MMIG: Multi-Modal Image Generator

Wenjin Liu
Hainan University
Haikou, Hainan Province, China
23110839000005@hainanu.edu.cn

Lijuan Zhou
Hainan University
Haikou, Hainan Province, China
zhoulijuan@hainanu.edu.cn

Ning Luo
Hainan University
Haikou, Hainan Province, China
luoning@hainanu.edu.cn

Bing Wei
Hainan University
Haikou, Hainan Province, China
weibing@buaa.edu.cn

Min Xu
Capital Normal University
Beijing, China
xumin@cnu.edu.cn

Shudong Zhang*
Hainan University
Haikou, Hainan Province, China
zsd@Hainanu.edu.cn

Abstract

Recently, text-guided image editing technology based on Stable Diffusion Model (SDM)[1] has developed rapidly. However, these methods have not yet achieved precise and detailed image editing by deeply integrating Large Language Model (LLM)[3] and SDM. Therefore, this paper proposes a new model, Multi-Modal Image Generator (MMIG), which combines LLM and SDM in-depth and can accurately edit the foreground and background of images using a long description with multiple concepts. MMIG consists of a multimodal LLM, a Multimodality mutual support module (MMSM), a Diffusion editing model (DEM), and BLIP2[6]. The multimodal LLM is used to achieve cross-modal alignment of input text prompts and images to generate accurate descriptions of target images. DEM can generate new images using the original images and text prompts generated by LLM. The MMSM module is used to enhance information fusion between LLM and DEM. BLIP2 is used to generate a textual description of the newly generated image. Further, preliminary experiments have been conducted on MMIG. Experimental results show that MMIG has excellent performance in image editing.

1. Introduction

In recent years, many image editing methods based on the diffusion model have emerged [1]. However, it is challenging to preserve the features of original images and text descriptions simultaneously to maintain the consistency between the theme and background while editing foreground targets and backgrounds.

To address these challenges, this paper proposes MMIG, which can use text prompts to guide more accurate image editing and provide precise descriptions for the generated images. Firstly, the original image and text prompts are separately input into the multimodal LLM. Mixture-of-Modality Adaptation (MMA) [3] is used for cross-modal alignment between input images and text. Then, the newly generated text descriptions from LLM and the original image are input into DEM. To enhance the information fusion between multi-modal LLM and DEM, MMSM is proposed. Further, to improve the generation of images, a semantic-based layered controlled image editing method [4], Attention Segregation Loss, and Attention Retention Loss [5] are adopted. Finally, BLIP2 [6] is used to generate an accurate description of the newly generated image.

2. Related work

With the continuous improvement of diffusion models in image processing tasks, a series of new text-guided image editing methods based on diffusion models have emerged[2, 4, 5]. These text-guided image editing methods have produced surprising results by striving to adjust text prompts to meet certain expected attributes of the generated image [4, 5]. However, current methods of text-guided image editing based on the diffusion model are dif-

*Corresponding author: zsd@Hainanu.edu.cn

ficult to capture multiple concepts in the text description and to fully retain the features of the original image. To address these challenges, a variety of approaches based on semantic editing and attention loss functions have been proposed [3, 4, 5, 6]. However, these methods all have bottlenecks and have not utilized the powerful capabilities of LLM to combine with SDM. Therefore, this paper proposes MMIG, which combines multi-modal LLM with SDM for text-guided image editing. In addition, MMSM is proposed to enhance the fusion of information between LLM and SDM to generate better images.

3. Method

The architecture of the MMIG proposed in this paper is shown in Figure 1. The calculation of MMIG is shown as follows:

$$z_d = f_{mllm}(x, w) = f_{llm}(f_{clip}(x), w) \tag{1}$$

where x is the input image; w is the text prompt for input; z_d is the new text description for generating the new image; $f_{mllm}(\cdot)$ is the calculation of the multimodal LLM; $f_{clip}(\cdot)$ is the calculation of the input image by the image encoder; and $f_{llm}(\cdot)$ is the calculation of input text by LLM.

Then, the text description z_d and text embeddings e_{llm} generated by LLM and the original input image are input into DEM to generate a new image, calculated as follows:

$$v_M = f_{MMSM}(e_{llm}, v_{DEM}) \tag{2}$$

$$I_{new} = f_{DEM}(x, z_d, v_M) \tag{3}$$

where v_M is the output of MMSM; $f_{MMSM}(\cdot)$ is the calculation of MMSM; e_{llm} and v_{DEM} are the text embeddings and feature vectors inputted into MMSM by LLM and DEM, respectively; $f_{DEM}(\cdot)$ is the calculation of the image generative model; and I_{new} is the newly generated image.

Afterward, an accurate text description of the newly generated image is provided by BLIP2, calculated as follows:

$$d_{new} = f_{BLIP2}(I_{new}) \tag{4}$$

where d_{new} is the newly generated image, and $f_{BLIP2}(\cdot)$ is the calculation for generating a new description of the new image. Finally, the overall process of MMIG is shown in the following equation:

$$[I_{new}, d_{new}] = f_{MMIG}(x, w) \tag{5}$$

where $f_{MMIG}(\cdot)$ represents the calculation of input image x and input text prompts w by MMIG.

3.1. Mixture-of-Modality Adaptation

In the proposed MMIG, MMA [3] is introduced to adapt and process visual language instructions of LLM to generate image descriptions that can accurately express image semantics and text prompts. The Multimodal LLM structure

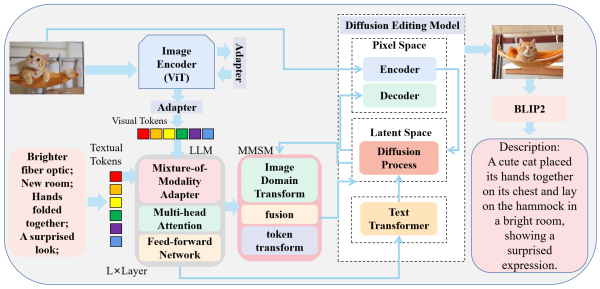

Figure 1. The architecture of MMIG.

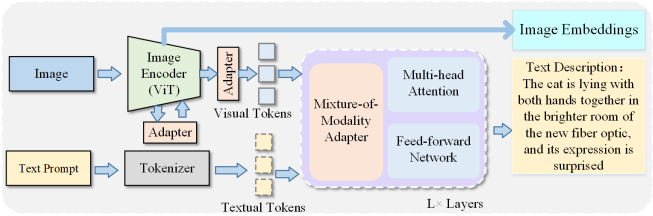

Figure 2. The structure of Multimodal LLM, which consists of an image encoder and a LLM with the MM-Adapter.

for processing image and text instructions is shown in Figure 2. MMA mainly includes Mixture-of-Modality Adapter (MM-Adapter) and Mixture-of-Modality Training (MMT). MM-Adapter can be defined as follows:

$$Z' = Z + s \times router(f_{a1}(Z), f_{a2}(Z); \widehat{w}) \tag{6}$$

where Z can represent single- or multi-modal features; s represents the scale factor; $router(\cdot)$ represents a routing function; $\widehat{w}$ is the route weight value; and $f_{a1}(\cdot)$ and $f_{a2}(\cdot)$ are RepAdapters.

Then, the entire multimodal LLM is jointly optimized through MMT, shown as follows:

$$\arg\min L_{MMT}(f_\Phi(I, T), R; \theta_a) \tag{7}$$

where R is the ground-truth response; $L_{MMT}(\cdot)$ is the loss function; $f_\Phi(\cdot)$ is the LLM; θ_a is the adaptation parameters; I is the input image; and R is the input text prompt.

Based on multimodal inputs, the LLM can progressively predict the next token, which can be represented as follows:

$$p_t = \prod_{s=1}^{S+1} p(R_s | Z, R_{0:s-1}; \theta_t, \theta_a) \tag{8}$$

where p_t is the probabilities of the predicted word; θ_t is the parameters of LLM; and θ_a is the parameters of adaptation modules.

3.2. Multi-modality mutual support module

To promote the image editing performance of DEM, a Multi-modal mutual Support module (MMSM), as shown

in Figure 3, is proposed by using LLM to assist DEM. It consists of an MLP (Multilayer Perceptron), a domain encoder, a domain decoder, and a cross-attention module. The calculation of MMSM is shown as follows:

$$e_3 = f_{MLP}(e_0) \tag{9}$$
$$e_2 = f_{DE}(e_1) \tag{10}$$
$$e_4 = f_{CA}(e_2, e_3) \tag{11}$$
$$e_6 = e_1 + f_{DD}(e_4) \tag{12}$$
$$L_{MMSM} = \frac{1}{2}(e_3 - e_2)^2 \tag{13}$$

where e_0 is the text embedding output by LLM; $f_{MLP}(\cdot)$ is the calculation of MLP; e_3 is the text embedding output through MLP; e_1 is the feature map output by DEM; $f_{DE}(\cdot)$ is the calculation of the domain encoder; e_2 is the feature vector output by the Domain encoder; e_4 is the weighted feature vector output by cross attention; $f_{CA}(\cdot)$ is the calculation of cross attention; $f_{DD}(\cdot)$ is the calculation of the domain decoder; e_6 is the feature map input into DEM; and L_{MMSM} is the loss functions of MMSM.

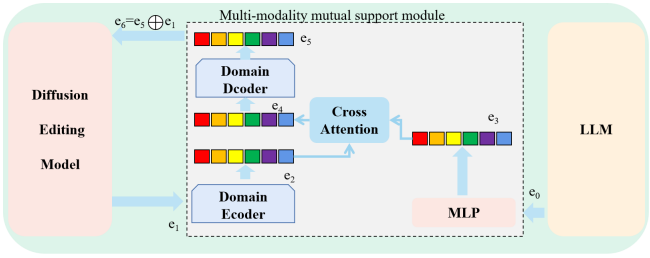

Figure 3. The structure of MMSM.

3.3. Diffusion editing model

The diffusion editing model (DEM) is based on SDM [1]. In DEM, a semantic-based layered controlled image editing method [4] and two loss functions are used to generate images more in line with text prompts [5]. The structure of DEM is shown in Figure 4. The Loss function of SDM is calculated as follows:

$$L_{SDM} = E_{\alpha(x),\varepsilon\sim N(0,1),t}[||\varepsilon - \varepsilon_\theta(z_t, t)||_2^2] \tag{14}$$

where $\varepsilon_\theta(z_t, t)$ represents an equally weighted sequence of denoising autoencoders; t=1...T; $\alpha(x)$ is the output of input x through the encoder; and z_t is a latent representation of the predicted noise of input x.

The adoptions of the semantic-based layered controlled image editing method [4] are as follows: first, the target image is obtained using the input text; next, the target image is decomposed into background and object attributes, which are input into the text encoder to obtain corresponding text

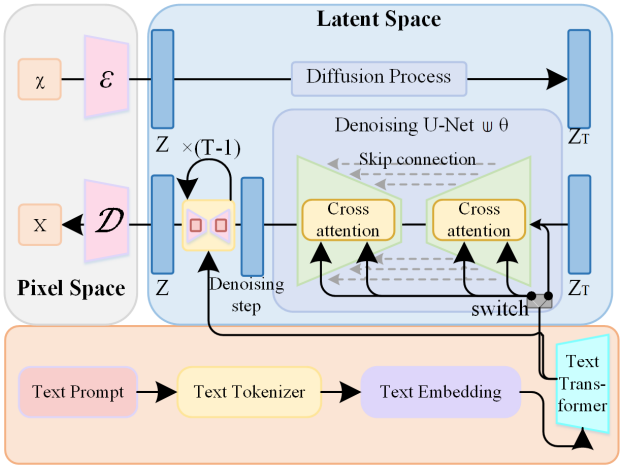

Figure 4. The structure of DEM. x represents the input image. $\tilde{x}$ represents the generated image. ε is the encoder. D is the decoder. Z is the latent representation. Z_T is the latent representation with added noise. T_θ is a domain-specific encoder.

vectors; subsequently, these two vectors undergo initial optimization to make them more aligned with the target image. The calculation is as follows:

$$[\hat{e}_a, \hat{e}_b] = \arg\min E_{x_t,\varepsilon\sim N(0,I)}[||M * (\varepsilon - f_\theta(x_t, t, [e_a, e_b]))||_2^2] \tag{15}$$

where $\hat{e}_a$ and $\hat{e}_b$ are text embeddings that describe object properties and background, respectively; t is the time step; x_t is the noisy version of the input; $f_\theta(\cdot)$ is the forward diffusion process using a pre-trained diffusion model; and M is the object mask.

Then optimize the text embeddings as follows:

$$e_{opt} = \alpha * \hat{e}_a + (1 - \alpha) * \hat{e}_b \tag{16}$$

where α is the weight value that describes object properties, which is 0.7.

In addition, the process for achieving arbitrary modifications and combinations of foreground target attributes and backgrounds is as follows:

$$L_{fg} = E_{x_t,\varepsilon\sim N(0,I)}[||M_t * (\varepsilon - f_\theta(x_t, t, e_{opt})||] \tag{17}$$

$$L_{bg} = E_{x_t,\varepsilon\sim N(0,I)}[||(1 - M_r) * (\varepsilon - f_\theta(x_t, t, e_{opt})||] \tag{18}$$

where M_t and $1 - M_t$ are the target and background of the target image, respectively; while M_r and $1 - M_r$ are the target and background of the reference image, respectively.

To address the issue of concept overlap in the given text prompts, Attention Segregation Loss [5] is utilized, which is calculated as follows:

$$L_{AS} = \sum_{m,n \in C, \forall m > n} [\frac{\sum_{i,j} \min([A_t^m]_{ij}, [A_t^n]_{ij})}{\sum_{i,j} ([A_t^m]_{ij} + [A_t^n]_{ij})}] \quad (19)$$

where t is time step; C is the concept set; A_t^m and A_t^n are a pair of cross attention maps of concepts $m, n \in C$ at the time step t; and $[A_t^m]_{ij}$ is the pixel value at position (i, j).

Furthermore, Attention Retention Loss [5] is used to explicitly force the model to retain conceptual information throughout the denoising process through consistency constraints. It is calculated as follows:

$$L_{AR} = \sum_{m \in C} [1 - \frac{\sum_{i,j} \min([A_{t-1}^m]_{ij}, [B_t^m]_{ij})}{\sum_{i,j} ([A_{t-1}^m]_{ij} + [B_t^m]_{ij})}] \quad (20)$$

where B_t^m is a proxy for ground truth; and A_{t-1}^m is the next time step's attention map.

Through this loss function, we force the diffusion model to retain all concepts from the previous time steps, thus reducing attention attenuation.

The loss function of DEM is shown as follows:

$$L_{DEM} = L_{SDM} + L_{fg} + L_{bg} + L_{AS} + L_{AR} \quad (21)$$

3.4. BLIP2

BLIP-2 [6] cleverly utilizes a Query Former (Q-Former) to connect the Vision Model and Language Model, enabling accurate descriptions of new images. Firstly, Q-Former is pretrained, leveraging guidance from the image encoder to facilitate learning. Then, the output of Q-Former is fed into a frozen LLM (Language Model) through a fully connected network to guide the generation of textual descriptions for images.

3.5. Training process

The training for MMIG is divided into five stages: in stage 1, we train MMSM separately; in stage 2, joint training for MMSM and multimodal LLM; in stage 3, joint training for MMSM and DEM; in stage 4, joint training for multimodal LLM, MMSM, and DEM; in stage 5, end-to-end training for MMIG. The total loss of the proposed MMIG is calculated as follows:

$$L_{Total} = L_{SDM} + L_{AS} + L_{AR} + L_{fg} + L_{bg} + L_{MMT} + L_{BLIP2} + L_{MMSM} \quad (22)$$

where L_{BLIP2} represents the loss function of BLIP2; L_{SDM} is the loss function of SDM; L_{AS} is Attention Segregation Loss; L_{AR} is Attention Retention Loss; L_{fg} and L_{bg} are the loss function for achieving arbitrary modifications and combinations of foreground target attributes and backgrounds; L_{MMT} is the loss function of MMT; and L_{MMSM} is the loss function of MMSM.

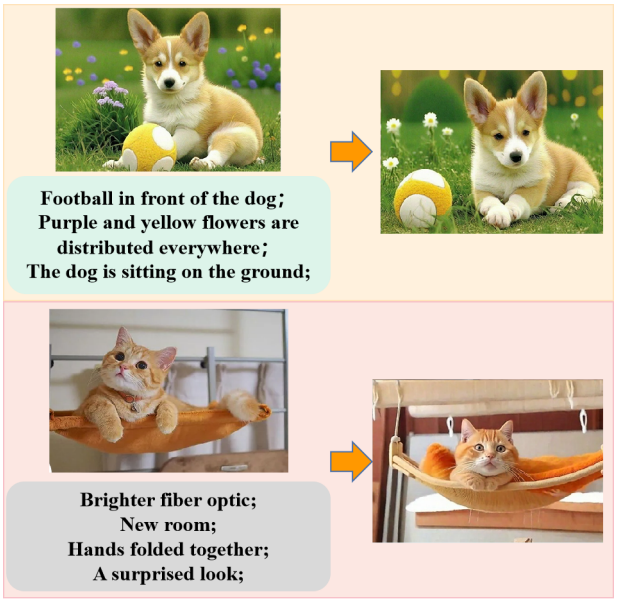

Figure 5. Experimental results of the separate training for MMSM in MMIG.

4. Experiment and results

4.1. Datasets and experiment settings

All experiments were conducted on a Linux server using Python language and PyTorch 1.10 framework. Specifically, the experiments were carried out on an NVIDIA A100 GPU with 80GB of VRAM.

4.2. Visualization

Currently, we complete separate training for MMSM. By inputting text prompts and images into MMIG, images with high similarity can be generated. The experimental results are shown in Figure 5. In the future, we will continue to conduct further stages of training to generate more similar images that can both highly preserve the original image features and maintain high consistency between specific subjects and backgrounds.

5. Discussion and conclusion

We are gonging to carry out a text-driven generative model of videos with high similarity and highly consistent background and theme in our future work.

References

[1] Rombach R, Blattmann A, Lorenz D, et al. High-resolution image synthesis with latent diffusion models[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022: 10684-10695.

[2] Hertz A, Mokady R, Tenenbaum J, et al. Prompt-to-prompt image editing with cross attention control[J]. arXiv preprint arXiv:2208.01626, 2022.

[3] Luo G, Zhou Y, Ren T, et al. Cheap and quick: Efficient vision-language instruction tuning for large language models[J]. arXiv preprint arXiv:2305.15023, 2023.

[4] Li P, Huang Q I, Ding Y, et al. LayerDiffusion: Layered Controlled Image Editing with Diffusion Models[J]. arXiv preprint arXiv:2305.18676, 2023.

[5] Agarwal A, Karanam S, Joseph K J, et al. A-STAR: Test-time Attention Segregation and Retention for Text-to-image Synthesis[J]. arXiv preprint arXiv:2306.14544, 2023.

[6] Li J, Li D, Savarese S, et al. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models[J]. arXiv preprint arXiv:2301.12597, 2023.