

# DeepAI: A Hybrid Method of Style Transfer Based on Instance Normalization and Feature Whitening

Noor Fazilla Abd Yusof, Siti Azirah Asmai, Goh Ong Sing

Fakulti Kecerdasan Buatan dan Keselamatan Siber, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.

DOI: <https://dx.doi.org/10.47772/IJRISS.2025.909000032>

Received: 17 August 2025; Accepted: 24 August 2025; Published: 27 September 2025

## ABSTRACT

Image style transfer has caught the interest and attention of researchers in artificial intelligence research. This emerging deep learning technique has shown impressive results for the past few years. Yet, a very limited number of application-ready platforms to leverage these image optimization techniques and visualize the artistic output. Therefore, we have developed a web-based image style transfer application - a platform for a user to experience the deep learning-based approaches to style transfer a digital image. We offered two (2) different methods for generating the style-transferred artwork. The DeepAI web application is available at the website (<https://deepai.asia>) which allows users to create unique artwork by embedding the style on their selected image.

**Keywords** -DeepAI; Style Transfer; Digital Images; Deep Learning; CNN

## INTRODUCTION

Real-to-synthetic artworks have gained significant popularity among scholars across various fields and domains [1]. Image abstraction and artistic stylisation are recent advances in computer graphics and image processing that utilise non-photorealistic rendering (NPR) to generate artistic stylised images [2], [3], [4], [5]. In artificial neural networks, image style transfer has emerged as a powerful technique capable of transferring the style of one image onto another [6], [7]. Image style transfer leverages feature extraction in convolutional neural network (CNN) to transfer stylistic characteristics from a style image to a content image. Generally, the process involves: (i) a content image as the base image, and (ii) a style image providing the artistic features. The technique transforms the textures and patterns of the style image while preserving the structural information of the content image [7]. The resulting style-transferred output is thus a novel artwork that combines the two sources into a single image.

This emerging deep learning technique can generally be divided into two (2) categories of style transfer methods: (i) image-iteration-based and (ii) model-iteration-based [8]. Studies [7], [8] have reported impressive results in image style transfer using different techniques. For instance, Gatys et al. [9] CNN-based texture synthesis and style conversion method; Li and Wand [10] the Markov random field (MRF) model and deep convolutional neural networks (dCNNs) for two-dimensional image synthesis; Ulyanov et al. [11] proposed a compact feedforward convolutional network trained on multiple texture samples; developed a feedforward network for image transformation [7]; while Lin et al. [12] applied a Generative Adversarial Network (GAN)-based approach for day-to-night image translation. More recently, researchers have also explored the use of text-image embedding models such as CLIP for style transfer applications [13]

Due to the impressive performance of deep learning-based image style transfer methods, they have been widely adopted across diverse domains [8], [14], [15]. For instance, VisWebDrone [16] applies style transfer in photogrammetry; AI-ID [17] utilises it for real-time web intrusion detection; FPT.AI [18] leverages it in text-to-speech (TTS) application; Blood Bowl [19] employs it in board game challenges; and nucleAIzer [20] uses it for nuclei identification in biomedical research. In addition, mobile applications such as Prisma (<https://prisma-ai.com>) have popularised image style transfer for photo editing and creative expression.

However, to the best of our knowledge, there are only a limited number of open-source solutions or platforms that support the application and visualisation of image style transfer. To address this gap, we have developed a web-based computer vision program, “DeepAI” - a platform that enables users to experience deep learning-based approaches for performing style transfer on digital images. DeepAI can be accessed publicly at (<https://deepai.asia>). Currently, the platform provides two (2) algorithm settings from which users may choose according to their preferences. In addition, the generated style-transferred images can be downloaded at different pixel resolutions.

In this paper, we present a web application for enhancing textures and patterns in digital images using deep learning-based style transfer. The contributions for this work are two-fold: (i) an enhanced technique for digital image style transfer; and (ii) the development of a web application for generating style-transferred artworks using deep learning. The remainder of this paper is organised as follows. Section II discusses background information related to style transfer. Section III presents the system architecture of the proposed web application. Section IV demonstrates the outcomes of deep learning-based style transfer and evaluates the performance of the DeepAI application. Finally, Section V concludes the paper.

## BACKGROUND

Here, we discuss several methods of style transfer, namely: (i) Neural Style Transfer, (ii) Per Style Per Model (PSPM), and (iii) Multiple Style Per Model (MSPM).

### Neural Style Transfer

The pioneering work by Gatys et al. [6] employed a pre-trained VGG-19 network to initialise the process with a noisy image, which was iteratively optimised. The loss function was defined by comparing the feature maps of the style and content images extracted from the immediate layers of the VGG network. Instead of direct feature comparison, a Gram matrix was used to represent the correlations between feature maps, thereby removing spatial information of style features while retaining their statistical properties.

Later research by Li and Wand [21] introduced the use of Markov Random Fields (MRFs) to compute the loss by decomposing both the style image and the stylised output into small patches. Each patch from the stylised image was matched with the most similar patch from the style image, and the differences were penalised. While this approach preserves more local texture details and often produces visually appealing results, its effectiveness depends heavily on the similarity between the content and style of images. In cases where the content and style images differ significantly, the method struggles, leading to confusion in patch matching and degraded visual quality.

### Per Style Per Model (PSPM)

Instead of training for every pair of style and content images, the Per Style Per Model (PSPM) approach uses a single feed-forward network to learn and stylise images [22], [23]. The process of this approach can be divided into three (3) tasks: (i) a feature extractor/encoder, (ii) a transfer network to learn the style features, and (iii) a decoder to reconstruct the image. Even though the processing speed is faster compared to [6], a separate transfer network still needs to be trained for each style.

### Multiple Style Per Model (MSPM)

Training a single network for each style leads to space and scalability issues. To address this limitation, researchers [24], [25], [26] introduced the Multiple Style Per Model (MSPM) approach, which combines multiple styles into a single transfer network. This method employs a Conditional Instance Normalisation (CIN) layer that swaps between similar styles to generate the corresponding stylised images. Since it conditions only a few layers, this technique significantly reduces both training time and storage requirements compared to training multiple individual models for multiple styles.

## Arbitrary Style Per Model (ASPM)

The main drawback of PSPM and MSPM is their limitation to generate styles that were not included during the training phase. In contrast, Style per Model (ASPM) can apply any chosen style to a content image in a single feedforward pass. This method operates similarly to patch-matching; it divides the feature maps of both the style and content images into small patches, then identifies and swaps the most similar pairs. Known as the Style Swap technique, this approach preserves details from both the content and style images at a high level [27]. This work aims to design an arbitrary style transfer system that allows users to freely select any image as the style reference. Consequently, PSPM can MSPM are not considered due to their limitations. Moreover, considering factors such as computational efficiency and space requirements for real-time applications, ASPM is identified as the most suitable option.

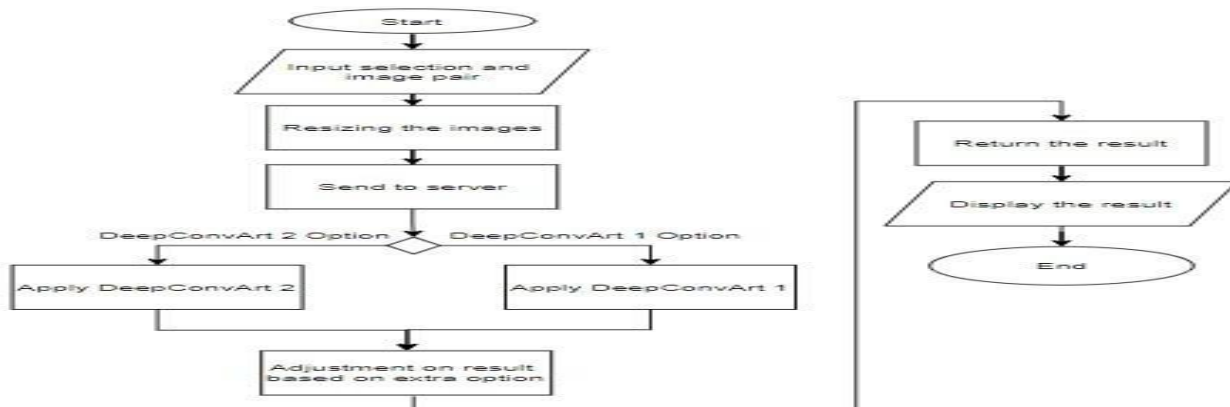


Fig. 1. Flowchart of the proposed Style Transfer System

## RESEARCH METHOD

In this section, we first show the flow design of the proposed style transfer system. Fig. 1 illustrates the overall process, starting from input selection to the final output display. Generally, the system allows the user to select both the input image and the preferred style image. Once selected, the images are processed and sent to the server for style transfer execution. The user may choose between two options of style transfer: (i) DeepConvArt 1, and (ii) DeepConvArt 2. We will discuss further on the DeepConvArt options in sections III-B & III-C. After the style transfer process, the system provides users with additional controls to adjust output settings according to their preferences. Finally, the generated style-transferred image is displayed on the screen as the final output.

### Front-end and Back-end Integration Setup

The system architecture of the proposed DeepAI application is illustrated in Fig. 2. The design adopts a modular structure consisting of two (2) main components: (i) the front-end deployment layer and (ii) the back-end processing layer, both of which are integrated to support seamless interaction, efficient computation, and scalable deployment.

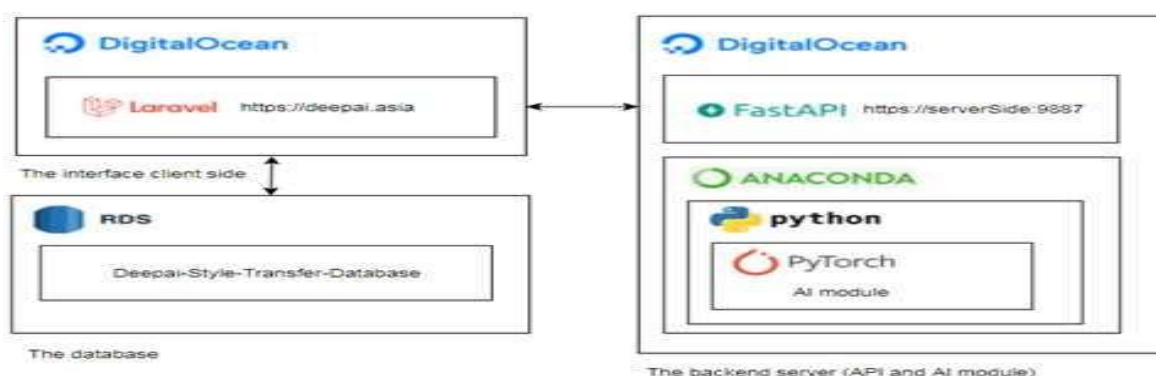


Fig. 2. System architecture of the proposed DeepAI application





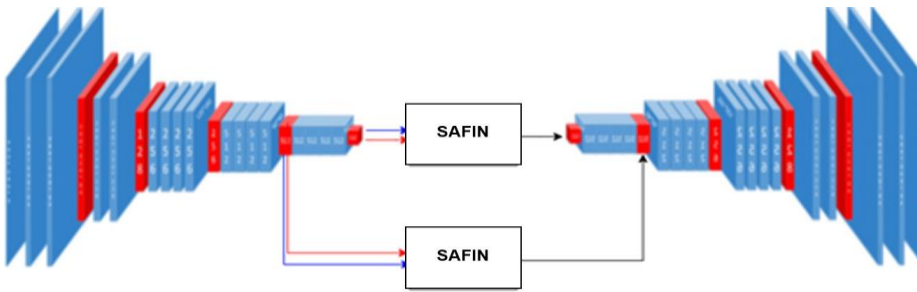


Fig. 4. The DeepConvArt-2 architecture

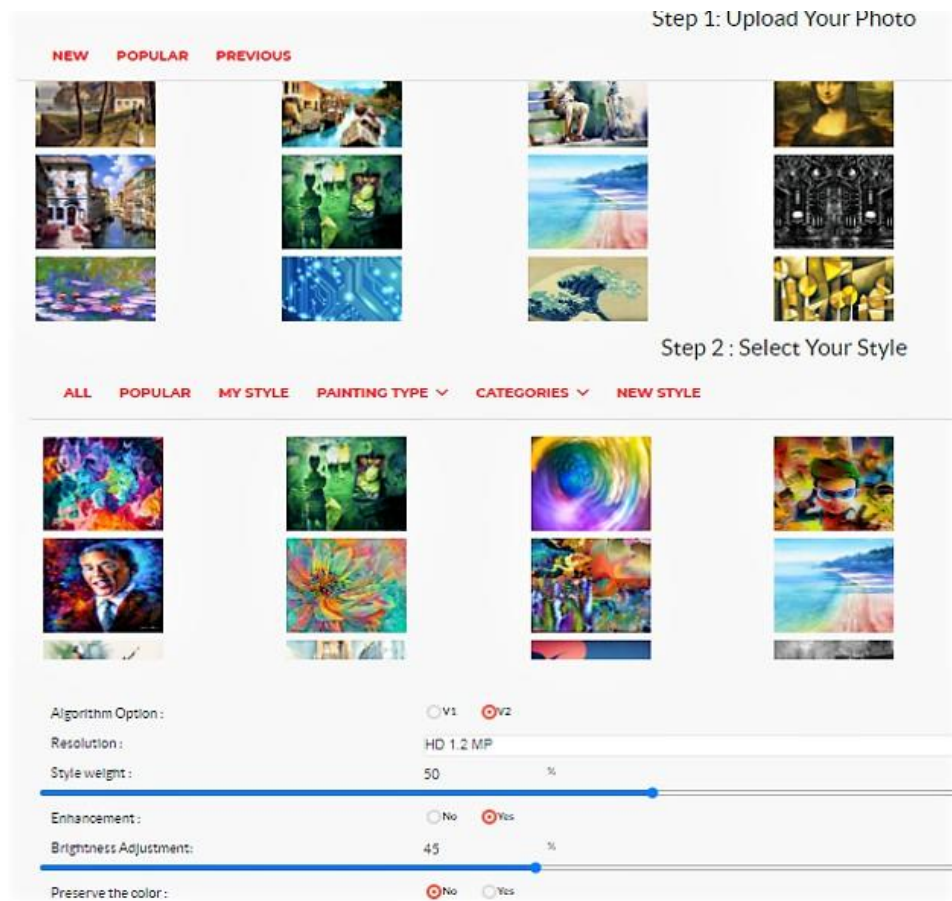


Fig. 6. The input page for content and style image.

### Algorithm 1: An algorithm for DeepConvArt 1

- A. Encode image into the VGG19 model
- B. Begin Image reconstruction and style transfer
  - feature whitening transformation
  - subtracting encoded content from its mean
  - dividing encoded content by its variance
  - calculate mean and variance
- C. Begin content enhancement

- multiplying encoded content by the stroke control degree
  - divide encoded style with stroke control degree
  - convert style by multiplying the result by the array style's variance
  - add result with the array style's mean
- D. Decode the image with the correlated layer of the decoder
- E. Colour-Preserving
- If the colour-preserving option is on; replace the colour channels of the stylised image with the content image

To mitigate this, we incorporate feature whitening before AdaIN. Whitening decorrelates feature maps, neutralising the native content style so that subsequent application of the style's mean and variance (via inverse whitening) is not distorted by residual content-specific statistics. This combination creates a purer feature space, thus yields more faithful style embedding. This step is particularly significant in cross-domain transfer, such as natural images and abstract artworks, where feature distributions are highly divergent.

The hybrid AdaIN-whitening approach thus balances content preservation and style fidelity. The whitening step emphasises style purity by suppressing unintended structural biases, while AdaIN reintroduces controlled stylistic variance that respects the content's global spatial structure. To support perceptual coherence, we adopt a pre-trained VGG-19 network [30] as an encoder, leveraging its well-validated ability to capture both low-level textures and high-level abstractions. The decoder then reconstructs the stylised image with improved sharpness and natural blending compared to single-normalisation approaches.

In short, the hybrid design of DeepConvArt 1 represents a deliberate methodological choice to overcome the shortcomings of existing AdaIN-only frameworks. By uniting whitening transformations with adaptive normalisation, the proposed model establishes a more reliable feature alignment mechanism that promotes both stylistic expressiveness and structural integrity of the synthesised images. This design choice is justified not only by theoretical considerations of statistical independence but also by the practical need for scalable, high-quality style transfer in diverse and complex artistic domains.

## Design of Style Transfer 2: DeepConvArt 2

In DeepConvArt 2 (Fig. 4), we developed a style transfer framework built upon double modules of Self-Attentive Factorised Instance Normalisation (SAFIN) [31]. This design extends the AdaIN-whitening approach of DeepConvArt 1 by incorporating attention-driven normalisation, which more effectively captures both local feature statistics and global semantic dependencies between content and style representations. The objective is to improve the fidelity of stylisation, particularly in scenarios where spatial coherence and fine-grained details are critical.

SAFIN differs fundamentally from AdaIN in its ability to factorise normalisation parameters and apply them selectively under the guidance of self-attention. Whereas AdaIN aligns mean and variance globally across channels, SAFIN leverages attention mechanisms to detect and weight the differences between encoded content and style array (Algorithm 2). This allows the network to emphasise semantically important regions of the content while suppressing less relevant features, thus achieving a more context-aware transfer of stylistic attributes. Prior work [31] has shown that such attention-driven normalisation significantly reduces artefacts and improves structural consistency in arbitrary style transfer tasks.

In DeepConvArt 2, the encoded content features are first whitened to neutralise inherent style biases. Parallel to this, a self-attention branch is introduced to process the style features, capturing both similarity and relational information between the two domains. The outputs of these branches are then merged and refined through a lightweight convolutional layer followed by ReLU activation, which adjusts feature interactions and

enhances non-linear adaptability. The use of double SAFIN modules further stabilises the style-content alignment by reinforcing feature calibration of multiple stages, ensuring that both coarse and fine-grained stylistic patterns are consistently embedded.

As in DeepConvArt 1, we employ a pre-trained VGG-19 network [30] as the encoder for feature extraction. VGG-19 provides hierarchical feature representations, which are essential for enabling the SAFIN modules to operate across multiple abstraction levels. The final decoded reconstructs the stylised image, producing results that exhibit improved texture fidelity, semantic consistency, and overall perceptual quality compared to single normalisation approaches.

---

**Algorithm 2:** An algorithm for DeepConvArt 2

---

- 1.0 Encode image into the VGG19 model
  - 2.0 Begin Image reconstruction and style transfer
    - feature whitening transformation
    - subtracting encoded content with its mean
    - dividing encoded content with its variance
    - get attention from encoded style and content
    - undergo a CNN layer to generate array A and B separately
    - calculate mean and variance
  - 3.0 Begin style converting
    - multiply encoded content with array A
    - add result with array B
    - multiply result with array style's variance
    - add result with array style's mean
  - 4.0 Image decoding
    - If layer = *conv*<sub>51</sub>; decode with first layer of decoder
    - If layer = *conv*<sub>41</sub>; concatenate with last layer's result; decode with second layer of decoder
    - else; decode with correlated layer of decoder
  - 5.0 Colour-Preserving
    - If colour-preserving option is on; replace colour channels of stylized image with content image
- 

In summary, DeepConvArt 2 demonstrates the effectiveness of integrating self-attention with factorised normalisation for image style transfer. The double SAFIN design enhances robustness by combining the strengths of whitening-based disentanglement with attention-driven feature selection. This yields outputs that maintain structural coherence while capturing stylistic nuances with higher fidelity, thereby representing a significant step forward in arbitrary style transfer design.

## RESULTS AND DISCUSSION

Here we present the main page and input page of DeepAI application in Fig. 5 and Fig. 6, respectively. On the main page of DeepAI, users can see the recent photos of the stylised image in the database. To style the new image, the user can simply upload the photo and select the style to be transferred in the photo. There are a few settings for the user to explore, i.e. the style weight, the size of images during processing, the stroke control degree, the result enhancement level, and the choice to do colour preserving, as shown in Figure 7. The application will process the request and show the result of the stylised image, as shown in Fig. 8. The stylised image can then be downloaded to the user's local drive. The produced artwork will depend on the settings chosen by the user. We show a few samples of different stylised artwork produced from DeepConvArt 1 and DeepConvArt2 in Fig. 9 and Fig. 10, respectively.



Fig. 5. The main page of DeepAI - a web application for image style transfer.

Algorithm Option : ☒ V1 ☐ V2

Resolution : HD 1.2 MP

Style weight : 50 %

Image Enhancement : None

Stroke Control: 100 %

Preserve the color : ☒ No ☐ Yes

Fig. 7. The setting options for DeepAI Style Transfer: DeepConv Art 1.

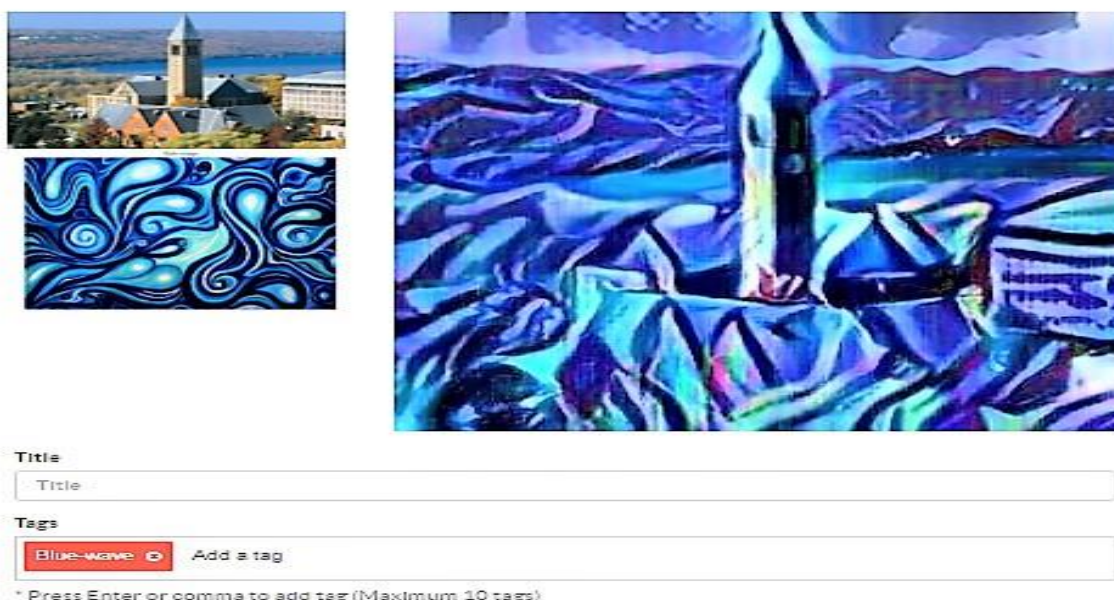


Fig. 8. The result page for stylized image.












Content	Style	Stylized result
		
		
		
		
		

Fig. 9. The Output of DeepAI Style Transfer: DeepConvArt 1

Content	Style	Stylized result
		
		
		
		
		

Fig. 10. The Output of DeepAI Style Transfer: DeepConvArt 2

## A. Evaluation

To evaluate the performance of algorithms and quality of produced artwork, we performed both quantitative and qualitative tests. We collected a dataset from the wikiArt dataset [30] as style images and used the coco2014 dataset [31] for content images. A total of 10 batches were evaluated, with each batch consisting of 10 pairs of images (content and style image).

We calculate the loss using the equation:

$$L_{c+s} = L_c + L_s \quad (1)$$

where  $L_c$  is content and  $L_s$  is style during the training phase. Assume  $L_c = ||g(t) - t||_2$  where  $t$  is the content image's feature (in computing the content loss we use the  $conv5_1$  layer's output which represents the content information of an image).  $g(x)$  is the style transfer process on input  $x$ . The content feature will be compared with the result's feature and get the difference as loss.

$$L_s = \|(g(t) - (s))/2 + \|(g(t) - (s))/2 \quad (2)$$

where  $s$  is the style image's feature in a single channel,  $(x)$  is the mean function and  $x()$  is the function to calculate the standard deviation. For computing the style loss, this process will be repeated for 5 times based on 5 layers ( $conv1_1$ ,  $conv2_1$ ,  $conv3_1$ ,  $conv4_1$ ,  $conv5_1$ ) where those 5 layers represent the style's features in different dimensions. As shown in Fig. 11, there are no significant changes in losses between batches, with less than a 25% difference between the highest and lowest values. This demonstrates that the algorithms are robust and consistently produce similar quality across stylized images. However, as illustrated in Fig. 12, DeepConvArt 2 yields a higher loss in each batch, which may be due to the way the DeepConvArt 2 algorithm updates the pixel values of an image. It might happen due to the way of DeepConvArt 2 algorithm updates the pixel value of an image.

As for the qualitative testing, we randomly selected 10 pairs of content and output images for each algorithm setting (DeepConvArt 1 and DeepConvArt 2). Respondents were then asked to rate the quality of the stylized images on a scale of 1 to 5, with 1 being the worst and 5 being the best. In total, 50 respondents participated in the evaluation. Based on the results shown in Fig. 13, we found that DeepConvArt 2 received a slightly higher average rating than DeepConvArt 1. This finding indicates that, from the perspective of human vision, the quality of artwork produced by DeepConvArt 2 is superior to that of DeepConvArt 1. However, this result contradicts our quantitative testing, where the loss values for DeepConvArt 2 are higher, as illustrated in Fig. 14. Here, we can conclude that using this application, certain style images work better than other, which cause the final artwork of the stylized image to degrade.

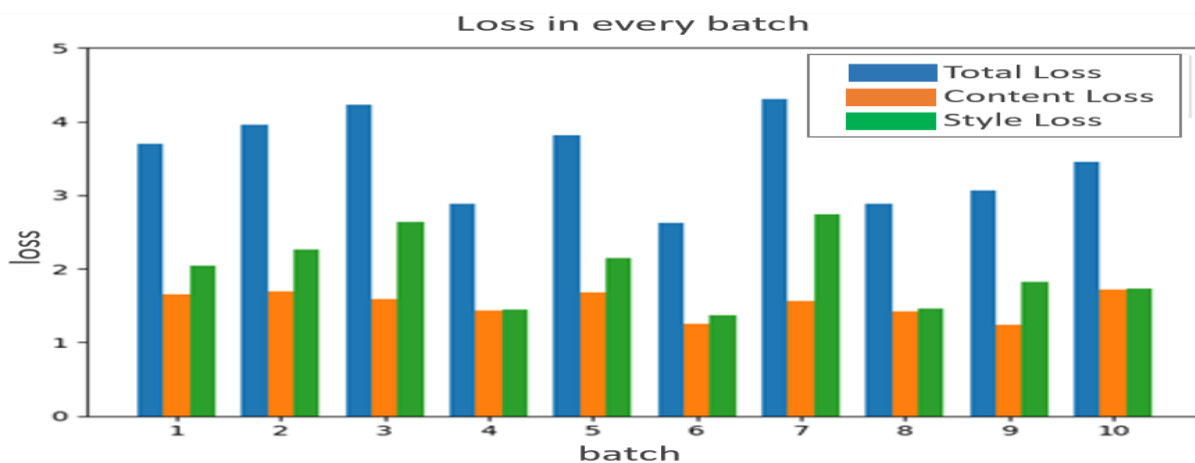


Fig. 11. The quantitative evaluation results: DeepConvArt 1

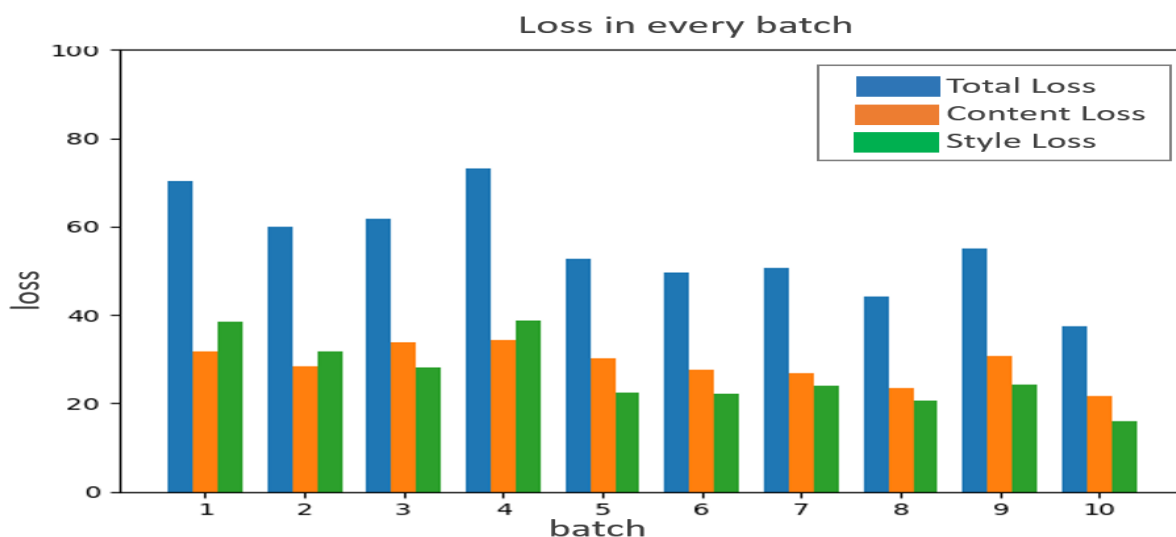


Fig. 12. The quantitative evaluation results: DeepConvArt 2

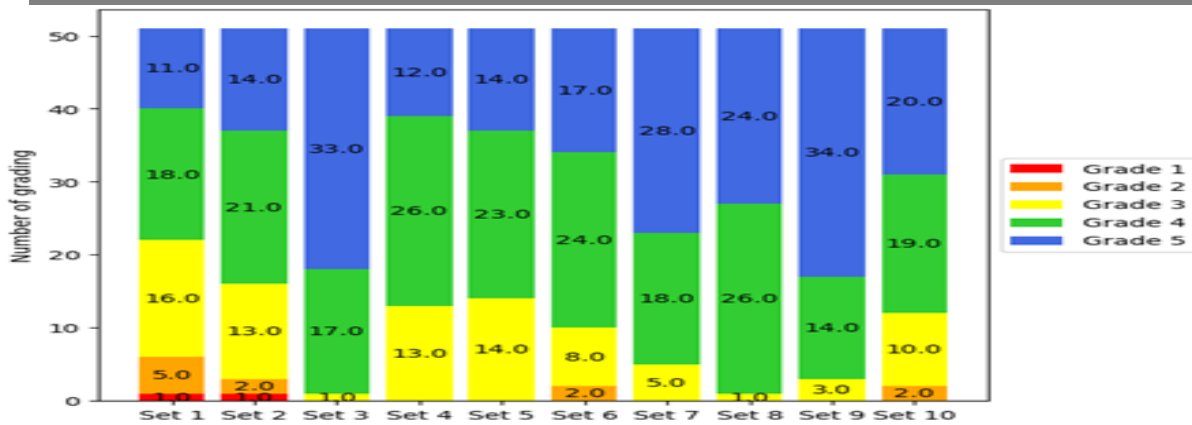


Fig. 13. The qualitative evaluation results: DeepConvArt 1

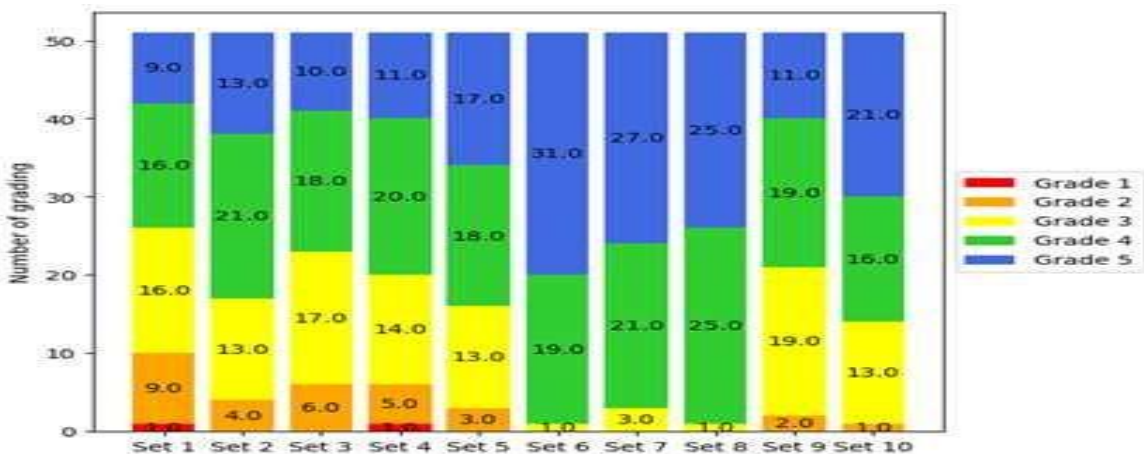


Fig. 14. The qualitative evaluation results: DeepConvArt 2

Despite the encouraging outcomes of our experiments, several limitations remain in the current study. First, the evaluation process revealed a discrepancy between quantitative and qualitative results. Specifically, while DeepConvArt 2 demonstrated a higher loss during training compared to DeepConvArt 1, it nevertheless achieved better average scores in human perceptual evaluations. This divergence suggests that the loss functions currently employed may not fully capture perceptual quality as experienced by human observers. Similar observations have been reported in prior studies where pixel-level loss measures fail to align with subjective visual assessments [32]. This limitation underscores the need for incorporating perceptually aligned loss functions such as perceptual loss or adversarial loss, which can better reflect human aesthetic judgement.

Second, the dataset used for qualitative testing was restricted to 10 randomly chosen content-style pairs, rated by 50 respondents. While this provides an initial insight, the relatively small and homogenous sample size may limit the generalizability of the findings. Future work should adopt larger, more diverse datasets encompassing a wide range of artistic styles and cultural variations, along with broader respondent pools to ensure statistical robustness and reduce potential bias in human evaluations.

Third, the current implementation is limited to two (2) algorithms, which both rely on a pre-trained VGG-19 network as the backbone encoder. While VGG-19 is widely used and effective for perceptual feature extraction [30], it is computationally heavy and may not represent the most efficient solution for real-time web applications. Lightweight architectures such as MobileNet, EfficientNet, or Vision Transformers could be investigated to balance efficiency with stylisation quality, especially for deployment in resource-constrained environments.

Fourth, the present system does not yet incorporate adaptive mechanisms to select or recommend suitable style images based on content characteristics. As observed in our results, certain style-content pairings led to degraded outputs, suggesting that mismatches between feature distributions of content and style images significantly influence the final quality. Future research could therefore focus on style-content compatibility



analysis, where style transfer is guided by learned metrics that predict which style images are likely to yield visually pleasing results. This could also open pathways to personalised style transfer by incorporating user preferences into the process.

Finally, the current study focused primarily on static images. Extending the framework to handle video style transfer while maintaining temporal consistency remains an open challenge. Furthermore, integration of additional algorithms (e.g. patch-based, attention-guided, or diffusion-based methods) could enhance both the diversity and controllability of stylised outputs.

Future work should pursue perceptually aligned loss functions to reconcile quantitative and qualitative outcomes. Larger and more diverse datasets with expanded user studies for robust evaluation. Additionally, exploration of alternative lightweight encoders to improve efficiency in web applications.

## CONCLUSION

DeepAI offers a fast and simple tool for performing style transfer on digital images. It serves as a useful starting point for the development of web applications that apply style transfer, particularly those aimed at community use. The application also holds potential as a teaching tool or educational platform to demonstrate and experiment with image style transfer using deep learning approaches. Currently, DeepAI supports only two algorithms—AdaIN-based and SAFIN-based. Future enhancements could include embedding additional algorithms and exploring alternative pre-trained networks beyond VGG-19 to improve performance and versatility. Moreover, further research should investigate the characteristics of style images that contribute to generating more visually appealing stylised artworks, particularly from the perspective of human perception.

## ACKNOWLEDGMENTS

The authors would like to thank Centre for Research and Innovation Management of University Technical Malaysia Melaka (UTeM) for sponsoring this work under the Grant Tabung Penerbitan Faculty dan Tabung Penerbitan CRIM UTeM.

## REFERENCES

1. J. Li, Q. Wang, H. Chen, J. An, and S. Li, "A Review on Neural Style Transfer," J Phys Conf Ser, vol. 1651, no. 1, pp. 3365–3385, 2020, doi: 10.1088/1742-6596/1651/1/012156.
2. C. Kumar, MP and Poornima, B and Nagendraswamy, HS and Manjunath, "A comprehensive survey on non-photorealistic rendering and benchmark developments for image abstraction and stylization," Iran Journal of Computer Science, vol. 2, no. 3, pp. 131--165, 2019.
3. C. Kumar, Pavan and Poornima, Basavaraj and Nagendraswamy, HS and Manjunath, "Structure preserving non-photorealistic rendering framework for image abstraction and stylization of low-illuminated and underexposed images," International Journal of Computer Vision and Image Processing (IJCVIP), vol. 11, no. 2, pp. 22--45, 2021.
4. J. Kim, Jong-Hyun and Lee, "Layered non-photorealistic rendering with anisotropic depth-of-field filtering," Multimed Tools Appl, vol. 79, no. 1, pp. 1291--1309, 2020.
5. A. and others Rosin, Paul L and Lai, Yu-Kun and Mould, David and Yi, Ran and Berger, Itamar and Doyle, Lars and Lee, Seungyong and Li, Chuan and Liu, Yong-Jin and Semmo, "NPRportrait 1.0: A three-level benchmark for non-photorealistic rendering of portraits," arXiv preprint arXiv:2009.00633, 2020.
6. 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.
7. L. Liu, Z. Xi, R. R. Ji, and W. Ma, "Advanced deep learning techniques for image style transfer: A survey," Signal Process Image Commun, vol. 78, no. February, pp. 465–470, 2019, doi: 10.1016/j.image.2019.08.006.
8. Z. He, K. Han, and Y. Li, "Review of Deep Learning-Based Style Transfer Research," IEEE, 2022, pp. 307–309. doi: 10.1109/iceib53692.2021.9686381.
9. L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks,"



- in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2414--2423. doi: 10.1109/ICRIEECE44171.2018.9008937.
10. C. Li and M. Wand, "Combining Markov Random Fields and Convolutional Neural Networks for Image Synthesis," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2479–2486.
11. D. Ulyanov, V. Lebedev, A. Vedaldi, and V. Lempitsky, "Texture networks: Feed-forward synthesis of textures and stylized images," 33rd International Conference on Machine Learning, ICML 2016, vol. 3, pp. 2027–2041, 2016.
12. C. T. Lin, S. W. Huang, Y. Y. Wu, and S. H. Lai, "GAN-Based Day-to-Night Image Style Transfer for Nighttime Vehicle Detection," IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 2, pp. 951–963, Feb. 2021, doi: 10.1109/TITS.2019.2961679.
13. Y. Wu, H. Zhao, W. Chen, Y. Yang, and J. Bu, "TextStyler: A CLIP-based approach to text-guided style transfer," Computers and Graphics (Pergamon), vol. 119, Apr. 2024, doi: 10.1016/j.cag.2024.103887.
14. L. Du, "How much deep learning does neural style transfer really need? An ablation study," Proceedings - 2020 IEEE Winter Conference on Applications of Computer Vision, WACV 2020, pp. 3139–3148, 2020, doi: 10.1109/WACV45572.2020.9093537.
15. J. Yim, Jonghwa and Yoo, Jisung and Do, Won-joon and Kim, Beomsu and Choe, "Filter style transfer between photos," in European Conference on Computer Vision, 2020, pp. 103--119.
16. N. Guimarães, L. Pádua, T. Adão, J. Hruška, E. Peres, and J. J. Sousa, "Viswebdrone: A web application for uav photogrammetry based on open-source software," ISPRS Int J Geoinf, vol. 9, no. 11, 2020, doi: 10.3390/ijgi9110679.
17. A. Kim, M. Park, and D. H. Lee, "AI-IDS: Application of Deep Learning to Real-Time Web Intrusion Detection," IEEE Access, vol. 8, pp. 70245–70261, 2020, doi: 10.1109/ACCESS.2020.2986882.
18. T. D. Chung, M. Driberg, M. F. Bin Hassan, and A. Khalyasmaa, "End-to-end Conversion Speed Analysis of an FPT.AI-based Text-to-Speech Application," LifeTech 2020 - 2020 IEEE 2nd Global Conference on Life Sciences and Technologies, pp. 136–139, 2020, doi: 10.1109/LifeTech48969.2020.1570620448.
19. N. Justesen, L. M. Uth, C. Jakobsen, P. D. Moore, J. Togelius, and S. Risi, "Blood bowl: A new board game challenge and competition for AI," IEEE Conference on Computational Intelligence and Games, CIG, vol. 2019-Augus, 2019, doi: 10.1109/CIG.2019.8848063.
20. R. Hollandi et al., "nucleAIzer: A Parameter-free Deep Learning Framework for Nucleus Segmentation Using Image Style Transfer," Cell Syst, vol. 10, no. 5, pp. 453-458.e6, 2020, doi: 10.1016/j.cels.2020.04.003.
21. C. Li and M. Wand, "Combining Markov Random Fields and Convolutional Neural Networks for Image Synthesis," Jan. 2016, [Online]. Available: <http://arxiv.org/abs/1601.04589>
22. J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in European conference on computer vision, Springer, 2016, pp. 694--711.
23. W. Huang, Haozhi and Wang, Hao and Luo, Wenhan and Ma, Lin and Jiang, Wenhao and Zhu, Xiaolong and Li, Zhifeng and Liu, "Real-time neural style transfer for videos," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 783--791, 2017.
24. M. Dumoulin, Vincent and Shlens, Jonathon and Kudlur, "A learned representation for artistic style," arXiv preprint arXiv:1610.07629, 2016.
25. G. Chen, Dongdong and Yuan, Lu and Liao, Jing and Yu, Nenghai and Hua, "Stylebank: An explicit representation for neural image style transfer," Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1897--1906, 2017.
26. Y.-F. Wang, Xin and Oxholm, Geoffrey and Zhang, Da and Wang, "Multimodal transfer: A hierarchical deep convolutional neural network for fast artistic style transfer," Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5239--5247, 2017.
27. M. Chen, Tian Qi and Schmidt, "Fast patch-based style transfer of arbitrary style," arXiv preprint arXiv:1612.04337, 2016.
28. X. Huang and S. Belongie, "Arbitrary Style Transfer in Real-Time with Adaptive Instance Normalization," Proceedings of the IEEE International Conference on Computer Vision, pp. 1501--1510, 2017, doi: 10.1109/ICCV.2017.167.

29. Y. Li, C. Fang, J. Yang, Z. Wang, X. Lu, and M. H. Yang, “Universal style transfer via feature transforms,” *Adv Neural Inf Process Syst*, vol. 2017-Decem, pp. 386–396, 2017.
30. K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–14, 2015.
31. A. Singh, S. Hingane, X. Gong, and Z. Wang, “SAFIN: Arbitrary Style Transfer with Self-Attentive Factorized Instance Normalization,” pp. 1–6, 2021, doi: 10.1109/icme51207.2021.9428124.
32. J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9906 LNCS, pp. 694–711, 2016, doi: 10.1007/978-3-319-46475-6\_43.