \sim P_{data}(x)} [(1 - \log D_Y G(x))] \quad (1)$$

2. Cycle-consistency loss: This loss function ensures that the transformed images can return to their original form with minimal difference. Supposing a flora image is first changed into a batik pattern and then changed back, the final image should look quite similar to the original. Eq. (2) computes the cycle-consistency loss.

$$L_{cyc}(G, F) = \mathbb{E}_{x \sim P_{data}(x)} [\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim P_{data}(y)} [\|G(F(y)) - y\|_1] \quad (2)$$

3. The total loss function combines the two losses above with a balance parameter λ , which determines how strongly the model maintains cycle consistency compared with its ability to produce realistic images. The total loss function is computed using Eq. (3).

$$L(F, D_X, D_Y,) = L_{GAN}(G, D_Y, X, Y) + (F, D_X, Y, X) + \lambda L_{cyc}(G, F) \quad (3)$$

CycleGAN employs a ResNet-based generator with nine residual blocks arranged in an encoder–residual block–decoder structure (Table 1). This design allows the model to maintain the floral image structure while applying the batik style. The discriminator uses a 70×70 PatchGAN (shown in Table 2), which examines small patches of the image to ensure that the generated batik textures appear detailed and realistic, similar to traditional batik.

Table 1. Cycle-Consistent Generative Adversarial Network (CycleGAN) generator architecture used for flora-to-batik style translation

Block	Layer	Output Channels	Kernel
Input	Red-Green-Blue (RGB) image	3	-
Encoder	Conv2D + InstanceNorm + ReLU	64	7×7
	Conv2D + InstanceNorm + ReLU (Downsampling)	128	3×3
	Conv2D + InstanceNorm + ReLU (Downsampling)	256	3×3
Residual (9 blocks)	Conv2D -> InstanceNorm -> ReLU -> Conv2D -> InstanceNorm	256	3×3
	ConvTranspose2D (3×3 , stride = 2) + InstanceNorm + ReLU	128	3×3
Decoder	ConvTranspose2D (3×3 , stride = 2) + InstanceNorm + ReLU	64	3×3
Output	Conv2D (7×7) + Tanh	3	7×7

Table 2. CycleGAN and Adaptive Instance Normalization CycleGAN (AdaIN-CycleGAN) discriminator architectures used for flora-to-batik style translation

Block	Layer	Output Channels	Kernel /Stride
Conv Block 1	Conv2D + LeakyReLU	64	$4 \times 4 / 2$
Conv Block 2	Conv2D + InstanceNorm + LeakyReLU	128	$4 \times 4 / 2$
Conv Block 3	Conv2D + InstanceNorm + LeakyReLU	256	$4 \times 4 / 2$
Conv Block 4	Conv2D + InstanceNorm + LeakyReLU	512	$4 \times 4 / 1$
Output	Conv2D	1	$4 \times 4 / 1$

The model is optimized using least squares GAN loss, cycle-consistency loss to maintain structural fidelity, and an optional identity loss to preserve color statistics when necessary. Training is conducted for 50 epochs with a batch size of 1 using the Adam optimizer (learning rate 2×10^{-4} , $\beta_1 = 0.5$, $\beta_2 = 0.999$) and employs loss weights $\lambda_{\text{cycle}} = 10$ and $\lambda_{\text{identity}} = 10$.

2.2.2 Enhanced Spatial Attention CycleGAN implementation

Figure 4 shows the ESA-CycleGAN model, which was developed as an extension of CycleGAN to address the weaknesses in unpaired image-to-image translation [28]. These problems include loss of edge details, fine textures, and pattern consistency in the style transfer results. But this model is highly relevant for converting flora images into Madura Batik motifs. That is because the conversion process requires the preservation of the natural structure of the flora objects (e.g., the contour of flowers or leaves), while also being able to incorporate the detailed ornamental patterns characteristic of Madura Batik. The ESA-CycleGAN architecture is based on the CycleGAN structure, which comprises two generators (G and F) and two discriminators (D_X and D_Y). However, the ESA-CycleGAN architecture is enhanced with two additional modules: the edge extraction module and the self-attention module. The edge extraction module adds a Canny edge detection map to the input of the generator, assisting the network in identifying contours and significant features in the source image. It provides contour information that guides the generator in retaining the basic shape of the flora during the transformation process. While the self-attention module is incorporated into the generator and discriminator to model dependencies between spatially distant regions of the image. When combined, they make the ESA-CycleGAN able to learn global color and texture and, on top of that, maintain precise spatial detail, resulting in the generated batik motif often appearing more harmonious throughout the image.

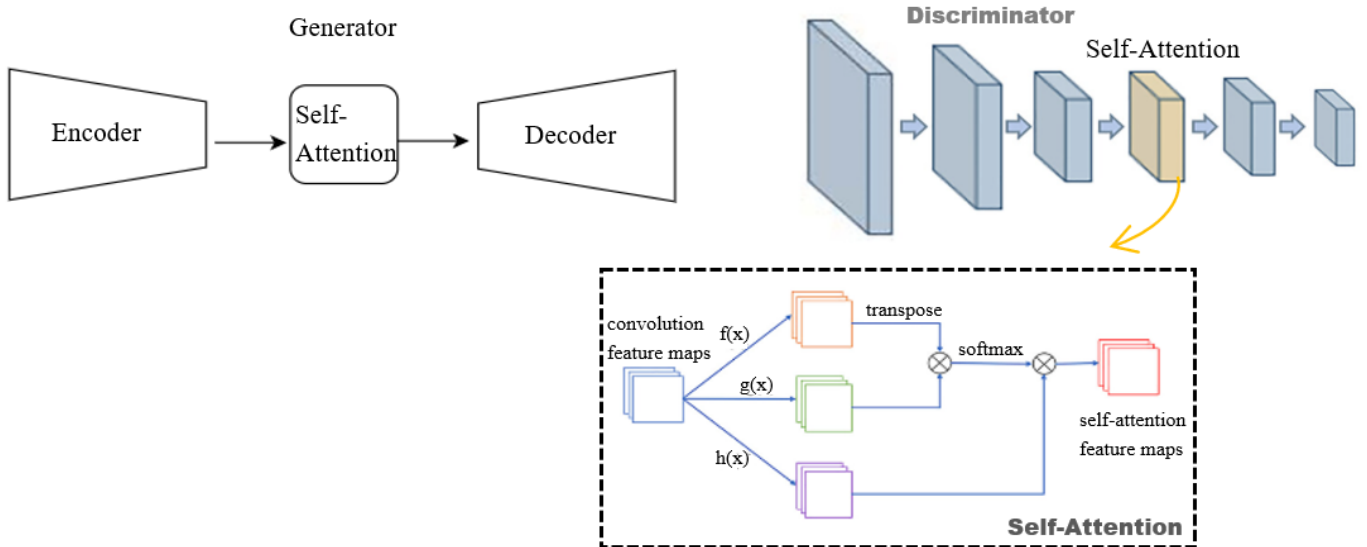


Figure 4. Enhanced Spatial Attention CycleGAN (ESA-CycleGAN) flora and batik Madurese

This makes ESA-CycleGAN an effective architecture for producing richly ornamented Madura batik motifs while retaining the original flora's natural character.

The Self-Attention Mechanism shows that the feature maps have three channels. The corresponding feature maps are produced by convolving the input with three 1×1

convolutional kernels for each branch. The feature map produced from the channel is transposed and calculated, subsequently undergoing matrix multiplication with the channel to yield a matrix resembling the covariance matrix, from which pixel correlation and similarity are evaluated. Thus, the attention feature map, particularly the feature

weights, is obtained from the Softmax output. Ultimately, it is multiplied pixel by pixel with the channel’s feature map to obtain the attention-weighted feature map, computed as Eq. (4).

$$O_j = \sum_{i=1}^N \frac{\exp(f(x_i)^T g(x_j))}{\sum_{i=1}^N \exp(f(x_i)^T g(x_j))} \times h(x_i) \quad (4)$$

- $f(x_i)$: the result of a 1×1 convolution on the input features to generate the query vector at position i .
- $g(x_j)$: the result of a 1×1 convolution on the input features to generate the key vector at position j .
- $h(x_i)$: the result of a 1×1 convolution on the input features to generate the value vector at position i .
- $\exp(f(x_i)^T g(x_j))$: computes the similarity between positions i and j using the dot product.
- The numerator and denominator form a Softmax function, ensuring that the attention weights at each position are positive and normalized.
- O_j : the final output of the attention aggregation process, representing a new feature representation at position j after considering the contributions of all positions i in the image.

Table 3. Enhanced Spatial Attention CycleGAN (ESA-CycleGAN) generator architecture used for flora-to-batik style translation

Block	Layer	Output Channels	Kernel
Input	Canny Edge Detection	4 (3 RGB + 1 edge)	-
	Concatenated with the RGB image (RGB + Edge)		
Encoder	Conv2D + InstanceNorm + ReLU	64	7×7
	Conv2D + InstanceNorm + ReLU (Downsampling)	128	3×3
	Conv2D + InstanceNorm + ReLU (Downsampling)	256	3×3
Residual (9 blocks)	Conv2D -> InstanceNorm -> ReLU -> Conv2D -> InstanceNorm	256	3×3
	Self-attention module	256	-
Decoder	ConvTranspose2D (3×3 , stride=2) + InstanceNorm + ReLU	128	3×3
	ConvTranspose2D (3×3 , stride=2) + InstanceNorm + ReLU	64	3×3
Output	Conv2D (7×7) + Tanh	3	7×7

ESA-CycleGAN expands the original CycleGAN model by adding two essential components to further reinforce structural preservation in its style translation: a canny-based edge extraction module and a self-attention module. The generator maintains the nine-residual-block ResNet-based encoder-residual-decoder structure, as summarized in Table 3, with the addition of self-attention after downsampling to capture long-range dependencies and preserve global motifs’ coherence. It also takes a concatenated edge map in addition to the RGB input to strengthen its retention of fine structural boundaries, such as the contours of the petals and ornamental edges. Meanwhile, the discriminator is based on a modified 70×70

PatchGAN architecture and has added a self-attention layer before the last convolutional block; see Table 4 for the improved sensitivity toward spatial relations and micro-texture realism typical of Madurese batik.

Training is done with least squares GAN loss to encourage stable adversarial learning; it combines a cycle-consistency loss ($\lambda_{\text{cycle}} = 10$), to ensure structural fidelity and an identity loss ($\lambda_{\text{identity}} = 10$), to stabilize color consistency. The model was trained for 50 epochs using the Adam optimizer (learning rate 2×10^{-4} , $\beta_1 = 0.5$, $\beta_2 = 0.999$) with a batch size of 4.

Table 4. ESA-CycleGAN discriminator architecture used for flora-to-batik style translation

Block	Layer	Output Channels	Kernel /Stride
Conv Block 1	Conv2D + LeakyReLU	64	$4 \times 4 / 2$
Conv Block 2	Conv2D + InstanceNorm + LeakyReLU	128	$4 \times 4 / 2$
Conv Block 3	Conv2D + InstanceNorm + LeakyReLU	256	$4 \times 4 / 2$
Self-attention	Query-Key-Value Attention	256	-
Conv Block 4	Conv2D + InstanceNorm + LeakyReLU	512	$4 \times 4 / 1$
Output	Conv2D	1	$4 \times 4 / 1$

2.2.3 Adaptive Instance Normalization CycleGAN implementation

AdaIN-CycleGAN is an extension of the CycleGAN architecture (Figure 5). This model enhances the generator by integrating the AdaIN mechanism directly into its residual blocks [29, 30]. Embedding AdaIN within the residual blocks allows the network to adaptively control style statistics (modifying the mean and variance of feature maps) while preserving the essential content structure.

This model relies on the AdaIN-CycleGAN architecture. That is because it works on two main ideas. The first is cycle consistency, which requires the network to recover the original image after a round-trip translation between domains. Second, AdaIN is used to align content features with style characteristics in a controlled manner. By combining cycle consistency and AdaIN, the model performs unpaired image-to-image translation that smoothly applies stylistic elements without losing the important structural details. For translating tasks with a large visual difference between source and target domains, this design is very suitable. For example, this approach can convert flora images into Madura Batik motifs.

AdaIN is analogous to style transfer. It adjusts the normalized channel-wise mean and variance of the content image to match those of the styled image, ensuring that the output image exhibits the same feature distribution as the ink painting. AdaIN lacks any trainable affine parameters. The supplied style image adaptively creates affine parameters. If x and y denote the feature maps of the content and style images, respectively, the AdaIN layer is as in Eq. (5):

$$AdaIN(x, y) = \sigma(y) \times \left(\frac{x - \mu(x)}{\sigma(x)} \right) + \mu(y) \quad (5)$$

- x : Feature from the content image (for example, a flora image).

- y : Feature from the style image (for example, a Madurese batik motif).
- $\mu(x)$: The mean value of the content feature x .
- $\sigma(x)$: The standard deviation of the content feature x .
- $\mu(y)$: The mean value of the style feature y .
- $\sigma(y)$: The standard deviation of the style feature y .

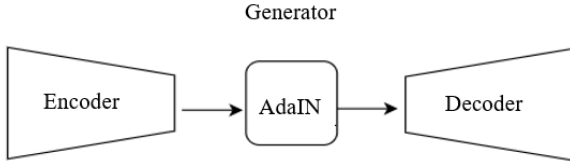


Figure 5. Adaptive Instance Normalization CycleGAN (AdaIN-CycleGAN) flora and Madurese batik

AdaIN-CycleGAN is an adaptation of the CycleGAN framework that integrates AdaIN to provide a more flexible and adaptive style control during floral-to-batik translation. Similar to the baseline architecture, the generator uses a ResNet-based encoder-residual-decoder structure with nine residual blocks, as summarized in Table 5. However, AdaIN-CycleGAN inserts AdaIN layers inside the residual block instead of the standard residual blocks, allowing the mean and variance of the feature maps to be dynamically aligned with the target batik style. This mechanism has facilitated smoother and more controlled style blending, especially in the case of floral textures combined with the color and ornamentation patterns of Madurese batik. As summarized in Table 2, the discriminator uses a similar PatchGAN-based architecture taken from CycleGAN, which evaluates local texture patches to ensure that the generated batik motifs preserve realistic micro-structural details.

Table 5. AdaIN-CycleGAN generator architecture used for flora-to-batik style translation

Block	Layer	Output Channels	Kernel
Input	Canny edge detection concatenated with the RGB image (RGB + edge)	4 (3 RGB + 1 edge)	-
Encoder	Conv2D + InstanceNorm + ReLU	64	7×7
	Conv2D + InstanceNorm + ReLU (Downsampling)	128	3×3
Encoder	Conv2D + InstanceNorm + ReLU (Downsampling)	256	3×3
	The encoder captures basic patterns from the input and gradually reduces the image resolution to learn deeper and more abstract features.		
AdaIN	ResidualBlock+AdaIN	256	3×3
Decoder	ConvTranspose2D (3×3 , stride = 2) + InstanceNorm + ReLU	128	3×3
	ConvTranspose2D (3×3 , stride = 2) + InstanceNorm + ReLU	64	3×3
Output	Conv2D (7×7) + Tanh	3	7×7

Training is set up similarly to CycleGAN with least squares GAN loss for stable optimization and cycle-consistency loss to preserve structural integrity during the bidirectional translation. Accordingly, AdaIN-CycleGAN was trained, according to the training script, for 50 epochs using the Adam optimizer (learning rate 2×10^{-4} , $\beta_1 = 0.5$, $\beta_2 = 0.999$) with a batch size of 1, and only cycle-consistency loss without any identity loss was used. These settings ensure stable adversarial learning while still allowing AdaIN to more adaptively modulate style statistics during the floral-to-batik translation.

Table 6. Comparison of CycleGAN, ESA-CycleGAN, and AdaIN-CycleGAN architectures and hyperparameters

Aspect	CycleGAN	ESA-CycleGAN	AdaIN-CycleGAN
Input channels	RGB (3 channels)	RGB + canny edge map (4 channels)	RGB only (3 channels)
Generator structure	Encoder \rightarrow 9 Residual Blocks \rightarrow Decoder	Encoder \rightarrow 9 Residual blocks \rightarrow Self-attention \rightarrow Decoder	Encoder \rightarrow AdaIN-Residual Blocks \rightarrow Decoder
Residual block type	Standard ResNet block with the InstanceNorm	Standard ResNet block with the InstanceNorm	Residual block with AdaIN + InstanceNorm
Normalization method	InstanceNorm	InstanceNorm	Adaptive Instance Normalization (AdaIN) and InstanceNorm
The style modulation method	None	Edge + self-attention enhances the structure	AdaIN aligns the mean/variance of content and style features
Texture preservation strategy	Relies on residual blocks	Edge guidance + attention preserve the fine structure	AdaIN preserves semantic structure through statistical alignment
Edge extraction module	None	Canny edge detection (concatenated before the encoder)	None
Self-attention (Generator)	No	Yes (after downsampling)	No
Self-attention (Discriminator)	No	Yes (before the final conv block)	No
Discriminator type	70×70 PatchGAN	70×70 PatchGAN + self-attention	70×70 PatchGAN
GAN loss	Least squares GAN	Least squares GAN	Least squares GAN
Cycle loss (λ_{cycle})	10	10	10
Identity loss ($\lambda_{identity}$)	Yes ($\lambda = 10$)	Yes ($\lambda = 10$)	None
Additional loss	None	Edge consistency loss	None
Epochs	50	50	50
Batch size	1	1	1
Learning rate	2×10^{-4}	2×10^{-4}	2×10^{-4}
Optimizer	Adam ($\beta_1 = 0.5$, $\beta_2 = 0.999$)	Adam ($\beta_1 = 0.5$, $\beta_2 = 0.999$)	Adam ($\beta_1 = 0.5$, $\beta_2 = 0.999$)

Table 6 provides an overview of the differences among CycleGAN, ESA-CycleGAN, and AdaIN-CycleGAN. The table includes information on the input setup, normalization layers, types of residual blocks, and the addition of attention or edge features. Table 6 also shows the loss functions and main hyperparameters used for training. When compared, these features clarify the differences among the three models. Comparing the results, the effects of these adjustments on the conversion of floral images into Madurese batik motifs can be seen.

2.3 Evaluation metrics

The PSNR is used to determine the amount of the original structure retained in the output, while the LPIPS examines the visual changes introduced during the style transfer. In many NST cases, a very high PSNR indicates that the result has not moved far enough from the input image, so the style influence is weak. When the PSNR becomes too low, the underlying content can start to degrade. LPIPS usually increases after the style is applied, although very high values can indicate that the image has developed artifacts or looks off. Considering both measurements helps explain how each model balances content preservation with stylistic changes.

3. RESULTS AND DISCUSSIONS

This study uses orchid photographs as the content images for transferring Madurese batik styles. Figure 6 shows one example from the test set. Orchids were chosen because they grow in many parts of Indonesia and show a wide variety of shapes and colors, which often influence batik artwork. Their petals and natural markings also provide the level of detail required to thoroughly test the models. Several Madurese batik motifs were included for the style images, representing the intricate designs common in coastal Madura.

The following subsection compares the results of the models using both PSNR and LPIPS, as well as visual examples, to see how they handle different floral inputs. These two types of evaluation help show the differences in how each model applies the batik style to the floral images. Figures 6-11 and Tables 7-9 provide examples of these outputs.

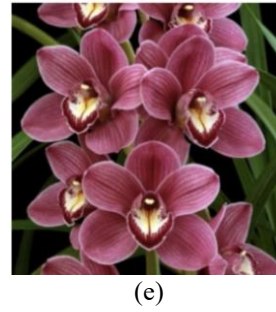
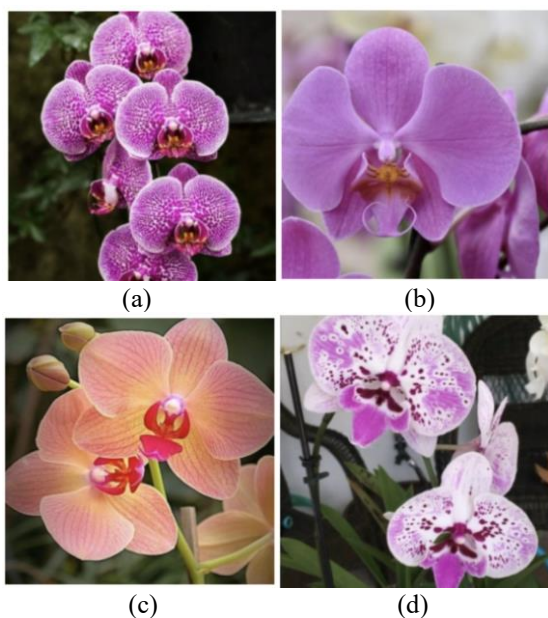


Figure 6. Flora image as content image: (a) *Phalaenopsis amabilis* ‘15’; (b) *Phalaenopsis amabilis* ‘1’; (c) *Phalaenopsis amabilis* ‘8’; (d) *Phalaenopsis amabilis* ‘6’; (e) *Cymbidium Clarisse Austin* ‘23’

3.1 Results of the quantitative evaluation

As shown in Table 7, the PSNR values indicate that ESA-CycleGAN produces the most stable results across all samples, indicating that it best preserves the flora structure. CycleGAN produces more varied and lower PSNR values, whereas AdaIN-CycleGAN produces values in the middle range but with less consistency. Because the PSNR reflects structural fidelity, ESA-CycleGAN provides the best content preservation.

Table 7. Comparison of the PSNR of CycleGAN, ESA-CycleGAN, and AdaIN-CycleGAN on five orchid images

Content Image	CycleGAN	PSNR	
		ESA-CycleGAN	AdaIN-CycleGAN
<i>Phalaenopsis amabilis</i> ‘15’ (PA 15)	4.757291	5.16	4.51
<i>Phalaenopsis amabilis</i> ‘1’ (PA 1)	9.125671	7.42	6.86
<i>Phalaenopsis amabilis</i> ‘8’ (PA 8)	6.876127	7.91	7.87
<i>Phalaenopsis amabilis</i> ‘6’ (PA 6)	4.609799	5.09	4.87
<i>Cymbidium Clarisse Austin</i> ‘23’ (CA 23)	6.952485	6.86	6.40

Note: PSNR: peak signal-to-noise ratio; LPIPS: learned perceptual image patch similarity

Table 8. Comparison of LPIPS of CycleGAN, ESA-CycleGAN, and AdaIN-CycleGAN on five orchid images

Content Image	CycleGAN	LPIPS	
		ESA-CycleGAN	AdaIN-CycleGAN
<i>Phalaenopsis amabilis</i> ‘15’ (PA 15)	0.701729	0.683	0.679
<i>Phalaenopsis amabilis</i> ‘1’ (PA 1)	0.692705	0.715	0.705
<i>Phalaenopsis amabilis</i> ‘8’ (PA 8)	0.518543	0.697	0.665
<i>Phalaenopsis amabilis</i> ‘6’ (PA 6)	0.516275	0.660	0.610
<i>Cymbidium Clarisse Austin</i> ‘23’ (CA 23)	0.469227	0.630	0.760

Table 8 shows that CycleGAN produces low and unstable LPIPS values, whereas ESA-CycleGAN produces moderate and consistent values. AdaIN-CycleGAN sometimes produces high values, indicating perceptual loss. Since LPIPS measures

visual similarity, the stable values of ESA-CycleGAN indicate a more controlled style application process that does not excessively alter the original form.

The best balance is achieved when the PSNR is neither too high nor too low, and the LPIPS increases but remains stable. This pattern shows that ESA-CycleGAN is most capable of maintaining the flora structure while applying the batik style, resulting in a level of visual change that remains recognizable. Thus, it is the most balanced model among the three.

3.2 Model-specific analysis

Figure 7 shows that CycleGAN produces different results across the test samples. The PSNR values ranged from 4.75 dB (*Phalaenopsis amabilis* ‘15’) to 9.12 dB (*Phalaenopsis amabilis* ‘1’). Some structural information is kept at the higher end, but in many cases, the content is not preserved well. The LPIPS values also varied widely, from 0.469 (*Cymbidium Clarisse Austin* ‘23’) to 0.701 (*Phalaenopsis amabilis* ‘15’). When orchid images contain complicated petal shapes, the translated versions often lose noticeable details.

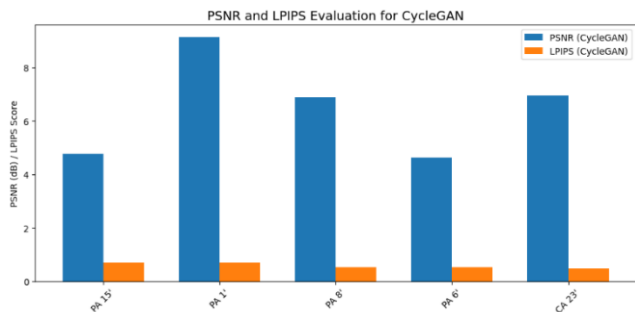


Figure 7. PSNR and LPIPS evaluation scores for CycleGAN across different orchid image samples

Note: PSNR: peak signal-to-noise ratio; LPIPS: learned perceptual image patch similarity

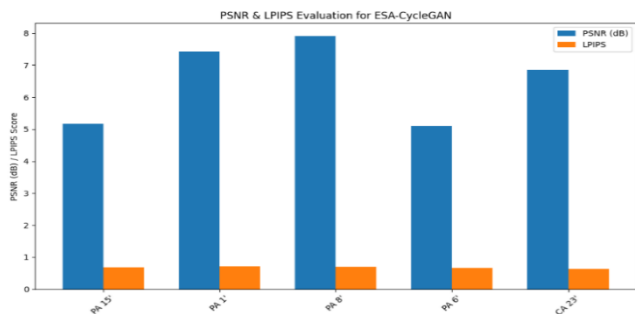


Figure 8. PSNR and LPIPS evaluation scores for ESA-CycleGAN across orchid image samples

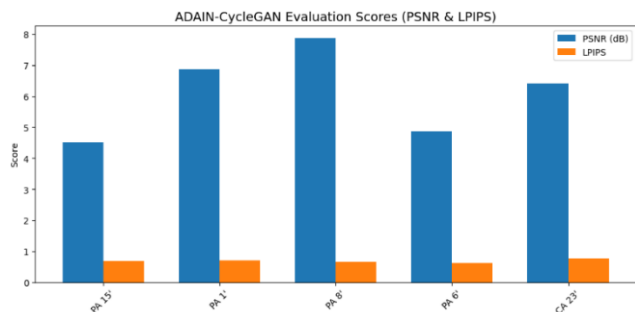


Figure 9. PSNR and LPIPS evaluation scores for AdaIN-CycleGAN across different orchid image samples

Figure 8 presents the results of the ESA-CycleGAN. The PSNR values fell within a tighter range, between 5.09 and 7.91 dB, and the LPIPS values remained fairly steady at 0.471–0.715. The edge-based features and attention layer help maintain the outlines and smaller textures in the flowers. Consequently, the generated batik images look cleaner and tend to follow the original floral shapes more closely. This trend is visible in many samples.

Figure 9 displays the output of AdaIN-CycleGAN. Here, the PSNR values fall between 4.51 dB (*Phalaenopsis amabilis* ‘15’) and 7.87 dB (*Phalaenopsis amabilis* ‘8’), and in some cases, the LPIPS score reaches 0.760 (*Cymbidium Clarisse Austin* ‘23’). The adaptive normalization gives the model more flexibility in adjusting the style, but it can also make the output sensitive to input feature variations. Artifacts may appear when this happens, which is not ideal for Madurese batik motifs that require a clear structure and consistent form.

3.3 Visual comparison and interpretation

Table 9 compares the visual output of CycleGAN, ESA-CycleGAN, and AdaIN-CycleGAN. In several cases, CycleGAN produces slightly blurred images with unclear contour lines that are not as clear as expected. This can be seen in samples such as *Phalaenopsis amabilis* ‘15’ and *Phalaenopsis amabilis* ‘6’, where the petal edges start to blend into the surrounding batik pattern. These visual issues are consistent with the observed lower PSNR scores for both images.

Table 9. Visual comparison of the style transfer results across the three models

Content Image	CycleGAN	ESA-CycleGAN	AdaIN-CycleGAN

In ESA-CycleGAN, the contours of the petals are better preserved, and the batik patterns are neatly arranged. The coloring result also appears to be more balanced. These results are reflected in the stable PSNR and LPIPS values in Tables 7

and 8, indicating that ESA-CycleGAN is the most effective method for the preservation of flora structure while applying the Madura batik style. While AdaIN-CycleGAN has smooth color blending, some of its samples, such as *Cymbidium Larisse Austin '23'*, show distortion in shape and overly dominant batik patterns. This is supported by the higher LPIPS values in a few samples in Table 8.

Overall, ESA-CycleGAN has the best balance between structural sharpness and stylization quality (Table 9). CycleGAN loses the contour details more easily; AdaIN-CycleGAN allows for flexible style but becomes less stable while maintaining flower morphology.

3.4 Discussion

Figures 10 and 11 show a comparison of PSNR and LPIPS for the three models on five orchid samples. These results help identify each model's strengths and limitations in preserving structure when the Madura batik style is applied. However, it often fails to maintain fine details such as petal contours and small patterns. This is evident from the low PSNR values in several samples and the inconsistent LPIPS values in Figures 10 and 11. The ESA-CycleGAN performs best in preserving the structure and texture. Petal contours are clearer, and the batik style is applied neatly. The graphs in Figures 10 and 11 show the stable values in almost all the samples. The main drawback of this model is its higher computational requirements owing to the use of edge extraction and self-attention.

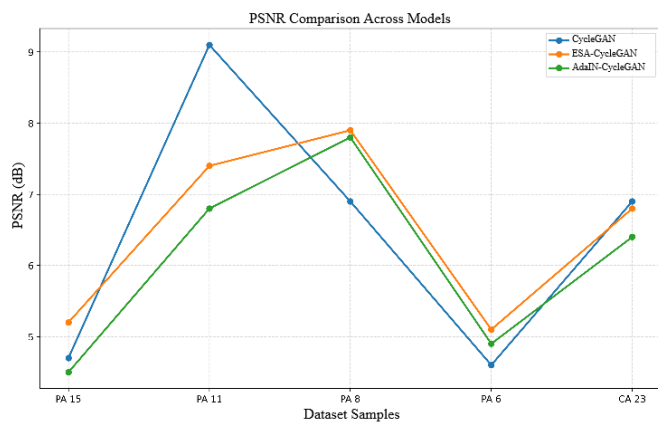


Figure 10. Comparison of the PSNRs of CycleGAN, ESA-CycleGAN, and AdaIN-CycleGAN across different orchid samples

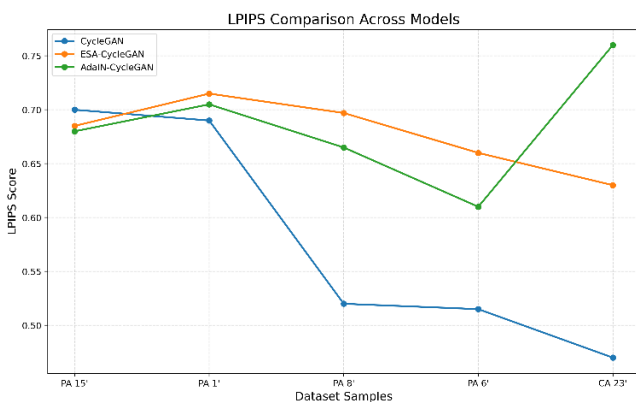


Figure 11. Comparison of LPIPS among CycleGAN, ESA-CycleGAN, and AdaIN-CycleGAN across different orchid samples

AdaIN-CycleGAN demonstrates more flexible style adaptation capabilities. This model can produce interesting visual variations; however, shape distortion is observed in some images. This can be seen in Figure 11's increased LPIPS values of certain samples. The model's sensitivity to feature variations makes it less stable for structural accuracy-requiring batik motifs.

4. CONCLUSION

This study examined how CycleGAN, ESA-CycleGAN, and AdaIN-CycleGAN transform floral images into Madurese batik motifs. From the tests we ran, ESA-CycleGAN generally produced the most coherent and culturally recognizable patterns, particularly in keeping the outlines of the flowers intact while still introducing batik-style elements. Although CycleGAN handled simpler images reasonably well, it tended to lose finer textures. AdaIN-CycleGAN often generated interesting color and style variations, although the shapes of the flowers became less clear at times.

The PSNR and LPIPS results were mostly in line with what we observed visually. ESA-CycleGAN showed steadier scores across samples, indicating that the combination of edge extraction and self-attention helps preserve structure. Both CycleGAN and AdaIN-CycleGAN displayed wider fluctuations in the metrics, especially on images with more complicated floral features.

Several things are worth exploring next. Feedback from batik artisans would be valuable in understanding how the generated motifs align with traditional expectations. A larger dataset with more diverse floral shapes and different regional batik styles could also help reveal where each model performs best or struggles. Attempting other generative models may also be useful to address the remaining limitations.

Overall, while the present results are encouraging, the work is still in its early stages. These methods could eventually become a supporting tool for documenting or experimenting with batik designs, but their role should complement rather than replace existing artistic practices.

ACKNOWLEDGMENT

This research was financially supported by DPPM, Risbang, Kemdiktisaintek, Republic of Indonesia (Grant No.: 120/C3/DT.05.00/PL/2025; B/022/UN46.1/PT.01.03/BIMA/PL/2025).

REFERENCES

- [1] Sari, I.P., Miftah, Z. (2020). Exploratory research on the myth of Batik Gentongan in Tanjung Bumi. In Proceedings of the 1st International Conference on Folklore, Language, Education and Exhibition (ICOFLEX 2019), pp. 36-39. <https://doi.org/10.2991/assehr.k.201230.007>
- [2] Triandika, L.S. (2023). The uniqueness of culture: Acculturation between religion and local culture on Indonesian Sumenep Batik Motifs. *Fikri: Jurnal Kajian Agama, Sosial dan Budaya*, 8(1): 41-53. <https://doi.org/10.25217/jf.v8i1.3180>
- [3] Triandika, L.S., Arifin, S., Rachmad, T.H. (2023). The

- meaning of Madura batik patterns in a review of visual communication, culture, and religiosity elements. *Sosial Budaya*, 20(1): 37-49. <https://doi.org/10.24014/sb.v20i1.22357>
- [4] Steelyana, E. (2012). Batik, a beautiful cultural heritage that preserve culture and support economic development in Indonesia. *Binus Business Review*, 3(1): 116-130. <https://doi.org/10.21512/bbr.v3i1.1288>
- [5] Gunawan, A.A., Bloemer, J., van Riel, A.C.R., Essers, C. (2022). Institutional barriers and facilitators of sustainability for Indonesian Batik SMEs: A policy agenda. *Sustainability*, 14(14): 8772. <https://doi.org/10.3390/su14148772>
- [6] Nuriyanto, L.K. (2022). Preservation of the Batik industry in Indonesia as part of the national identity. *International Journal of Science and Applied Science: Conference Series*, 6(2). <https://doi.org/10.20961/ijsascs.v6i2.73912>
- [7] Kunjuraman, V., Mohd Radzi, N.A., Arimbi, D.A. (2025). Revitalizing the Batik industry in Indonesia: A scenario assessment. *Changing Societies & Personalities*, 9(3): 826-847. <https://doi.org/10.15826/csp.2025.9.3.355>
- [8] Poon, S. (2020). Symbolic resistance: Tradition in Batik transitions sustain beauty, cultural heritage and status in the era of modernity. *World Journal of Social Science*, 7(2): 1-10. <https://doi.org/10.5430/wjss.v7n2p1>
- [9] Mohd Salleh Anuar, N.N., Abdul Latiff, D.I., Mohd Fathir, M.F. (2025). Challenges that impact youth engagement with batik industries on social media. *'Abqari Journal*, 32(2): 149-157. <https://doi.org/10.33102/abqari.vol32no2.671>
- [10] Rosalina, R., Sahuri, G. (2024). Unraveling Indonesian heritage through pattern recognition using YOLOv5. *Computer Science and Information Technologies*, 5(3): 265-271. <https://doi.org/10.11591/csit.v5i3.p265-271>
- [11] Utami, B.S., Maharani, P.I. (2024). Local wisdom as the basis of visual identity in handwritten Batik from Sendang Pengilon Salatiga. *Gondang: Jurnal Seni dan Budaya*, 8(2): 367-383. <https://doi.org/10.24114/gondang.v8i2.57569>
- [12] Winarno, E., Solichan, A., Ramdani, A.P., Hadikurniawati, W., Septiarini, A., Hamdani, H. (2025). Enhanced Semarang Batik classification using deep learning: A comparative study of CNN architectures. *Bulletin of Electrical Engineering and Informatics*, 14(5): 3544-3557. <https://doi.org/10.11591/eei.v14i5.9347>
- [13] Joseph, M., Richard, J., Halim, C.S., Faadhilah, R., Qomariyah, N.N. (2021). Recreating traditional Indonesian Batik with neural style transfer in AI artistry. In 2021 International Conference on ICT for Smart Society (ICISS), Bandung, Indonesia, pp. 1-8. <https://doi.org/10.1109/ICISS53185.2021.9533197>
- [14] Tang, X., Yu, H.J., Feng, Q. (2024). Capturing style: Going beyond traditional artistic conventions through neural style transfer—Evidence from Malaysia Batik. In Proceedings of the Eleventh International Symposium of Chinese CHI, New York, NY, USA, pp. 497-503. <https://doi.org/10.1145/3629606.3629660>
- [15] Ihsan, A.F. (2021). A study of batik style transfer using neural network. In 2021 9th International Conference on Information and Communication Technology (ICoICT), Yogyakarta, Indonesia, pp. 313-319. <https://doi.org/10.1109/ICoICT52021.2021.9527490>
- [16] Dubey, A., Thilagavathi, P., Dhawan, A., Srivastava, S., Vayelapelli, M., Shukla, B.S. (2025). Neural style transfer as an artistic methodology. *ShodhKosh: Journal of Visual and Performing Arts*, 6(4s): 390-399. <https://doi.org/10.29121/shodhkosh.v6.i4s.2025.6844>
- [17] Pangestu, H.G., Yunus, A.P., Khomsah, S., Choo, Y.H., Ito, T. (2025). Experimental exploration of neural style transfer: Hyperparameter impact and VGG feature dynamics in batik motif generation. In the 2025 International Conference on Artificial Life and Robotics (ICAROB2025), Oita, Japan, pp. 761-766. <https://doi.org/10.5954/ICAROB.2025.OS26-9>
- [18] Lifindra, B.H., Herumurti, D., Yuniarti, A. (2024). A comparison of VGG architecture convolutional layers in migrating batik style into fractal shape. In 2024 International Conference on Smart Computing, IoT and Machine Learning (SIML), Surakarta, Indonesia, pp. 268-273. <https://doi.org/10.1109/SIML61815.2024.10578183>
- [19] Zhang, J., Jiang, Y. (2023). Style transfer technology of batik pattern based on deep learning. *Journal of Fiber Bioengineering and Informatics*, 16(1): 57-67. <https://doi.org/10.3993/jfbim02171>
- [20] Liu, L., Sun, X.Y. (2024). A study on Miao batik pattern design based on style transfer algorithms. In 2024 4th International Conference on Artificial Intelligence, Robotics, and Communication (ICAIRC), Xiamen, China, pp. 953-957. <https://doi.org/10.1109/ICAIRC64177.2024.10900223>
- [21] Pan, H.J., Idzwan bin Ismail, A., Alwi, A., Mahmuddin, M. (2025). The application and optimization of style transfer neural network based on deep learning in fashion design. *Systems and Soft Computing*, 7: 200277. <https://doi.org/10.1016/j.sasc.2025.200277>
- [22] Khalida, R. (2022). Generate Asian Games 2018 mascot for batik motif with neural style transfer. *International Journal of Advanced Research in Computer and Communication Engineering*, 11(1): 150-155. <https://doi.org/10.17148/IJARCC.2022.11124>
- [23] Adali, F., Akbar, A.S., Mahendra, D. (2025). Neural style transfer and clothes segmentation for creating new batik patterns on clothing design. *Scientific Journal of Informatics*, 12(1): 43-52. <https://doi.org/10.15294/sji.v12i1.19554>
- [24] Sirisha, U., Kumar, C.K., Narahari, S.C., Srinivasu, P.N. (2025). An iterative PRISMA review of GAN models for image processing, medical diagnosis, and network security. *Computers, Materials & Continua*, 82(2): 1757-1810. <https://doi.org/10.32604/cmc.2024.059715>
- [25] Zhu, J.Y., Park, T., Isola, P., Efros, A.A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, pp. 2242-2251. <https://doi.org/10.1109/ICCV.2017.244>
- [26] Kurniawati, A., Damayanti, F., Purnawan, I.K.A., Permana, Y. (2025). Color transformation of Madurese Batik to Balinese style using CycleGAN. *AIP Conference Proceedings*, 3372: 040007. <https://doi.org/10.1063/5.0299197>
- [27] Huang, B., Mo, L. (2025). SegCycle-SPADE: An end-to-end framework for semantic segmentation-based automated extraction and artistic reconstruction of

- traditional craft patterns using conditional GAN. *PLoS One*, 20(11): e0329100. <https://doi.org/10.1371/journal.pone.0329100>
- [28] Wang, L., Wang, L.D., Chen, S.B. (2022). ESA-CycleGAN: Edge feature and self-attention based cycle-consistent generative adversarial network for style transfer. *IET Image Processing*, 16(1): 176-190. <https://doi.org/10.1049/ipr2.12342>
- [29] Zhang, F.Q., Gao, H.M., Lai, Y.P. (2020). Detail-preserving CycleGAN-AdaIN framework for image-to-ink painting translation. *IEEE Access*, 8: 132002-132011. <https://doi.org/10.1109/ACCESS.2020.3009470>
- [30] Xu, W.J., Long, C.J., Wang, R.S., Wang, G.H. (2021). DRB-GAN: A dynamic ResBlock generative adversarial network for artistic style transfer. In 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, pp. 6363-6372. <https://doi.org/10.1109/ICCV48922.2021.00632>