Vol.2023-CG-191 No.9
2023/9/16

Sketch-Guided Two-Stage Text-to-Image Generation with Spatial Control

Tianyu Zhang[1,a)] Haoran Xie[1,b)]

Abstract: Recent text-to-image diffusion models can produce high-quality images based only on textual prompts. However, it is difficult to correctly interpret instructions specifying the layout of a compositional space using only text. We propose a sketch-based method to control the spatial relationship of corresponding objects in image generation and solve the issue of object loss in diffusion models. Our proposed method uses a pre-trained text-to-image diffusion model as the image generator and employs sketches as spatial guidance. Specifically, we divide the proposed model into two stages. In the feature extraction stage, sketches are segmented into individual objects using the image segmentation approach, and the obtained bounding boxes and labels are then used as spatial-guided inputs to the attention layers of the diffusion models. In the image generation stage, the proposed model utilizes a pre-trained text-to-image diffusion model as the generator to generate corresponding images. We evaluate the proposed method quantitatively and qualitatively with several experiments, validating the spatial control of the proposed method. In addition, we further demonstrate its versatility by changing the position relationships and relative scales in sketches.

Keywords: Image generation, Sketch-guided, Two-stage model, Diffusion model

1. Introduction

Image generation is currently in a stage of rapid development with new methods constantly emerging. The development of deep learning-based approaches, particularly Variational Autoencoders (VAE), autoregressive models, and Generative Adversarial Networks (GAN), has advanced and improved image generation approaches. In addition, conditional generative models allow additional conditions to be specified during image generation to increase the control and flexibility of the generation process, such as providing sketches or text descriptions to control the features of generated images.

The diffusion model is undoubtedly one of the most revolutionary technologies that have surfaced in the past few years. Such as Denoising Diffusion Probabilistic Models (DDPM)[5], Denoising Diffusion Implicit Models (DDIM)[12], and Stable Diffusion (SD)[10]. These models have disrupted the long-standing dominance of GANs in the demanding field of image synthesis and have demonstrated promise across various domains, such as computer vision, multi-modal modeling, and natural language processing. In addition, diffusion models can amplify the productivity of professional artists greatly and have attracted widespread interest from the general public in practical applications such as art design and creation.

Despite the successes, the powerful pre-trained diffusion models still lack a high level of control that can guide the spatial properties of complex images. Lengthy and intricate text descriptions are often required for complex images, involving complex semantic relationships and multiple objects. Generating models struggle to maintain consistency and coherence when faced with long textual descriptions, resulting in issues of blurry or inaccurate generated results and object loss. In fact, in Stable Diffusion[10], current state-of-the-art image generators struggle to effectively comprehend straightforward layout instructions specified in text form. This is mainly because diffusion models belong to the category of probabilistic generative models, where the core idea is to iteratively generate real images from noisy images. At each step, the model focuses on updating the image's pixel values without considering the pixels' positional information.

As mentioned above, the current diffusion models face the following issues: 1) the text prompts are difficult to describe the semantic information, especially in complex images; 2) text-to-

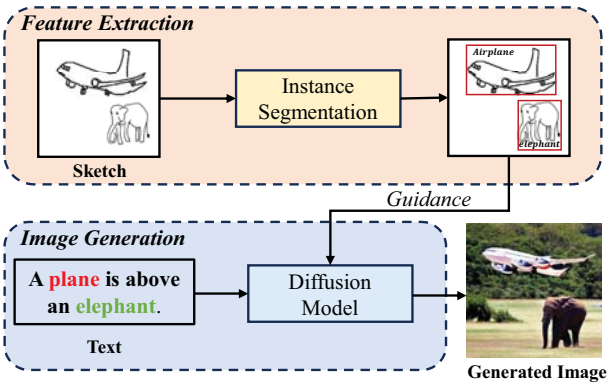

Fig. 1 Based on the diffusion model, the proposed method is guided by the sketch's segmentation. The proposed method does not necessitate any further training of the pre-trained text-to-image diffusion model.

[1] Japan Advanced Institute of Science and Technology, Nomi, Ishikawa 923–1211, Japan
[a)] s2110414@jaist.ac.jp
[b)] xie@jaist.ac.jp

image generation models lack spatial control of generated results; 3) diffusion models may lose the objects that depicted in text prompts. To solve these issues, we propose the sketch-based image generation method with two-stage latent diffusion model. As shown in Figure 1, we try to intervene in the image generation process by adding sketches as new control conditions and altering the attention layers in the diffusion process. In the first stage, we utilize instance segmentation to extract object locations and labels from sketches and encode them as spatial guidance of the generation process. In the second stage, the pre-trained LDM generates images according to the input text prompts, where the objects' positions and scales will follow the spatial guidance of the sketches. Our proposed method gets reliable layout control without the need for additional training, while still maintaining the quality of the generated images.

The main contributions of this work are listed as follows:

- We propose a sketch-based image generation model, which intervenes with the spatial properties in attention layers of diffusion models to control the spatial relationship in generated objects.
- The proposed model can effectively improve the object loss issue that occurs in the diffusion models.

2. Related Works

2.1 Conditional Image Generation

Compared with traditional unconditional generation methods, conditional image generation introduces additional input conditions, enabling the generator to generate images with specific properties based on conditional information. In previous studies, conditional image generation based on GANs is a common method. AniFaceDrawing[6] adopted a latent space exploration method of StyleGAN with shadow guidance to generate high-quality anime portraits. Unlike the mature conditional image generation of GANs, the conditional image generation of diffusion models is still under exploration. ControlNet[14] puts forward a neural network structure designed to control pre-trained diffusion models, facilitating the integration of supplementary input conditions. Another approach encodes conditional information into latent embeddings, which are then mapped to intermediate layers of U-Net via cross-attention layers. In this way, GLIGEN[8] implements bounding boxes, reference images, and keypoints as conditional information to control image generation based on the latent diffusion model.

The advantage of conditional image generation is that it provides greater control, enabling users to generate images with specific properties as desired. However, conditional image generation also faces some challenges, such as the accuracy and completeness of conditional information, the diversity, and scale of training data, etc.

2.2 Diffusion Model

Diffusion models first introduced by Sohl-Dickstein et al.[11] and later advanced by Song et al.[13] and Ho et al.[5]. In recent times, numerous text-image models of significant scale have surfaced, such as Stable Diffusion[10], demonstrating unprecedented semantic generation.

Diffusion models mainly consist of two processes: the forward diffusion process and the reverse denoising process (inference process). In the forward diffusion process, a random image is sampled from the data distribution, and Gaussian random noise is gradually added to the image through a fixed process until it becomes pure noise. In reverse denoising, starting from pure noise, the process gradually restores it to a real image. Specifically, the generation process starts with a random noisy image, and the image is updated based on the current image state and the known noise at every time step. The diffusion model gradually restores the image to its original state through multiple iterations and gradually reduces the noise intensity.

Diffusion models can well preserve the texture and details of the image, and the generated results have good visual effects. However, diffusion models require multiple diffusion steps and sampling iterations, resulting in a long training time.

3. Conditional Generation with Latent Diffusion Model

In this section, we first introduce the preliminaries of the latent diffusion model (LDM) in Section 3.1 and the attention mechanism in Section 3.2. Our framework and implementation details will be discussed in Section 4.

3.1 Latent Diffusion Model

The difference between LDM and the DDPM is that LDM does not directly operate on the images but operates in the latent space. LDM calls this method perceptual compression. LDM reduces the dimensionality of the data by projecting it into a low-dimensional, efficient latent space, in that high-frequency, imperceptible details are abstracted away. Perceptual compression is typically employed to reduce computational complexity, save storage space, and improve the efficiency of model training and inference.

LDM trained an AutoEncoder, including an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$. After the image x is compressed by the encoder $\mathcal{E}$ to latent representation z, the diffusion process is performed on the latent representation space. Given a latent sample z_0, the Gaussian noise is progressively Increased to the data sample during T steps in the forward process, producing the noisy samples z_t, where the timestep $t = \{1, \ldots, T\}$. As t increases, the distinguishable features of x_0 gradually diminish. Eventually when $T \to \infty$, x_T is equivalent to a Gaussian distribution with isotropic covariance. Finally, LDM infers the data sample z from the noise z_T and $\mathcal{D}$ restores the data z to the original pixel space and gets the result images $\widetilde{x}$.

Specifically, given an image $x \in \mathbb{R}^{H\times W\times 3}$ with height H, width W in RGB space , LDM first utilizes an encoder $\mathcal{E}$ to encode the image x into a latent representation space:

$$z = \mathcal{E}(x) \tag{1}$$

where $z \in \mathbb{R}^{h\times w\times c}$ with height h and width w, the constant c represents the number of channels. The encoder $\mathcal{E}$ downsamples the image by a factor $f = H/h = W/w$. Then $\mathcal{D}$ recover the image from the latent representation space:

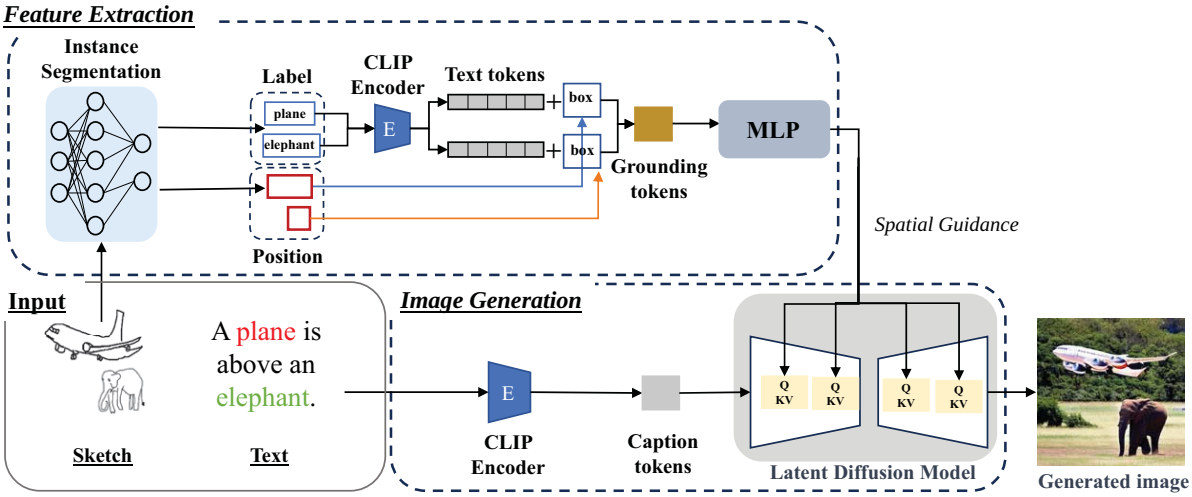

Fig. 2 The framework of our model. The model first extracted the sketch's features and introduced them into the attention layers with caption tokens to generate the images.

$$\tilde{x} = \mathcal{D}(z) = \mathcal{D}(\mathcal{E}(x)) \tag{2}$$

3.2 Attention Mechanism

LDM can be used to explore conditional image generation, which is mainly obtained by expanding the conditional denoising autoencoder $\epsilon_\theta(z_t, t, y)$. y is the conditional information that controls the process of image generation.

Specifically, LDM implements $\epsilon_\theta(z_t, t, y)$ by adding a cross-attention mechanism to the U-Net backbone network. To easily introduce various types of conditioning y (such as text, layout, sketch, etc.), LDM introduces a domain-specific encoder τ_θ, which is used to map y to an intermediate representation $\tau_\theta(y)$.

Finally, LDM integrates the conditional information into the middle layer of U-Net through cross-attention layers mapping. The implementation of the cross-attention layer is as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right) \cdot V \tag{3}$$

, with $Q = W_Q^{(i)} \cdot \varphi_i(z_t)$, $K = W_K^{(i)} \cdot \tau_\theta(y)$, $V = W_V^{(i)} \cdot \tau_\theta(y)$ of dimension d . where $\varphi_i(z_t)$ is an intermediate representation of U-Net, N is the latent's index dimension. W_Q, W_K, and W_V are learnable projection matrice in LDM.

In this case, the attention maps M can be calculated as follows,

$$\text{M} = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right) \tag{4}$$

The attention map M controls the spatial distribution of values V, which contains rich semantic information.

The spatial arrangement and shapes of objects in the generated image are contingent on the cross-attention maps[3]. Interestingly, The image's structure is established during the initial stages of the diffusion process. Most importantly, the degree to which attention is injected into the diffusion process affects the quality of the generated results. However, applying the injection throughout all diffusion steps does not necessarily achieve the optimal result.

4. Sketch-Guided Image Generation

We discuss the detailed composition of our proposed two-stage image generation model with spatial control in this section. We first give an overview of our proposed model in Section 4.1. We also introduce the two stages in our proposed model. In the feature extraction stage, we utilize instance segmentation to extract object locations and labels from sketches and encode them as spatial guidance of the generation process (introduced in Section 4.2). In the image generation stage, the pre-trained LDM generates images according to the input text prompts, where the objects' positions and scales will follow the spatial guidance of the sketches (introduced in Section 4.3).

4.1 Framework Overview

Our goal is to generate high-quality images with the spatial guidance of human-drawn sketches. In the proposed two-stage model, the feature extraction stage constrains a set of constraints (position, label, etc.) extracted from the sketch and introduces them into LDM to influence the position and shape generation. The image generation stage leverages the generative capabilities of the latent diffusion model to generate images following the spatial guidance from the feature extraction stage.

As shown in Figure 2, we use both a sketch and a text prompt as inputs. The text prompt is encoded into text embeddings by the encoder of CLIP, while sketch serves as a conditional input and undergoes instance segmentation. We employ the pre-trained DeepLab-V2 model to obtain corresponding labels and bounding boxes. The labels are encoded into temporary text tokens by the encoder of CLIP and combined with the upper left and lower right coordinates of the bounding boxes to form the final grounding tokens. Finally, the grounding tokens are inputted into the attention layers of the LDM to provide spatial guidance for image generation.

4.2 Feature Extraction Stage

In the feature extraction stage, we focus on extracting spatial information from the sketch for spatial control in the conditional generation. Inspired by SketchyScene[15], We employ the segmentation model based on DeepLab-v2 as the segmenter $\mathcal{S}$ to

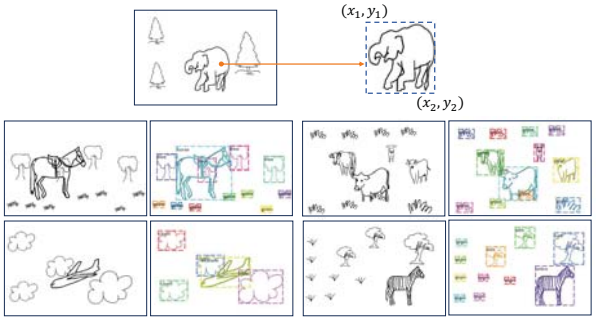

Fig. 3 The visualized results of feature extraction stage. We divide the sketches into labels and bounding boxes and capture the coordinates of the top-left and bottom-right corners of the bounding boxes.

complete the instance segmentation, which is customized for segmenting sketches. Therefore, for the input sketch x_s, it can be expressed as $\mathcal{S}(x_s)$.

As shown in Figure 3, after segmentation, the corresponding bounding boxes and labels of the objects can be obtained. The labels l represent the corresponding names, such as "cow", "tree", and "airplane". The bounding boxes b represent the coordinates$[x_1, y_1, x_2, y_2]$, where (x_1, y_1) represents the top-left coordinate and (x_2, y_2) represents the bottom-right coordinate. The segmentation can be expressed as

$$(l, b) = \mathcal{S}(x_s) \tag{5}$$

The labels will be encoded by the CLIP text encoder as the text tokens. The text tokens will be combined with coordinates as grounding tokens and inputted to LDM for conditional control. Thus, We define our model as a composition of the caption and grounding tokens:

$$I = (c, e) \tag{6}$$

$$e = \mathcal{S}(x_s) = (l, b) \tag{7}$$

where I is the generated image, c is the caption tokens and e is the grounding tokens.

4.3 Image Generation Stage

In the image generation stage, the pre-trained LDM generates images according to the input text prompts with spatial guidance from the feature extraction stage. Therefore, during the image generation stage, we use spatial guidance to influence the spatial generation of objects on the attention maps in the initial stages of the inference process. Subsequently, we employ LDM to generate images only based on text prompts.

4.3.1 Cross-Attention

As shown in Equation 3, LDM integrates the conditional information into the middle layer of U-Net through cross-attention layers mapping. In the original latent diffusion model, Q comes from visual tokens generated from latent seeds, and both K and V come from caption tokens in the text.

In our model, we kept K and V unchanged and still included the feature information in the caption. Inspired by the GLIGEN[8], We fuse the obtained grounding tokens and visual tokens as Q to query in the attention layers. Thus the Q can be expressed as

$$Q = v + \beta \times tanh(\gamma) \times e \tag{8}$$

where v is the visual tokens from the latent seeds, β is a gated parameter that will be introduced in Section 4.3.2 and γ is a learnable scalar.

4.3.2 Gated Parameter β

For a diffusion process with T time steps, we can set a fixed time step αT to divide the inference process, where α is a constant. When time steps $t \leq \alpha T$, this indicates that the diffusion process is early, at which point we condition the control via set $\beta = 1$. At this time, the Q of attention layers will be composed of grounding tokens e and original visual tokens v:

$$Q = v + tanh(\gamma) \times e \tag{9}$$

When $t \geq \alpha T$, the model set $\beta = 0$. In this situation, we use the original generation ability of LDM for image generation. Thus, the Q of attention layers will be the original visual tokens v:

$$Q = v \tag{10}$$

Note that the model in this situation has nothing to do with the additional input conditions, and the model maintains the original generation ability.

In summary, since the degree to which attention is injected into the diffusion process affects the quality of the generated results, we divided the image generation stage into two steps by β:

$$\begin{cases} \beta = 1, & t \leq \alpha T \quad \text{spatial guidance} \\ \beta = 0, & t > \alpha T \quad \text{Standard inference} \end{cases} \tag{11}$$

We explored the impact of different α on generation in the section 5.2.

5. Experiment and Results

We conduct qualitative and quantitative experiments to verify the image quality and sketch input consistency of our model's generated images. In Section 5.1 we introduce the implementation details of our experiment. We present the results of our qualitative evaluations (Section 5.2) and quantitative experiments (Section 5.3).

5.1 Implementation Details

Both stages of our model are implemented on the Ubuntu system, i7-13700KF CPU, and a single NVIDIA RTX4090 GPU. In the conducted experiments, we use the pre-trained LDM-V1.4 as the proposed image generator. All of our sketch images from the SketchyCOCO dataset[2], which include 14 categories of objects and 3 categories of background freehand sketches.

In quantitative comparison with LDM, we use the 300 randomly sampled sketches in the SketchyCOCO dataset to generate 300 images for evaluation. We provided the corresponding text prompts manually. In quantitative comparison with GANs, we utilize the 200 randomly sampled sketches with a single object in the SketchyCOCO dataset for evaluation.

5.2 Qualitative Evaluation

As shown in Figure 4, the state-of-the-art mainstream text-to-image method cannot further control the position information of the generated image. Our model uses the sketch as additional

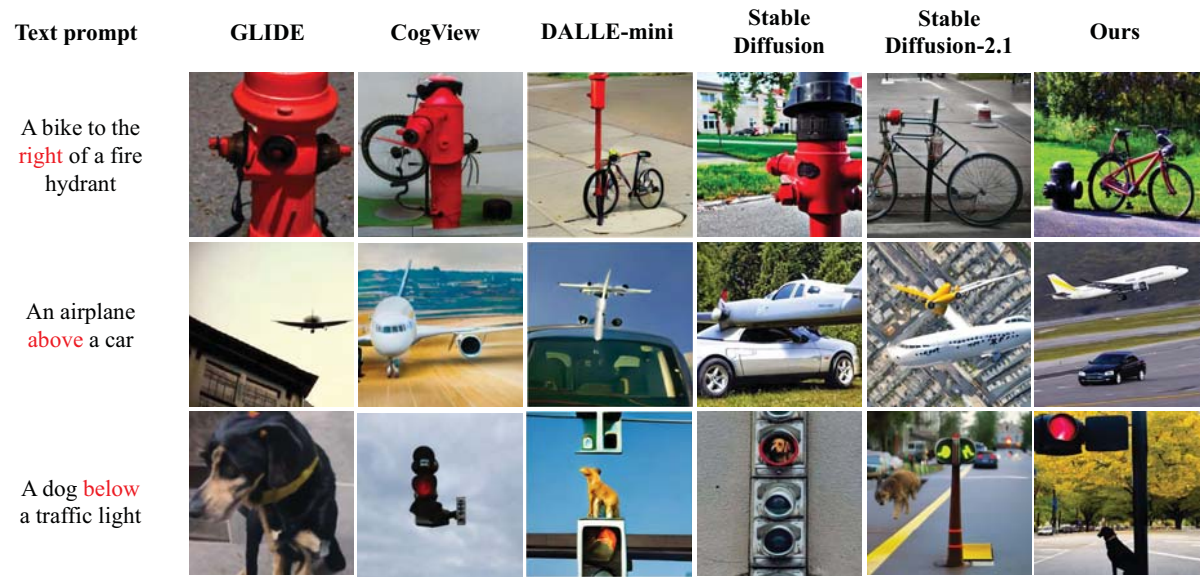

Fig. 4 The generated results of typical text-to-image generative models. Most text-to-image models can not comprehend the corresponding spatial relationships in the text prompt. However, we finish spatial guidance for the objects of generated images by the cross-attention maps.

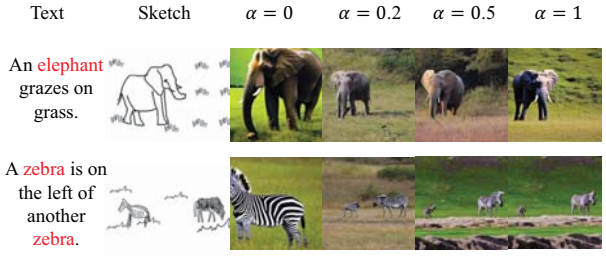

Fig. 5 The generated images with different α values. In the first and second rows, we consider the condition with a single object and two objects. In the third row, we verified situations that are unreasonable in reality.

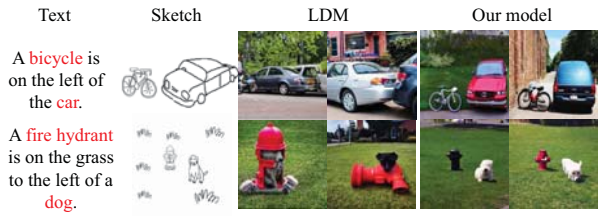

Fig. 6 Our model can effectively improve the object loss issue that occurs in the original LDM model.

supplementary information to control the position generation of the image. It is verified that our image has achieved a relatively good effect in terms of spatial control, and all desired objects can appear in the corresponding position.

We tried to explore the influences of different α value settings in our model. As shown in Figure 5, the position is only controlled in the early stage of the diffusion process, and the image can also be generated according to the bounding boxes and labels of the sketches. Objects in the generated images can already appear in the correct position.

We conducted further exploratory experiments to verify that our model can help to improve the object loss issue. As shown in Figure 6, When there are multiple objects in the semantics, the pre-trained text-to-image LDM model will have situations of semantic loss and disordered positions. After adding our sketch as an auxiliary, the generated image can contain the correct number of objects and have the corresponding position information.

As illustrated in the top of Figure 7, under the given text prompt, we deliberately alter the positions of objects in the sketches to contradict the positional information described in the text. The generated images consistently depict objects positioned according to our sketches, even if it contradicts the text. As shown in the bottom of Figure 7, we change the scale of the objects in the sketches with the text prompt constant, the corresponding objects in the generated image will change accordingly, even if the generated image does not conform to realistic logic at all.

5.3 Quantitative Comparsions

We compare our model with the state-of-the-art methods on the sketch-to-image task (pix2pix[7], SketchyGAN[1], and Sketchy-COCO[2]). We also conduct a comparison study between our model with the LDM to demonstrate the usefulness of our model for spatial control. Since prior text-to-image methods do not support taking sketches as input, it is not fair to compare with them on this metric. Thus, we only report metrics for the LDM as a reference. We employ Fréchet Inception Distance[4] (FID) as a metric to assess the quality of the generated images. We use the YOLO score[9] to evaluate grounding accuracy (the correspondence between the input bounding box and generated entity).

As shown in Table 1, our model (27.34 FID value) performs less favorably in terms of FID score compared to LDM (21.42 FID value). This is largely due to the influence of the sketches' spatial control on the model's generation capability. However, our model has succeeded in terms of YOLO scores (21.6, 42.0, 21.7 scores are better than 0.5, 2.4, and 0.4 scores of LDM), indicating that our model can generate corresponding objects at the desired positions.

We also conduct the comparison experiment between our pro-

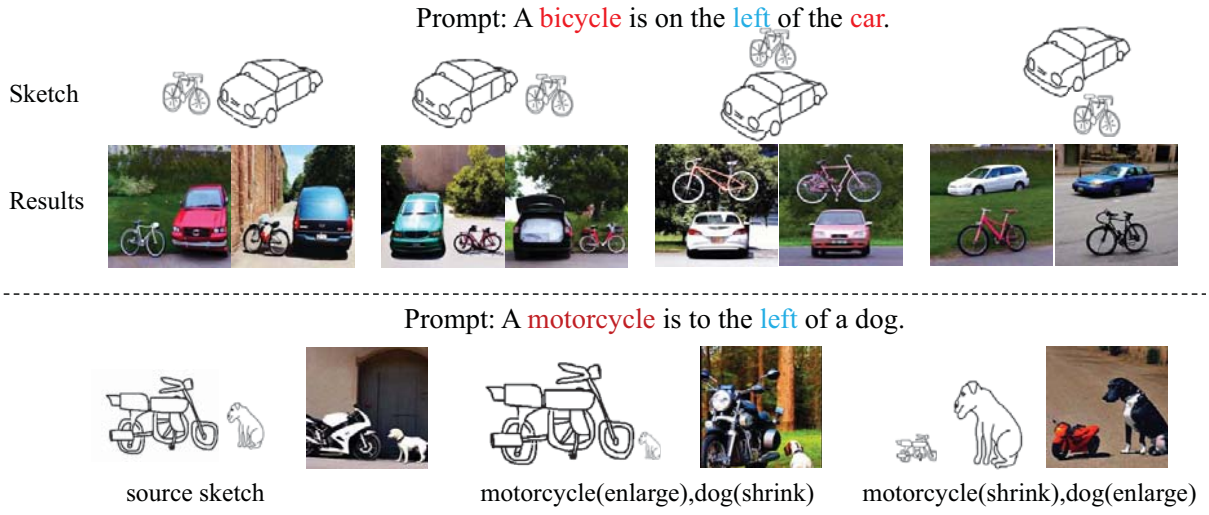

Fig. 7 We verified that the generated image will follow the spatial guidance of our sketch even if it contradicts the input text.

	FID(↓)	YOLO score(mAP / AP_{50} / AP_{75})(↑)
LDM	21.42	0.5 / 2.4 / 0.4
Our model	27.34	21.6 / 42.0 / 21.7

Table 1 Comparison between the pre-trained LDM[10] and our model.

Model	FID(↓)
pix2pix	143.1
SketchyGAN	141.5
SketchyCOCO	87.6
Our model	**21.04**

Table 2 We compare the proposed method with sketch-to-image generation methods in image quality.

posed and several image generation methods in FID value. Since the sketchyGAN and pix2pix are not models for complex images, we just utilize the single object sketches in this situation. As shown in Table 2, our proposed model gets the best result (21.04 FID value) in image quality than previous work, due to the strong generative ability of the diffusion model.

6. Conclusion

In this work, we proposed sketch-based spatial control by the pre-trained latent diffusion model without fine-tuning or training. Our proposed model has two stages, the feature extraction stage and the image generation stage. In the feature extraction stage, the sketches are segmented by the pre-trained segmentation model to obtain the labels and bounding boxes for spatial guidance. In the image generation stage, the model use pre-trained LDM to generate images. Our method can obtain the generated image, whose objects' spatial information (position and scale) are consistent with the sketches', and effectively solve the object loss issue of the original LDM.

Our model still has many limitations. First, our current model is limited to using the sketch to control spatial information and does not fully use all the advantages of sketches. Moreover, the proposed model utilizes the freehand sketch as additional conditional information, the semantics and intentions of sketches may not be unambiguous and are limited by the user's drawing level.

References

[1] Chen, W. and Hays, J.: Sketchygan: Towards diverse and realistic sketch to image synthesis, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9416–9425 (2018).
[2] Gao, C., Liu, Q., Xu, Q., Wang, L., Liu, J. and Zou, C.: Sketchycoco: Image generation from freehand scene sketches, *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5174–5183 (2020).
[3] Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y. and Cohen-Or, D.: Prompt-to-prompt image editing with cross attention control, *arXiv preprint arXiv:2208.01626* (2022).
[4] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. and Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium, *Advances in neural information processing systems*, Vol. 30 (2017).
[5] Ho, J., Jain, A. and Abbeel, P.: Denoising diffusion probabilistic models, *Advances in Neural Information Processing Systems*, Vol. 33, pp. 6840–6851 (2020).
[6] Huang, Z., Xie, H., Fukusato, T. and Miyata, K.: AniFaceDrawing: Anime Portrait Exploration during Your Sketching, *ACM SIGGRAPH 2023 Conference Proceedings*, SIGGRAPH '23, New York, NY, USA, Association for Computing Machinery, (online), DOI: 10.1145/3588432.3591548 (2023).
[7] Isola, P., Zhu, J.-Y., Zhou, T. and Efros, A. A.: Image-to-image translation with conditional adversarial networks, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134 (2017).
[8] Li, Y., Liu, H., Wu, Q., Mu, F., Yang, J., Gao, J., Li, C. and Lee, Y. J.: GLIGEN: Open-Set Grounded Text-to-Image Generation, *arXiv preprint arXiv:2301.07093* (2023).
[9] Li, Z., Wu, J., Koh, I., Tang, Y. and Sun, L.: Image synthesis from layout with locality-aware mask adaption, *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13819–13828 (2021).
[10] Rombach, R., Blattmann, A., Lorenz, D., Esser, P. and Ommer, B.: High-resolution image synthesis with latent diffusion models, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695 (2022).
[11] Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N. and Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics, *International Conference on Machine Learning*, PMLR, pp. 2256–2265 (2015).
[12] Song, J., Meng, C. and Ermon, S.: Denoising diffusion implicit models, *arXiv preprint arXiv:2010.02502* (2020).
[13] Song, Y. and Ermon, S.: Generative modeling by estimating gradients of the data distribution, *Advances in neural information processing systems*, Vol. 32 (2019).
[14] Zhang, L. and Agrawala, M.: Adding conditional control to text-to-image diffusion models, *arXiv preprint arXiv:2302.05543* (2023).
[15] Zou, C., Yu, Q., Du, R., Mo, H., Song, Y.-Z., Xiang, T., Gao, C., Chen, B. and Zhang, H.: Sketchyscene: Richly-annotated scene sketches, *Proceedings of the european conference on computer vision (ECCV)*, pp. 421–436 (2018).