Coloring Pekalongan Batik Using a Madura Dataset: A Comparative Study of GAN and Caffe-Based CNN Models

Muhamad Machrus Ali Wahyudi[1*], Arik Kurniawati[2], Fitri Damayanti[3], I Ketut Adi Purnawan[4]

[1,3]Information Systems Department, Universitas Trunojoyo Madura, Bangkalan, East Java, Indonesia
[2]Informatics Department, Universitas Trunojoyo Madura, Bangkalan, East Java, Indonesia
[4]Information Technology Deparment, Universitas Udayana, Badung, Bali, Indonesia
Email: [1*]machrusaw99@gmail.com, [2]arik.kurniawati@trunojoyo.ac.id, [3]fitrid@trunojoyo.ac.id, [4]adipurnawan@unud.ac.id

(Received: 17 Sep 2024, revised: 30 Sep 2024, accepted: 1 Okt 2024)

Abstract

Madura Batik, as one of Indonesia's valuable cultural heritages, is known for its unique characteristics involving the use of bright colors such as red, yellow, and green, as well as traditional motifs that often feature elements of nature like flowers, leaves, and animals. Each motif in Madura Batik reflects the rich philosophy, values, and stories of Madura culture. This batik is also famous for its production process, which is largely carried out manually using traditional dyeing techniques. However, with the advancement of technology, there is a growing need to integrate technological innovations into the batik dyeing process without losing its traditional essence. This research combines Generative Adversarial Networks (GAN) models and compares them with Caffe-based pretrained Convolutional Neural Networks (CNN) to create new color variations in Pekalongan batik images. The input for the models is grayscale batik images, which are then processed to generate colorful outputs. The dataset used consists of 519 Madura batik images, with a distribution of 80% for training, 20% for validation, and 10 images for testing. The preprocessing process includes resizing, normalization, and batching to accelerate model convergence. Performance evaluation is conducted using FID, MSE, PSNR, and SSIM metrics. The results show that the GAN model with 100 epochs produces better image quality compared to the Caffe-based pretrained CNN model, particularly in terms of visual and structural similarity. In conclusion, the GAN method offers great potential for innovation in batik coloring without compromising its traditional motifs.

Keywords: Madura Batik, Pekalongan Batik, Coloring, Grayscale, GAN, Caffe-based Pretrained CNN

I. INTRODUCTION

Batik is one of the most valuable cultural possessions of Indonesia. As an art form, batik has high aesthetic value and had been part of Indonesian culture since time immemorial. Every pattern and motif in batik reflects deep philosophies, values, and stories from various regions across Indonesia [1]. One of these regions with a strong tradition in batik is Madura. The Madura batik displays characteristics of bright colors and a particular pattern, reflecting the life of the Madurese people, the great cultural variety combined with nature's beauty in this area.

Batik Madura has different characteristics from any other region. The color often used is bright, like red, blue, green, and yellow, combined with plant and animal motifs and various geometric elements [2]. Because most of the stages are still manually achieved by traditional dyeing methods, the making of Madura batik requires great skill and precision. While Madura batik does have its own charm, it faces difficulties in being maintained amidst modernization.

Along with technological development, the effort to combine technology with batik art is continuously pursued. One of them is the use of deep learning methods, such as Generative Adversarial Networks (GAN), to make innovations in batik coloring. The GAN model is a state-of-the-art deep learning method for synthesizing new realistic data based on training data [3]. GAN can be applied in batik coloring, resulting in new color variations different from traditional coloring results, while maintaining the heart of the motif and character of Madura batik. In general, GAN consists of two main components: generator and discriminator. The job of the generator is to generate new images from given inputs, while the discriminator acts as a kind of tester, which tells the system if an image generated by the generator is real or fake.

Although GAN has been applied to many fields of image coloring, doing it with batik-with all its very intricate patterns and rich cultural symbolism-is a challenge. In coloring batik, the result should be very pleasing to the senses but also accurate in terms of philosophy and preservation of values in batik motifs. This gap presents the necessity of further optimization of the GAN architecture for correct processing of colorized complex motifs, which can be depicted in Madura batik, but keeping the originality of the designs.

Besides the GAN model, this study uses the Caffe model with a pre-trained approach to make a comparison in coloring. Caffe is an open-source deep learning framework with libraries containing multiple neural network architectures, among them the CNN. The Caffe model applies the pre-trained model that has been trained on large datasets like ImageNet [4]. However, the model could be adapted for batik coloring by utilizing weights from a pre-trained model that was initially intended for general image classification. A pre-trained approach has an advantage in that it can increase the pace of CNN model training on smaller batik datasets due to a model having an already pre-trained foundation for recognizing visual features. This model was not trained for batik coloring, though, so testing must be compared to the GAN model, which is specially devised to generate new color variations of batik images.

A similar related study is “Automating Monochrome Image Coloring Using Generative Adversarial Network Method”, in which the work involves coloring monochrome images of flowers using the Generative Adversarial Network method. The PSNR result on test data for validation was within a range of 51.75 dB to 73.6 dB at the 50th epoch, with an average of 61 dB for PSNR. These results reflect that GAN has relatively good colouring-the colours of all generator-produced images are somewhat different from that of the original ones. Also, in the paper “Automatic Coloring of Sketches Using Conditional GAN Method to Accelerate the Coloring Process” [5], this automatic coloring of the images of sketches was also done using Conditional GAN. In the validation, the least value of FID was 58.389 with epoch 66 and a learning rate of 0.01, while the highest was 81.554 with an epoch of 19 and learning rate of 0.0001. From these results, good coloring could be achieved, and thus it also can put shading effects on the sketch images. The third study, “SC-FEGAN: Face Editing Generative Adversarial Network with User's Sketch and Color” [6], demonstrates image editing on faces using sketches and color. Evaluation results provide a PSNR value to have the highest score in value of 31.1687 and an SSIM value of 0.9671, showing good performance in coloring.

The recommendation of this study is to apply GAN to color the Madura batik images. GAN is preferred since it does not change the motif of Madura batik but generates new colors. Although GAN has been used widely in image colouring, this research points out that very few studies tackle batik coloring. This implies not only the processing of visual coloring but also the retaining of its philosophical values and intricate traditional motifs. Research objectives: The coloring of Madura batik with GAN is performed for new color varieties. Additionally, GAN is used to attempt color generation for Pekalongan batik based on Madura batik colors. This research also compares the coloring result of GAN with the CNN model using Caffe to look for the superior coloring method. It fills in a certain research gap that is highly necessary within the optimization of the method GAN for rich pattern and culturally important batik motifs.

II. RESEARCH METHOD

The method used in this research is Generative Adversarial Network (GAN). Figure 1 illustrates the steps involved in this study.

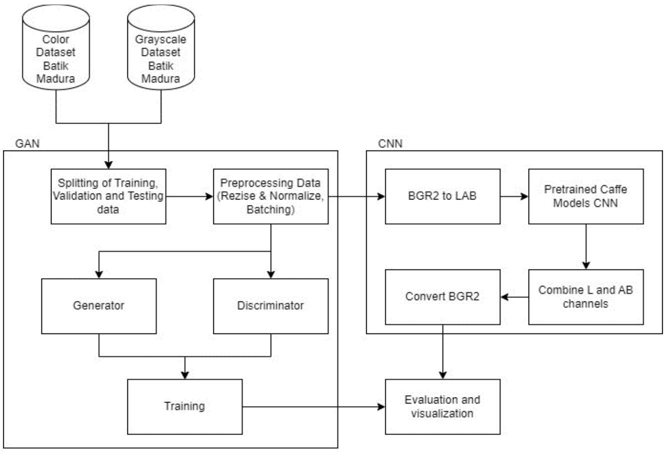

Figure 1. Research Architecture

A. Data Collection

The dataset of Madura batik images was directly captured using a camera, web scraping, and from Kaggle [7]. Collection through the internet is done to get a wider variety of batik patterns.

The dataset used is composed of 519 Madura batik images, with very rich pattern variations. Generally, Madura batik includes a number of bright colors such as red, yellow, and green. Another special characteristic of this batik is that its patterns include a lot of flowers and leaves, typical in its tradition.


Figure 2. Dataset Visualization

Figure 2 is an illustration of the dataset on Madura batik used within the research. This visualization provides a broad overview of the characteristic batik pattern variability in colors that potentially exist within the dataset, serving as an important visual reference throughout the research.

B. Preprocessing

After all, 519 images in the collection, a number of problems were found in the height and width of the Madura batik images. The images were resized to 256x256 pixels. These images had been normalized in the range of [0, 1] by dividing the original pixel value ranging from [0, 255]. This normalization would help speed up convergence of the model while training. Further, in cases where the images are grayscale and had only one channel, an extra channel was added to them so as to make all the images three-channel format images-like RGB. The final step in the pre-processing of the dataset is to batch the data. This allows the model to train on smaller segments, thereby making it more memory-efficient and facilitating parallel training.

C. Development of GAN Model

The Generative Adversarial Network is a framework in machine learning that was designed to generate new data similar to existing data. It was introduced by Ian Goodfellow and his team in 2014, and basically it consists of two important components competing with each other: the generator and discriminator [8].

The Generative Adversarial Network is basically a framework of machine learning that was designed to generate new data similar to the already existing data. The idea was first introduced by Ian Goodfellow and his team in the year 2014, and basically this consists of two important components competing with each other: generator and discriminator.

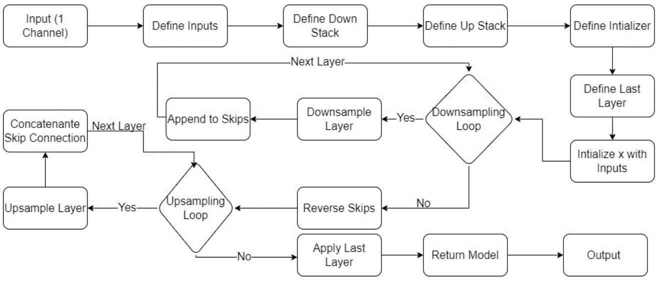

Figure 3. Structure Generator GAN

Figure 3 is generator architecture in the GAN model using a U-Net-based approach. First, it takes input through a single-channel image and passes that through successive downsampling to capture essential features at a lower resolution. While doing so, at each layer, the features are preserved in skip connections, which later get reused during upsampling with an aim to enhance resolution of the image. This generator structure has been informed with information both from the downsampling and upsampling stages, which produces high-quality outputs that are highly detailed; hence, it is effective in such tasks as segmentation or image resolution enhancement [9].

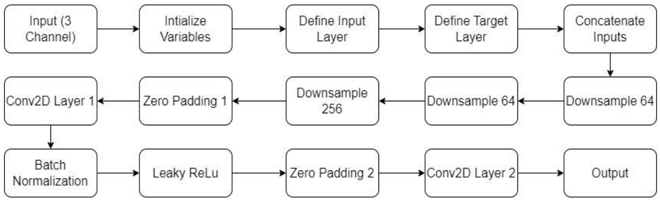

Figure 4. Structure Discriminator GAN

Figure 4 is discriminator architecture of the GAN model. Given an RGB three-channel input image, initializing, and then concatenating it with a target-this is the architecture which consists of a number of convolutional layers, Conv2D, along with zero padding. It provides downsampling and helps in the extraction of major features present in an image. The network also utilizes batch normalization for stability and the Leaky ReLU activation function to avoid the problem of the vanishing gradient. At the end, it decides whether an image is real or generated by a generator [10].

The training process involves a minimax game between the generator and discriminator. Essentially, the generator tries to come up with images to confuse the discriminator, while the discriminator is enhanced further to get an increasingly better job of distinguishing between real and generated images [3]. The loss function is a sum of adversarial losses of the discriminator and reconstruction losses of the generator, such that both are trained together until the model generates images whose visual quality is as good as that of the original ones.

D. Training

Training of the models is done using 40 and 100 epoch schemes. The training is done based on the specifications of Google Colab with 12GB RAM and a 15GB GPU T4. There are major components involved in this area where the training process of the GAN model is taking place: the generator and discriminator. Both of these components interact side by side at every single training cycle. Where the generator is trained in generating colored images from gray-scale input, the discriminator is trained to classify real color images from images created through the generator. Training that takes place in every iteration uses a batch of gray-scale images for the generator and real color images for the discriminator. It is designed so that the generator generates realistic colored images, while the discriminator predicts if these images are real or generated.

Both the components will apply the loss function. If the generator happens to generate images the discriminator is able to easily distinguish, it gets a penalty. In similar fashion, if the discriminator does not discriminate correctly between real and generated images, it gets penalized. Training is done for a number of epochs where, in every epoch, loss values of a generator and discriminator are noted down [11]. Such training stops either when a model starts to be optimal, or the early-stopping criteria are satisfied, meaning that no salient improvement of the loss of the generator has taken place for several epochs. The 40-epoch scheme of training was aimed to get results in the least time possible, whereas the 100-epoch scheme was employed to check how it will perform with the maximum allowed time and for much deeper evaluation of quality of the results generated by the model [12].

E. Development of CNN Models Based on Caffe Models

Data preprocessing for the CNN model is constantly done in a way appropriate for training, as it was done for the GAN model: resizing, normalizing, and batching. Images that initially come in BGR format get converted into LAB format, more suitable for color analysis and to achieve realistic coloring results. It uses a pre-trained Caffe-based CNN model in which it was taught important features from a big dataset that include the most important visual ones [13]. Then, the L-channel will be combined with the AB color channels through the merging process after conversion to achieve the final colored image.

F. Validation and Evaluation

Below, the performance of the used GAN model is evaluated based on important metrics, including FID, MSE, PSNR, and SSIM. This is done in all ways to check how well the image of high-quality colored Madura batik, as generated from GAN, matches with the original-colored image.

1) This is the statistical distance of feature distribution between generated images from GAN and original images. A low value of FID will reflect that the generated image is closer in quality to the original images. [14]. FID is calculated using the formula as in Equation 1.

$$FID = \| \mu_r - \mu_g \|^2 + Tr\left(\Sigma_r + \Sigma_g - 2\sqrt{\Sigma_r \Sigma_g}\right) \quad (1)$$

Where:
- μ_r and μ_g are the means of the feature distributions of the original image and the image generated by GAN.
- Σ_r and Σ_g are the covariance matrices of the distributions of the original image and the image generated.

2) MSE: This estimates the mean average of the squared errors of the original and colored image pixels. The MSE is less; hence, the truth of colorization is more [4]. MSE is calculated using the formula as in Equation 2.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(I_o(i) - I_g(i))^2 \quad (2)$$

Where:
- n is the number of pixels in the image.
- $I_o(i)$ is the intensity of pixel i in the original image.
- $I_g(i)$ is the intensity of pixel i in the generated image.

3) PSNR (Peak Signal-to-Noise Ratio): The image quality ratio is from the maximum signal to the actual color precision of possible noise—the higher the PSNR, the better the quality [4]. PSNR is calculated using the formula as in Equation 3.

$$PSNR = 10log_{10}\left(\frac{MAX_I^2}{MSE}\right) \quad (3)$$

Where:
- MAX_I is the maximum pixel intensity value (e.g., 255 for an 8-bit image).
- MSE is the Mean Squared Error between the original image and the generated image.

4) SSIM was obtained by comparing the structural similarities of the original image and the colorized image; a higher value for SSIM indicates better visual quality, much closer to the original image [8]. SSIM is calculated using the formula as in Equation 4.

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (4)$$

Where:
- μ_x and μ_y are the means of the images x and y.
- σ_x^2 and σ_y^2 are the variances of the images$x$ and y.
- σ_{xy} is the covariance between the images x and y.
- C_1 and C_2 are constants used to maintain the stability of the calculation when the denominator approaches zero.

III. RESULT AND DISCUSSION

A. Dataset Splitting

Out of the total 519 images of Madura batik, this research uses test data from 10 images. The rest of the 509 images were then divided into 80% for training data and 20% for validation data. This then gives 407 training images and 102 to be used as validation images. Therefore, the total general dataset to be used in this research shall total 407 images for the training set, 102 images for the validation set, and 10 more to be tested for predictions.

B. GAN Model Evaluation

After the dataset is divided, the GAN model training process is carried out to obtain the loss results from the generator and discriminator.

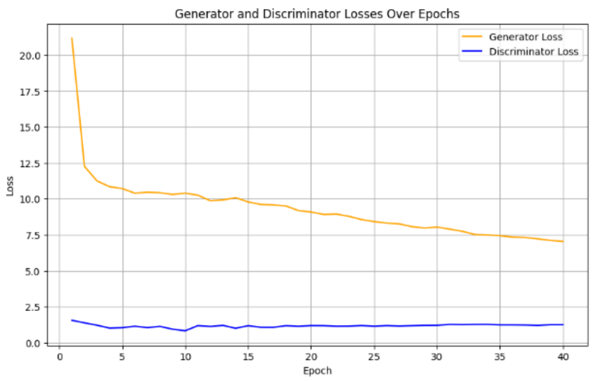

Figure 5. Generator and Discriminator Loss Graph for 40 Epochs

Figure 5 is loss from generator and discriminator over 40 epochs. It seems like the loss for the generator decreases indefinitely for the whole duration of the training. For this, the first epoch loss for a generator is 21.1625, and it is at 1.5603 for a discriminator. Across the epochs, these losses began to lessen and finally stopped at 7.0440 for the generator and at 1.2541 for the discriminator in the last epoch. The fact that this reduction in the generator's loss, systematic in nature, goes on shows increasing quality within the generated images as they approach the original images. At the same time, lowering the loss by the discriminator tells us now the model has started to be more effective at distinguishing which is real and which is generated.

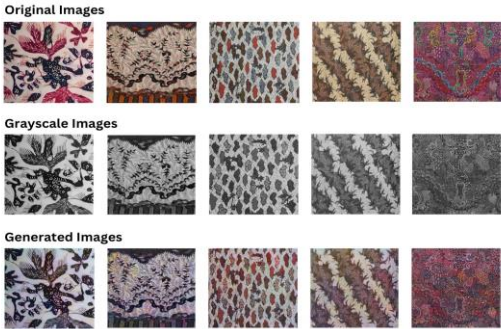

Figure 6. Madura Batik Results Using GAN for Epoch 40

As shown in Figure 6, it presents the Madura batik images generated from the GAN model. The models have been trained with the schemes of 40 epochs using both the original and grayscale input images. The results are different from the original images, especially color-wise.

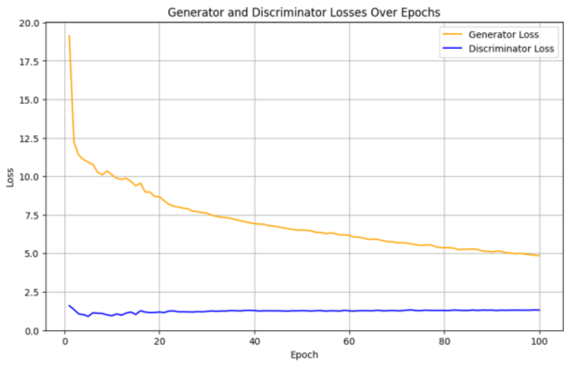

Figure 7. Generator and Discriminator Loss Graph for 100 Epochs

Figure 7 is generator and discriminator loss graphs after 100 epochs of training. In this the generator showed a smooth decreasing behavior in its loss. The losses for the generator and discriminator were 19.1397 and 1.5985, respectively in the first epoch; these further during the process showed a fall to 4.8503 in the case of the generator and 1.3112 in the case of the discriminator during the last epoch. This shows stability in the loss of the generator and actually reflects its better quality in generating images closer to the original ones. In turn, the loss of the discriminator should decrease to show how efficiency for this model towards distinguishing real images from those generated increases.

Figure 8. Madura Batik Results Using GAN for Epoch 100

Figure 8 shows the result of generating Madura batik images from Generative Adversarial Networks. Both the original and grayscale input images have used a 100-epoch scheme. The output using the 100-epoch scheme is closer to the original images. Quality assessment for the generated images has been done by three vital metrics namely Frechet Inception Distance, Mean Squared Error, and Structural Similarity Index.

Table 1. GAN Performance Evaluation Metrics

	EPOCH	FID	MSE	PSNR	SSIM
GAN	40	110.531	37.09	14.10	0.83
	100	**90.623**	**22.35**	**28.67**	**0.87**

Table 1 is a comparison of the model performance under two different training schemes, namely 40 and 100 epochs, for the GAN model. The modalities are evaluated with respect to four important metrics: Frechet Inception Distance, Mean Squared Error, Peak Signal-to-Noise Ratio, Structural Similarity Index.

By the 40th epoch, the FID was 110.531-skewed to large distances between the generated images and the original image-signifying that, still, quality is not very good for the generated images. The MSE remained at 37.09; there is a relatively big average squared error, with still some significant difference remaining between the original and generated images. While the value of PSNR is 14.10, from the above SSIM metric, it goes without saying that the value is 0.83, which signifies that though the pictures are pretty good in structural appearance, they still differ in visual detail.

The Frechet Inception Distance fell in aggression to 90.623 with the scheme of 100 epochs, hence depicting that the quality of the images had fully improved. Also, the Mean Squared Error plummeted to 22.35, showing a shrinkage in the average error. While the PSNR increased significantly to about 28.67, the visual quality was much better and SSIM reached 0.87, which designates an improvement in the structural similarity of the images between the original and the generated ones. This could be taken as the evidence that with more epochs of training, GAN learns to improve the quality of generation toward the original.

C. Experiment on Pekalongan Batik

Since some experiments have been conducted and new colors generated for Madura batik, the next step would be tests on Pekalongan batik. This is done to explore the capability of the GAN models in generating color innovations from the various types of batik trained using the Madura batik dataset.

Other than the GAN model, there is also an implemented CNN model on Caffe that is used for comparison of coloring results among different approaches.

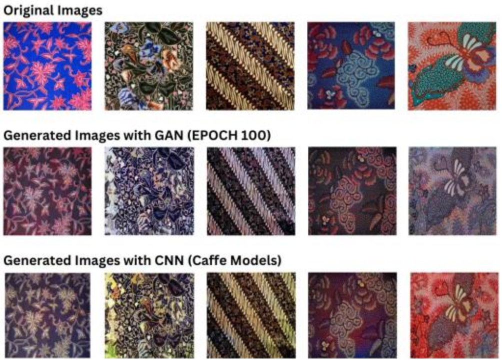

Figure 9. Results of Pekalongan Batik from GAN and Caffe Models

Figure 9 is the result of the new coloring on Pekalongan batik with the Generative Adversarial Network and Caffe-based CNN methods. Comparison is made to the original images. Although some likenesses of images to the original images are obtained from the model based on Caffe, the GAN model indeed gets much closer to generating a new color variation with artistic value. Being able to compare the coloring results of both GAN and Caffe models makes some metrics handy: Frechet Inception Distance, Mean Squared Error, Peak Signal-to-Noise Ratio, and Structural Similarity Index. These would help show which of the two methods is more or less effective in generating new innovative and qualitative colors.

Table 2. Performance Evaluation Metrics of GAN and Caffe Models on Pekalongan Batik

METHODE	FID	MSE	PSNR	SSIM
GAN (EPOCH 100)	**190.215**	**48.03**	**44.82**	**1.06**
Pretrained CNN (Caffe model)	230.461	70.49	28.05	0.89

Table 2 is the comparison of performance of two models of image colorization: GAN with a 100-epoch scheme and CNN with Caffe, evaluated on Pekalongan batik. The two models evaluated against the four following keys are Frechet Inception Distance, Mean Squared Error, Peak Signalto Noise Ratio, and Structural Similarity Index.

In the GAN model trained for 100 epochs, the FID is 190.215, which means it is comparatively closer to the original images rather than from the CNN method. Yet, the depictable gap is remarkable. Its MSE is 48.03, meaning a big difference exists still between the original and generated images, though improved from the CNN method. The signal to noise ratio is low at 44.82, and visual quality is pretty good. SSIM—1.06: Its value over the maximum means an extremely high structural similarity between the generated and the original images, hence it has a very close structural resemblance.

Similarly, the FID for the Caffe method is 230.461, which showed a greater distance between the generated and original images and thus low quality in similarity to the original image. This is because it has a higher MSE at 70.49, which could also mean that the average error difference is large. A PSNR of 28.05 is low and, therefore, bad for visual quality, leading to increased noise compared to GAN results; in the meantime, SSIM equals 0.89, which allows one to speak of a lower structural similarity, so the images created by CNN are less similar to the original for structurally qualitative measures.

These results are indicative that the performance of the GAN method in 100 epochs yielded better quality images compared to those obtained by Caffe-based CNN both from a visual and structural perspective.

IV. CONCLUSION

This research has successfully demonstrated how GAN is able to innovate colorization on Madura and Pekalongan batik by using a few batik datasets. The GAN model performed better than the Caffe-based CNN model, especially for batik image colorization. As can be understood, this generative model is very good at producing highly realistic images while maintaining the original motifs and characteristics of Madura batik. FID, MSE, PSNR, and SSIM-evaluation metrics-all proved that GAN always outperformed CNN in colorization with respect to visual quality and performance. In GAN, a smaller value of FID means smaller difference between the generated distribution and original images. The capability of GAN to generate new variations of color demonstrates its flexible and creative potential in solving challenging batik colorization problems.

The evaluation depicted that GAN with 100 epochs outperformed GAN with 40 epochs. From the loss graph, it can be analyzed that in this training, the generator and discriminator losses continuously decreased. The loss of the generator decreased from 21.1625 in epoch 1 to 4.8503 in epoch 100, while the loss of the discriminator reduced from 1.5603 to 1.3112. This decrease testifies to the better generation capability of the network and discriminator efficiency of distinguishing real versus generated images.

On the other hand, huge improvements were demonstrated in the quality evaluation of the images after 100 epochs. Concretely, FID dropped from 110.531 at epoch #40 to 90.623 at epoch #100, which means that the generated images became closer to the real ones. The MSE went from 37.09 to 22.35, which is interpreted as smaller average error. The PSNR increased from 14.10 to 28.67; this said, generated images looked much better. Regarding SSIM, an increase from 0.83 to 0.87 means that the generated and real images had better structural similarity.

Experiments with Pekalongan batik also demonstrated that GAN was used effectively to create color innovations on various types of batik. The FID was 190.215, MSE was 48.03, PSNR was 44.82, and SSIM was 1.06 for the GAN colorization results at 100 epochs. By comparison, using a CNN model implemented on Caffe, the FID is 230.461, while MSE is 70.49, PSNR is 28.05, and SSIM is 0.89. Thus, these confirm that the GAN at 100 epochs outperforms the CNN in both visual and structural similarities.

However, there are also several factors in the significant improvements mentioned above. In the FID and MSE improvements of Madura batik, it is improved at 100 epochs while the values still show quite a gap from the original images, which means that the generated results are not fully optimal. The difference in quality between Madura and Pekalongan batik probably says that GAN models may need further tuning for each type of batik.

Future work would do well to explore the development of GAN models with architectural variations or additional training techniques that could serve to improve generation quality. Other work might include deeper studies on how different training parameters and pre-processing techniques can affect the quality of generated images. Finally, it would be interesting to test this on more batik datasets for comparison with other generative methods to provide more insight into how well GAN works in batik colorization innovation.

ACKNOWLEDGEMENT

This research was financially supported by an Institutional Research Grant from the Institute of Research and Community Services (LPPM) at Universitas Trunojoyo Madura in 2024

REFERENCES

[1] K. H. Suradiradja and K. Nano, “Perbandingan Model Deep Learning Untuk Prediksi Klasifikasi Jenis Batik,” *InComTech J. Telekomun. dan Komput.*, vol. 14, no. 2, pp. 97–105, 2024.

[2] L. Franca, J. Bahasa, dan Pengajaranya, E. Pebri Ila Rachma, and Q. Lailyn Amrullah, “Menduniakan Batik Tulis Madura Lewat Semiotika Motifbatik Yang Unik Dan Khas : Diplomasi Budaya,” *Ling. Fr. J. Bahasa, Sastra, dan Pengajaranya*, vol. 8, no. 1, pp. 11–25, 2024.

[3] A. Satriawan, B. Imran, and S. Erniwati, “Identifikasi Kemiripan Foto Asli dan Sketsa Menggunakan Model Generatif Adversarial Network (GANs),” *J. Kecerdasan Buatan dan Teknol. Inf.*, vol. 2, no. 3, pp. 122–127, 2023, doi: 10.69916/jkbti.v2i3.36.

[4] Y. Al Najjar, “Comparative Analysis of Image Quality Assessment Metrics: MSE, PSNR, SSIM and FSIM,” *Int. J. Sci. Res.*, vol. 13, no. 3, pp. 110–114, 2024, doi: 10.21275/sr24302013533.

[5] R. R. Kalendesang and D. H. Setiabudi, “Pewarnaan Otomatis Sketsa Gambar Menggunakan Metode Conditional GAN Untuk Mempercepat Proses Pewarnaan,” *J. Infra*, vol. 10, no. 2, 2022.

[6] Y. Jo and J. Park, “SC-FEGAN: Face editing generative adversarial network with user’s sketch and color,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2019-Octob, pp. 1745–1753, 2019, doi: 10.1109/ICCV.2019.00183.

[7] Sandi Indika Saputra, “Batik Madura Dataset,” *Kaggle*, 2024. https://www.kaggle.com/datasets/ndisan/batik-madura (accessed Aug. 06, 2024).

[8] M. Ricky and M. E. Al Rivan, “Implementasi Deep Convolutional Generative Adversarial Network untuk Pewarnaan Citra Grayscale,” *J. Tek. Inform. dan Sist. Inf.*, vol. 8, no. 3, pp. 556–566, 2022, doi: 10.28932/jutisi.v8i3.5218.

[9] T. Tirtawan, E. K. Susanto, P. C. S. W. Lukman Zaman, and Y. Kristian, “Batik Clothes Auto-Fashion using Conditional Generative Adversarial Network and U-Net,” *3rd 2021 East Indones. Conf. Comput. Inf. Technol. EIConCIT 2021*, pp. 145–150, 2021, doi: 10.1109/EIConCIT50028.2021.9431867.

[10] N. T. Tran, T. A. Bui, and N. M. Cheung, “Dist-gan: An improved gan using distance constraints,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11218 LNCS, pp. 387–401, 2018, doi: 10.1007/978-3-030-01264-9_23.

[11] A. B. J. Malyan, R. R. Putra, E. Laila, A. T. Wardhana, M. Fikri, and I. G. T. Isa, *Epoch Tuning Hyperparameter in Fire Image Classification at University Sjakhyakirti*, vol. 1. Atlantis Press International BV, 2023. doi: 10.2991/978-94-6463-118-0_37.

[12] I. Denata, T. Rismawan, and I. Ruslianto, “Implementation of Deep Learning for Classification Type of Orange Using The Method Convolutional Neural Network,” *Telematika*, vol. 18, no. 3, p. 297, 2021, doi: 10.31315/telematika.v18i3.5541.

[13] S. Shaban, R. P. Singh, and D. M. Mehra, “Age and Gender Prediction using Caffe Model and OpenCV,” *Int. J. Innov. Res. Comput. Sci. Technol.*, no. 1, pp. 15–21, 2022, doi: 10.55524/ijircst.2022.10.1.4.

[14] D. Torbunov *et al.*, “UVCGAN: UNet Vision Transformer cycle-consistent GAN for unpaired image-to-image translation,” *Proc. - 2023 IEEE Winter Conf. Appl. Comput. Vision, WACV 2023*, pp. 702–712, 2023, doi: 10.1109/WACV56688.2023.00077.