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

Development of Double-Tail Generative Adversarial Network with Adaptive Style Transfer for Anime Background Production with Makoto Shinkai's Stylization

Agus Purwanto[1*], Kusrini[2], Ema Utami[3], David Agustriawan[4]
[1, 2, 3, 4]Informatics Study Program, Faculty of Computer Science, Universitas Amikom Yogyakarta, Indonesia

Abstract.
Purpose: Traditionally, 2D anime production involves the expertise of experienced animators and is labor-intensive and time-consuming. Generative adversarial networks (GANs) have been developed to create high-quality anime over the years. However, the developed GANs still have caveats, such as the presence of artifacts, high-frequency noise, color and semantic structure mismatches, blurring, and texture issues. Additionally, research on AI-generated anime images with a particular style is still lacking. Thus, this study aimed to develop double-tail generative adversarial network (DTGAN) with adaptive style transfer to generate quality anime background images aligning with Makoto Shinkai's anime style.
Methods: A dataset of real world and anime images was collected and preprocessed. The training was run, and an inference process was done to generate background images with the anime style of Makoto Shinkai using DTGAN with adaptive style transfer. Evaluations of the images produced were performed using visual comparison and quantitative analysis using Fréchet Inception Distance (FID) and peak signal-to-noise ratio (PSNR).
Result: Compared to other methods, the images generated by DTGAN with adaptive style transfer had the lowest FID and highest PSNR values of.38.7 and 19.4 dB, respectively. Visual comparison of the images against other methods and real anime image of Makoto Shinkai demonstrated that images from DTGAN had the best quality that matched Makoto's style, as observed from color, background preservation, photorealistic style, and light contrast.
Novelty: These findings suggest that DTGAN with adaptive style transfer using adaptive instance normalization (AdaIN) and linearly adaptive denormalization (LADE) outperforms other methods, highlighting its practical use for 2D anime production.

Keywords: Double-tail generative adversarial network, Anime background production, AdaIN, LADE, Makoto shinkai
Received February 2025 / **Revised** May 2025 / **Accepted** June 2025

INTRODUCTION

Two-dimensional animation is commonly found in everyday life, such as in social media, video games, and films. Traditional 2D animation creation is labor-intensive and time-consuming, leading to high costs [1-3]. The process of creating animation requires drawing skills and creativity to complete the animation design. On average, an animator needs to draw 12 illustrations per second of movement, showcasing the time and effort required for 2D animation [4]. Moreover, the hand-drawing technique can only be done by an experienced animator.

Technological advances have driven the transformation of the 2D animation industry by implementing approaches that accelerate and simplify the animation production process. One of the aspects that has attracted attention is the application of image models generated by artificial intelligence (AI) [1, 2, 5]. AI-generated image models have played an important role in several aspects of animation design and production automation, simplifying the animation production process, from character design to background artwork. This allows content creation to be faster and more efficient [6]. In addition, AI-generated image models can assist in generating and refining visual elements, providing animators with new tools to improve their work and actualize their creative visions [3].

*Corresponding author.
Email addresses: agus@amikom.ac.id (Purwanto)*, kusrini@amikom.ac.id (Kusrini), ema.u@amikom.ac.id (Utami), david.agustriawan@i3l.ac.id (Agustriawan)
DOI: 10.15294/sji.v12i1.20850

In the context of image production, AI-generated image models have demonstrated remarkable capabilities in generating realistic and artistic images and characters [7]. The use of AI in image generation has been a critical transformation in various fields, especially in art and design. Technologies such as generative adversarial networks (GANs) have been used to create visuals that previously could only be produced by experienced animators [5, 8]. GANs have become the leading technology in AI-generated images due to their ability to produce highly realistic and detailed images. In 2D animation, including anime, GANs have been used to automate and enrich the creative process and offer new ways to design complex characters and backgrounds. The style of creativity in creating 2D anime images refers to the style of anime or Japanese animation styles [9].

In this study, the anime production was made in Makoto Shinkai anime style. Makoto Shinkai is a globally renowned filmmaker and animator who has produced famous works, such as "Your Name" (*Kimi no Na Wa*), The Garden of Words "(*Kotonoha no Niwa*)", and "Weathering with You" (*Tenki no Ko*). He is also known for his voice acting, editing, digital graphic animation, and artwork creation [10, 11]. Makoto Shinkai's style was chosen as the object of the study due to the uniqueness of his creative art style. His work can be clearly distinguished from other animated works; therefore, his work represents the main standard of animated work [11]. The anime style characteristics of Makoto emphasize on composition and background, the combination of nuanced sound effects, and detailed attention to everyday life that is considered ordinary. Mokoto's style is known to 'ground' the audience as much as possible in reality, although there are metaphysical elements and dream-like animations that appear in his work through a photorealistic style. He also often makes the foreground and background of the animation appear out of focus, as well as plays with light reflection or light effect [10].

Various GAN algorithms have been developed to generate anime images, including StyleGAN, CycleGAN, and AnimeGAN [2, 5, 12-17]. StyleGAN has been used to generate anime with an emphasis on preserving anime characteristics. Nonetheless, StyleGAN still has disadvantages, one of which is the presence of artifacts [14]. Meanwhile, CycleGAN has been applied in the translation of human face images to anime but there are still limitations in terms of image clarity and suitability of the background [15, 16]. Similarly, animeGAN and its derivatives also have drawbacks in generating anime images, including weak anime style, blurry details, and feathering edge [8, 17]. In addition, the production of anime-style background images is challenging due to the complexity of visual details in background images, such as lighting, color, and texture. This challenge highlights the need to develop a GAN algorithm with style transfer to generate background images with a specific anime style.

Previous studies have demonstrated the advantages of double-tail generative adversarial network (DTGAN) in generating anime images from real-world images [17]. DTGAN has a generator with two output paths: (a) the main tail, which refines the generated image by removing artifacts and noise, and (b) the support tail, which generates coarse-grained anime images. DTGAN also has linearly adaptive denormalization (LADE) to prevent further artifacts. The improvement of the resulting image quality is also supported by implementing loss functions in the form of region smoothing loss functions that reduce complex textures and fine-grained revision loss functions that remove noise and keep edges clear [18, 19]. Furthermore, the application of color reconstruction loss with Lab color space and grayscale input were able to improve grayscale style loss [20, 21]. These advantages show the potential of developing DTGAN to produce anime images.

Nonetheless, DTGAN with adaptive style transfer consisting of LADE and adaptive instance normalization (AdaIN) has never been used to generate a specific anime style. Therefore, this study aimed to develop DTGAN with adaptive style transfer to generate quality background images with Makoto Shinkai's anime style. The development of this algorithm contributes to the field of AI-generated images and has practical implications in anime-style production, enabling the production of quality anime images with a specific anime style.

METHOD

In this study, we developed DTGAN with adaptive style transfer, consisting of AdaIN and LADE. The model development was performed on a computer with a GPU NVIDIA RTX 2080, 10 GB VRAM, and an 8-core CPU with 32 GB DDR4. The stepwise experimental approach is depicted in Figure 1.

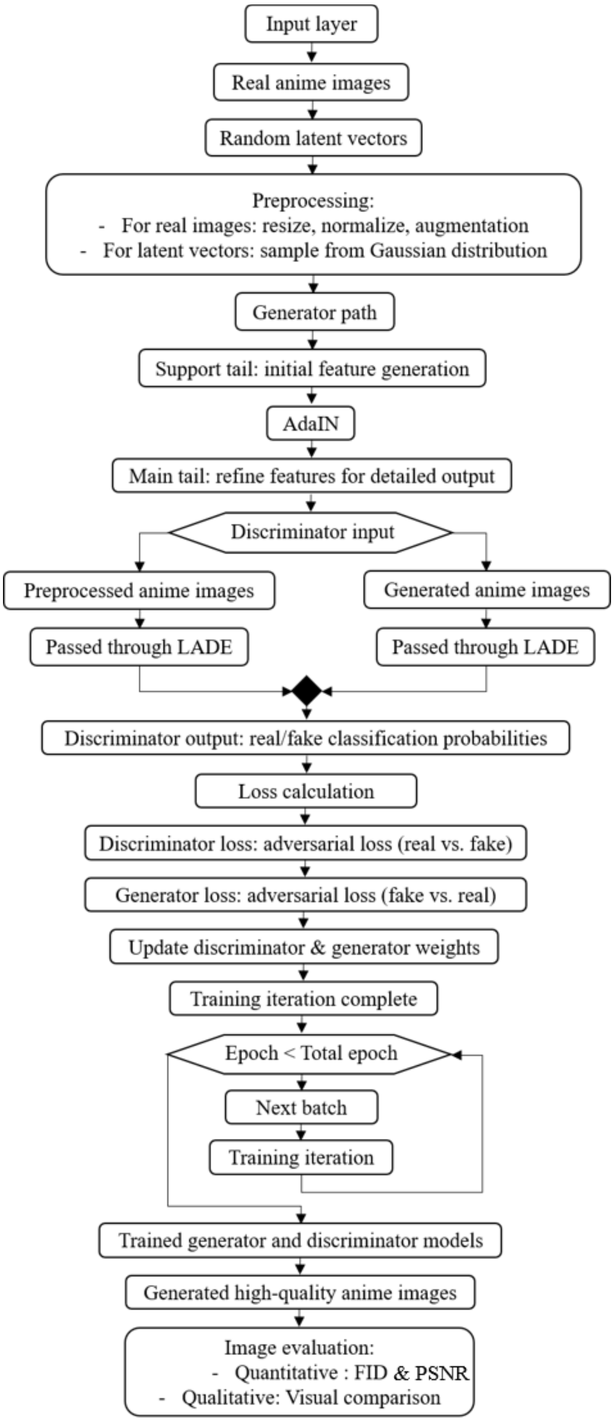

Figure 1. Experimental approach

The Architecture of DTGAN with adaptive style transfer

The DTGAN architecture is presented in Figure 2. DTGAN primarily comprises two discriminators and a generator. The generator used in DTGAN has two outputs, i.e., main and support tails, where the two output tail structures are identical. The main tail eliminates artifacts and noise from the resulting image, while the support tail produces preliminary anime-stylized images and revises them before using the revised images as auxiliary labels of the main tail [17]. The blue and green cube-likes illustrate the convolution modules, consisting of a LReLU activation layer, LADE layer, and a convolutional layer. EA, part of the main tail, is the external attention module, while VGG19 refers to the pre-trained VGG19 network. $G_{s0}1(p)$ and $G_{s1}(p)$ are the two outputs from the support tail, while $G_{s0}(p)$ and $G_{s1}(p)$ are the unsmoothed and smoothed images, respectively. Since the non-parametric differentiable guided filter preserves edge and, at the same time, retains the image's global semantic structure, the guided filter was employed to smooth $G_{s0}1(p)$ to obtain $G_{s1}(p)$. To remove noise and visual artifacts on $G_{s0}1(p)$, the fine-grained revision module is used. The ground truth is the high-quality anime-style images that the module produces, and the primary objective of the generator is to create the mapping from the input image to the ground truth. The DTGAN with adaptive transfer style was chosen to ensure that the images produced have the appropriate background anime style and high quality, similar to Makoto Shinkai's anime style.

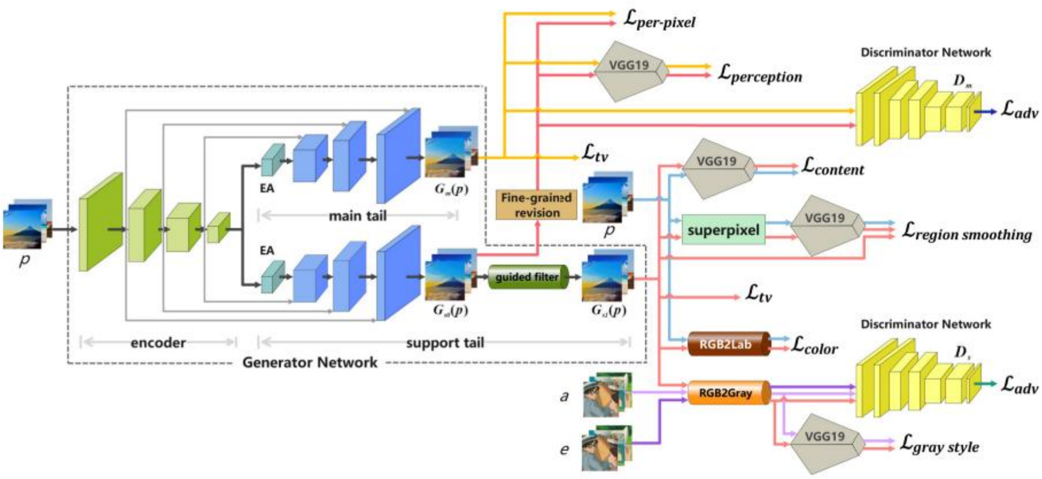

Figure 2. The architecture of double-tail generative adversarial network (DTGAN) with adaptive style transfer (P, a, and e represent real-world, anime, and anime images with blurred edges, respectively) [18]

The adaptive style transfer mechanism in this model relies on AdaIN and LADE. AdaIN is responsible toward the style integration process (lighting and color) into the background style feature. AdaIN utilizes the style image to produce the normalization parameters (γ and β), which are then employed to normalize the content picture. These parameters achieve both style transfer and normalization. Meanwhile, LADE functions to reduce visual artifacts that can interfere with the style accuracy of the background image. This effectively helps the model retain the input photo content and obtain results that are similar to the real anime image. LADE collects global distribution information from its own features by linearly weighting them via point convolution. As a result, LADE can better understand the feature data distribution and steer clear of artifacts.

Dataset and data preprocessing

The real-world background image dataset used consisted of 6,656 images of natural scenery (mountains, rivers, beaches) and urban areas, including buildings. The collected dataset was diverse and of high quality, and they were taken from the Common Objects in Context (COCO) dataset, which are accessible publicly. The real-world images were used as the content base for style transfer. These images provide the structural integrity needed for the model to learn content preservation while adapting to anime styling. Meanwhile, the Makoto Shinkai anime image dataset used consisted of 1,650 images, where the images were screenshots of the film *Kimi no Nawa* (Your Name) taken from Danbooru2020. As the data from Danbooru2020 is publicly available, no permission is required to the original creator (Makoto Shinkai). The dataset included a range of anime aesthetics, capturing various color schemes, line art styles, and textures that are common in anime backgrounds.

To standardize and enhance the dataset for style transfer training, these preprocessing steps were applied [17]:

a. Image resizing: Each image was resized to 256x256 pixels to match the input size required by the model. This step helped maintain consistent model performance across different image sources.
b. Data augmentation: Random horizontal flip, rotation, and perspective distortion were conducted to add variability in orientation, improve the model's robustness to orientation changes, and make the model invariant to slight perspective changes, respectively. Augmentation was also done by applying color jitter to adjust brightness, contrast, saturation, and hue. The data augmentation supported the model learning to adapt to color variations while maintaining style fidelity.

c. Normalization: All images were normalized to a range of [-1, 1], suitable for Tanh activation functions. Normalization is crucial to maintain stability in pixel values, especially during generator and discriminator updates.
d. Channel alignment for model compatibility: The anime-style and real-world images were standardized to three RGB channels to ensure compatibility with the VGG model used for feature extraction and the dual-generator architecture.

Training model

Training was set up using TensorFlow framework, hyperparameters, and loss functions (content, style, and adversarial losses). The training parameters were batch size of 8, epoch of 100, and learning rate of 0.0001 and 0.0002 for generator and discriminator, respectively (Table 1). Content loss was applied to ensure the preservation of the real-world image content while style loss was applied to ensure that the style image produced aligned with the reference style. Meanwhile, adversarial loss was applied to ensure a realistic image is preserved.

Table 1. Parameter Setting for the Model

Parameter	*FID score*
Batch size	8
Number of epoch	100
Generator learning rate	0.0001
Discriminator learning rate	0.0002

Trainings for background and anime were conducted as follows:
a. Model initialization: Initialized the generator and discriminator.
b. Optimization: Adam optimizer was used to update the model weights.
c. Training loop: A batch of real-world and anime-style images was taken, and then passed through the generator and discriminator. The loss functions were computed and optimized, followed by updating the generator and discriminator weights.
d. Iteration: Iterations were performed until the model reached sufficient convergence.

Inference with style user control

This step consisted of the following:
a. User interface implementation: Consisted of the user interface and integration with the model. The user interface allowed the user to select the desired anime style while the integration with the model employed user input to set the style parameters in the AI model.
b. Inference process: Features were extracted from the real-world image and the reference style using the encoder. Then, AdaIN was applied to integrate the style features into the content features. The decoder generated the final anime image that matched the user's chosen style.

Results evaluation

Images generated by the models were evaluated qualitatively and quantitatively as follows:
a. Qualitative analysis: Qualitative analysis was performed through visual comparison, i.e., comparing real images, anime-style reference images, and the model-generated outputs in terms of shape, color, content preservation, and style alignment. This analysis is crucial for assessing both stylistic accuracy and content preservation [22].
b. Quantitative analysis: Quantitative analysis was conducted by calculating Fréchet Inception Distance (FID) and peak signal-to-noise ratio (PSNR) score [23, 24]. FID is a widely used metric for evaluating the quality and realism of images generated by GANs. It compares the distribution of features in real anime images with those in generated anime-style images, indicating how closely the model's outputs resemble the target anime style. FID calculates the Wasserstein-2 distance, also known as Fréchet, between multivariate Gaussians fitted to the Inception-v3 network embedding space of the generated images and the 'real images' [24]. Based on this, the means and covariances for both types of data were estimated according to Eq. (1) below:
c.

$$FID(r,g) = \left|\left|\mu_r - \mu_g\right|\right|_2^2 + T_r\left(\sum_r + \sum_g + 2\left(\sum_r x \sum_g\right)^{\frac{1}{2}}\right) \qquad Eq.\ (1)$$

Where (μ_r; Σ_r) and (μ_g; Σ_g) are the average and covariance of real data and distribution model.

Since FID has a blind spot in discerning image quality, PSNR, a commonly used parameters to quantitatively assess image quality, was also utilized [23, 24]. PSNR measures the ratio of signal peak to noise between two monochrome images I and K. The higher the PSNR (in dB), the better the quality of the resulting image. The PSNR was calculated using Eq. (2), which related to the mean-square error (MSE) (Eq. (3)):

$$\mathrm{PSNR}(\mathrm{I},\mathrm{K}) = 10\log_1 0(\frac{Max_1^2}{MSE}) = 20\log_1 0\,(Max_1^2) - 20\log_1 0\,(MSE_{I;K}) \qquad \textit{Eq. (2)}$$

$$MSE_{I;K} = \frac{1}{mn}\sum_{i=0}^{m-1} x \sum_{i=0}^{n-1}\big(l(m.n) - K(m,n)\big)^2 \qquad \textit{Eq. (3)}$$

Where MAX_1 is the maximum possible pixel image value and MSE represents the cumulative squared error between the compressed and the original image

RESULTS AND DISCUSSIONS

Dataset was taken from publicly available resources. The number of real world image datasets (6,656) used was much higher than the number of anime image datasets (1,650). Data preprocessing, which involved data augmentation for anime data set, increased the number of anime dataset by four times, reaching 6,600 —close to the number of real world data images. Having a balance number of database is essential since imbalance in the dataset number can cause the generator to over-fit the real-world structure and not perform well in anime style applications, potentially leading to content-style mismatches during training. GANs trained on imbalanced datasets are prone to mode collapse, where the generator produces limited images without sufficient diversity [25].

DTGAN was selected in this research because of its advantages in generating anime images from real-world images [17]. The architecture of DTGAN and the loss functions allow DTGAN to generate coarse-grained anime images, reduce complex textures, and refine the generated image by removing artifacts and noise as well as keeping edges clear. In this study, the image results of the developed algorithm were compared to other advanced algorithms to generate anime-style images, such as AdaIN, CartoonGAN, AnimeGAN, AnimeGANv2, and White-box.

A comparison of the resulting image quality was carried out to determine which algorithm produced the best image. The AI-generated images are presented in Figure 3. AdaIN resulted in the poorest image qualities with blurred images, unclear edges, and color production that were the least similar to the real-world images. CartoonGAN, AnimeGAN, and AnimeGANv2 generated better image qualities than AdaIN. However, the image qualities were still subpar compared to images generated using White-box and DTGAN with adaptive style transfer as reflected from the color produced. This finding is in line with a report by [17].

The poor image quality of AdaIN and AnimeGAN found in this study has been observed previously. AnimeGAN was reported unable to apply anime characteristics to the resulting images due to limited integration of anime-specific knowledge. There was also local color deviation [26]. On the other hand, AdaIN cannot be applied independently to produce quality images [17]. Although AnimeGANv2 was developed to improve the drawbacks of AnimeGAN, the model still has weaknesses in terms of blurry details and feathering edges [17]. These drawbacks were also observed in this study. Additionally, CartoonGAN has the disadvantage of losing the original color content, which in line with the findings of [27].

Compared to White-box, DTGAN with adaptive style transfer generated better images with precise edges, actual brightness, authentic colors, and abstract details, as well as minimal artifacts. Although there is a slight color difference to the actual images, DTGAN with adaptive style transfer had an excellent style transfer accuracy in terms of photorealistic style and images that were not very focused or detailed, aligning with Makoto Shinkai style [10].

Figure 3. AI-generated images with anime style
(The first row of images is the real image photos used as the input images, followed by images generated from AdaIN, CartoonGAN, AnimeGAN, AnimeGANv2, White-box, and DTGAN + AdaIN + LAD)

Another set of images was generated to assess whether the models could display the play of light contrast (Figure 4), which is one of the characteristics of Makoto Shinkai's visual style. AdaIn and AnimeGAN produced the poorest image qualities with distorted shapes, inability to produce straight lines, and lack of color preservation. AnimeGAN2 produced better object images with anime stylization than AdaIn and AnimeGAN despite its inability to preserve color, as the images tended to generate greyish color. Similarly, CartoonGAN better preserved the content and color of the image. However, the color of the real images and photos was still lost. The low image qualities found here further corroborates reports from [17, 24, 25]. Whitebox and DTGAN with adaptive style transfer had the best image quality and stylization, as well as produced excellent color contrast, aligning with Makoto Shinkai's visual style. Close observations of the clouds revealed the presence of artifacts on the clouds, indicating that further optimization is needed.

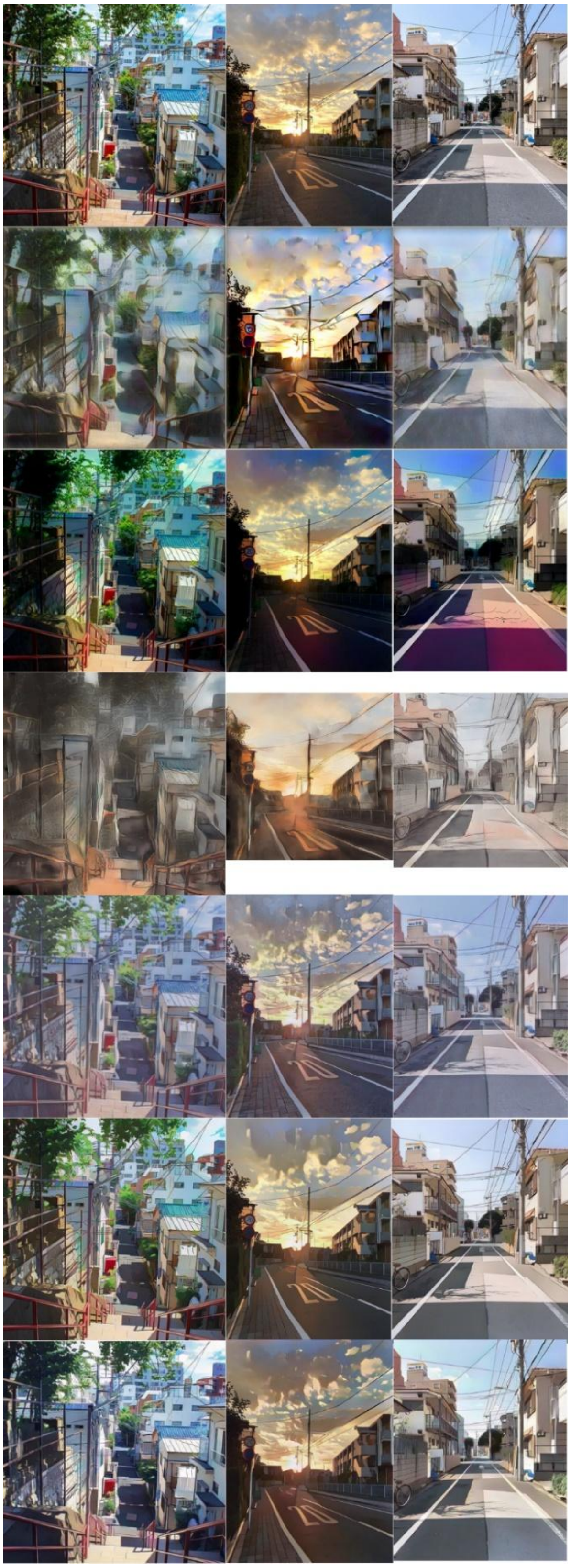

Figure 4. AI-generated images capture the light contrast in the background environment with anime style (The first row of images is the real image photos used as the input images, followed by images generated by AdaIN, CartoonGAN, AnimeGAN, AnimeGANv2, White-box, and DTGAN + AdaIN + LADE)

Figure 5 shows a direct comparison between real-world, DTGAN with adaptive style transfer mechanism, and Makoto Shinkai images. DTGAN with adaptive style transfer mechanism effectively conserved the content of the real-world images including buildings, stairs, plants, trellises, power lines, sky, and shadows. The only aspect that was slightly different between the two images was that the natural lighting of DTGAN is similar to that of anime's. Although there were differences of the objects in the Makoto Shinkai anime compared to the other images, it was clear that DTGAN with adaptive style transfer mechanism successfully implemented Makoto Shinkai's style. This can be observed from the similar depiction of buildings, stairs, and plants between the two, which in accordance with Makoto Shinkai's photorealistic style. This is the first study that reported image comparison between the results of GAN algorithm with Makoto's style images.

Figure 5. Image comparison between real-world images (left), DTGAN images with adaptive style transfer mechanism (center), and Makoto Shinkai images (right)

Image assessment was also conducted quantitatively using FID and PSNR to avoid bias from human observation. The FID method has been widely used because of its consistency with human inspection and sensitivity to small changes in images' actual distribution [24]. Specifically, FID measures density differences between two distributions in the high dimensional feature space of an Inception Net [28]. This metric has excellent ability to discriminate and, at the same time, has good computational efficiency and robustness [29]. The FID value obtained using only DTGAN (without AdaIn and LADE) was 44.1 (Table 2). This value was lower than the FID values of CartoonGAN, AnimeGAN, AnimeGANv2, and White-box (Table 2), indicating that only DTGAN produced better images. More interestingly, when DTGAN was combined with AdaIN + LADE, the FID value improved, reaching 38.7. This result shows that the image generated by DTGAN + AdaIN + LADE (DTGAN with adaptive style transfer) outperformed the other methods, as it achieved the lowest FID [30]. Despite its advantages in assessing images, FID lacks the ability to evaluate image quality effectively [24]. As a result, PSNR was utilized to mitigate FID's drawback. Similar to FID, PSNR has been widely used as a quantitative parameter, providing a classical pixel-wise fidelity metric, where higher PSNR indicates lower reconstruction error [23]. White-box had the lowest PSNR value, followed by CartoonGAN and animeGAN. Although the white-box generated images that were decent, the low value of PSNR indicates that the noise is high and interferes with the image reconstruction. This contradictory result may occur since PSNR mainly focuses on pixel-level differences and may not fully reflect the perceived visual quality of the image [31]. Although PSNR has weaknesses, it remains commonly used because of its meaningful insight and simplicity [31]. AnimeGANv2 exhibited a higher PSNR value than AnimeGAN, further confirming the improvement of AnimeGANv2 over its predecessor [17]. DTGAN without AdaIN + LADE had a similar PSNR value to AnimeGANv2, indicating similar image reconstruction capability. DTGAN with adaptive style transfer achieved the highest PSNR value of 19.4, indicating better image quality compared to the other methods. This also demonstrates that DTGAN with adaptive style transfer is capable of reducing noise in the input image. However, the PSNR value is slightly below the standard value of PSNR (20–25 dB) [30], suggesting that method still requires optimization. Overall, the FID and PSNR values in this experiment align with the findings from [17], who also reported that the images generated by DTGAN had the best quality when compared to other state-of-the-art modeling, such as CartoonGAN, AnimeGAN, and White-box in animation production, with FID value of 40.7. They also reported that DTGAN could maintain the color and brightness of the input image and display clear edges. In addition, DTGAN in that study had the best KID value of 1.5, indicating that DTGAN produces images that are most similar to anime images. The superiority of DTGAN + AdaIN +

LADE compared to other methods in generating animated photos highlights the usefulness of this method in anime image production.

Table 2. FID and PSNR Scores of AI-Generated Images

No.	*Method*	*FID Score*	*PSNR Score (dB)*
1	DTGAN only	44.1	18.1
2	CartoonGAN	65.2	16.3
3	AnimeGAN	59.8	17.2
4	AnimeGANv2	51.6	18
5	White-box	78.3	15.5
6	DTGAN + AdaIN + LADE	38.7	19.4

DTGAN has a generator with two outputs: (a) main tail, which refines the resulting image by removing artifacts and noise, and (b) support tail, which produces a coarse-grained anime image. In order to facilitate the main tail's learning process, support tail functions generate and edit preliminary anime-styled images before employing the updated images as auxiliary labels. On the other hand, the supporting tail helps the main tail create the final anime-styled image. In the end, the support tail is considered only as an accessory employed in training by the main tail since the support tail is discarded following training completion [18].

AdaIN utilizes the style image to produce the normalization parameters (Ɣ and β), which are then employed to normalize the content picture. These parameters achieve both style transfer and normalization. In contradiction, LADE generates the normalization parameters from the content image. That being said, the style image is not involved in the normalization process. As a result, LADE is simply utilized to execute the adaptive normalization operation. Therefore, LADE is classified as an internal normalization approach since the normalization parameters provided by the content image are employed to normalize it. AdaIn, on the other hand, is classified as an external normalization method as the normalization parameters provided by the style image are used to normalize the content image. Furthermore, the use of LADE allows for the global data distribution obtainment from all channels, so the distribution can be used to direct the instance normalization from AdaIN. Hence, LADE evades the artifacts produced by AdaIn in the resulting images. All in all, DTGAN with adaptive style transfer can generate high-quality background images with anime stylization, which is suitable for Makoto Shinkai's stylization.

Although the research results show the advantages of the DTGAN with adaptive style transfer compared to other methods, there are still shortcomings of this approach, where the algorithm requires large computation. In terms of image results, the light contrast still does not fully resemble the style of Makoto Shinkai. Hence, further investigation is required. In addition, there are still some difficulties if this method is applied to higher image resolutions. To overcome this, additional research is needed to increase inference speed and reduce model complexity.

CONCLUSION

The present study demonstrates that DTGAN with adaptive style transfer using AdaIN + LADE outperforms other methods. The use of AdaIN is intended to transfer the style features of the style image to the content image using normalization, and LADE eliminates the artifacts produced by AdaIn, allowing the production of better-quality background images with Makoto Shinkai stylization. Our results contribute to the development of anime-style production, which usually takes time and requires hand-crafted expertise. With the creation of an algorithm that produces quality anime background images, the production of high-quality anime images that matches the intended style is possible. Furthermore, the algorithm can be applied in artmaking or films and further enriches research related to generative models.

REFERENCES

[1] S. Ruan, “Anime Characters Generation with Generative Adversarial Networks,” *2022 IEEE International Conference on Advances in Electrical Engineering and Computer Applications, AEECA 2022*, pp. 1332–1335, 2022, doi: 10.1109/AEECA55500.2022.9918869.

[2] A. Purwanto, E. Utami, and D. Agustriawan, “A Comprehensive Literature Review on Generative Adversarial Networks (GANs) for AI Anime Image Generation,” in *2024 IEEE International Conference on Artificial Intelligence and Mechatronics Systems (AIMS)*, IEEE, 2024, pp. 1–6.

[3] Z. Wang *et al.*, “StyleAdapter: A Single-Pass LoRA-Free Model for Stylized Image Generation,” pp. 1–16, 2023, [Online]. Available: http://arxiv.org/abs/2309.01770

[4] S. Chen and M. Zwicker, “Improving the Perceptual Quality of 2D Animation Interpolation,” in *Computer Vision – ECCV 2022*, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, Eds., Cham: Springer Nature Switzerland, 2022, pp. 271–287.

[5] E. Brown, “GANs in Animation: The Dawn of Automated Creativity,” *Animation Technology Review*, vol. 29, no. 1, p. 45 60, 2021.

[6] W. Yuanliang and Z. Zhe, “Integration effect of artificial intelligence and traditional animation creation technology,” *Journal of Intelligent Systems*, vol. 33, no. 1, p. 20230305, 2024.

[7] P. Johnson, “Application of AI in 2D Animation: A Comparative Study,” *Journal of Animation Studies*, vol. 15, no. research gate, pp. 45–50, 2021.

[8] J. Chen, G. Liu, and X. Chen, “Animegan: A novel lightweight gan for photo animation,” in *Artificial Intelligence Algorithms and Applications: 11th International Symposium, ISICA 2019, Guangzhou, China, November 16–17, 2019, Revised Selected Papers 11*, Springer, 2020, pp. 242–256.

[9] J. Clements, *Anime: A history*. Bloomsbury Publishing, 2023.

[10] U. Aiman, “Unveiling the Familiar: Exploring Makoto Shinkai’s Anime Art,” *Indian Journal of Social Science and Literature (IJSSL)*, vol. 3, no. 2, pp. 36–40, 2023.

[11] Y. Chen and W. Cheng, “An Analysis of Aesthetics in Makoto Shinkai’s Animated Films,” in *SHS Web of Conferences*, EDP Sciences, 2024, p. 04007.

[12] M. I. Ahmed, M. R. Hasan, K. M. I. Uddin, and R. Khan, “Photo-to-Cartoon Translation with Generative Adversarial Network,” *7th International Conference on Trends in Electronics and Informatics, ICOEI 2023 - Proceedings*, no. Icoei, pp. 265–269, 2023, doi: 10.1109/ICOEI56765.2023.10125959.

[13] S. Kumari and K. Aggarwal, “Scope of generative adversarial networks (GANs) in image processing: A review,” *Int. J. Health Sci*, vol. 6, no. S6, pp. 724–733, 2022.

[14] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, “Analyzing and improving the image quality of stylegan,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8110–8119.

[15] L. Quan and H. Zhang, “Facial animation using CycleGAN,” *International Conference on Electrical, Computer, Communications and Mechatronics Engineering, ICECCME 2021*, no. October, pp. 1–5, 2021, doi: 10.1109/ICECCME52200.2021.9591087.

[16] C. Chen, L. Wu, and Y. Xue, “Style changing of the Real human face to cartoon face by Cycle-GAN,” *Proceedings - 2022 International Conference on Machine Learning and Intelligent Systems Engineering, MLISE 2022*, pp. 410–414, 2022, doi: 10.1109/MLISE57402.2022.00088.

[17] G. Liu, X. Chen, and Z. Gao, “A Novel Double-Tail Generative Adversarial Network for Fast Photo Animation,” *IEICE Trans Inf Syst*, vol. E107.D, no. 1, pp. 72–82, Jan. 2024, doi: 10.1587/transinf.2023EDP7061.

[18] Y. Wang, S. Luo, L. Ma, and M. Huang, “RCA-GAN: an improved image denoising algorithm based on generative adversarial networks,” *Electronics (Basel)*, vol. 12, no. 22, p. 4595, 2023.

[19] Y. Yan *et al.*, “Fine-grained attention and feature-sharing generative adversarial networks for single image super-resolution,” *IEEE Trans Multimedia*, vol. 24, pp. 1473–1487, 2021.

[20] G. M. Shalini, R. Anand, T. Sankeerth, H. V. Reddy, S. Hariharan, and V. Kukreja, “Integrated Colorization and Styling Approach for Enhancing Image Clarity,” in *2024 3rd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*, IEEE, 2024, pp. 798–804.

[21] Y. Dong, W. Tan, D. Tao, L. Zheng, and X. Li, “CartoonLossGAN: Learning surface and coloring of images for cartoonization,” *IEEE Transactions on Image processing*, vol. 31, pp. 485–498, 2021.

[22] A. Aggarwal, M. Mittal, and G. Battineni, “Generative adversarial network: An overview of theory and applications,” Apr. 01, 2021, *Elsevier Ltd*. doi: 10.1016/j.jjimei.2020.100004.

[23] S. Xu, V. Dutta, X. He, and T. Matsumaru, “A transformer-based model for super-resolution of anime image,” *Sensors*, vol. 22, no. 21, p. 8126, 2022.

[24] A. Borji, “Pros and cons of GAN evaluation measures: New developments,” *Computer Vision and Image Understanding*, vol. 215, p. 103329, Jan. 2022, doi: 10.1016/J.CVIU.2021.103329.

[25] L. Wang, W. Chen, W. Yang, F. Bi, and F. R. Yu, “A State-of-the-Art Review on Image Synthesis with Generative Adversarial Networks,” 2020, *Institute of Electrical and Electronics Engineers Inc.* doi: 10.1109/ACCESS.2020.2982224.

[26] Z. Lu *et al.*, “Seeing is not always believing: Benchmarking Human and Model Perception of AI-Generated Images,” no. NeurIPS, pp. 1–34, 2023, [Online]. Available: http://arxiv.org/abs/2304.13023

[27] Kim J. and S. Park, "AniGANs and CartoonGANs: Revolutionizing Character Design in Animation," *International Journal of AI and Art*, vol. 5, no. 3, p. 134 135, 2020.
[28] T. Kynkäänniemi, T. Karras, M. Aittala, T. Aila, and J. Lehtinen, "The role of imagenet classes in fr\'echet inception distance," *arXiv preprint arXiv:2203.06026*, 2022.
[29] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, "A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications," Jan. 2020, [Online]. Available: http://arxiv.org/abs/2001.06937
[30] X. Hou, T. Liu, S. Wang, and L. Zhang, "Image Quality Improve by Super Resolution Generative Adversarial Networks," in *2021 2nd International Conference on Intelligent Computing and Human-Computer Interaction (ICHCI)*, IEEE, 2021, pp. 117–121.
[31] M. Prodan, G. V. Vlăsceanu, and C.-A. Boiangiu, "Comprehensive evaluation of metrics for image resemblance," *Journal of Information Systems & Operations Management*, vol. 17, no. 1, pp. 161–185, 2023.