Computer Science and Information Systems 22(3):1331–1346 https://doi.org/10.2298/CSIS241212045S

Leveraging AI and Diffusion Models for Anime Art Creation: A Study on Style Transfer and Image Quality Evaluation

Chao-Chun Shen[1], Shun-Nian Luo[2], Ling Fan[3,*], Chenglin Dai[4]

[1]School of Art Design and Media, Sanda University, China
ghinishen@163.com
[2]School of Information Science and Technology, Sanda University, China
snluo@sandau.edu.cn
[3]Shanghai Technical Institute of Electronics & Information College, China
FL0514@126.com
[4]School of Information Science and Technology, Sanda University, China
ad88105506@163.com

Abstract. The remarkable advancements in artificial intelligence (AI)-driven image generation technologies have brought about a profound transformation across various industries, particularly in new media, video production, and gaming. AI-generated content (AIGC) has emerged as a game-changing, cost-efficient solution for companies seeking high-quality visual assets while operating within constrained budgets and having limited access to traditional human resources. Through the use of sophisticated algorithms, AIGC enables the creation of stunning visuals without relying on conventional, labor-intensive workflows. Among the most prominent techniques, diffusion models have played a pivotal role in the development of AI image generation tools, giving rise to both proprietary platforms like Midjourney and open-source alternatives such as Stable Diffusion. These technologies continue to evolve, benefiting from the collaborative contributions of global programming communities.

This study focuses on advancing the capabilities of Stable Diffusion, an open-source AI image generation model, to address prevalent challenges in style consistency and image quality. By integrating Python and harnessing cutting-edge AI techniques, such as DreamBooth and embedding methods, the research aims to enhance the model's ability to replicate and embed distinct artistic styles. Specifically, the study targets the unique art style of the popular mobile game "Arknights" as a training objective, applying advanced techniques to refine the system's output. The proposed approach demonstrates significant improvements over the baseline model, showcasing enhanced performance in generating style-consistent anime imagery. This research contributes to the evolving landscape of AI-driven art generation, offering novel insights into the application of diffusion-based technologies within creative industries. By utilizing DreamBooth and embedding for style transfer and injection, the study achieves notable efficiency, drastically reducing the time required to train a new model. Ultimately, this work paves the way for more specialized and customizable AI systems in art creation, pushing the boundaries of what AI can achieve in the realm of creative expression.

[*] Corresponding author

Keywords: AI-Generated Content (AIGC); Diffusion Model; Academic Affairs Management

1. Introduction

The development of AI-generated art can be traced back to advancements in artificial intelligence and computer vision technologies. Early computer graphics (1960s-1980s) focused on technical and scientific drawing, laying the groundwork for later AI art. The rise of neural networks in the 1980s, particularly with the introduction of the backpropagation algorithm, set the stage for deep learning. In the 2010s, deep learning breakthroughs, such as Convolutional Neural Networks (CNNs) and Generative Adversarial Networks (GANs), enabled significant progress in image generation. The advent of style transfer techniques in 2015 further advanced AI art, allowing the fusion of artistic styles with content images, opening new possibilities for AI-driven artistic creation.

Since the dawn of human civilization, art has always accompanied our development. Humans receive external stimuli like sound waves, electromagnetic waves, and pressure via their senses, process them through the central nervous system, and present them in an abstract form in their minds. These abstract concepts are expressed through diverse art forms, such as sculpture and painting. Among them, painting is the most fundamental art expression. Humans transmit the light received by their eyes to the visual center of the cerebral cortex as neural signals. After countless neurons in the visual center process these neural impulses, they become what humans see and imagine, ultimately being presented on a canvas through their hands and feet.

Researchers studying AI painting are also simulating the human brain's operation to build neural network models, enabling AI to learn to draw images that conform to human cognition and aesthetics [1]. The objective of this study is to explore how to achieve image style transfer and artistic style injection using state-of-the-art technologies with a small model, thus improving image quality. This study focuses on combining or enhancing existing style transfer and injection techniques for fast style transfer in small-model-generated images. In existing research, there are no cases of combining Embedding and Dreambooth technologies. Additionally, this paper proposes using FID and MLE simultaneously for cross-analysis to evaluate image quality, which are the key technical innovations of this study.

AI-generated illustrations directly address these issues, helping companies in the social media and gaming industries provide fast and convenient services. They aid startups in cost savings and enable mature art companies to manage art outsourcing [2]. Aiming at commercial companies' need to create anime illustrations and character stand-alone images, an anime image generation system based on the diffusion model solves problems like low output efficiency, inconsistent art styles, high commissioning costs, and unstable delivery times in traditional artist commissioning. It meets commercial demands for efficiency, economy, stability, and convenience in product output, adapting to new business models. This project conducts a basic experiment based on an AI painting system to draw some anime images, specifically avatars, to explore the commercial applicability of AI painting [3].

Taking the open-source model Stable Diffusion as an example, some AI painting models may produce unstable output images due to model architecture or training process instabilities, leading to significant quality differences and inconsistencies among generated images. Meanwhile, insufficient training data, monotonous samples, or overly strong regularization during training may cause the model to repeatedly generate a large number of extremely similar or detail - less images. This situation is known as "mode collapse" in the industry. The image distribution under mode collapse is clearly far from the real image distribution. To reduce the model's training cost, this paper aims to improve the original Stable Diffusion model, enhancing its stability and the quality of output images.

This paper will leverage the foundational components of the open-source model Stable Diffusion and employ Dreambooth and embedding techniques to fine-tune the model by injecting custom themes for stylistic adjustments. The model will be evaluated using Maximum Likelihood Estimation (MLE) and Fréchet Inception Distance (FID) as metrics. The training objective is to develop a model capable of generating style-consistent images similar to the game's character illustrations, with better performance in maximum likelihood estimation compared to the original model.

2. Literature Review

This section introduces the technical principles and evaluation criteria for model performance of the AI drawing model applied in this design [4]. It mentions the basic principle of the industry's mainstream image generation model, stable diffusion, which utilizes noise as random numbers to generate images.

The Diffusion model is a neural network model that takes descriptive text, random noise, and a sequence of time steps as input and outputs an image. By repeatedly applying denoising operations to the generated random noise image, the Diffusion model produces an output image at each step, which serves as the input for the next denoising operation. This process continues until the number of denoising steps reaches a predefined total, at which point the resulting image is the model's final output. The descriptive text determines the style and content of the image, while the random noise serves as the initial input. Each generated image is assigned a time step number (decreasing sequentially), with the core process being the generation of an image from noise. As illustrated in Figure 1:

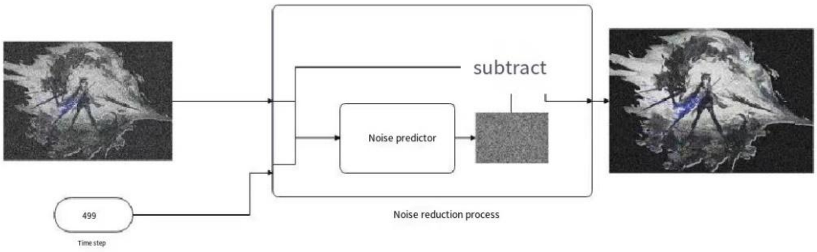


Fig. 1. Forward Propagation Flowchart of the Diffusion Model

The process of continuously subtracting the noise predicted by the noise predictor from the random noise eventually results in a brand-new image being generated from a completely random noise input [5]. This process of generating an image from noise is referred to as reverse diffusion.

The training of the noise predictor, on the other hand, is known as forward diffusion. Its principle is depicted in Figure 2:

By using both the original image and a version of the image with added noise as input, the noise predictor is trained to output the added noise. Thus, the essence of Diffusion technology lies in teaching neural networks to reverse the process of adding noise to images, thereby enabling image generation. Its principle is depicted in Figure 2:

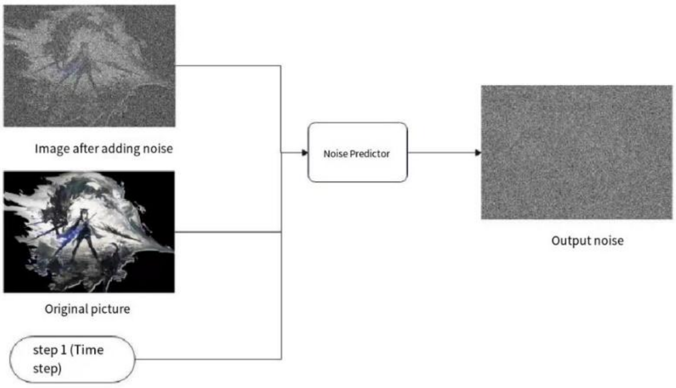


Fig. 2. Reverse Propagation Flowchart of the Diffusion Model

To quantitatively evaluate the quality of images generated by a model, researchers primarily use Maximum Likelihood Estimation (MLE) to assess both the model and the generated images. The idea behind Maximum Likelihood Estimation is that, for a given set of observed data x, we aim to find the parameter θ* among all possible θ$_1$, θ$_2$, ..., θ$_n$ that maximizes the probability of generating the observed data. This leads to the formula 1:

$$\theta^* = \underset{\theta}{argmax} p(x|\theta) \qquad (1)$$

In the training process of AI-generated images, we randomly sample mmm data points from the real image distribution $Pdata(x)$, then compute the probability of each data point occurring under the model's generated image distribution $P_\theta(x)$, and multiply these probabilities together. The θ that maximizes this final probability is the parameter θ^* that makes the neural network produce images closest to real images, as shown in formula 2:

$$\theta^* = \underset{\theta}{argmax} \prod_{i=1}^{m} P_\theta(x^i) \qquad (2)$$

FID (Fréchet Inception Distance) is a metric used to evaluate the quality of generative models.

FID is calculated based on the Fréchet distance between two probability distributions: one representing the distribution of real images and the other representing the distribution of images generated by the generator model. Specifically, FID quantifies this gap by computing the feature representations of these two distributions within a pre-trained Inception network and then calculating the Fréchet distance between these representations.

A lower FID value indicates a smaller gap between the generated images and the real image distribution, thus indicating better performance of the generative model. Based on the FID value, the performance of the generative model can be interpreted and evaluated.

The calculation of FID typically involves the following steps: computing the feature means and covariance matrices for both the real image dataset and the generated image dataset, as shown in formulas 4 and 5:

$$\left\|\mu_{real} - \mu_{gen}\right\|_2^2 \qquad (4)$$

$$d^2\left(\Sigma_{real}, \Sigma_{gen}\right) = \left\|\Sigma_{real} + \Sigma_{gen} - 2\left(\Sigma_{real}\Sigma_{gen}\right)^{\frac{1}{2}}\right\|F \qquad (5)$$

Then, the Fréchet distance between them is calculated, reflecting the similarity between the two distributions.

$$\mathrm{FID} = \left\|\mu_{real} - \mu_{gen}\right\|_2^2 + Tr\left(\Sigma_{real} + \Sigma_{gen} - 2\left(\Sigma_{real}\Sigma_{gen}\right)^{\frac{1}{2}}\right) \qquad (6)$$

Table 1. Functional Modules Table

Neural Network Layers	Function
Preprocessing Layer	Uniformly process the size, pixel, and other attributes of the images being trained
Noise Predictor (Forward Diffusion Layer)	By inputting both the original image and the image with added noise, it trains its ability to predict the noise.
Reverse Diffusion Layer	By combining with the noise output by the noise predictor, it achieves the effect of generating an image from complete noise.
Dreambooth Model	Inject custom art style into the existing image drawing model.
FID（Fréchet Inception Distance）	FID is used to calculate the closeness between the distribution of generated images and that of real images.
Maximum Likelihood Estimation Layer	Among the existing parameters, Maximum Likelihood Estimation (MLE) is utilized to identify a set of neural network parameters that make the distribution of images generated by the model closest to the distribution of real images.

3. Methodology

This section primarily introduces the primary modules and hierarchical structure of the image generation system. The system comprises the Rendering Module, Style Injection Module, and Data Assistance Module.

3.1. System Implementation

The system adopts the stable diffusion model as its fundamental architecture, utilizing the open-source model darkSushiMixMix as the initial training model. These two open-source components have already implemented the functions of the Forward Diffusion Layer and Reverse Diffusion Layer, enabling the system to randomly generate images with various styles and details, fine-tuned according to different prompt words. However, the current primitive system lacks optimization for specific art styles and suffers from severe overfitting and extreme lack of diversity when inputting numerous prompt words. The purpose of this system is to build upon these two modules, enabling the neural network to learn the prompt word "Arknights" and, upon inputting this prompt, output images resembling the art style of the "Arknights" series of illustrations. Additionally, the system aims to inject the essence of the Arknights art style into the model.

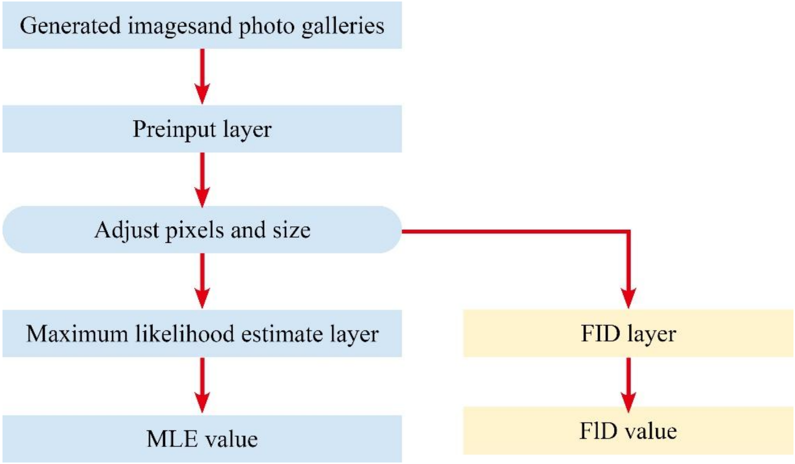

Generated imagesand photo galleries
Preinput layer
Adjust pixels and size
Maximum likelihood estimate layer
MLE value
FID layer
FID value

Fig. 3. Flowchart of the Data Assistance Module

As mentioned above, Figure 3 illustrates the operation process of the model evaluation system, which inputs the image, initializes, and calculates MLE and FID.

3.2. Data Assistance Module

The Data Assistance Module comprises the Preprocessing Layer, MLE Calculation Layer, and FID Layer. The input of this module is the generated images and a real image set, and the output is the MLE and FID values for the generated images compared to the real image set.

The Preprocessing Layer aims to standardize the dimensions and pixels of input images, simplifying MLE and FID value calculations. First, it uses the Python imread() command from cv2 to load generated and real images in color format. Then, the input images are resized to 512*512 pixels, enabling the MLE and FID layers to extract more representative features and enhance the reliability of the final MLE and FID outputs.

The MLE Layer takes the images processed by the Preprocessing Layer as input. Its output is the KL divergence between these images and the image set. After extracting the feature vectors, the MLE value is calculated. Since the MLE value equals the KL divergence value [6], the program directly computes the KL divergence between the generated and real image sets.

The FID Layer receives images from the Preprocessing Layer, extracts their feature vectors, and calculates the FID value. In FID calculations, the feature vector distributions of real and generated images are regarded as two high - dimensional Gaussian distributions, and their Fréchet Distance is computed. This approach allows FID to offer a more comprehensive evaluation of image quality.

3.3. Image Rendering Module

This module is composed of the open-source generative model framework, Stable Diffusion, and the generative model DarkSushiMixMix_225D, which is a Diffusion Model within the Stable Diffusion framework. In the Stable Diffusion framework, various Diffusion Models can be employed to achieve diverse artistic styles for image generation, such as realism, ink painting, abstraction, and more [7]. These models are typically implemented based on deep neural networks, necessitating the use of corresponding pre-trained models for parameter initialization and inference [8].

The Style Injection Module leverages Dreambooth and Embedding techniques to inject desired keywords and artistic styles into DarkSushiMixMix_225D without retraining the model from scratch. Current large AI models, also known as foundation models, have been trained on billions of data points, making them highly generalized, versatile, and practical for various drawing scenarios. However, these models often struggle to meet specific requirements for detail control or particular drawing styles. To address this issue efficiently in terms of time and cost, researchers have proposed fine-tuning techniques for large models, including Dreambooth and Embedding.

Dreambooth, introduced by Google in August 2022, is a novel deep learning technique for fine-tuning existing text-to-image models [9]. Its goal is to generate more detailed and personalized output images by fine-tuning pre-trained text-to-image models. It enables users to "feed" custom image information to the model and generate diverse images through simple names and prompts, while preserving the critical visual features desired by the user. By fine-tuning the model with just 3-5 images and text prompts, Dreambooth can effectively generate new images that accurately replicate the appearance of the input images. This study utilizes the Dreambooth website provided by Google. After providing the training set, the corresponding Dreambooth file is automatically generated after a period of time, which can then be imported for use [10].

In a trained model, the Text Coder functions as a dictionary, combining input text and word vectors to guide the UNET in initializing noisy images [11].

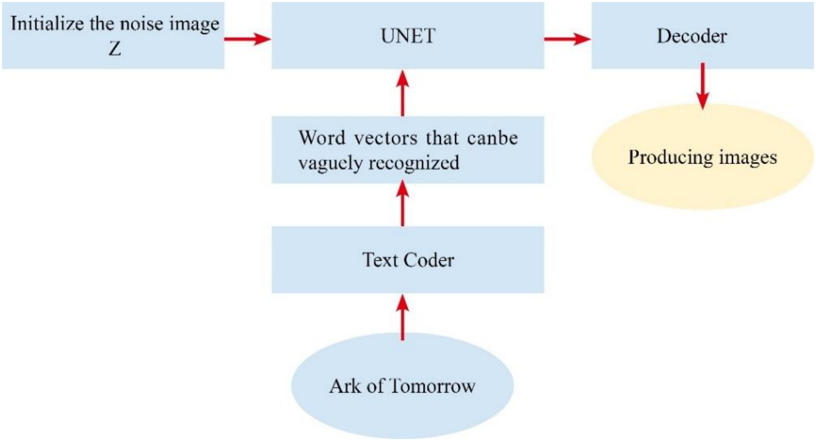

Initialize the noise image Z
UNET
Decoder
Word vectors that canbe vaguely recognized
Producing images
Text Coder
Ark of Tomorrow

Fig. **4**. Flowchart of Embedding Algorithm

However, when encountering novel keywords that are not part of the Text Coder's vocabulary, traditional approaches would require retraining the entire model, significantly reducing its flexibility. To address this, the Embedding algorithm is introduced. It trains the Text Coder to find word vectors that share similar characteristics and styles with the new keywords, enabling fine-tuning of the model without altering its fundamental structure or Text Coder.

4. Model Evaluation

This section generated 100 images using the original generative model, and calculated 100 MLE and FID values respectively. Through the scatter plot, an intuitive distribution of image quality was obtained [12]. The corresponding indicators for evaluating model performance were then obtained by calculating the average values of MLE and FID. After that, the Embedding and Dreambooth technologies, which can cleverly avoid the difficulties of large model training, such as high resource consumption and poor effect, were introduced [13].

4.1. Introduction to the Real Image Dataset

The system employs 200 images of Arknights characters collected from the authoritative source station, Prts. Arknights, as the training targets for the model to generate images, serving as the real image dataset. To ensure the stability of model training, all selected images for the real image dataset exhibit the following characteristics: 1) They belong to the mature and consistent art style of Arknights. 2) There are no apparent drawing inconsistencies or errors. 3) The backgrounds are clean, devoid of redundant interfering elements. All images are in JPG format.

As depicted in Figures 5 and 6, the anime images produced by the original DarkMix model exhibit issues such as unclear hierarchical structures, indistinct art styles, and severe lack of details. In Figure 5, the female character's facial details are severely lacking, with important features like the nose and mouth missing [14]. Figure 6 shows a girl with a bizarre hairstyle that seemingly blends a ponytail with loose hair, and an unnatural connection between the head and body. It is evident that the initial model-generated images have rough craftsmanship, distorted characters, and dull expressions. Additionally, many details in the characters' clothing and hair are lost, making the images aesthetically unpleasing to the average viewer.

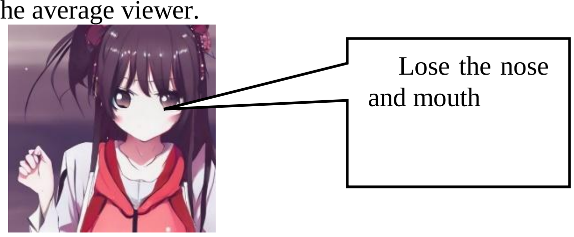


Fig. 5. Generated image of the original model 1

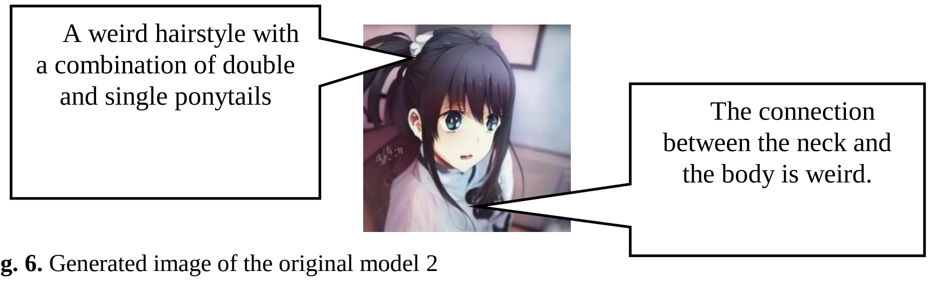

A weird hairstyle with a combination of double and single ponytails
The connection between the neck and the body is weird.

Fig. 6. Generated image of the original model 2

We randomly generated 100 images with the original model and input them into a data auxiliary module to calculate MLE and FID. The results are presented in the scatter plot in Table 4-4, where the horizontal axis represents the maximum likelihood estimate (MLE) values obtained by comparing the 100 generated images with the real image dataset, and the vertical axis corresponds to FID values. Since the maximum MLE is equivalent to the minimum KL divergence, we use KL divergence as a substitute for MLE for convenience. It is evident that points closer to the lower left corner indicate higher similarity between generated and real images. For the original model, MLE values mostly ranged from 0.5 to 1.5, with an average of 1.36882, while FID values were generally distributed between 95 and 115, averaging 105.6795. Using the MLE and FID scatter plot provides a more intuitive evaluation of model performance.

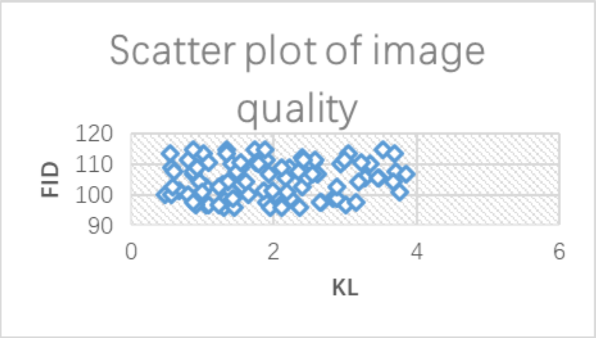
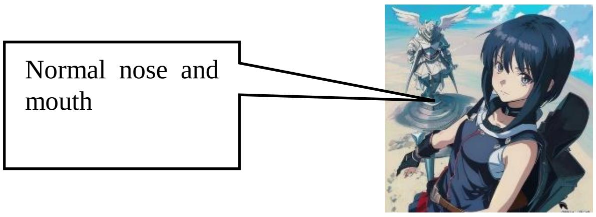


Fig. 7. Scatter plot of image quality

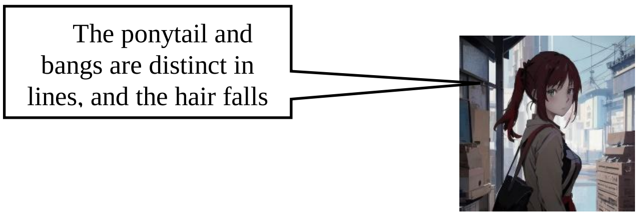
4.2. Output Image Set and Performance of the Current Model

After injecting the art style using the aforementioned methods, as depicted in Figures 8 and 9, we observe improved image detail richness, clearer and smoother lines, and more vivid character expressions [15]. Figure 8 shows a female character with normal noses and mouths, distinct lines, and high recognizability. Figure 9 reveals a girl with a three-dimensional bangs and ponytail, and naturally flowing hair, eliminating the chaotic appearance. It is evident that the images generated by the existing model exhibit refined craftsmanship, well-defined body curves, and clear expressions. Additionally, more details are present in the characters' clothing and hair, making the images aesthetically pleasing to the average viewer.



Normal nose and mouth

Fig. 8. Output Image 1 from the Current Model



The ponytail and bangs are distinct in lines, and the hair falls

Fig. 9. Output Image 2 from the Current Model

As shown in Figure 10, it can be observed that the anime images generated by the current model have the highest distribution of MLE values within the range of 0.2 to 1, with an average of 1.15472. The FID values are generally distributed within the range of 80 to 120, with an average of 100.9565. It can be noticed that both the average values of MLE and FID have decreased, and most of the scatter points are closer to the lower left corner compared to before. Therefore, it can be preliminarily judged that the current model is superior in performance to the original model.

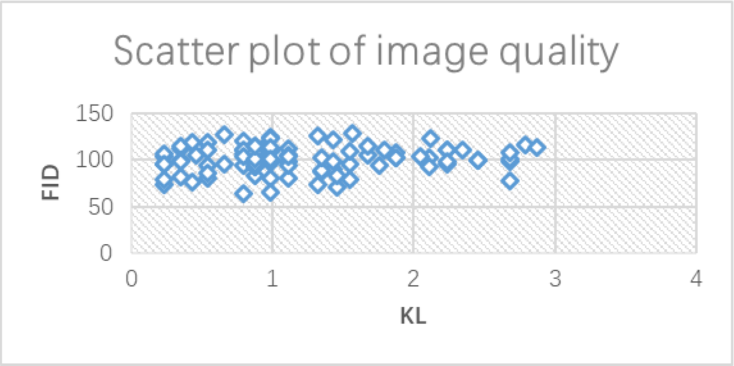


Fig. 10. Scatter plot of image quality

4.3. Implementation of Data Assistance Module

The data assistance module loads and preprocesses images from a folder, unifies their sizes, and passes the image list to the FID and MLE calculation layers. Finally, it generates a corresponding scatter plot based on the data provided by the FID and MLE layers. The functional flow of this module is illustrated in Figure 11.

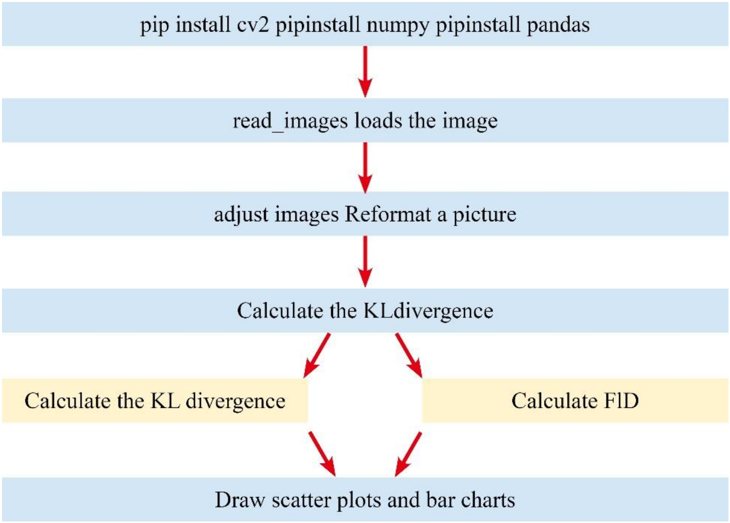

pip install cv2 pipinstall numpy pipinstall pandas
read_images loads the image
adjust images Reformat a picture
Calculate the KLdivergence
Calculate the KL divergence
Calculate FlD
Draw scatter plots and bar charts

Fig. 11. Data Assistance Module Flowchart

The primary function of the preprocessing layer is to uniformly process the images from the input data assistance module by reading them into an image list from a specified folder using the cv.imread() function and resizing each image to 512*512 pixels [16]. After reading the images, you can call the custom testing function show_image(image_paths) to display the images, verifying whether they have been successfully loaded into the system. The function show_image() creates a window to display the image and waits for any user input before closing the window. Once the read_images() function is verified to be working correctly, the image list is fed into the adjust_images() function for preprocessing. This function iterates through each image in the list and resizes it to 512*512 pixels using the cv2.resize() function, finally placing all the images into a new list.

The FID (Fréchet Inception Distance) and MLE (Maximum Likelihood Estimation, approximated by KL divergence) calculation layers are primarily designed to convert the list of preprocessed images into one-dimensional arrays, extract their feature vectors, and utilize dedicated FID and KL divergence computation functions to obtain the FID distance and KL divergence between images. These metrics are then plotted on a scatter plot for visualization.

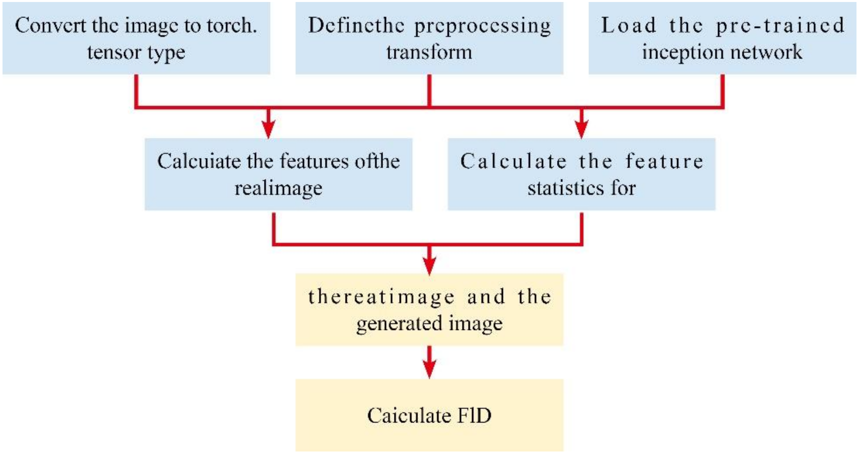

Convert the image to torch. tensor type
Definethe preprocessing transform
Load the pre-trained inception network
Calcuiate the features ofthe realimage
Calculate the feature statistics for
thereatimage and the generated image
Caiculate FID

Fig. 12. FID Layer Function Flow Chart

Figure 12 illustrates the operation process of the FID calculation system, which initializes the model for computation, calculates the features of the two images, and computes their FID.

5. Conclusions

The purpose of this system is to optimize the existing image generation model technology, aiming to obtain a model with better image generation performance [17]. Additionally, the system proposes to adopt both FID and MLE as indicators for evaluating the quality of the generated model and uses these two metrics to optimize the model. Based on the original diffusion model framework, Dreambooth and Embedding technologies are employed as optimizations, addressing issues such as long training time, difficulty in training, high energy consumption, and poor results successfully lightening the problem.

Specifically, Dreambooth endows the model with more powerful detail depiction and style transfer capabilities [18]. The Embedding technology successfully enables the system to create content in a specific style based on the keyword "Arknights."

Ultimately, the images generated by this model successfully surpass the original model's images in terms of FID and MLE indicators, demonstrating that this system has better image generation performance.

The optimization of the original image generation system using Embedding and Dreambooth technologies has been achieved, enabling the model to surpass the original model's image quality in specific art styles. A method for evaluating the model's performance using a combination of MLE and FID with scatter plots has been proposed. After optimization, the MLE of the model decreased by approximately 2.4, and the FID decreased by approximately 5, indicating that the distribution of images generated by the optimized model is closer to the real distribution of images than before.

Overlapping the two scatter plots reveals that most of the points from the optimized model are significantly closer to the bottom left and more dispersed than those of the original model [19]. This suggests that the distribution of images generated by the optimized model is closer to the real distribution of images but there are instances where the quality of generated images declines after adjustment, indicating reduced stability in the new model. It is speculated that this is due to overfitting of the model to certain specific images. Thus, while lightweight injection techniques like Dreambooth and Embedding significantly improve training efficiency and avoid the high costs of retraining large models, they may also somewhat reduce model stability [20].

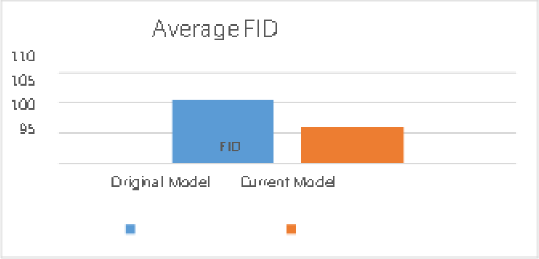


Fig. 13. Average FID

As shown in Figure 13, after optimization with Dreambooth and Embedding, the average FID decreased by approximately 5. FID, as an important indicator of the distance between two images (i.e., image similarity), indicates that the lower the FID, the more similar the two images are, meaning the generated image is closer to a real image. Table 5-1 proves that from the FID evaluation metric, the image generation quality of the current model is slightly higher than that of the original model, demonstrating better specialized generation capabilities.

As shown in Figure 14, after optimization with Dreambooth and Embedding, the average KL decreased by approximately 0.2. KL, as an indicator of the similarity between an image and the distribution of an image set, is also a crucial metric for image similarity. A lower KL indicates a higher likelihood that the image is real, meaning the generated image is closer to a real image. Table 5-1 proves that from the KL evaluation metric, the image generation quality of the current model is slightly higher than that of the original model, excellently fulfilling the task of generating images in specific art styles.

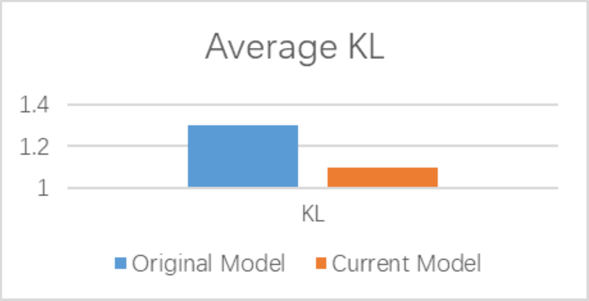


Fig. 14. Average KL

Our research still has many limitations, such as overfitting issues and room for improvement in image quality. To address these problems, we have considered the following improvement methods, such as increasing the model size and enhancing the diversity of the training set images.

Acknowledgment. The research is conducted under the Humanities and Social Sciences Planning Fund Project of the Ministry under the title of “Research on the Social Media Communication of Mainland China's Image in Taiwan”, Project No. 24YJAZH122, with special thanks.

References

1. Alex Krizhevsky, Ilya Sutskever, Geoffrey E.Hinton. ImageNet Classification with Deep Convolutional Neural Networks. (2012).
2. Dennis Elbrachter, Dmytro Perekrestenko, Philipp Grohs, and Helmut Boelcskei. 2021. Deep Neural Network Approximation Theory. IEEE Transactions on Information Theory (2021).
3. Daneshfar, Fatemeh, Ako Bartani, and Pardis Lotfi. "Image captioning by diffusion models: a survey." *Engineering Applications of Artificial Intelligence* 138 (2024): 109288.
4. Paiva, José Carlos, José Paulo Leal, and Álvaro Figueira. "Comparing semantic graph representations of source code: The case of automatic feedback on programming assignments." Computer Science and Information Systems 00 (2024): 4-4.
5. Brokman, Jonathan, et al. "MONTRAGE: Monitoring Training for Attribution of Generative Diffusion Models." *European Conference on Computer Vision*. Cham: Springer Nature Switzerland, 2024.
6. Alimisis, Panagiotis, et al. "Advances in diffusion models for image data augmentation: A review of methods, models, evaluation metrics and future research directions." *Artificial Intelligence Review* 58.4 (2025): 1-55.
7. Osorio, Pedro, et al. "Latent diffusion models with image-derived annotations for enhanced ai-assisted cancer diagnosis in histopathology." *Diagnostics* 14.13 (2024): 1442.
8. Blake Bullwinkel, Kristen Grabarz, Lily Ke, Scarlett Gong, Chris Tanner, and Joshua Al-len. 2022. Evaluating the fairness impact of differentially private synthetic data. arXiv preprint arXiv:2205.04321 (2022).
9. Turner, D. Bruce. "A Diffusion Model for an Urban Area." Journal of Applied Meteorolo-gy and Climatology, vol. 3, no. 1, 1964, pp. 83–91.
10. Rahman, Abidur, et al. "Implementation of diffusion model in realistic face generation." *2024 9th International Conference on Image, Vision and Computing (ICIVC)*. IEEE, 2024.
11. Rogers, Everett M. "A Prospective and Retrospective Look at the Diffusion Model." Jour-nal of Health Communication, vol. 9, no. S1, 2010, pp. 13–19.
12. Juan Miguel Lopez Alcaraz and Nils Strodthoff. 2022. Diffusion-based time series imputation and forecasting with structured state space models. arXiv preprint arXiv:2208.09399 (2022).
13. Kidder, Benjamin L. "Advanced image generation for cancer using diffusion models." *Biology Methods and Protocols* 9.1 (2024): bpae062.
14. Samy Bengio and Yoshua Bengio. 2000. Taking on the curse of dimensionality in joint distributions using neural networks. IEEE Transactions on Neural Networks (2000).
15. Croitoru, Florinel-Alin, et al. "Diffusion Models in Vision: A Survey." IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 45, no. 9, 2023, pp. 10850–10869. DOI: 10.1109/TPAMI.2023.3261988.
16. Moser, Brian B., et al. "Diffusion models, image super-resolution, and everything: A survey." *IEEE Transactions on Neural Networks and Learning Systems* (2024).

17. Tomer Amit, Eliya Nachmani, Tal Shaharbany, and Lior Wolf. 2021. SegDiff: Image segmentation with diffusion probabilistic models. arXiv preprint arXiv:2112.00390 (2021).
18. Peter, Ojonugwa Oluwafemi Ejiga, Md Mahmudur Rahman, and Fahmi Khalifa. "Advancing AI-Powered Medical Image Synthesis: Insights from MedVQA-GI Challenge Using CLIP, Fine-Tuned Stable Diffusion, and Dream-Booth+ LoRA." *Conference and Labs of the Evaluation Forum*. 2024.
19. Turker, Anil, and Ender M. Eksioglu. "3D convolutional long short-term encoder-decoder network for moving object segmentation." *Computer Science and Information Systems* 21.1 (2024): 363-378.
20. Andrew Campbell, Joe Benton, Valentin De Bortoli, Tom Rainforth, George Deligiannidis, and Arnaud Doucet. 2022. A continuous time framework for discrete de-noising models. arXiv preprint arXiv:2205.14987 (2022).

ChaoChun Shen is currently Professor in the School of Art Design and Media, Department of Network and New Media, Sanda University. His research interests include AIGC in new media, Social media big data, content algorithm, News communication analysis. He has served as chief editor and general manager of new media company, there are more than 40 related papers, including SCI, EI, CSSCI. He is member of IEEE.

Shun Nian Luo graduated with a Doctor of Philosophy in Information Management from Dayeh University, Taiwan. He has previously worked at institutions in Taiwan such as Chinfon Securities, Mitsukoshi-MetLife Insurance, and Taojiang University of Science and Technology. Since 2020, he has been working at the School of Information Science and Technology, Shanghai Sanda University. He has won awards including the Third Prize in the College Middle-aged Group of the 7th Shanghai Teachers' Calligraphy Competition and the Excellent Instructor Award in the Practice Competition of Huawei ICT Competition - Shanghai Division. His research areas include information security, system analysis and design, Unity game engine, 3D animation, robot ROS (Robot Operating System), and embedded systems.

Ling Fan is an associate professor at the School of Shanghai Technical Institute of Electronics & Information College of designt and art. Her areas of expertise include gamified teaching, AI-based art creation, new media communication, artificial intelligence, and art education.

Chenlin Dai is a student who has been admitted to the master's program at the University of Southern California (USC) this year. He majored in software engineering during his undergraduate studies.

Received: December 12, 2024; Accepted: March 07, 2025.