Journal of Artificial Intelligence Practice (2024)
Clausius Scientific Press, Canada

DOI: 10.23977/jaip.2024.070404
ISSN 2371-8412 Vol. 7 Num. 4

HCM: Icon art design based on diffusion model

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

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

Keywords: Diffusion model, Icon Design, HCM model, Image Generation

Abstract: With the advancement of AI-generated image technology, the field of icon design has increasingly incorporated computational methods as design references. Compared to direct design by human designers, AI technology can significantly reduce time and labor costs. In response to this, this research propose the High-Quality Customized Model (HCM) for controllable stylized icon design. This research introduce the Icon-ControlNet module to achieve precise control over icon generation, ensuring high levels of customization. Additionally, article employ the IconIC module to reduce computational resource consumption and enhance generation efficiency. This article have also constructed the IconData dataset, comprising 25,000 finely annotated medium-sized images. Through extensive ablation experiments, the results were evaluated by FID and IS, effectively demonstrating the advantages of HCM in terms of icon clarity, style transfer, and diversity. This model provides a novel solution for the automation and personalization of icon design.

1. Introduction

In the digital era, icons, as a crucial component of visual language, face challenges in traditional design due to the time-consuming and labor-intensive nature of manual drawing. This often results in difficulties in ensuring consistency and efficiency, which in turn limits the pace of icon development. The Diffusion model leverages deep learning technology to automatically capture icon features, promoting efficiency and diversity in icon design, thereby advancing innovation in digital design.

Various technologies and methods have already been proposed and applied in the field of image generation. The GAN[1] model employs an adversarial training process between a generator and a discriminator, optimizing the generator's output for fast inference but often encountering issues such as gradient vanishing and mode collapse, which degrade the quality and diversity of generated images. The VAEs[2] model combines deep learning with Bayesian inference, maximizing the marginal log-likelihood of input data, but offers limited diversity compared to GANs. The DDPM model uses Markov chains and reparameterization techniques for modeling and optimization, producing high-quality and diverse images, though its slow sampling speed results in high computational costs and time consumption.

In this study, we propose the High-Quality Customized Model (HCM), which integrates the Icon-ControlNet module to create a model capable of generating diverse and stylized icons(As shown in Figure 1). This article introduce the IconIM (Icon Information Model) to enable the generation of

icons according to specific category styles. To facilitate this, constructed a new multi-style icon dataset to support the development of this novel icon generation model. Meanwhile developed IconData, a medium-sized dataset containing 25,000 icons across eight categories. Our research incorporates additional control conditions and the qrcode-monster module, significantly enhancing the flexibility and controllability of image generation. Our model allows users to precisely control the content and style of generated icons by specifying semantic labels, target styles, or constraints. This effectively addresses potential issues in sample diversity encountered by traditional generation models, while maintaining stylistic consistency with reference images[3] .

Through extensive ablation studies, comparative experiments, and evaluation metrics tests, the HCM model demonstrates advantages in stylization, artistry, and aesthetic quality of generated icons. The model quickly adapts to and integrates different styles, and compared to existing text-guided models, our HCM model exhibits breakthrough developments in clarity and diversity of generated icons, achieving successful style transfer. In this study, our contributions are as follows:

- We propose the Icon-ControlNet module, which significantly enhances the model's grasp of design details in icon generation, improving control over generated images.
- We introduce the HCM model, utilizing the IconIM and IconIC modules to make icon generation more controllable and diverse, while reducing computational resources during training.
- We construct the ICOND dataset, a medium-sized, finely annotated dataset of 25,000 high-definition icons, categorized into eight styles, with detailed annotations for each category.

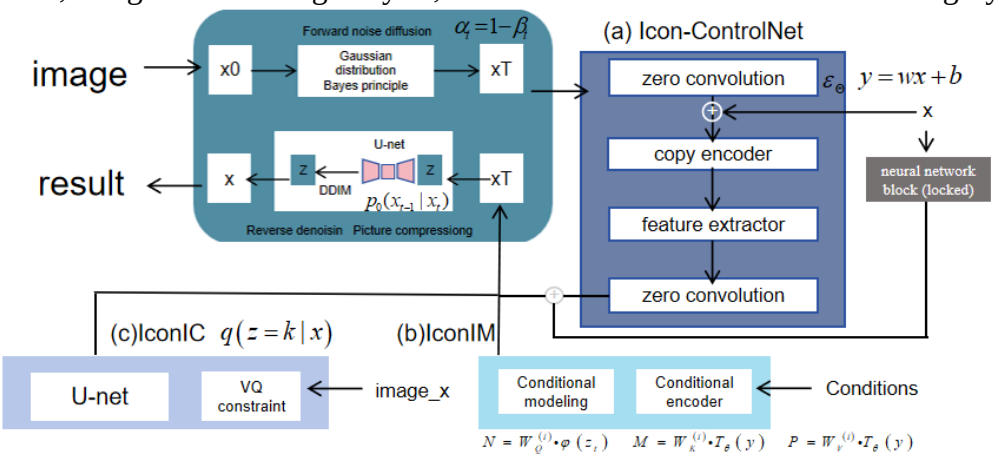

Figure 1: HCM generates an icon flowchart

2. Related work

2.1 Evolution of Icon Generation Technology

In recent years, large-scale text-to-image models[4] have achieved significant breakthroughs in the evolution of artificial intelligence, enabling the synthesis of high-quality and diverse images from given text prompts. The VAEs[5] model, which utilizes the optimization of the variational lower bound and reconstruction error to evaluate differences using KL divergence, enhances the scientific nature of icon generation compared to traditional methods. However, experiments have shown that images generated by this model often suffer from low quality and blurred details. Our model focuses on improving the quality of image generation, producing high-definition images with complete detail representation.

2.2 Evolution of Diffusion Models

Image generation technology has continuously evolved through iterations of diffusion models, progressing from the foundational DDPM framework to the diffusion GAN, which integrates GAN concepts, and finally stabilizing in the widely adopted Stable Diffusion model. The DDPM[6] model proposed by Berkeley, with its U-Net structure, performs exceptionally well in generating realistic and diverse images, though it faces challenges such as high computational costs, difficulty in real-time inference, and sensitivity to hyperparameters. Jascha Sohl-Dickstein's Diffusion model[7] , based on parameterized Markov chains and utilizing Langevin dynamics and Gaussian noise for image generation, struggles with over-complexity and imprecision in interpreting input conditions, which impacts image generation quality. CompVis's introduction of stability coefficients and an optimized network structure in the Stable Diffusion method has resulted in high stability, fast training speeds, and strong controllability, albeit at the expense of some sample diversity and slower generation speeds. Our model incorporates the IconIM and IconIC modules, allowing for conditional modeling based on user requirements, thereby achieving sample diversity while reducing memory footprint.

2.3 ControlNet Model

The emergence of Generative Adversarial Networks (GANs), Conditional GANs (CGANs), and Stable Diffusion models has paved the way for the development of ControlNet. The ControlNet[8] model achieves conditional control over pretrained image generation models by locking the parameters of large pretrained models, duplicating their encoding layers, and combining trainable copies with zero convolution layers, thereby generating images that better match user-specified conditions. The CLIP[9] model enables text-guided image generation through cross-modal learning, but it is highly dependent on large amounts of training data, and its black-box nature reduces interpretability and transparency. Models like MakeAScene[10] and SpaText[11] require high precision and complexity in segmentation masks, making the training and inference processes relatively complex. Lvmin Zhang's[12] research on reusing deep and shallow encoding layers, connected to neural structures with "zero convolution," demonstrates that ControlNet can effectively control stable Diffusion under single or multiple conditions. Our model integrates a skeletal control network module, enhancing flexibility and controllability, improving icon generation efficiency, enhancing model stability, simplifying the training and inference process, and reducing discrepancies with real images.

3. Model introduction

This article aim to improve the model's ability to understand abstract concepts and generate images in an icon style. We build IconData, a medium-sized multi-style icon dataset that explicitly extracts fine-grained style characteristics from the icon style types given for reference. Based on noise training module and predictive generation module, HCM introduces IconIC module and IconIM module, and integrates icon-ControlNet to realize style transfer, which makes the Icon generation process more controllable, and the generated samples more authentic and referential.

3.1 IconData

In the data collection phase, article implemented a diversified strategy that integrated efficient web crawling technology with manual collection methods. This approach enabled us to comprehensively gather icon resources from prominent media software platforms, specialized design websites, and

reputable open data sets, ensuring not only a substantial volume of data but also its diversity and timeliness. In total, approximately 28,000 icon materials were amassed.

In the data screening phase, article initially eliminate fuzzy, repetitive, or low-quality icons through manual inspection to ensure that the dataset encompasses a diverse array of icon types, styles, and application scenarios. This approach is aimed at enhancing the model's generalization capabilities while improving both the purity and representativeness of the dataset. Following this rigorous screening process, 25,000 high-quality icons were retained for future utilization.

In the data processing phase, all icons were processed using Photoshop, LR, and other reworking software to eliminate interference caused by irrelevant factors such as advertising and mosaic. The icons were normalized, adjusted to a uniform size (512*512 pixels), and converted to PNG format.

In order to establish a more efficient icon generation model, all data are divided into eight categories (as shown in Figure2) according to the recognized icon classification types, namely: simple style, abstract style, three-dimensional style, hand-drawn style, flat style, skeutrified style, linear style and pixel style. There are about 3,000 icon pictures in each category.

During the data annotation phase, article harness the WD1.4 tagger to streamline the annotation process. The WD1.4 tag extender, a cutting-edge tool leveraging the power of Stable Diffusion and DeepDanbooru, excels at utilizing neural network architectures to effortlessly discern the underlying elements and salient features within uploaded images. This sophisticated system subsequently assigns weighted tags based on their significance, automatically generating a hierarchy of relevant labels. By incorporating the WD1.4 labeler into our workflow, we've successfully forged ahead in creating a robust, multi-label icon dataset that boasts unparalleled quality. This not only significantly alleviates the onerous task of manual labeling but also establishes a solid foundation upon which future research endeavors and applications can thrive.

In the model training phase, article divided the data set strictly according to the scientific method, which was divided into training set, verification set and test set, the proportion was 70%, 20% and 10% respectively. In the training process, L1 loss function was used to measure the difference between the generated icon and the target icon, and Adam optimizer adaptatively adjusted the learning rate. We set the total number of steps to 500,000. In order to verify the model performance and avoid overfitting, multiple rounds of training were conducted, and the quality of the model generated icons was evaluated by verification sets after each round. The number of training rounds, learning rate, batch size and other parameters are constantly adjusted to optimize the training results to achieve the best effect.

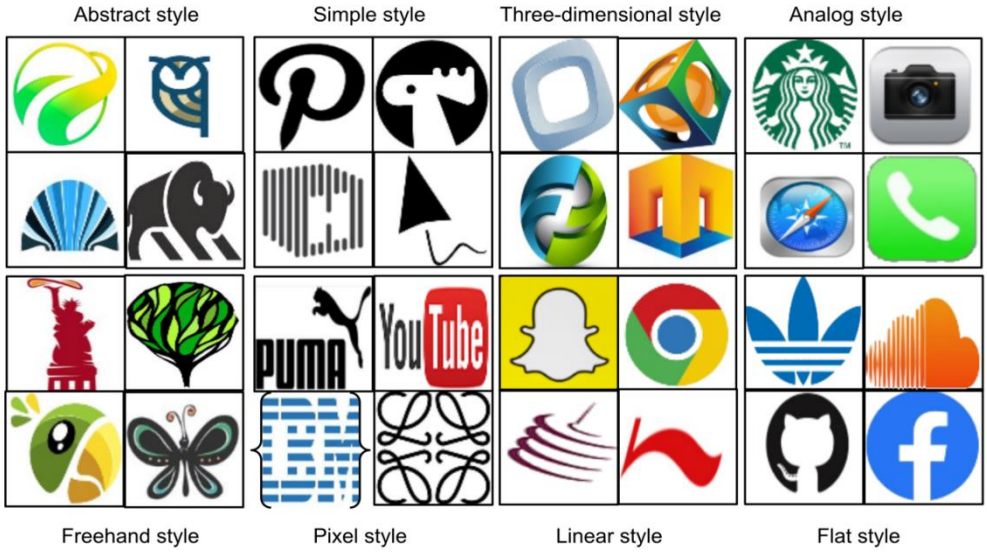

Figure 2: IconData shows part of the data set in different styles

3.2 HCM MODEL

3.2.1 Noise distribution training module

In the training process, to ensure that the image noise processing steps follow the consistency and there is a certain robustness between each classification, to provide a natural prior distribution, the trained weights make the model have stronger generalization ability, to avoid the noise or distortion of the data to affect the image generated by the model, we introduce Gaussian distribution to achieve the effect. A Gaussian distribution is a continuous probability distribution defined by the mean μ and variance. Its probability density function is:

$$p\left(x \mid \mu, \sigma^2\right) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{1}$$

The μ represents the center of the distribution, and the σ^2 represents the dispersion of the distribution. In the process of Diffusion, Gaussian distribution, as the main method of noise addition, can gradually transform the original image data into pure noise, which simulates the natural degradation of data.

$$\begin{aligned} x_t &= \sqrt{1-\beta_t}\, x_{t-1} + \beta_t \varepsilon \\ &= \sqrt{\alpha_t}\, x_{t-1} + \sqrt{1-\alpha_t}\, \varepsilon \\ &= \sqrt{\alpha_t \alpha_{t-1}}\, x_{t-2} + \sqrt{1-\alpha_t \alpha_{t-1}}\, \varepsilon \quad .... \end{aligned} \tag{2}$$

Gaussian noise is gradually added to the initial data x0, generating a series of noisy data x1,x2,... , xT. At each step, the mean and variance of the noise are determined by the predefined hyperparameters (βt) and the current data point. The introduction of Gaussian distribution in the model not only allows the model to capture the real distribution characteristics of the data more accurately, but also greatly simplifies the complexity and computation of the model. For people, the generated icon model will be closer to reality, improve the accuracy and readability of information; The application of Gaussian distribution simplifies the process of designers, improves the efficiency of research, and can be more focused on the exploration of creativity and core problems.

3.2.2 Prediction generation module

In HCM, article set up a predictive generation module to assist continuous iterative de-noising with Bayes theorem, effectively improve the robustness of HCM to noise, enhance the stability of generation and the generalization ability of the model, and achieve high-quality image generation in the potential space.

In order to measure the difference between the distribution of the generated image and the distribution of the real image and guide model optimization, we use KL divergence[13] to measure the difference between the predicted distribution of the model and the real distribution, and optimize the model as part of the loss function to improve its accuracy and generalization ability.

$$D_{KL}\left(P \| Q\right) = \sum_i P(i) \log\left(\frac{P(i)}{Q(i)}\right) \tag{3}$$

HCM uses the prediction generation module to estimate the probability distribution of the data

xt−1 of the previous time step given the noisy data xt and time step t of the current time step. According to the current noisy image data and prior knowledge, the prediction generation module can infer the image data of the previous time step (that is, less noise), which makes the generated sample more close to the real data distribution, and improves the quality of icon generation and the overall sample diversity.

3.2.3 IconIC

HCM introduces the IconIC module to form an integrated and efficient compression technology, reducing the storage and transmission requirements of images, and achieving a perfect balance between image quality, storage requirements and transmission efficiency.

VQ constraints are used in IconIC to map continuous image feature vectors to discrete codebooks. VQ constraint selects representative vectors (i.e. code words in codebook) by clustering method, and maps original data vectors to these representative vectors to achieve data compression.

$$q\left(z=k\mid x\right)=\begin{cases}1 \;\; for \;\; k=\arg\min_j \parallel z_e\left(x\right)-e_j \parallel_2 \\ 0 \;\; otherwise\end{cases} \tag{4}$$

x refers to the original data vector that needs to be quantized, and q is the quantizer. Combined with VQ constraints, IconIC can intelligently identify redundant information and key features in images, thus achieving finer and more efficient compression. Using IconIC, compared with traditional JPEG algorithms, HCM not only has a higher compression rate, but also can retain more image details and clarity.

3.2.4 IconIM

IconIM module can generate images or texts that meet specific requirements according to input conditions, capture the characteristics of input factors, increase model flexibility and controllability, and complete cross-modal generation. HCM can realize personalized, customized and stylized icon generation. According to different input conditions, information is encoded by condition encoder. For example, text information conditions are encoded by transformer model such as BERT. After obtaining the conditional coded information, each layer needs to calculate attention.

The attention mechanism involves selectively focusing on certain relevant things while ignoring others to capture important information more efficiently.

$$\text{ﻩ} = \text{ﻩ}_{\text{ﻭ}}^{(\text{ﻩ})}\text{ﻩ}(\text{ﻩ}_{\text{ﻭ}})\text{ﻩ} = \text{ﻩ}_{\text{ﻭ}}^{(\text{ﻩ})}\text{ﻩ}_{\text{ﻭ}}(\text{ﻩ})\text{ﻩ} = \text{ﻩ}_{\text{ﻭ}}^{(\text{ﻩ})}\text{ﻩ}_{\text{ﻭ}}(\text{ﻩ}) \tag{5}$$

N comes from the feature encoding Zt, while M and P come from the conditional information y. HCM pays attention to the generation of customized icons in exclusive style. IconIM module can understand various input conditions, and then encode and output important information to improve the precision of sample generation.

3.3 Icon-ControlNet

3.3.1 Holding module

The original weight of the model is fixed, avoiding overfitting to maintain the generalization ability of the model, and it is easier to control the stability of the training process and the basic image generation quality. When introducing ControlNet into a pre-trained neural network module, the policy is to keep the parameters of the original module (whose parameter set is Θ) unchanged and locked,

that is, to freeze them to retain the maturity and generalization that it has been trained on large amounts of image data.

$$L=\mathrm{E}_{z_0,t,c_t,c_f,\varepsilon\sim N(0,1)}\left[\varepsilon-\varepsilon_\theta\left(z_t,t,c_t,c_f\right)\right] \tag{6}$$

At the same time, we create a trainable copy of the module, and HCM uses the trainable copy to flexibly deal with a variety of input conditions to build a deep and robust backbone network architecture, improve robustness, and do not overfit or catastrophic forgetting problems. Locking in raw parameters ensures that HCM retains production-grade performance and stability from processing billions of image data, converging faster during training, increasing training efficiency, further improving the efficiency of icon design, and reducing repetitive labor costs for designers.

3.3.2 Initialization module

The control network uses a zero initialization layer to connect network blocks. Due to the initialization of Zero Convolution, the entire Icon-ControlNet model remains the original model, and Icon-ControlNet can smoothly transition to the new control conditions at the initial stage of training, reducing the initial shock of training. Rapid adaptation to new control conditions in a limited time ensures the robustness of training.

The output of the zero initialization layer is y, the input feature map is x, the weight is w, and the bias is b.

$$\begin{cases}\text{weight：}\ y=wx+b\\ \text{Gradient derivation：}\ \partial y/\partial w=x,\partial y/\partial x=w,\partial y/\partial b=1 \qquad w=0,x\neq 0\end{cases} \tag{7}$$

Through continuous iteration, the zero convolution module gradually becomes a common convolution layer with non-zero weights, and the parameter weights are continuously optimized.

3.3.3 Canny algorithm

Canny edge detection algorithm can accurately identify the edge information in the image, reduce the blur and noise in the generated icon, and improve the clarity and quality of the icon. The Canny algorithm[14] , as a component of ControlNet for edge detection, first smooths the image using a Gaussian filter to reduce the impact of noise on edge detection.

$$G(x,y)=\frac{1}{2\pi\sigma^2}\mathrm{e}^{-\frac{x^2+y^2}{2\sigma^2}} \tag{8}$$

(x,y) is the coordinates of the pixels in the image, and σ is the standard deviation of the Gaussian kernel, which determines the smoothness of the filter. Then Sobel or other gradient operators are used to calculate the gradient intensity and direction of the image.

$$\begin{aligned}G_x&=\frac{\partial f}{\partial x}=f\left(x+1,y\right)-f\left(x,y\right)\\ G_y&=\frac{\partial f}{\partial y}=f\left(x,y+1\right)-f\left(x,y\right)\end{aligned} \tag{9}$$

The Canny module[15] in HCM can identify and extract the edge information in the original image very accurately and carefully to reduce noise interference, which makes the generated icon can highly restore the detailed features of the original image. By adjusting the low threshold and high threshold

parameters of the Canny module, you can flexibly control the fine level of icon edge detection.

Canny algorithm enables HCM to have high accuracy, noise suppression, smooth edge connection and efficient calculation, improve icon clarity, purity, beauty and generation efficiency, greatly enhance the flexibility and controllability of icon design, so that we can capture and integrate design intent with unprecedented accuracy. Revolutionary convenience and efficiency for both users and icon designers.

4. Experiment

This article describe in detail the HCM detail adjustment methods and build effects for generating specific style icons. In order to achieve the best result of HCM generation under comprehensive conditions, this paper discusses the icon generation effect of the basic model under different parameters. In addition, compare the generation results with other generation models under the same conditions. In each stage of the experiment, a number of evaluation indicators such as IS and FID were used to evaluate.

4.1 Experimental setup

Parameter Settings. Under study configured several key parameters to optimize the performance of the HCM model, fine-tuning for the unique characteristics of each type of icon. The parameters include the number of iterations, learning rate, noise scheduling strategy, and the dimension of conditional text embedding to balance the detail richness and computational efficiency of the generated icons.

Evaluation index. The IS measures the diversity and clarity of the generated , the FID assesses the similarity of the generated icons to the real icon set, and the composite indicators reflect the usefulness and beauty of the icons, ensuring a comprehensive and in-depth understanding of the model's performance.

Sampling method. Considering factors such as style type and experimental test, chose DPM++ 2M as the sampling method.

4.2 Ablation study

In order to deeply understand the importance of each key component in our proposed icon generation model and its impact on the final performance, we adopted a group strategy to display the generation results of different styles of icons at different stages, and gradually integrated multiple cutting-edge technologies. These experimental modules observe and analyze the resulting performance changes by gradually removing or replacing specific parts of the model, significantly optimizing image detail, realism, and variety.

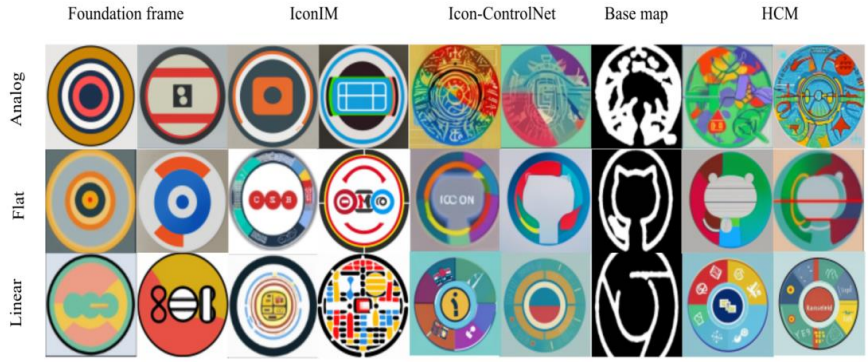

Figure 3: IconIM, Icon-ControlNet module, HCM generation sample

First, define a base model configuration (as shown in Figure3) that contains all the basic components that we believe are critical to generating high-quality icons, such as generator architecture, discriminator architecture, loss function, and training strategy, and set circles as the base condition for icon generation. This basic model will serve as the starting point for all subsequent experiments.

In the second experiment, IconIM was added. As can be seen from the figure (as shown in Figure3), IconIM makes the image quality of the generated result clearer and the content more specific, and dynamically adjusts the generation process according to preset conditions better. The model can generate ICONS that meet specific needs, have diversity and style consistency.

The third set of experimental Icon-ControlNet module is the innovation point model of this paper. Before running, we select qrcode_monster, select different preprocessors for different styles, and adjust the parameters of the model. Adding Icon-ControlNet module to model generation results (as shown in Figure3), Icon-ControlNet has a strong influence on Icon generation, improving the model's ability to control input, and increasing the diversity and detail richness of generated icons. The display effect of the icon is more colorful, artistic and stylized.

The fourth set of experiments is our HCM model, and the basic framework is fused with IconIM and Icon-ControlNet to build a highly integrated and efficient Icon generation model. We can clearly see the influence of the two modules on the generated results from the figure, which shows the flexibility and control ability of the model.

The results of four groups of experiments show that HCM has controllability and sample diversity in generating stylized icons. The dual control mechanism of HCM ensures the perfect combination of diversity and precision of the generated icons, which greatly improves the practical value and visual appeal of the generated results. HCM shows that icon creation will develop in a more intelligent and personalized direction.

4.3 Generated result evaluation

In the study of image generation model, it is very important to evaluate the quality of the generated image. FID is used to calculate the Fréchet distance between the generated image and the real image in the feature space of the Initiation-V3 neural network. To calculate the FID value, we first input the real icon image and the generated icon image respectively into the pre-trained Inception-v3 network. The FID value is calculated according to the formula of FID.

$$FID(p,q) = \| \mu_p - \mu_q \|_2^2 + Tr\left(C_p + C_q - 2\sqrt{C_p C_q}\right) \tag{10}$$

PSNR is an index to evaluate the quality of image compression or image reconstruction. High PSNR values generally indicate higher image quality and less noise.

$$PSNR = 10 \times \log^{\left(\frac{\max^2}{MSE}\right)} \tag{11}$$

Table 1: Different styles of generating sample evaluation index values

style	Basicmodel+IconIM		Basicmodel+Icon-ControlNet		HCM	
	FID	PSNR	FID	PSNR	FID	PSNR
Analog	2.12	14.150	2.03	14.958	1.95	15.452
Linear	2.96	12.952	2.64	13.231	2.83	14.386
Flat	2.54	11.650	2.87	11.470	2.07	12.689

Looking at the values in the table (as shown in Table1), the lower FID value proves that the icon

generated by our model is closer to the real icon. The PSNR index shows that the image sample generated by HCM has higher quality and lower distortion.

Through the study of this experiment, we found that the icon generation method of HCM model performs well in FID and PSNR indicators, and can generate high-quality images close to the real icon distribution, and realize the stylized migration. In the future, we will further explore how to optimize the model structure and parameter Settings to further improve the quality of the generated images, and combine more evaluation indicators to comprehensively evaluate the performance of the generated images.

4.4 Detail inquiry experiment

The influence of changing key hyperparameter exploration on model results. The number of training rounds refers to the number of iterations required to train the model. In each training round, the model traverses the entire data set once, learning multiple iterations of the data to gradually understand the characteristics of the data and improve performance.Set the number of rounds and save the rounds, obtain the loss function of the same classification data set under different training rounds and generate results, and test the optimal number of training rounds of the model.

Figure 4: The effect of HCM generated ICONS under different training rounds

Under different training rounds, different loss functions and generation effects can reflect the learning progress and performance changes of the model during training. A low value of the loss function means that the model is gradually optimized to better fit the training data. The experimental results show that the loss function of the second, fourth, sixth, eighth and tenth rounds is 0.0258, 0.0145, 0.0296, 0.0195 and 0.0203, respectively. The loss function of the fourth round is the lowest and the loss function of the sixth round is the highest. By comparing ten generated images (as shown in Figure4), the calculation IS shows that the model icon sample with four training rounds has high confidence and high entropy, indicating that the images generated by the model in the fourth round have high quality, uniform distribution and diversity. Therefore, the test shows that the training rounds suitable for the skeuomorphic style are 4 rounds. Comparative experiments provide key information during model training to understand when the model reaches its optimal state and how to adjust training strategies to further optimize model performance. In actual operation, the parameter values are adjusted in real time according to the generated effect to ensure that the model has excellent performance in practical application.

4.5 Contrast experiment

In order to comprehensively evaluate the performance of HCM on icon generation tasks, this article designed a comparison experiment and selected several representative models. In the experiment, set the same style and prompt words as the premise, and select FID and user satisfaction as the scoring criteria.

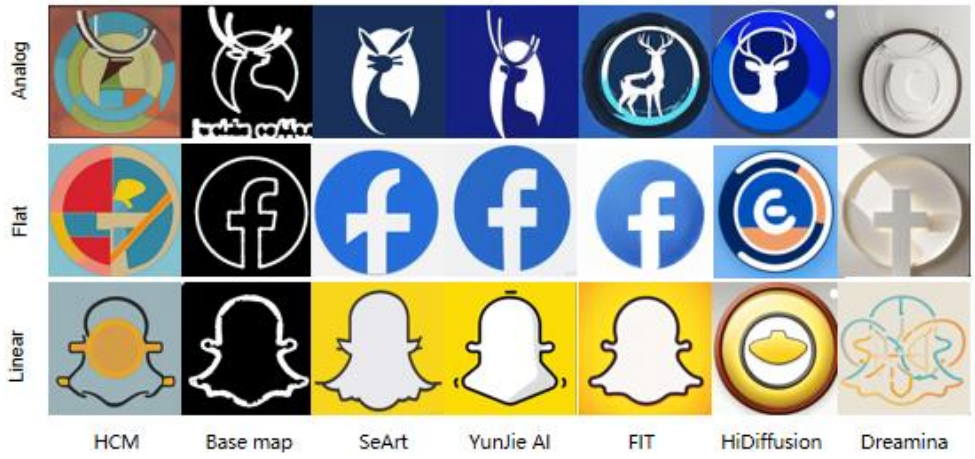

Figure 5: Comparison between HCM and similar image generation models

As shown in Figure5, under the same generation conditions, most samples generated by other models lack diversity, and only a small amount of modifications to the original image are slightly insufficient in accurately controlling the details of the generated icons. In contrast, HCM significantly improves the ability of the model to generate icons based on text descriptions and sketches by introducing additional condition control modules. The generated icons are closer to the input conditions in terms of shape, color, and detail, and more color and shape changes are made on the original image. Overall comparison, HCM performs best in the fine control of icon generation tasks, and provides an efficient and controllable generation method for icon design.

5. Conclusions

In the article construct a medium-sized dataset for collecting stylized icons and propose a novel stylized icon generation model that can help designers create new icons in a controllable style. The advantages of HCM model in the icon generation task were verified through ablation experiment and comparison experiment. The generated icon has the characteristics of stylization, diversification and artistic style, and the generation process is efficient and rapid reasoning is realized. In future work, research team will further explore how to optimize HCM model parameters and training strategies to improve the quality and efficiency of generating icons.

References

[1] Zhou T , Li Q , Lu H ,et al.GAN review: Models and medical image fusion applications[J].Information Fusion, 2023.
[2] Child R. Very Deep VAEs Generalize Autoregressive Models and Can Outperform Them on Images[J]. 2020. DOI: 10.48550/arXiv.2011.10650.
[3] Chen J, Pan Y, Yao T, et al. Controlstyle: Text-driven stylized image generation using diffusion priors[C]// Proceedings of the 31st ACM International Conference on Multimedia. 2023: 7540-7548.
[4] Ruiz N, Li Y, Jampani V, et al. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation[C]// Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023: 22500-22510.
[5] Kingma D P, Welling M .Auto-Encoding Variational Bayes[J].arXiv.org, 2014.DOI:10.48550/arXiv.1312.6114.

[6] Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models[J]. Advances in neural information processing systems, 2020, 33: 6840-6851.
[7] Sohl-Dickstein J, Weiss E, Maheswaranathan N, et al. Deep unsupervised learning using nonequilibrium thermodynamics[C]//International conference on machine learning. PMLR, 2015: 2256-2265.
[8] Cong Zhao, Pinghua Chen. Icon Shape Generation Based on Generative Adversarial Network[J]. Computer Science and Application , 2020, 10(3): 456-463. https://doi.org/10.12677/CSA.2020.103047
[9] Zhang L, Rao A, Agrawala M. Adding conditional control to text-to-image diffusion models[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 3836-3847.
[10] 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.
[11] 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.
[12] Avrahami O, Hayes T, Gafni O, et al. Spatext: Spatio-textual representation for controllable image generation[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 18370-18380.
[13] Lin J, Jiang Z, Guo J, et al. IconDM: Text-Guided Icon Set Expansion Using Diffusion Models[C]//ACM Multimedia 2024.
[14] Vaswani A. Attention is all you need[J]. arXiv preprint arXiv:1706.03762, 2017.
[15] Canny J. A computational approach to edge detection[J]. IEEE Transactions on pattern analysis and machine intelligence, 1986 (6): 679-698.