Scientific Journal of Informatics
Vol. 12, No. 1, Feb 2025

p-ISSN 2407-7658 https://journal.unnes.ac.id/journals/sji/index e-ISSN 2460-0040

Neural Style Transfer and Clothes Segmentation for Creating New Batik Patterns on Clothing Design

Farhan 'Adali[1*], Agus Subhan Akbar[2], Danang Mahendra[3]
[1, 2, 3]Department of Information System, Universitas Islam Nahdlatul Ulama Jepara, Indonesia

Abstract.
Purpose: Applying the original batik image style to other object images and generating new batik patterns that applied to clothing.
Methods: This research uses the Neural Style Transfer method to apply object images to batik to produce new batik patterns, and Clothes Segmentation is used to select areas of clothing in the image so the new batik patterns can be applied to clothing images. And Testing using SSIM, LPIPS and PSNR metrics. This research uses Google Colab, batik image data, and clothing mockup images taken from the internet.
Result: This study shows high average results on SSIM, LPIPS and fair results on PSNR. The results show that the similarity is relatively high with high detected noise.
Novelty: This research develops a new approach in the field of batik pattern innovation and its application to clothing design images. The novelty of this research lies in the implementation of Neural Style Transfer and Clothes Segmentation, which results in a method of exploring new batik patterns and applying them to clothing design images.

Keywords: Neural style transfer, Fashion design, Batik
Received January 2025 / **Revised** March 2025 / **Accepted** April 2025

INTRODUCTION

Batik is one of the Indonesian Cultures. Etymologically, the word batik comes from the Javanese language, namely "tik", which means the verb of making dots, which later developed into the term "batik"[1]. The United Nations Educational, Scientific and Cultural Organization (UNESCO) has designated batik as an intangible cultural heritage of Indonesia[2]. There are several processes for making batik on cloth, namely by using a small-tipped canting. Batik produced from this technique is called batik tulis. This batik process takes a month or more and can make a variety of beautiful art and patterns so that it is maintained and preserved. The next technique is a batik cap or print. This technique uses copper plates to make batik on cloth, which takes 2 to 3 days[3].

Each batik pattern has a meaning representing beauty and visual art. Over time, batik has undergone various innovative changes, such as manufacturing techniques and its function in the art or fashion industry. In recent years, digital technology has contributed to the development of creativity in the field of art and design.

One of the technologies that is often implemented in the field of art and visuals is Neural Style Transfer (NST), NST has always been an interesting and inspiring research topic driven by scientific and industrial challenges. A large amount of research has been conducted in this field[4]. This method was first raised by Leon A. Gatys, Alexander S. Ecker and Matthias Bethge in their research entitled "A Neural Algorithm of Artistic Style" in 2015[5]. This research discusses how to exchange styles in two images by moving the Artistic Style/Style of one image (Style image) into the content of another image (Content image)[6]. Neural Style Transfer is emerging as a technology with potential in the fashion industry, especially in the field of innovative design. According to the following research[7], NST has transformed digital art by allowing individuals to incorporate distinctive artistic styles into their images, enabling realistic changes in portrait images. This study also shows that NST can produce personalized output.

*Corresponding author.
Email addresses: farhanadali5@gmail.com (Farhan)*, agussa@unisnu.ac.id (Agus), danang.umk@gmail.com (Danang)
DOI: 10.15294/sji.v12i1.19554

A study of Wang et al.[8], introduced a localized style transfer method specifically for clothing, which showed impressive results in preserving the shape of original cloth and enabling design personalization to transfer patterns into clothing images. According to the object discussed, which is batik, will become a Style image and transfer its artistic style to the Content image, this Content image uses other objects. It will produce new batik patterns by maintaining the batik style in other object images. In this research we combine[7], [8] to produce design innovations and direct application to clothing images.

The new batik pattern (the result of Neural Style Transfer) can be applied to Fashion Design using clothes segmentation, which functions to create a clothing object detection mask in the image[9], [10]. By separating the Background and Fashion design objects, then applying the new batik pattern (the result of Neural Style Transfer) to the Masking output, and combining the masking results on the Fashion Design as before. Using a design method like this can increase the effectiveness and efficiency of trial in exploring batik patterns in fashion designs. As for the quality results of this method is tested and compared with the results of designs made using Adobe Photoshop with the same NST batik results.

METHODS

In general, this research combines the NST-generated image on a mockup or clothing image segmented using clothing segmentation and generates NST image on the clothing image. The datasets used are JPG, JPEG, and PNG format images downloaded from free platforms on the internet, then processed using several steps applied in this research.

Ingredients

Google colab

Google Colab is a Google research product where users run Python code through browser access, and it is very suitable for machine learning, data analysis, and education. Besides, its provides free GPU computing with certain limitations. There is also paid access, which certainly offer more access[11]. The advantage of this platform lies in the entire computing process is run on Google servers, and it does not require configuration on the device.

Library

This research uses several libraries with different functions: Matplotlib library for image visualization, NumPy library for array computing, OpenCV library for image processing, PyTorch library for machine learning and neural network implementation, TensorFlow library for pre-trained image models, Pillow library for opening and storing images [12].

Neural style transfer

Neural Style Transfer is a method in the field of computational art uses machine learning algorithms to combine two images: First image is a style image, and another one is a content image to create a new image that has content of the content image and art of the style image[13].

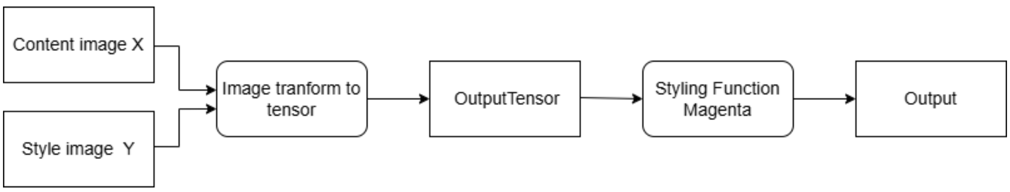


Figure 2. Flow diagram of neural style transfer

In this research, we use TensorFlow. TensorFlow is an open-source library in the Python programming language that is often used in deep learning [14], because it can apply Neural networks, Loss functions and Optimization.

To begins this procedure, the image must be loaded from a file for used in the TensorFlow model. Using `tf.io.read file` to read the image file, format the image into an RGB tensor using `tf.image.decode_image`, then normalize image pixel values to the range [0, 1]. Then, resize image while maintaining the original proportions by setting maximum dimension (this study uses 512 as maximum dimension to balance visual quality and efficiency), and finally, add batch dimensions to prepare image for inclusion in the model. Next, Matplotlib is used to display the image. The next process is to convert tensor into an image in a format that can be saved and displayed by returning pixel values to [0, 255], converting tensor into a NumPy array, and if the tensor has more than three dimensions then only the first batch will be used. Finally, the NumPy array is converted to an image using `PIL.Image.fromarray`.

In Tensorflow, there is Magenta Arbitrary image Stylization, which is a project developed by Google and focuses on art and creativity. TensorFlow allows Magenta to run Convolutional Neural Networks (CNN) models on images, which Magenta models were prepared in TensorFlow Hub, such as Neural Style Transfer, so users do not need to create or train models from scratch[15]. This process loads the image file that has been processed at the beginning. After successfully applying TensorFlow and Magenta, the next step is to display the results of the Neural Style Transfer, which becomes the output Content image and Style image after the process carried out by the Magenta model. Results will be in the form of tensors and converted into images using tensor_to_image.

Clothes segmentation

In clothes segmentation section, this research uses the Unet model. Unet is one of the well-known image segmentation algorithms. Initially, Unet was invented and first used for biomedical image segmentation [10]. Main purpose of this architecture is to provide a solution to overcome the challenges of limited annotated data in the medical field. The selection of Unet model was based on its ease of application, efficiency, and flexibility. Unet is designed to work effectively even with a limited amount of training data. Due to its architecture, Unet is able to handle small objects properly while maintaining speed and accuracy, which is often a challenge in other segmentation techniques[16].

Currently, Unet has been developed and can be used in many segmentation fields, the clothing field is one of them. The Unet used for this clothing image segmentation is “Unet_2020-10-30” model provided on Google Colab. This model can be used after installing Clothes_Segmentation. By using this model, masking can be used on images to detect clothing.

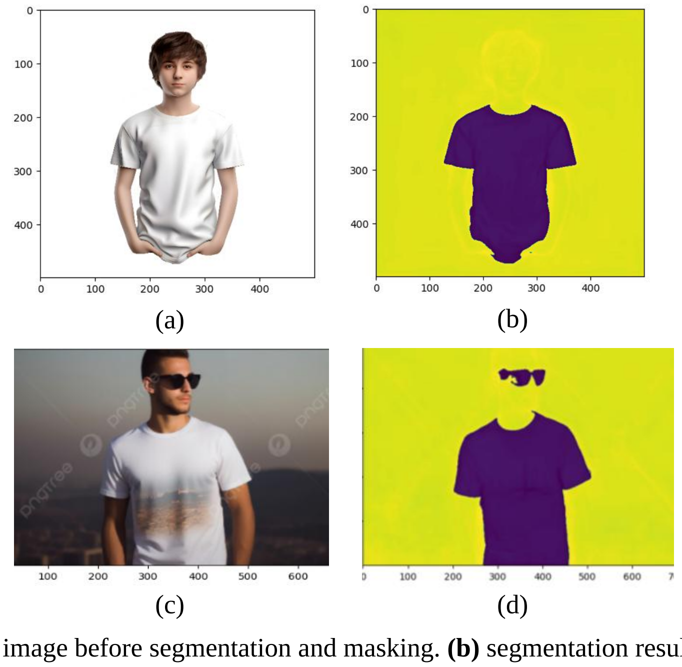
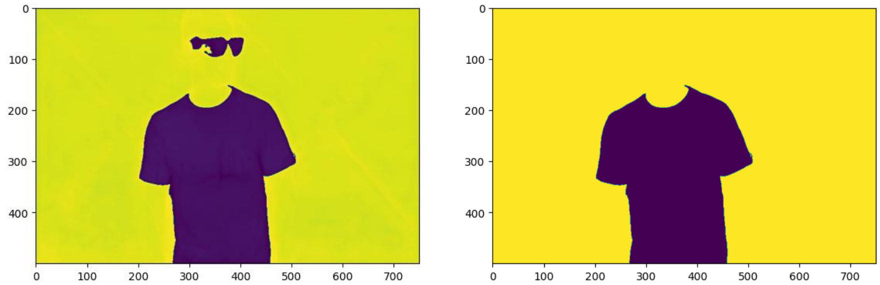


Figure 3. **(a)** Design image before segmentation and masking. **(b)** segmentation result of (a). **(c)** different image from (a) by wearing glasses. **(d)** segmentation result of (c).

It can be seen in figure d that glasses are detected and entered into segmentation, we handle this by using contour in filtering the mask. A contour is boundary or line that describes the shape of an object in image. Contour is commonly used for image segmentation for medical purposes such as cells and organs, it is used to limit certain objects to get a more relevant image with less noise [17]. That is also what we apply to this clothing segmentation by applying contour to filter objects such as glasses.



(a) (b)

Figure 4. (a) Segmentation result before contouring. (b) the result after applying contour to limit the segmentation.

With this result, the clothing segmentation has been filtered and does not detect accessories that are separated from the clothing design so that the next process can be carried out.

Adobe photoshop

Adobe Photoshop is a very popular editing software includes various advanced editing features. In addition, this program is considered easier to use by researchers and has many features that make it easy to use[18]. Its function has penetrated the fields of Design and Fashion, such as batik. Designers often use Adobe Photoshop to visualize their batik designs on mockup before production [19]. The purpose of using Adobe Photoshop is to apply the Batik Patterns produced by NST by utilizing features such as in research[19]. The advantage of this software in applying the design to a mockup is to maintain mockup's texture so it looks more realistic. This research applies Photoshop with the minimum possible features by only using the batik from NST to the design by segmenting the clothing area and then applying NST batik image after adjusting batik image's size to equalized with the size of mockup image and combining the results to be realistic by Multiply. And not using advanced features such as shadow, exposure and others.

Structure similarity index measure

SSIM is a method of calculating and measuring the similarity between two images with strong perception as the quality of human perception, called as Human Visual System (HVS)[20]. This algorithm calculates Luminance, Contrast and Structure in the image[21], so there will be a difference in the value of the results if one of the three aspects is different. The flow of this SSIM calculation can be seen in the following diagram.

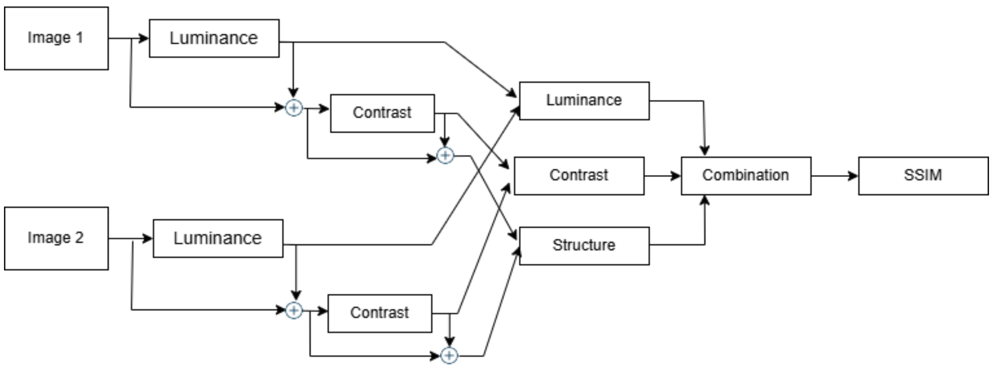


Figure 5. SSIM flow diagram

SSIM can also used in calculating the difference between the compressed image and the pre-compressed image to see the difference in quality between two images, as in the following research[6].
The mathematical formulation of SSIM formulated as below, where Luminance, Contrast, and Structure are calculated:

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \quad (1)$$

$$c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \tag{2}$$

$$s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}, \tag{3}$$

C_1 and C_2 are small constants used to stabilize the division when the denominator approaches zero. Typically: $C_1 = (K_1 \cdot L)^2$, $C_2 = (K_2 \cdot L)^2$, and $C_3 = C_2/2$. Where K_1 and K_2 are small constants traditionally set to 0.01 and 0.03, and L is the maximum value of the pixel (for example, 255 for an 8-bit image). As a Result, the three components are combined into a unique expression weighted with α, β, and γ;

$$SSIM(x,y) = [l(x,y)]^\alpha \cdot [c(x,y)]^\beta \cdot [s(x,y)]^\gamma . \tag{4}$$

In the application, you can use the scikit-image library in the metrics module by importing structural_similarity from skimage.metrics, and the output will show a value from 0 to 1, where the closer to 1, considered to the more similar.

Peak signal to noise ratio

PSNR is a signal processing measurement that compares the received or processed signal to the original source signal. This comparison allows us to identify possible signal's disturbances or distortions that cause differences results[22]. PSNR is the metric most often used to assess the quality of compression results. The signal represents the original data, while the noise represents the compression error. PSNR is an estimate of human perception of the quality of image reconstruction during comparison[23]. The ratio between two images is calculated in decibels. Since the dynamic range of signals is very wide, PSNR is usually calculated as the logarithmic term of the decibel scale (dB)[24]. PSNR calculation will be easily defined through MSE, mathematically the MSE and PSNR values can be formulated as follows;

$$MSE = \frac{1}{m\,n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2. \tag{5}$$

$$\begin{aligned} PSNR &= 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) \\ &= 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \\ &= 20 \cdot \log_{10}(MAX_I) - 10 \cdot \log_{10}(MSE) \end{aligned} \tag{6}$$

MAX_I is the maximum pixel value of the image. Images with 8 bits per sample, the MAX_I value is 255. While the value of 16 bit image, the MAX_I is 65535. The application of PSNR can use the skimage.metrics library just like SSIM where the module has function that automatically calculates the PSNR between two images.

Learned perceptual image patch similarity

LPIPS is a metric that calculates the similarity between two images by extracting image's features, then calculating the distance between the feature vectors to compute the perceptual similarity[25]. This metric is calibrated based on human judgment using data from the two-alternative forced choice (2AFC) test[26]. In testing this metric, the process involves transforming images, equalizing image dimensions and using Alex.Net architecture on LPIPS, which is trained to understand human perception of visual similarity. The LPIPS similarity value will show that the lower the value, the more similar the images are.

Likert scale

The Likert scale is a measurement method used to measure the opinions or perceptions of respondents. This scale is often used in surveys and research[6].

RESULTS AND DISCUSSIONS

After applying described method in previous section, the results of processed image's design with NST seen in Table 1.

Table 1. Result of styling content with style image using NST

No.	Content image	Style image	NST result
1			
2			
3			
4			
5			

Based on above results, The image focuses on maintaining the content of "Content image" and adding the artistic style of "Style Image" so that it blends and becomes a new batik patterns. As part of the evaluation of patterns generated by Neural Style Transfer, an interview was conducted with a CEO and a fashion designer from the batik industry, CV. Rajasa Mas Jaya Cilacap. According to the feedback given, the patterns produced by this technology are considered attractive and still maintain the distinctive characteristics of adapted batik, especially in terms of color. They also highlighted that this innovation positively impacts both the efficiency of the design process and production workflow. Compared to conventional methods that require lengthy manual research, repeated pattern revisions, and additional costs for pattern designers, NST offers a faster and more cost-effective solution. From a business perspective, the use of NST has the potential to reduce production time and increase design variety more flexibly. Meanwhile, from a design perspective, NST allows a wider exploration of patterns without losing the basic characteristics of batik. Therefore, this technology is seen as a good approach to support innovation and the development of modern batik designs.

The result of Table 1 will be applied to fashion design images that have been processed using clothes segmentation and will look like in Figure 6.

(a) (b) (c)

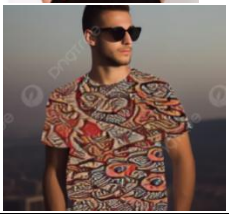
(d) (e)

Figure 6. (a-e) is the application of the pattern of data in table 1 rows 1 – 5 to clothes, respectively.

This research uses images taken from the internet and applies the new Batik NST results with fashion design. Then the obtaining results will be calculated as the level of similarity with the results of design using Adobe Photoshop with the same Batik NST results. This similarity measurement uses SSIM and LPIPS then assesses the difference in noise produced using PSNR.

Table 2. Comparison

Ours	Photoshop	SSIM	LPIPS	PSNR
		0.8254	0.0288	20.36741980139766 dB
		0.7042	0.0575	14.813716825896483 dB
		0.9651	0.0282	25.202649825497776 dB

		0.4241	0.1190	16.642554963037213 dB
		0.8739	0.0568	20.908969600229675 dB
Average		0.75854	0.05806	19.5870

Based on above results, the differences can be seen visually in texture of the clothes, where this method cannot maintain the texture of clothes such image designed using Adobe Photoshop. This is the reason why the similarity metrics value shows different results.

The similarity value calculated by SSIM and LPIPS shows that the design image created by this method is not completely similar to the results produced using Professional Software. The average results of SSIM and LPIPS show 0.75854 and 0. 05806. This result is quite good considering the SSIM value is closer to 1, the more similar it will be. And the lower LPIPS value, the more similar it will be. In contrast, results shown in PSNR where the noise difference between two images produced is indeed very high, so the PSNR value is low. However it should be noted that PSNR is not suitable for assessing image quality that depends on visual perception[27], but only used to calculate the difference in reconstruction between images 1 and 2, so the PSNR results show a fairly high noise difference between the two images assessed in Table 2.

After calculating the similarity of images using metrics, the next step is evaluation using the Likert scale. We also conducted a Survey on 30 respondents with 5 questions from "very dissimilar" to "very similar" with 5 levels, where each question represents 1 comparison. Table 3 shows the result of the survey.

Table 3. Questionnaire responses

	Freq (1)	**Freq (2)**	**Freq (3)**	**Freq (4)**	**Freq (5)**	**Total**	**Mean**	**Median**	**Mode**	**Std Dev**
Q1	0	1	13	13	3	30	3.6	4	4	0.72
Q2	0	0	6	17	7	30	4.03	4	4	0.67
Q3	0	1	7	12	10	30	4.03	4	4	0.85
Q4	1	1	7	12	9	30	3.9	4	4	0.99
Q5	0	2	12	14	2	30	3.53	4	4	0.73

Based on the table above, the highest mean values are observed in Q2 and Q3, both at 4.03, indicating that respondents generally perceive the compared images in these Questions as highly similar. On the other hand, Q5 has the lowest mean score of 3.53, suggesting that respondents find this comparison less similar compared to the others. For Q1 mode, Freq scores 3 and 4 both appear 13 times, so the data is bimodal. However, both scores are modes. Additionally, The standard deviation values are relatively low, indicating that responses are fairly consistent across all questions. However, Q4 has the highest value at 0.99, which suggests a greater variation in perception among respondents regarding the similarity of the images in that question. Overall, the results suggest that most respondents perceive the compared images as similar, with Q2 and Q3 being the most consistent in high similarity ratings. Meanwhile, Q5, despite having the lowest score, still falls within the mid-to-high similarity range.

As seen in Figure 6, the design image processed using this method does not show the texture of the clothes. An alternative to bringing up the texture is Blend the image, where the colour intensity of the batik will be decreased, so the texture will be visible, but this causes the original colour of the clothes will be shown. As in the following example;

Figure 7. Blended Result

The texture may be slightly visible, but the color of batik will fade. The previous results only show part of the batik and do not show the entire object that is adapted into batik, such as the image of a peacock batik, but the application results are only partial and not completely visible. Therefore, this experiment is continued by making the peacock batik still look like batik in general by using the Scale Factor on Tile fill image to replicate the peacock batik in design image.

(a) (b)

Figure 8. (a) is result after applying Scale factor. (b) is blended result of scale factor.

Scale factors and blends can be changed as needed and adjusted manually, giving option to explore the design more freely.

CONCLUSION

This design method cannot completely replace the utilization of professional design software. However, it is an alternative for designers and digital artists to expand their exploration and trials in combining an object with batik to get new batik patterns.

NST in the fashion industry can be integrated into textile production to create dynamic patterns and utilized in e-commerce so that customers can customize clothing designs in real-time. However, the main challenges in its commercialization and scalability include the need for clear regulations regarding design copyright and the adoption of technology by the traditional fashion industry that still relies on conventional methods. Education and gradual integration are needed for NST to be widely accepted. This can be an opportunity for innovation in creativity, design personalization and production efficiency.

REFERENCES

[1] R. Y. Saputra, S. B. Kurniawan, P. Rintayati, and E. Mindrati, "Motif Batik dalam Pendidikan Karakter Pasa Siswa Sekolah Dasar Kabupaten Ngawi," *J. Basicedu*, vol. 5, no. 2, pp. 596–604, 2021, doi: 10.31004/basicedu.v5i2.762.

[2] I. De Carlo, Masiswo, I. R. Salma, M. Triwiswara, and D. K. Syabana, *Batik Nusantara: Kumpulan Motif*. 2020.

[3] A. A. Trixie *et al.*, "Trixie," *A John Steinbeck Encycl.*, p. 401, 2006.

[4] Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu, and M. Song, "Neural Style Transfer: A Review," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 11, pp. 3365–3385, 2020, doi: 10.1109/TVCG.2019.2921336.

[5] L. Gatys, A. Ecker, and M. Bethge, "A Neural Algorithm of Artistic Style," *J. Vis.*, vol. 16, no. 12, p. 326, 2016, doi: 10.1167/16.12.326.

[6] C. Yulian, H. Santoso, I. Wasito, and . H., "Batik-Image Style Transfer Using Neural Style Transfer and Convolutional Autoencoder," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 10, no. 4, pp. 593–607, 2023.

[7] K. Kashyap and S. Fargose, "Dynamic Neural Style Transfer for Artistic Image Generation using

VGG19".

[8] H. Wang, H. Xiong, and Y. Cai, "Image Localized Style Transfer to Design Clothes Based on CNN and Interactive Segmentation," *Comput. Intell. Neurosci.*, vol. 2020, 2020, doi: 10.1155/2020/8894309.

[9] H. Tian, Y. Cao, and P. Y. Mok, "DETR-based Layered Clothing Segmentation and Fine-Grained Attribute Recognition," *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2023-June, pp. 3535–3539, 2023, doi: 10.1109/CVPRW59228.2023.00360.

[10] Ó. S. Solà, "GRAU D ' ENGINYERIA INFORMÀTICA Facultat de Matemàtiques i Informàtica Universitat de Barcelona Object Detection vs . Semantic Segmentation on Fashion RGB Images," 2022.

[11] G. I. E. Soen, M. Marlina, and R. Renny, "Implementasi Cloud Computing dengan Google Colaboratory pada Aplikasi Pengolah Data Zoom Participants," *JITU J. Inform. Technol. Commun.*, vol. 6, no. 1, pp. 24–30, 2022, doi: 10.36596/jitu.v6i1.781.

[12] A. Joshi and H. Tiwari, "An Overview of Python Libraries for Data Science," *J. Eng. Technol. Appl. Phys.*, vol. 5, no. 2, pp. 85–90, 2023, doi: 10.33093/jetap.2023.5.2.10.

[13] S. N. Paradilaga, "Penerapan Kombinasi Pola Dua Gambar Dengan Metode Neural Style Transfer Untuk Aplikasi Mobile," 2020.

[14] S. A. Sanchez, H. J. Romero, and A. D. Morales, "A review: Comparison of performance metrics of pretrained models for object detection using the TensorFlow framework," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 844, no. 1, 2020, doi: 10.1088/1757-899X/844/1/012024.

[15] B. O. F. Technology, C. Engineering, U. The, and S. Of, "IMPLEMENTATION OF DEEP DREAM & NEURAL STYLE TRANSFER ALGORITHM," no. May, 2023.

[16] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image Segmentation Using Deep Learning: A Survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3523–3542, 2022, doi: 10.1109/TPAMI.2021.3059968.

[17] X. Chen, B. M. Williams, S. R. Vallabhaneni, G. Czanner, R. Williams, and Y. Zheng, "Learning active contour models for medical image segmentation," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2019-June, pp. 11624–11632, 2019, doi: 10.1109/CVPR.2019.01190.

[18] G. Syahputra and A. J. Edwards, "Transforming History Teaching: Using Adobe Photoshop E-Posters to Teach the Battle of November 10, 1945," *J. Educ. Technol. Learn. Creat.*, vol. 2, no. 1, pp. 29–40, 2024, doi: 10.37251/jetlc.v2i1.980.

[19] A. Febrian, Q. J. Adrian, N. Husna, and M. W. A. Mawardi, "Pelatihan Dasar Adobe Photoshop dan Penggunaan AI pada CV Barata Batik (Basic Training of Adobe Photoshop and Use of AI at CV Barata Batik)," vol. 5, no. 1, pp. 131–138, 2024.

[20] I. Bakurov, "Structural Similarity Index (SSIM) Revisited : a Data-Driven Approach Structural Similarity Index (SSIM) Revisited : a Data-Driven Approach," 2019.

[21] H. B. Sumarna, E. Utami, and A. D. Hartanto, "Tinjauan Literatur Sistematik tentang Structural Similarity Index Measure untuk Deteksi Anomali Gambar," *Creat. Inf. Technol. J.*, vol. 7, no. 2, p. 75, 2021, doi: 10.24076/citec.2020v7i2.248.

[22] F. A. Fardo, V. H. Conforto, F. C. de Oliveira, and P. S. Rodrigues, "A Formal Evaluation of PSNR as Quality Measurement Parameter for Image Segmentation Algorithms," no. 0, pp. 1–11, 2016.

[23] K. Suriyan, N. Ramaingam, S. Rajagopal, J. Sakkarai, B. Asokan, and M. Alagarsamy, "Performance analysis of peak signal-to-noise ratio and multipath source routing using different denoising method," *Bull. Electr. Eng. Informatics*, vol. 11, no. 1, pp. 286–292, 2022, doi: 10.11591/eei.v11i1.3332.

[24] U. Sara, M. Akter, and M. S. Uddin, "Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study," *J. Comput. Commun.*, vol. 07, no. 03, pp. 8–18, 2019, doi: 10.4236/jcc.2019.73002.

[25] S. Ghazanfari, S. Garg, P. Krishnamurthy, F. Khorrami, and A. Araujo, "R-LPIPS: An Adversarially Robust Perceptual Similarity Metric," 2023.

[26] M. Kettunen, E. Härkönen, and J. Lehtinen, "E-LPIPS: Robust Perceptual Image Similarity via Random Transformation Ensembles," 2019.

[27] D. R. I. M. Setiadi, "PSNR vs SSIM: imperceptibility quality assessment for image steganography," *Multimed. Tools Appl.*, vol. 80, no. 6, pp. 8423–8444, 2021, doi: 10.1007/s11042-020-10035-z.