ISSN 2616-5767 Vol.7, Issue 6: 145-154, DOI: 10.25236/AJETS.2024.070621

Design of art calligraphy image generation based on the diffusion model

Peiqi Yuan[1,*], Yekuan He[2]

[1]*Physics Department, Capital Normal University, Beijing, 100080, China*
[2]*Institute of International Education, Guangzhou College of Technology and Business, Foshan, 528100, China*
[*]*Corresponding author*

Abstract: *With the advancement of image generation technology, there has been a growing interest in automating the creation of stylized fonts using computers. Traditionally, designers have had to create multiple font styles to meet client demands, which required significant human and material resources. To address this challenge, we propose the Font Model Manager (FMM) model. This paper introduces the Type ControlNet and Type Condition Information Model, which enhance the precision of the font generation process and improve the accuracy of the generated images. Additionally, FMM incorporates a Type Image Compression Model, which reduces the computational time and storage costs required for training by compressing images, thereby increasing training efficiency. Furthermore, we have developed a comprehensive, accurately labeled, and high-resolution Typeface Image dataset, filling a gap in the market's available data. To evaluate the model's effectiveness, we employed Peak Signal-to-Noise Ratio (PSNR) as the primary metric, achieving an average value of 9.52 dB, which surpasses the performance of comparable models on the same dataset and ensures the visual quality of the generated font images. Overall, these advancements significantly improve the accuracy and efficiency of stylized font generation, meeting the market's demand for diverse font styles.*

Keywords: *FMM model, image processing, image compress model, diffusion model, controlnet*

1. Introduction

The use of deep learning to generate stylized fonts provides designers and artists with innovative tools and technical support, while also fostering greater creativity in industries such as brand marketing and educational training, thereby driving the development of these related sectors.

The process of using deep learning to generate stylized fonts generally involves data collection, model selection and training, as well as the generation and optimization stages. In the past, CNNs were used to generate stylized fonts by employing convolutional neural network models to learn the features and styles of input fonts, thereby generating new font images or glyphs with similar styles. However, the performance of CNNs heavily depends on the tuning of various hyperparameters, such as learning rates, network layers, and the number of filters, making the optimization process complex and reliant on extensive experimentation and experience. GANs [1][2], on the other hand, generate stylized fonts by learning the distribution characteristics and styles of input font samples to create new font images or glyphs with similar styles. However, GANs can sometimes be limited by the diversity and coverage of training data, resulting in generated fonts that lack diversity or innovation.

To address these challenges, we propose the Font Flow Mark Model (FMM), a diffusion-based model for generating stylized fonts. FMM offers a more stable generation process, high-quality and realistic image outputs, and advantages in data retention and theoretical foundation. Unlike GANs, FMM generates images by controlling the incremental addition of random noise, leading to a more stable generation process and avoiding issues like mode collapse and unstable training. The images generated by FMM typically exhibit high quality and realism, capturing the details and complex stylistic features of the input data while maintaining a natural and lifelike appearance. Additionally, FMM does not require paired input-output data, making it more flexible and applicable to various types of generation tasks in practical scenarios. With greater stability and the avoidance of mode collapse, FMM utilizes the diffusion process to control the gradual addition of noise, providing a more stable training process and consistent generation results.

ISSN 2616-5767 Vol.7, Issue 6: 145-154, DOI: 10.25236/AJETS.2024.070621

Through ablation studies, comparative experiments, and evaluation metric tests, both qualitative and quantitative results indicate that the FMM model excels in generating fonts with stylization and artistry. The model quickly adapts to and integrates different styles, and compared to existing text-guided models, FMM demonstrates breakthrough developments in clarity and diversity of generated fonts. Our contributions in this study are as follows:

- We constructed the Typeface Image dataset, a medium-sized, high-definition icon dataset that is finely categorized and annotated.
- We designed the Type ControlNet, significantly enhancing the model's control over design details and image information.
- We introduced the Font Flow Mark Model, incorporating the Typeface Optimization Module to enhance the controllability and diversity of generated fonts, enabling more accurate style transfer.

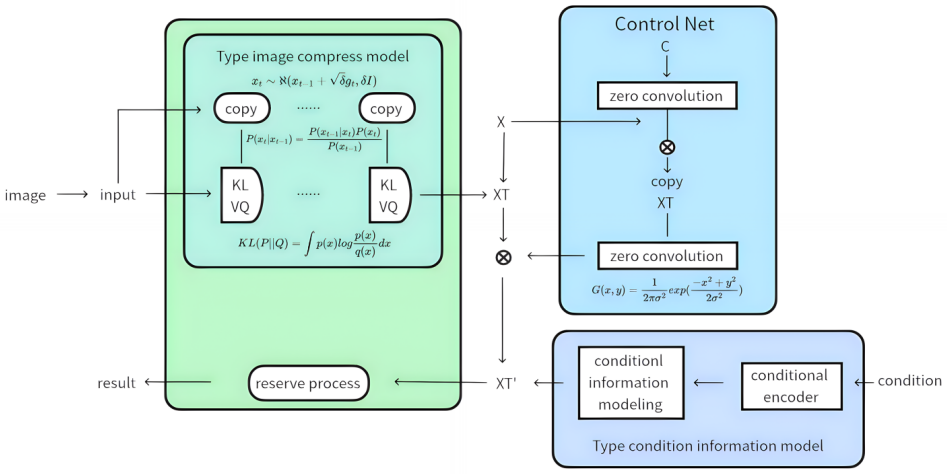

Figure 1: Flow chart of the experiment

2. Related work

2.1 Development of Stylized Font Technology

Historically, AI-generated methods for stylized font design have predominantly relied on GAN and CNN-based approaches. Colin Chen [3] introduced Cycle-GAN, which utilizes adversarial training between the generator and discriminator to continuously improve the quality of generated images. Although GANs have shown remarkable performance in style transfer and generalization, there remains room for improvement in their generation efficiency. Umi Laili Yuhana [4] proposed the Multi-Scale Information CNN, which extracts features from input font images using convolutional layers and combines them with style transfer techniques to generate stylized fonts. While CNNs excel in image recognition and processing, achieving ideal results in stylized text generation typically requires the integration of additional technologies. Unlike these traditional methods, our approach introduces an image compression module to reduce computational resources during training and incorporates conditional information modeling to enhance the precision of stylized image generation.

2.2 Diffusion Models

The evolution of image diffusion models, from the initial DDPM to Diffusion GAN and then to modern Stable Diffusion, reflects the progressive advancement and optimization of image generation technologies. The Resampling Diffusion Model proposed by Wing-Fung Ku [5] generates images by progressively adding Gaussian noise and learning how to reverse it, demonstrating exceptional performance in producing high-quality images, though it presents challenges in training difficulty. Building on DDPM, Luan Thanh Trinh [6] introduced the Latent Denoising Diffusion GAN, which combines the discriminator of GAN with the denoising process of DDPM, accelerating the denoising process and achieving significant advantages in image quality and diversity. However, this increases the complexity of the process. Stable Diffusion further optimizes the approach by iteratively adding random

ISSN 2616-5767 Vol.7, Issue 6: 145-154, DOI: 10.25236/AJETS.2024.070621

noise to input images, showcasing excellent generation capability and training stability. Nevertheless, it still faces challenges related to the complexity of hyperparameter tuning and inference costs. Our model offers relatively straightforward inference costs and parameter control.

2.3 Skeletal Control Networks

Controlling Image Diffusion Models optimize the generation process and enhance the model's controllability and generation capability by introducing additional control conditions or guiding signals. These control conditions encompass several aspects, such as text prompt techniques: Alec Radford [7] proposed CLIP, which optimizes generation results by adjusting CLIP features and modifying the cross-attention mechanism, although it still has limitations in training data diversity and the richness of generated data. Spatial information processing methods: Oran Gafni [8] introduced Make-A-Scene, which encodes segmentation masks as tokens or maps them into localized token embeddings to control image generation, but it consumes significant computational resources and is challenging to control the generation results; image information control methods: Rinon Gal [9] proposed Textual Inversion, which fine-tunes diffusion models using a small number of example images to personalize the generated content, though it presents difficulties in control. To enhance generation capability and stability, our FMM model uses 3D elements to assist in 2D image generation, resulting in richer data outcomes and improved control over generation effects.

3. Models

TI (Typeface image) data set is constructed, and the weights of each style are trained after the data is screened by AI and manual. FFM conforms to Gaussian distribution and uses Bayesian methods to improve the robustness and generalization ability of the model. TC (Type controlnet) is introduced to control the contour of the generated result, and TOM (typeface optimization module) is introduced to save training time and control style details.(as shown in Figure1)

3.1 DataSet

The initial collection of the dataset relied on Python's Requests and BeautifulSoup libraries, and captured about 30,000 images from multiple online platforms such as forums, Xiaohongshu, and JD.com. This experiment focused on stylized fonts, so after data collection, an initial manual screening was performed to eliminate images that were not relevant to the stylized font theme.

Figure 2: Display of the Chinese style pattern, handwriting style, decorative style, bronze style, science fiction style, comic book style, hip hop style and flower style dataset

After the initial screening, the images in the dataset also contain some useless junk videos and advertisements. We used deep learning models such as Mask R-CNN for image processing, applied image segmentation technology to identify and eliminate large color block areas that accounted for more than 50% of the image, and processed advertisements and mosaics in the image. After this series of cleaning operations, the dataset has about 20,000 images remaining and is ready for the experimental phase.

ISSN 2616-5767 Vol.7, Issue 6: 145-154, DOI: 10.25236/AJETS.2024.070621

In the post-processing phase of the dataset, we used Python's scikit-image library to size the images, sizing all images to a multiple of 512 pixels on one side to ensure that the images have uniform size and proportion in subsequent processing. The Apowersoft Watermark Remover was used to remove the watermark from all the images and keep the original content and quality of the images as much as possible. After the above processing, the remaining 2w images were further classified. According to the style of the images, they are divided into 8 categories: national style, handwriting style, decorative style, bronze style, science fiction style, comic style, hip hop style, and flower style (some of which are shown in Figure2). Each category contains 2000 images for the experimental section.

In order to improve the accuracy of image classification, we used the DeepDanbooru project developed based on EfficientNet proposed by Mingxing Tan[10], and then developed the WD1.4 tagger. WD1.4 labels and classifies images efficiently through the process of label extraction, model comparison and weight ranking.

In the model training phase, we divided the data set into training set, verification set and test set in a ratio of 7:2:1. The L1 loss function is used to evaluate the difference between the generated font and the target font during training, and the Adam optimizer is selected to dynamically adjust the learning rate. The total number of training steps is set at 500,000. In order to effectively verify the model performance and prevent overfitting, we conduct multiple rounds of training, and evaluate the quality of the generated ICONS through verification sets after each round of training. By gradually adjusting hyperparameters such as learning rate and batch size, the training process is optimized, thus improving the accuracy and robustness of the model.

3.2 Main frame

3.2.1 Training frame

The core idea of FFM optimization based on DDPM is to learn the data distribution by simulating the data diffusion process, which is composed of forward diffusion process and backward de-noising process. The forward process is to gradually add noise to the original data through a series of random steps, and finally obtain a completely random noise sample. Each step in this process adds Gaussian noise to the current state of the data. Specifically, suppose the initial data is an image or signal,then the data after t step diffusion x_t can be expressed as: $x_t = \sqrt{\bar{a}_t}x_0 + \sqrt{1-\bar{a}_t}\epsilon$. $\epsilon \sim \aleph(0, I)$represents noise taken from a standard Gaussian distribution, $\bar{a}_t$ is the accumulation decay function, usually defined as: $\bar{a}_s = \prod_{s=1}^{t} a_s$,$a_s$is the attenuation coefficient of each step. The Gaussian distribution is used as a conditional distribution to describe the evolution of data points from one time step to the next.Specifically, suppose that at time step t, the distribution of data points is x_t, and its update can be expressed by the following Gaussian formula:

$$x_t \sim \aleph\left(x_{t-1} + \sqrt{\delta} \cdot g_t, \delta \cdot I\right) \tag{1}$$

x_{t-1} is the data point of the previous time step $t-1$, δ is the hyperparameter that controls the size of the diffusion step, and g_t is a random vector sampled $\aleph(0, I)$ from the standard normal distribution.At each time step t, Bayes' theorem is used to update the distribution of data points based on the known conditional distribution and the data points of the previous time step. Specifically, if the x_{t-1} data point of the previous time step $t-1$,x_t is the data point of the current time step t, Bayes' theorem can be expressed as:

$$p(x_t|x_{t-1}) = \frac{p(x_{t-1}|x_t)p(x_t)}{p(x_{t-1})} \tag{2}$$

$p(x_{t-1}|x_t)$ is the probability x_{t-1} under a given condition x_t, described by a Gaussian distribution, $p(x_t)$ is the prior distribution x_t at the current time step t, assumed to be a standard Gaussian distribution,$p(x_{t-1})$ is the prior distribution at the previous time step $t-1$, calculated by forward propagation.The Gaussian distribution of the forward process enables the FMM to perform the addition and removal of noise stably, which is closer to the original style and more conducive to style control. The Bayesian method allows the prior distribution of parameters to be introduced into FMM, and the prior of the initial data distribution can be modeled, rather than just relying on a single deterministic initial distribution, which is more flexible.

3.2.2 Generative process

The goal of the FMM backward de-noising process is to gradually recover x_t from the noise state to the original data x_0. This process also depends on the Gaussian distribution. At each step t, the model

ISSN 2616-5767 Vol.7, Issue 6: 145-154, DOI: 10.25236/AJETS.2024.070621

predicts the noise ϵ_t and uses that prediction to backsample x_{t-1}.

$$x_{t-1} = \frac{1}{\sqrt{a_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{a}_t}}\epsilon_t\right) + \sigma_t z \tag{3}$$

$z \sim \aleph(0, I)$ is additional Gaussian noise, β_t and α_t is a parameter calculated based on a predefined time step t. In backward inference, we usually focus on the distribution of the model's parameters or hidden variables. For example, if we want to infer what the initial random noise distribution of the model looks like when it generates the data, then this distribution is our target. The backward process also satisfies the Gaussian distribution, which makes FMM better learning ability, better generation effect, and more conducive to the generation of stylized fonts.

3.2.3 Typeface optimization module

Although some methods exist to reduce the number of sampling steps for diffusion models, such as DiffusionGAN and DDPM, training a good diffusion model still requires a lot of GPU resources, mainly because the training and inference process of the model is based on the pixel space. In order to solve the problem of DDPM, FMM adopts the image compression module to optimize it by using the autoencoder as the sensing image compression module, in which the autoencoder includes an encoder and a decoder. The encoder is responsible for compressing the image x to a potential representation z, and the decoder reconstructs the potential representation to the image space. The autoencoder introduces KL[11][12] and VQ[13][14] constraints during training, and retains the spatial dimension of the image. In stable diffusion, VQ constraints are often used to ensure that the generated sample falls in a discrete coding space rather than a completely continuous latent space, which can be achieved by mapping the generator's output to a fixed number of discrete points set.

KL constraints are often used to ensure that the difference between the distribution of the generator output and the target distribution is minimized. In stable diffusion, KL constraints can help generators get closer to the target distribution when generating data, thus improving the quality and authenticity of generated samples. In probabilistic modeling, KL divergence is used as a part of the loss function in the generative model. Its theoretical significance is to measure the difference between two probability distributions. The greater the KL divergence, the greater the difference between them. When the KL divergence is small, it indicates that the degree of difference between the two is small. By minimizing the KL divergence, you can ensure that the generated data matches the real data distribution in statistical properties.

$$KL(P||Q) = \int p(x) \log \frac{p(x)}{q(x)} dx \tag{4}$$

In FFM, both VQ constraints and KL constraints are methods to improve the quality, continuity, and authenticity of generated samples by limiting generator output or optimizing the generation process. They operate at different levels, and VQ constraints focus more on discrete representations of potential Spaces, ensuring that the generated samples have a controllable structure and continuity, effectively preventing the model output from being too scattered. However, KL constraints pay more attention to the degree of fit between the generated data distribution and the target distribution. Improve the authenticity and quality of the generated samples, and enhance the stability and training efficiency of the model.

FMM introduces external condition information into the generation process through the condition information model to guide the generator to generate a specific type of data sample. The Multi-Modality Cross-Attention mechanism proposed by Xi Wei[15] and the Frame-Action Cross-Attention mechanism proposed by Zijia Lu[16] are introduced, and different types of conditional variables are introduced. Through these conditional variables, More refined control and customized generation of generated data can be achieved. The conditional information is encoded by a conditional encoder. After obtaining the conditional encoded information, each layer uses the following formula to calculate attention.$Q = W_Q^{(i)} \cdot \varphi_i(z_t), K = W_K^{(i)} \cdot \tau_\theta(y), V = W_V^{(i)} \cdot \tau_\theta(y)$,$Q$ from feature coding Z_t, K,V from conditional information y.The introduction of TOM improves the training efficiency and further optimizes the design of stylized fonts.

ISSN 2616-5767 Vol.7, Issue 6: 145-154, DOI: 10.25236/AJETS.2024.070621

3.3 Control network module

3.3.1 Model structure

ControlNet proposed by Lvmin Zhang is built on Stable Diffusion, by introducing additional control signals to guide the generation process, the core idea of ControlNet is to add a branch condition in each step of Stable Diffusion. This branch condition takes additional input and translates it into an effect on the behavior of the model.

The weight of the FFM is copied into two identical parts, a "locked" copy and a "trainable" copy. Suppose you lock all the parameters of the training in θ, and then clone it into a trainable copy θ_c, the replicated θ_c uses the ControlNet model and is trained using the external condition vector c,this signal c can be any type of data, such as segmentation, edge, depth, etc.The control signal is processed through a special encoder $f(c;\theta_c)$, where θ_c is the encoder parameter. There are two zero convolution modules in the minimum unit structure of the ControlNet model. They are 1×1 convolution modules, and both weight and bias are initialized to zero. In the training process, zero convolution modules gradually become common convolution layers with non-zero weight, and the weight of parameters is constantly optimized. Keep updating and learning.

ControlNet updates the formula at each time step $t$as follows: $x_{t-1} = x_t + \alpha_t \cdot \epsilon_\theta\left(x_t, t, f(c;\theta_c)\right)$, $\epsilon_\theta\left(x_t, t, f(c;\theta_c)\right)$is the learning target of the diffusion model, namely the predicted noise ϵ, in this formula, α_t is a time-dependent scaling factor used to adjust the process of de-noising.

3.3.2 Canny

Canny Algorithm proposed by Vladyslav Yevsieiev [17] is an accurate edge detection method, whose core steps include noise suppression, gradient calculation, non-maximum suppression, double threshold detection and edge connection.

Noise suppression is the smoothing of the image using a Gaussian filter to reduce noise. The formula of Gaussian filter is: $G(x,y) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$,the gradient calculation σ is to use the Sobel operator to calculate the gradient of the image in the horizontal and vertical directions,gradient magnitude $M = \sqrt{G_x^2 + G_y^2}$,direction $\theta = \arctan\left(\frac{G_x}{G_y}\right)$.

Non-maximum suppression is a non-maximum suppression of each pixel to determine whether it is a local maximum. That is, check for a local maximum in the direction of the gradient. Assuming the θ gradient direction, according to the θ difference, the gradient direction is divided into four main angles (0°, 45°, 90°, 135°), check the neighborhood pixels in these directions to decide whether to keep the current pixel as the edge. Double threshold detection uses two threshold values T_H and T_F to classify edges, in which pixels higher than T_H are considered strong edges, pixels lower than T_F are suppressed and not considered edges, and pixels between T_H and T_F are marked as weak edges and need to be further processed. The edge connection is to connect the weak edge with the strong edge. Only when the weak edge pixel is adjacent to the strong edge pixel, the weak edge pixel is retained, forming a continuous edge. The introduction of Canny algorithm enhances the flexibility and controllability of stylized font design, bringing revolutionary convenience and efficiency to users and designers.

4. Experiment

4.1 Training of data sets

Starting from the data source, the WD1.4 labeler is used to label the original data set, and the results of automatic labeling are reviewed one by one, the inconsistent labels are removed, and the mislabeling or missing labels are corrected. The annotated data set is preprocessed, the image resolution is adjusted, and a variety of data enhancement techniques are applied to enrich the diversity of samples. The addition of new samples helps the model learn more robust feature representation and prevents overfitting in the training process.

During the training process, we selected the most suitable bottom film, adjusted the resolution of the training picture to 512,512, and set save_every_n_epochs to 2 and max_train_epochs to 10 according to the size and complexity of the data set. Set train_batch_size to 1 to ensure that the model can fully learn the feature information in the data. At the same time, we also adopt the strategy of dynamically adjusting

ISSN 2616-5767 Vol.7, Issue 6: 145-154, DOI: 10.25236/AJETS.2024.070621

the learning rate to optimize the training process. At the beginning of training, we set a large learning rate to speed up the convergence of the model. As the training progresses, we gradually reduce the learning rate to avoid excessive fluctuations in the model near the optimal solution. In addition, we also used the TensorBoard visualization tool proposed by Bo Pang[18] to monitor and record the training process in real time, so that we could find and solve potential problems in time.

Finally, after the training was completed, we used an independent test set to conduct a comprehensive and objective evaluation of the model, and conducted detailed analysis and comparison of the model's performance in different categories and different difficulties.

4.2 Ablation experiment

In the in-depth exploration of image generation technology, we carried out a series of rigorous experiments based on FFM model to evaluate and optimize the performance of the model under different artistic styles (as shown in Figure3). Through the group design of the system, this series of experiments gradually integrate a variety of advanced technologies to achieve a significant improvement in image generation quality.

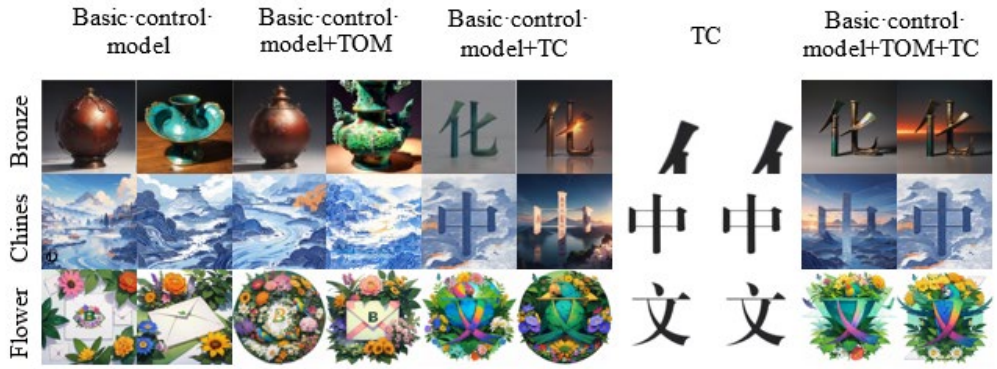

Figure 3: Bronze style, national style, flower style in different control conditions generated pictures

The experiment was divided into four experimental groups, each representing a different stage and depth of technology integration. The first set of experiments, as a benchmark test, independently applied the basic control model. Through this set of experiments, we preliminarily evaluated the performance stability and consistency of the basic control model under different styles. In the second group of experiments, TOM was introduced as the first step of technology fusion, which carried out more detailed and accurate regulation on the generated image content, enhanced the expressive power of image content, improved the controllability of the generation process, and laid the foundation for further integration of subsequent technologies. The third set of experiments integrated TC into the image generation process, used external cues to precisely regulate the structure and details of the image, combined with detailed parameter adjustment, we successfully achieved high-precision control of the generated image, significantly improving the structural clarity and detail richness of the image. In the fourth set of experiments, we build a highly integrated image generation system by combining the basic control model, TOM and TC. The system not only inherits the technical advantages of each component, but also shows excellent performance beyond a single technology under the synergy.

Through the comprehensive evaluation of the generated images, we found that from the first group to the fourth group, the quality of the images showed a significant upward trend, which was specifically manifested as more vivid style, more accurate content and more exquisite details. This result not only validates the effectiveness of technology fusion in the field of image generation, but also provides valuable inspiration for future research direction.

4.3 Index evaluation

In the generation of stylized fonts by font streamer model, the comparison of image evaluation indicators is extremely important. This experiment takes PSNR as an evaluation indicator to explore the difference in the effect of font streamer model on the generation of different styles. PSNR is a measure of the difference between the quality of the image and the original image. The higher the PSNR, the closer the image quality is to the original image.

ISSN 2616-5767 Vol.7, Issue 6: 145-154, DOI: 10.25236/AJETS.2024.070621

$$PSNR = 1 \quad 0 \quad \times \log_{10}\left(\frac{MaxValue^2}{MSE}\right) \tag{5}$$

MSE is the mean square error of the graph, and MaxValue is the maximum value of the image pixel, $MSE = \frac{1}{m \times n}\sum_{i=1}^{m}\sum_{j=1}^{n}[I(i,j) - K(i,j)]^2$,where I(i,j) and K(i,j) are the pixel values of the original image and the processed image at the position (i,j) respectively, and m and n are the height and width of the image.

Table 1: MSE and PSNR results of bronze style, national pattern style and flower style under different control conditions

	Basic control model+TMO		Basic control model+TC		Basic control model+TOM+TC	
	MSE	PSNR	MSE	PSNR	MSE	PSNR
Bronze style	4989.900	11.150 dB	6565.376	9.958 dB	3930.590	12.186 dB
Chinese style	8065.433	9.065 dB	10256.826	8.021 dB	7549.006	9.352 dB
Flower style	9172.921	8.506 dB	9555.811	8.328 dB	8653.149	8.759 dB

According to Table1, it can be concluded that the model shows a high PSNR when generating diversified font styles. This result indicates that the model has achieved a high accuracy in simulating real font styles, and the image quality generated by the model is highly similar to that of real samples.

We plan to further explore optimization strategies for the model architecture, including but not limited to adjustments at the network level, selection of activation functions and customization of loss functions, to further improve the quality and diversity of the generated images. At the same time, more diversified evaluation systems are introduced, such as SSIM, FM Score, etc., to comprehensively and objectively measure the comprehensive performance of the generated images in terms of style consistency, detail retention and visual beauty.

4.4 Contrast experiment

Figure 4: Stylized font results generated by FFM, Dreamina, FiT, HiDiffusion, MiracleVis

There are a variety of small programs for stylizing fonts on the market. We compared their results with those generated by FMM, selected several groups of stylistic fonts as samples (as shown in Figure4), and evaluated their visual quality in detail, including font clarity, style consistency, detail performance, and overall beauty. The fonts produced by FMM maintain a high level of clarity and sharpness in a variety of styles. Dreamina, by contrast, produces fonts that are prone to blurring or distortion at smaller sizes; The style of fonts generated by FiT is not obvious; The font generated by HiDiffusion has certain style fluctuation between different samples. The results of stylized fonts generated by MiracleVis are highly variable. Overall, the typefaces generated by our model are more visually striking and more creative and artistic in their stylistic expression.

4.5 Control weight adjustment experiment

During the generation process, we found that the control weight in the experimental group with TC had a great impact on the image generation result. In the model, the basic control model was mainly responsible for providing global style information, while TC refined the local features of the generated image by introducing spatial control information.

ISSN 2616-5767 Vol.7, Issue 6: 145-154, DOI: 10.25236/AJETS.2024.070621

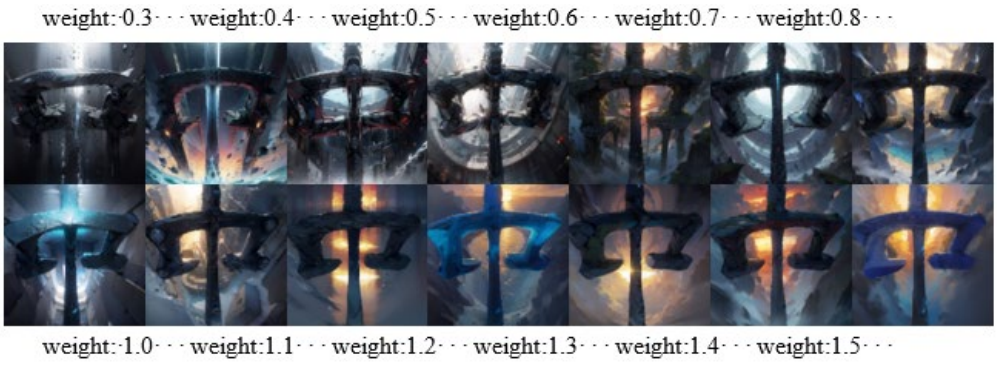

Figure 5: The result of FFM generated image under different TC weights

With the increase of control, the prompt information in TC has a greater impact on the result. When the control weight is small, the basic control module of the result generation is the dominant part, and the font outline of the generated image is fuzzy and the style is clear. When the control weight is large, TC is the dominant part, and the result of the generated image is clear, but the style is fuzzy. Therefore, it is necessary to adjust the control weight to a reasonable proportion to generate the optimal stylized font. We start the experiment with a small control weight, gradually increasing its value and observing the changes in the generated image (as shown in Figure5). Record the characteristics of the resulting image under each weight, especially the sharpness and degree of stylization of the font outline. By comparing the generated results under different weights, we can find a balance point that can maintain the clear outline of the font and maintain the stylistic characteristics.

5. Conclusion

FFM is an innovative deep learning architecture designed to generate highly stylized font images, cleverly blending the underlying control model, TOM, and TC. This integration enhances the flexibility and adaptability of the model, and improves the precision and artistic sense of font image generation. The introduction of TOM is a highlight in FFM. Through the built-in condition control mechanism, the model can flexibly respond to the given external conditions or inputs, dynamically adjust its output content, and ensure that the generated font image strictly follows the user-specified style, size, layout and even emotional color conditions. The integration of TC is another important embodiment of FFM's technological innovation. Based on the large pre-trained text-to-image diffusion model, TC further explored and learned the art of conditional control, reusing the source model's large pre-training layer, and building a new encoder for capturing and parsing specific content conditions, ensuring that the model can accurately understand and execute complex font style instructions while maintaining the original high-quality image generation capability. A large number of rigorous experimental data prove the superiority of FMM, compared with the use of conditional control model alone, only add TOM or TC enhanced conditional control model, FFM in all aspects of font image generation have shown a better performance, the generated picture style is bright, rich in details, and can accurately reflect the diversified needs of users.In the future, FFM will further optimize real-time generation capabilities, expand font style and language support, and enhance adaptive learning capabilities to adapt to changing design trends and promote continued innovation and application of stylized font generation technology.

References

[1] Ghadekar P, Gundawar A, Kamnapure S, et al. Improving Image Quality of Noisy Images Through Denoising and Style GAN Technique[C]//2023 7th International Conference On Computing, Communication, Control And Automation (ICCUBEA). IEEE, 2023: 1-6.

[2] Rajodiya P, Samruddha S, Alex S A, et al. Enhancing Image Fidelity through Denoising and Style GAN Techniques with Serial and Parallel Computation[C]//2024 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE). IEEE, 2024: 1-6.

[3] Chen C, Wu L, Xue Y. Style changing of the Real human face to cartoon face by Cycle-GAN[C]//2022 International Conference on Machine Learning and Intelligent Systems Engineering (MLISE). IEEE, 2022: 410-414.

[4] Yuhana U L, Edo G, Syarif H. Enhancement of Blurred Indonesian License Plate Number Identification Using Multi-Scale Information CNN[C]//2023 3rd International Conference on Smart

ISSN 2616-5767 Vol.7, Issue 6: 145-154, DOI: 10.25236/AJETS.2024.070621

Generation Computing, Communication and Networking (SMART GENCON). IEEE, 2023: 1-6.
[5] Ku W F, Siu W C, Cheng X, et al. Intelligent painter: Picture composition with resampling diffusion model[C]//2023 IEEE International Conference on Image Processing (ICIP). IEEE, 2023: 2255-2259.
[6] Trinh L T, Hamagami T. Latent Denoising Diffusion GAN: Faster sampling, Higher image quality[J]. IEEE Access, 2024.
[7] Radford A, Kim J W, Hallacy C, et al. Learning transferable visual models from natural language supervision[C]//International conference on machine learning. PMLR, 2021: 8748-8763.
[8] Gafni O, Polyak A, Ashual O, et al. Make-a-scene: Scene-based text-to-image generation with human priors[C]//European Conference on Computer Vision. Cham: Springer Nature Switzerland, 2022: 89-106.
[9] Gal R, Alaluf Y, Atzmon Y, et al. An image is worth one word: Personalizing text-to-image generation using textual inversion[J]. arXiv preprint arXiv:2208.01618, 2022.
[10] Tan M, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks[C]// International conference on machine learning. PMLR, 2019: 6105-6114.
[11] Vieillard N, Kozuno T, Scherrer B, et al. Leverage the average: an analysis of kl regularization in reinforcement learning[J]. Advances in Neural Information Processing Systems, 2020, 33: 12163-12174.
[12] Xiong W, Dong H, Ye C, et al. Iterative preference learning from human feedback: Bridging theory and practice for rlhf under kl-constraint[C]//Forty-first International Conference on Machine Learning. 2024.
[13] Ying Z, Mandal M, Ghadiyaram D, et al. Patch-vq:'patching up'the video quality problem[C]// Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021: 14019-14029.
[14] Yan W, Zhang Y, Abbeel P, et al. Videogpt: Video generation using vq-vae and transformers[J]. arXiv preprint arXiv:2104.10157, 2021.
[15] Wei X, Zhang T, Li Y, et al. Multi-modality cross attention network for image and sentence matching[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 10941-10950.
[16] Lu Z, Elhamifar E. Fact: Frame-action cross-attention temporal modeling for efficient action segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024: 18175-18185.
[17] Pavlova M A , Timofeev V , Bocharov D ,et al.Low-parameter method for delineation of agricultural fields in satellite images based on multi-temporal MSAVI2 data[J].Computer Optics, 2023.DOI: 10.18287/-6179-co-1235.
[18] Pang B, Nijkamp E, Wu Y N. Deep learning with tensorflow: A review[J]. Journal of Educational and Behavioral Statistics, 2020, 45(2): 227-248.