

DiffEditor: Boosting Accuracy and Flexibility on Diffusion-based Image Editing

Chong Mou^{1,3} Xintao Wang² Jiechong Song¹ Ying Shan² Jian Zhang^{1,3*}

¹School of Electronic and Computer Engineering, Shenzhen Graduate School, Peking University

²ARC Lab, Tencent PCG ³Peking University Shenzhen Graduate School-Rabbitpre AIGC Joint Research Laboratory

{eechongm, xintao.alpha}@gmail.com, yingsshan@tencent.com, {songjiechong, zhangjian.sz}@pku.edu.cn

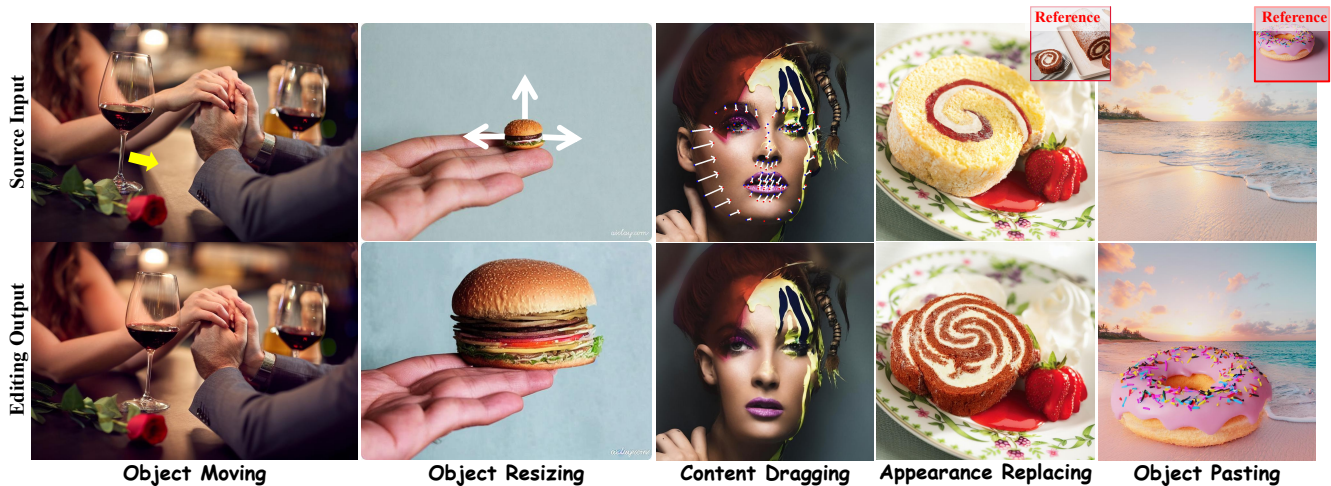


Figure 1. Our **DiffEditor** can perform various fine-grained image editing tasks. Given an image, users can select an object to move or resize. Or they can drag content with points. Moreover, users can choose a reference image for object pasting and appearance replacing.

Abstract

Large-scale Text-to-Image (T2I) diffusion models have revolutionized image generation over the last few years. Although owning diverse and high-quality generation capabilities, translating these abilities to fine-grained image editing remains challenging. In this paper, we propose **DiffEditor** to rectify two weaknesses in existing diffusion-based image editing: (1) in complex scenarios, editing results often lack editing accuracy and exhibit unexpected artifacts; (2) lack of flexibility to harmonize editing operations, e.g., imagine new content. In our solution, we introduce image prompts in fine-grained image editing, cooperating with the text prompt to better describe the editing content. To increase the flexibility while maintaining content consistency, we locally combine stochastic differential equation (SDE) into the ordinary differential equation (ODE) sampling. In addition, we incorporate regional score-based gradient guidance and a time travel strategy into the diffusion sampling, further improving the editing

quality. Extensive experiments demonstrate that our method can efficiently achieve state-of-the-art performance on various fine-grained image editing tasks, including editing within a single image (e.g., object moving, resizing, and content dragging) and across images (e.g., appearance replacing and object pasting). Our source code is released at <https://github.com/MC-E/DragonDiffusion>.

1. Introduction

Text-to-image (T2I) diffusion models [29, 32, 33, 35] have become the mainstream of image generation, praised for their high-quality and diverse generation capability. The pre-trained T2I models can serve as a good generation prior and can be used in various ways, e.g., image editing. Since the excellent text-to-image ability, numerous diffusion-based image editing methods are implemented based on the text guidance [5, 6, 11–13, 16]. However, the generated results of T2I models are usually sensitive to the quality of text [37]. Therefore, text-guided image editing struggles to achieve fine-grained content manipulation.

Recently, DragGAN [30] provides a user-friendly way

*Corresponding author

This work was supported by National Natural Science Foundation of China under Grant 62372016.

to manipulate the image content by point dragging. However, limited by the capacity of GAN [8] models, DragGAN cannot edit general images. Inspired by this interactive editing mode, DragDiff [39] and DragonDiff [28] are proposed based on the pre-trained T2I diffusion model [33]. Empowered by the diverse generation capabilities of the base model, they can perform fine-grained editing on general images. However, their editing process lacks flexibility, as shown in Fig. 2. Concretely, the image editing operation of transforming a lion from a closed mouth to a widely open mouth conflicts with the LORA [34] in DragDiff, resulting in failure. The visual cross-attention designed in DragonDiff also makes it struggle to imagine new content (e.g., mouth) that is not present in the source image, causing failure too. In addition, these two methods and most diffusion-based image editing methods employ ordinary differential equations (ODE) [41] solver, a deterministic sampling process. Although ODE can better maintain the consistency between the edited results and the source image, its determinacy also limits flexibility during the editing process. Compared to ODE, stochastic differential equations (SDE) [15] is a stochastic sampling process. Some works [48, 49] study the latent space of SDE for accurate image editing. Unlike these works, we aim to utilize the stochasticity in SDE to improve the flexibility of diffusion-based image editing, as shown in the last image of Fig. 2.

Another insight is that although DragDiff and DragonDiff utilize feature correspondence in the pre-trained T2I diffusion model to achieve fine-grained image editing, the role of the text input is ignored in their frameworks. Here, we raise a question: *Does the text have no effectiveness in fine-grained image editing, or is there another more suitable form of text input?* In addition to the text prompt, DALL-E2 [32] presents a novel attempt to generate images conditioned on the image prompt, i.e., using images to describe images. Subsequently, some multimodal works [20, 24, 25] and object-customization works [22, 47, 51] are proposed to support image prompts for more detailed content description. Inspired by these works, we introduce image prompts into the fine-grained image editing process, improving editing quality through more detailed content descriptions. In addition, we combine regional score-based gradient guidance and a time travel strategy into diffusion sampling, which further enhances the editing quality.

In summary, this paper has the following contributions:

- We present a novel attempt to introduce the image prompt to fine-grained image editing tasks. In conjunction with the image editing algorithm, this design can provide a more detailed description of the editing content, thus improving the editing quality.
- We consider both the flexibility and content consistency of image editing by proposing regional SDE sampling and regional score-based gradient guidance. We also intro-



Figure 2. Illustration of editing flexibility limitations in DragDiff [39] and DragonDiff [28], as well as our improvement.

duce a time travel strategy in diffusion-based image editing to improve the editing quality further.

- Extensive experiments demonstrate that our method can achieve state-of-the-art performance on various fine-grained image editing tasks (i.e., content dragging, object moving, resizing, pasting, and appearance replacing, as shown in Fig. 1) with attractive complexity.

2. Related Work

2.1. Diffusion Model

Diffusion model [15] is a thermodynamics-driven [40, 42] algorithm, including a diffusion process and a reverse process. In the diffusion process, an image \mathbf{x}_0 is gradually added Gaussian noise as $q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\sqrt{\alpha_t}\mathbf{x}_0, (1-\alpha_t)\mathbf{I})$, where α_t linearly decreases from 1 to a sufficiently small number to encourage $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$. The reverse process is to iteratively recover \mathbf{x}_0 from \mathbf{x}_T by training a denoiser, conditioned on the current noisy image \mathbf{x}_t and time step t :

$$\mathbb{E}_{\mathbf{x}_0, t, \epsilon_t \sim \mathcal{N}(0, \mathbf{I})} [\|\epsilon_t - \epsilon_\theta(\mathbf{x}_t, t)\|_2^2], \quad (1)$$

where ϵ_θ is the function of the denoiser. DDIM [41] defines the diffusion sampling as $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\sqrt{\alpha_{t-1}}\mathbf{x}_0 + \sqrt{1-\alpha_{t-1}-\delta_t^2} \cdot \frac{\mathbf{x}_t - \sqrt{\alpha_t}\mathbf{x}_0}{\sqrt{1-\alpha_t}}, \alpha_t^2\mathbf{I})$, which is a non-Markovian process and can be formulated as:

$$\mathbf{x}_{t-1} = \underbrace{\frac{\mathbf{x}_t - \sqrt{1-\alpha_t}\epsilon_\theta^t(\mathbf{x}_t)}{\sqrt{\alpha_t}}}_{\text{"predicted } \mathbf{x}_0 \text{"}} + \underbrace{\sqrt{1-\alpha_{t-1}-\sigma_t^2} \cdot \epsilon_\theta^t(\mathbf{x}_t)}_{\text{"direction pointing to } \mathbf{x}_t \text{"}} + \underbrace{\sigma_t \epsilon}_{\text{"noise"}}, \quad (2)$$

where $\sigma_t = \eta \sqrt{(1-\alpha_{t-1})/(1-\alpha_t)} \sqrt{1-\alpha_t/\alpha_{t-1}}$. When $\eta = 1$ for all t , it becomes DDPM [15], i.e., a stochastic differential equation (SDE). As $\eta = 0$, the sampling process becomes deterministic, i.e., an ordinary differential equation (ODE). Most diffusion-based image editing works rely on ODE to achieve better content consistency. [48, 49] explore SDE in diffusion-based image editing.

Most current works focus on conditional diffusion generation, e.g., text conditions [29, 33], greatly revolutionizing the community of image generation. Although promising T2I generation quality, the generated results are sensitive to text and usually require tedious prompt design [37]. In addition to text condition, DALL-E2 [32] presents the

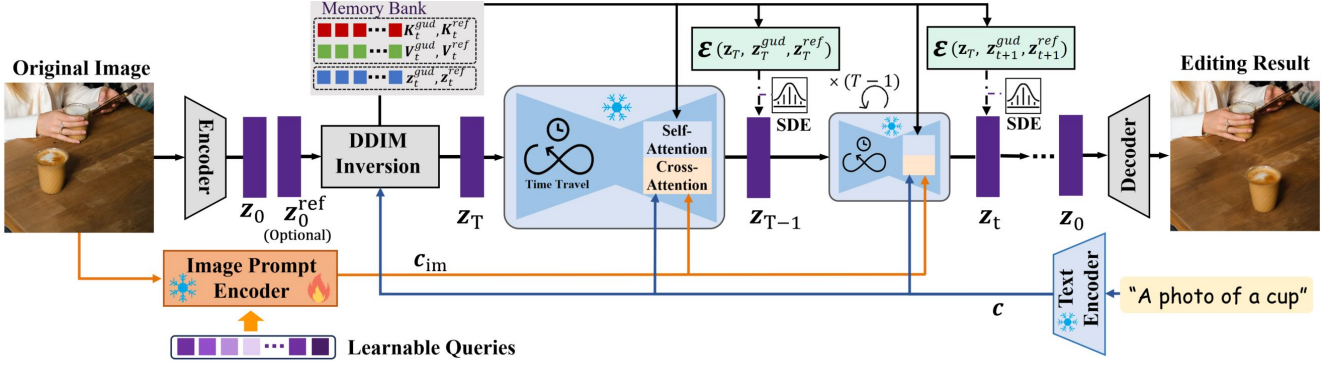


Figure 3. Overview of our proposed DiffEditor, which is composed of a trainable image prompt encoder and a diffusion sampling with editing guidance that does not require training.

first attempt to generate images guided by image prompts. ELITE [47], Bilp-Diffusion [22], and IP-Adapter [51] present the learning of image prompts for object customization. However, the effectiveness of image prompts in fine-grained image editing has hardly been studied.

2.2. Image Editing

The primary objective of image editing is to manipulate the content of a given image in a controlled manner. Previous methods [1–3] usually invert images into the latent space of GANs [8] and then edit the image by manipulating latent vectors. Recently, DragGAN [30] presents a point-dragging formulation for fine-grained image editing. However, limited by the capability of GANs, these methods have weaknesses in model generalization and image quality. Motivated by the success of text-to-image diffusion models [33], various text-guided image editing methods [4, 6, 13, 18, 27] are proposed. The commonly used editing strategies are (1) adding noise and then denoising with target description [7, 18, 21, 27, 46]; (2) using cross-attention maps as an editing medium [11, 13, 16]; (3) using text as editing instructions [6]. However, the correspondence between the text and image in T2I models is weak, making it difficult to achieve fine-grained image editing. Recently, DragDiff [39] and DragonDiff [28] achieve fine-grained image editing based on the feature correspondence [45] in the pre-trained StableDiffusion (SD) [33]. Specifically, DragDiff uses LORA [34] to maintain content consistency and optimizes the latent z_t in a specific diffusion step. DragonDiff is built based on the score-based [44] gradient guidance [9] and a visual cross-attention design for drag-style image editing without model tuning.

3. Method

3.1. Preliminary: Score-based Editing Guidance

From the continuous perspective of score-based diffusion [43, 44], the external condition y can be combined in a conditional score function, *i.e.*, $\nabla_{x_t} \log q(x_t|y)$, to sam-

ple from a more enriched distribution. The conditional score function can be further decomposed as:

$$\begin{aligned} \nabla_{x_t} \log q(x_t|y) &= \nabla_{x_t} \log \left(\frac{q(y|x_t)q(x_t)}{q(y)} \right) \\ &\propto \nabla_{x_t} \log q(x_t) + \nabla_{x_t} \log q(y|x_t), \end{aligned} \quad (3)$$

where the first term is the unconditional denoiser, *i.e.*, $\epsilon_\theta^t(x_t)$. The second term refers to the conditional gradient produced by an energy function $\mathcal{E}(x_t, y) = \log q(y|x_t)$, measuring the distance between current state x_t and condition y . Here, we reformulate Eq. 3 as:

$$\tilde{\epsilon}_\theta^t(x_t) = \epsilon_\theta^t(x_t) + \eta \cdot \nabla_{x_t} \mathcal{E}(x_t, y), \quad (4)$$

where η refers to the learning rate. Recently, Self-Guidance [11] and DragonDiff [28] convert image editing operations into gradient guidance for image editing tasks. The energy function \mathcal{E} in Self-Guidance is built based on the correspondence [13] between image and text features. DragonDiff constructs the energy function based on image feature correspondence [45] in pre-trained SD, which can achieve more accurate drag-style editing tasks. In this paper, we aim to boost the accuracy and flexibility of diffusion-based image editing with the DragonDiff framework.

3.2. Overview

An overview of our image editing pipeline is presented in Fig. 3. Specifically, given an image x_0 to be edited, we first employ it as image prompts and use an image prompt encoder to embed it. These image embeddings cooperate with text embeddings to form a better description to guide the diffusion process. Then we use DDIM inversion [41] to transform x_0 into a Gaussian distribution z_T in the latent space of the pre-trained SD [33]. If the reference image x_0^{ref} exists (*i.e.*, in appearance replacing and object pasting), it will also be involved in the inversion. In this process, we follow the design in DragonDiff [28] to store some intermediate features (K_t^{gud} , V_t^{gud} , K_t^{ref} , V_t^{ref}) and latent (z_t^{gud} , z_t^{ref}) at each time step in a memory bank, which is

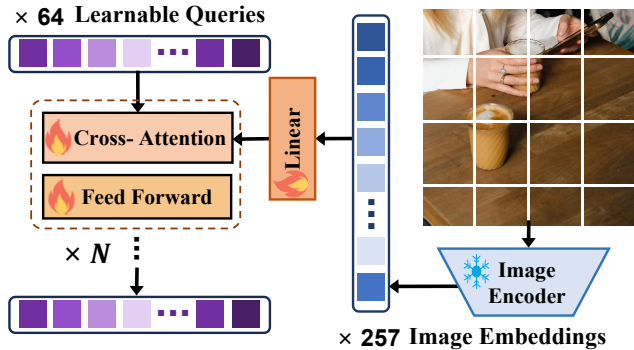


Figure 4. Illustration of the design of our image prompt encoder.

used to guide subsequent image editing. Note that "gud" and "ref" represent the information of the source and reference image in the inversion process, respectively. In the subsequent generation sampling, we step forward with the cooperation of score-based editing guidance, visual cross-attention, and image prompt. In this process, some elaborately designed strategies (*e.g.*, regional gradient guidance, regional SDE, and time travel) enhance the editing further.

3.3. Content Description with Image Prompt

Although several fine-grained image editing methods [28, 39] are based on the T2I diffusion model, the role of prompts is ignored as a simple description. Compared to text prompts, image prompts [22, 32, 51] can provide a more detailed content description. In this paper, we find that the image prompt can improve the quality of fine-grained image editing, especially in some complex scenarios.

Inspired by IP-Adapter [51], the architecture of our image prompt encoder is shown in Fig. 4. Concretely, given an input image x_0 , the pre-trained CLIP [31] image encoder embed it to 257 tokens. Then, a linear layer is used to adjust the channel dimension, and a QFormer [23] (without self-attention layer) module is employed to adjust the token numbers to 64 by 64 learnable queries. The QFormer module consists of N (8 by default) submodules, each composed of a cross-attention layer and a feed-forward network (FN). 64 learnable queries serve as queries to extract information from 257 image tokens that act as keys and values. Finally, 257 image tokens are composed into 64 embedding tokens (c_{im}), and then they are input into the same cross-attention module as text tokens (c) in the SD. To build classifier-free guidance [14] like the text condition, the conditional and unconditional image prompts are jointly trained by randomly dropping (*i.e.*, set image to zero) during training. Finally, image tokens and text tokens are processed separately with the query Q in the cross-attention module, and the results are added together:

$$\text{Att}(Q, K', V', K'', V'') = \mathcal{S}\left(\frac{Q(K')^T}{\sqrt{d}}\right)V' + \gamma \cdot \mathcal{S}\left(\frac{Q(K'')^T}{\sqrt{d}}\right)V'', \quad (5)$$

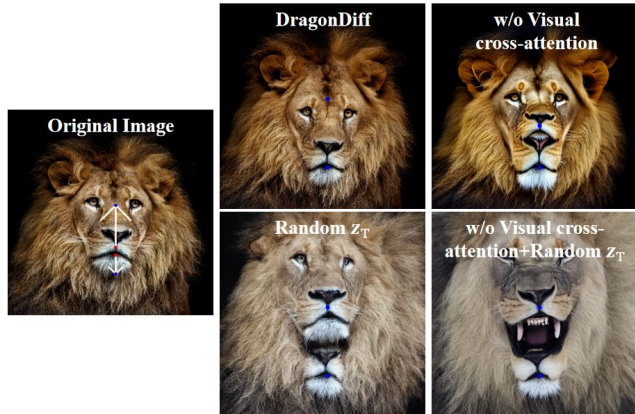


Figure 5. The impact of different components on the editing flexibility of DragonDiff [28].

where (K', V') and (K'', V'') refer to the keys and values from the text and image prompt, respectively. γ is a weight to balance these two terms. \mathcal{S} is the function of Softmax. Note that in tasks with reference images (*i.e.*, object pasting and appearance replacing), K'' and V'' are formed by the concatenation of image tokens from the source image and the reference image. During training, we fix the parameters in the pre-trained SD and CLIP image encoder, and we only optimize the linear embedding and QFormer by \mathcal{L}_2 loss:

$$\mathbb{E}_{x_0, t, \epsilon_t \sim \mathcal{N}(0,1)} \left[\|\epsilon_t - \epsilon_\theta^t(z_t, c, c_{im})\|_2^2 \right]. \quad (6)$$

After training, this module can be integrated into pre-trained SD for various editing tasks, as demonstrated in this paper.

3.4. Sampling with Regional SDE

Maintaining consistency between editing results and original images is a great challenge in fine-grained image editing. Most methods adopt a deterministic sampling process (ODE) and utilize DDIM inversion for sampling initialization. In addition, DragDiff [39] uses LORA [34] to constrain the output content, and DragonDiff [28] uses visual cross-attention to maintain content consistency. However, these strategies also compromise editing flexibility, *e.g.*, hindering the imagination of new content to harmonize editing operation as shown in Fig. 2. Our further experiments on DragonDiff show that reducing the content consistency strength can improve editing flexibility. As seen in Fig. 5, the editing flexibility is improved when we randomly initialize the sampling starting point z_T or remove visual cross-attention. When we apply both reductions, the editing objective can be achieved flexibly, but the content consistency is severely compromised. Therefore, in this paper, we explore how to improve editing flexibility without significantly impacting content consistency. In the sampling process (*i.e.*, Eq. 2) of DragonDiff, $\sigma_t = 0$, which is a deterministic ODE sampling. This leads to the final result being highly dependent on z_T and the information injected

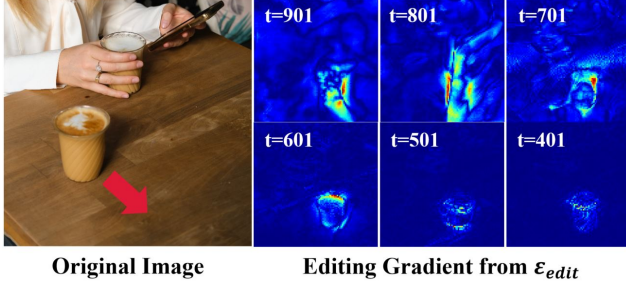


Figure 6. Editing gradient from \mathcal{E}_{edit} in different sampling steps.

by visual cross-attention. Our solution is to introduce randomness (*i.e.*, $\sigma_t > 0$) during the sampling process, while this randomness is controlled within local editing areas and specific time intervals. Here, we use $\mathbf{z}_{t-1} = \mathcal{F}(\mathbf{z}_t; \sigma_t)$ to simplify Eq. 2. Our regional SDE sampling is defined as:

$$\mathbf{z}_{t-1} = \mathbf{m}_{edit} \cdot \mathcal{F}(\mathbf{z}_t; \eta_1(t)) + (1 - \mathbf{m}_{edit}) \cdot \mathcal{F}(\mathbf{z}_t; \eta_2(t)),$$

$$(\eta_1(t), \eta_2(t)) = \begin{cases} (0.4, 0.2), & t \in \tau_{SDE} \\ (0, 0), & t \notin \tau_{SDE} \end{cases} \quad (7)$$

where \mathbf{m}_{edit} locates the editing area. τ_{SDE} is the time interval for applying regional SDE. After using this sampling strategy, we can accurately inject flexibility to produce satisfactory results, as shown in the last image of Fig. 2.

3.5. Editing with Gradient Guidance

Regional gradient guidance. In DragonDiff [28], the energy function \mathcal{E} consists of two parts, *i.e.*, editing \mathcal{E}_{edit} and content consistency $\mathcal{E}_{content}$. Although their target areas are independent of each other, the scope of the gradient guidance they generate is global and overlapping, resulting in mutual interference. Concretely, in Fig. 6, we visualize the editing gradient produced by \mathcal{E}_{edit} in the object moving task. As can be seen, the gradient guidance gradually converges to the editing area as the diffusion sampling proceeds. During this process, there are some activations outside the editing area, and these imprecise activations can affect the content consistency in these unedited areas (details are presented in Sec. 4.3). To rectify this weakness, we use the editing region mask \mathbf{m}_{edit} to locally combine \mathcal{E}_{edit} and $\mathcal{E}_{content}$. Finally, the conditional term in Eq. 3 is defined as:

$$\nabla_{\mathbf{z}_t} \log q(\mathbf{y}|\mathbf{z}_t) = \mathbf{m}_{edit} \cdot \nabla_{\mathbf{x}_t} \mathcal{E}_{edit} + (1 - \mathbf{m}_{edit}) \cdot \nabla_{\mathbf{x}_t} \mathcal{E}_{content}, \quad (8)$$

where \mathbf{y} is the editing target. During the sampling, we only add guidance in the first n time steps.

Time travel. DragDiff [39] treats \mathbf{z}_t as a learnable parameter and iteratively optimizes it within a diffusion step t . In contrast, DragonDiff [28] incorporates score-based gradient guidance into each sampling step, *i.e.*, Eq. 4. However, applying editing guidance through Eq. 4 once at each

Algorithm 1: Proposed DiffEditor

```

1 Require:
2 pre-trained SD [33]  $\epsilon_\theta$ ; image to be edited  $\mathbf{x}_0$ ; mask
  of the editing region  $\mathbf{m}_{edit}$ ; gradient-guidance
  steps  $n$ ; time interval  $\tau_{SDE}$  for SDE; time travel
  interval  $\tau_{TT}$  and the internal iterations  $U$ .
3 Initialization:
4 (1) Compute text embedding  $\mathbf{c}$  and image image
  embedding  $\mathbf{c}_{im}$ 
5 (2) invert  $\mathbf{x}_0$  to  $\mathbf{z}_T^{gen}$  and build the memory bank
6 for  $t = T, \dots, 1$  do
7   if  $t \in \tau_{TT}$  and  $t\%2 == 0$  then
8      $U_{cur} = U$ 
9   else
10     $U_{cur} = 1$ 
11   for  $u = 1, \dots, U_{cur}$  do
12     noise prediction:  $\hat{\epsilon}_t = \epsilon_\theta(\mathbf{z}_t, t, \mathbf{c}, \mathbf{c}_{im})$ ;
13     if  $T - t < n$  and  $t\%2 == 0$  then
14       compute  $\mathcal{E}_{edit}$  and  $\mathcal{E}_{content}$  by [28];
15       compute  $\nabla_{\mathbf{z}_t} \log q(\mathbf{y}|\mathbf{z}_t)$  by Eq. 8;
16       inject gradient guidance by Eq. 3;
17       compute  $\mathbf{z}_{t-1}$  by Eq. 7;
18     else
19       compute  $\mathbf{z}_{t-1}$  by Eq. 2 ( $\sigma_t = 0$ );
20     if  $U_{cur} > 1$  then
21        $\mathbf{z}_t = \frac{\mathbf{z}_{t-1} - \sqrt{1 - \alpha_t} \epsilon_\theta(\mathbf{z}_t, t, \mathbf{c})}{\sqrt{\alpha_t}}$ 
22  $\mathbf{x}_0 = Decoder(\mathbf{z}_0)$ ;
23 Output:  $\mathbf{x}_0$ 

```

sampling step lacks refinement for editing, especially in some complex scenarios. *Can we combine the advantages of DragDiff and DragonDiff to build recurrent guidance in the score-based diffusion [44]?* To address this issue, we build time travel to perform rollbacks, *i.e.*, $\mathbf{z}_t \leftarrow \mathbf{z}_{t-1}$, during the sampling process. This strategy has been empirically shown to inhibit the generation of disharmonious results when solving hard generation tasks [26, 52]. However, the rollback strategy (*i.e.*, $\mathbf{z}_t \sim \mathcal{N}(\sqrt{1 - \beta_{t-1}}\mathbf{z}_{t-1}, \beta_{t-1}\mathbf{I})$) in these works is not suitable in fine-grained image editing tasks. This is because random noise \mathbf{I} can introduce significant uncertainty, undermining the content consistency of editing results. To ensure the accuracy of rollback, we use deterministic DDIM inversion [41] to roll back \mathbf{z}_{t-1} to \mathbf{z}_t . During sampling, the time travel is performed U (3 in our design) times for each guidance step in a time interval τ_{TT} .

Due to the guidance enhancement from regional guidance and time travel, we can achieve editing with fewer guidance time steps, *i.e.*, we introduce gradient guidance every two time steps in sampling. Finally, the algorithm logic of our DiffEditor is defined in Alg. 1.

Table 1. Quantitative evaluation on face manipulation with 68 and 17 points. The accuracy is calculated by MSE distance between edited points and target points. The initial distance (*i.e.*, 57.19 and 36.36) is the upper bound, without editing. FID [38] is utilized to quantize the editing quality of different methods. The time complexity is computed on the ‘1 point’ dragging.

	Preparing complexity↓	Inference complexity↓	Unaligned face	17 Points↓ From 57.19	68 Points↓ From 36.36	FID↓ 17/68 points
UserControllableLT [10]	1.1s	0.04s	✗	32.32	24.15	51.20/50.32
DragGAN [30]	50.22s	6.28s	✗	15.96	10.60	39.27/39.50
DragDiff [39]	42.37s	19.52s	✓	22.95	17.32	38.06/36.55
DragonDiff [28]	3.53s	15.00s	✓	18.51	13.94	35.75/34.58
DiffEditor (Ours)	3.53s	13.88s	✓	17.05	11.52	33.10/33.02

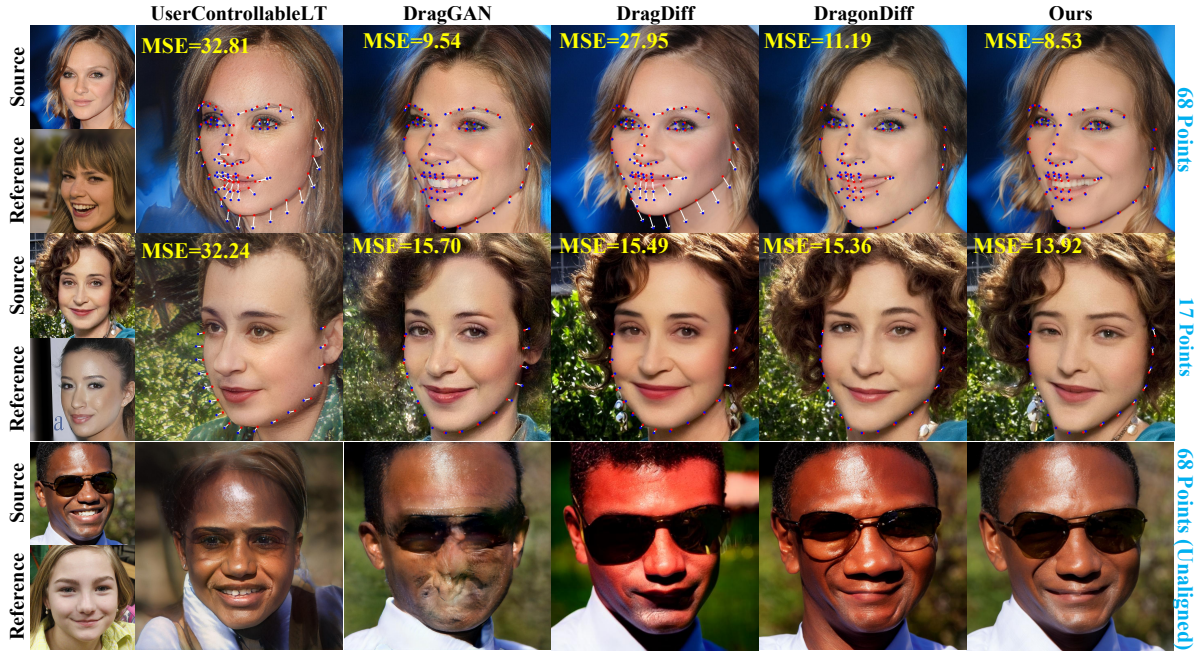


Figure 7. Qualitative comparison between our DiffEditor and other methods in face manipulation. The current and target points are labeled with red and blue. The white line indicates distance. The MSE distance between the result and the target is labeled in yellow.

4. Experiments

4.1. Implementation Details

We choose Stable Diffusion V1.5 [33] as the base model for image editing. During image prompt training, we use the training data from LAION [36] and process the image resolution to 512×512 . We choose Adam [19] optimizer with an initial learning rate of 1×10^{-5} . The batch size during training is set as 16. The training process iterates 1×10^6 steps on 4 NVIDIA A100 GPUs. We use the same embedding module to process image prompts in different applications. The inference adopts DDIM sampling with 50 steps, and we set the classifier-free guidance scale as 5.

4.2. Comparison

Time complexity. We divide the time complexity of different methods into preparing and inference stages. The preparing stage involves Diffusion/GAN inversion and model tuning. The inference stage generates the editing re-

Table 2. Quantitative evaluation on object pasting, object moving, and appearance replacing. The result is calculated by CLIP [31] \uparrow distance between editing results and target descriptions.

	Pasting	Moving	Replacing
Pain-by-example	0.265	-	-
Self-Guidance	-	0.246	0.243
DragonDiff	0.260	0.282	0.263
Ours	0.274	0.288	0.281

sult from latent representation. The time complexity for each method is tested on one point dragging, with the image resolution being 512×512 . All times are tested on an NVIDIA A100 GPU with Float32 precision. The results in Tab. 1 present the attractive preparing complexity of our method, and the inference complexity is lower than existing diffusion-based methods, *i.e.*, DragDiff and DragonDiff.

Performance. First, we evaluate our method on content dragging by comparing it with some well-known GAN-based methods (*i.e.*, UserControllableLT [10], Drag-

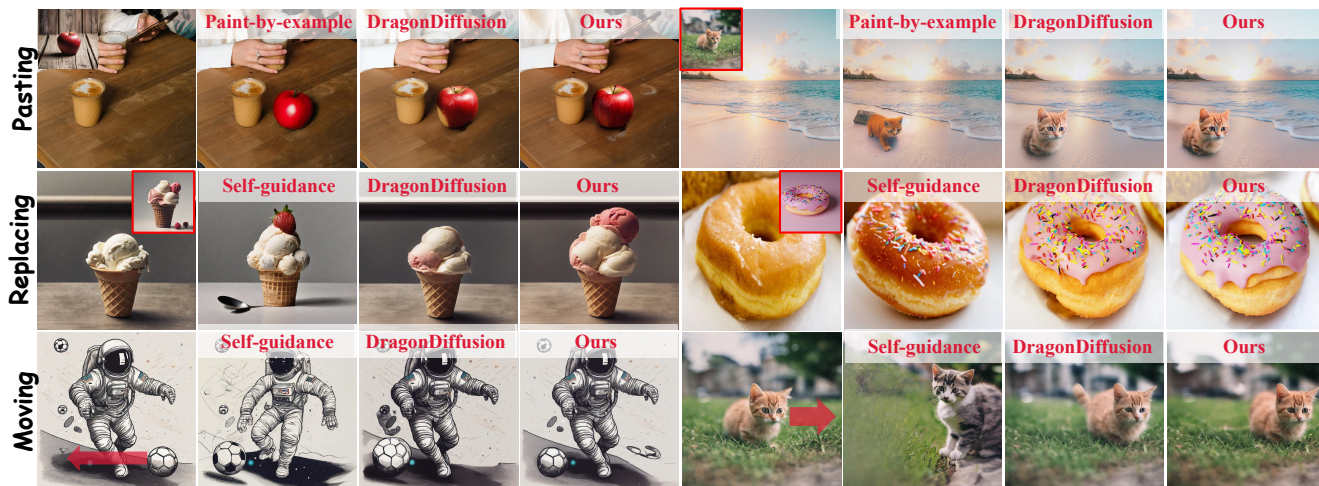


Figure 8. Visual comparison between our method and other methods on appearance replacing, object pasting and object moving tasks.

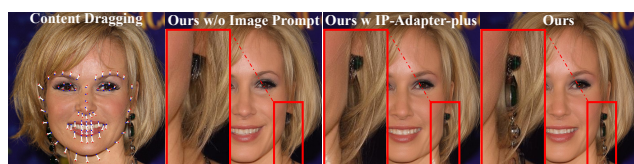


Figure 9. Visual comparison with IP-Adapter-plus [51].

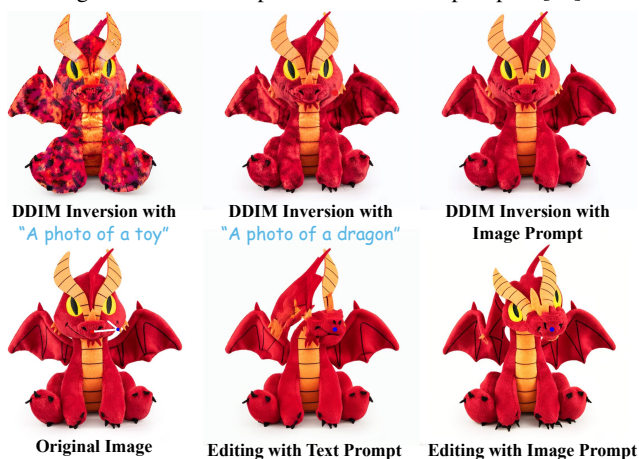


Figure 10. The first and second rows show the effectiveness of the image prompt in DDIM inversion and image editing, respectively.

GAN [30]) and recent diffusion-based methods (*i.e.*, DragDiff [39], DragonDiff) on the keypoint-based face manipulation. We used the same test set as DragonDiff, *i.e.*, 800 aligned faces from the CelebA-HQ [17] training set. We evaluate the editing performance under 17-point guidance and 68-point guidance. To quantify editing accuracy, we calculated the MSE distance between the landmarks of the edited result and the target landmarks. In addition, we calculate FID [38] between the editing results and the CelebA-HQ training set to represent the image quality. The quantitative comparison is presented in Tab. 1. One can see that our method has a significant improvement in accuracy and

generation quality compared to other diffusion-based methods, achieving comparable editing accuracy to DragGAN. Although DragGAN has higher editing accuracy on aligned faces, its base model is specifically trained for aligned faces and cannot edit general faces, as shown in the last row of Fig. 7. The qualitative comparison in Fig. 7 shows that our method has high editing accuracy and content consistency while maintaining good flexibility. For example, in the case where teeth need to be imagined, our DiffEditor can produce more natural results. In contrast, DragDiff and DragonDiff have difficulties in imagining new content.

In addition to content dragging, we also compare with Paint-by-example [50] in object pasting, and we compare our method with Self-Guidance [11] and DragonDiff in object moving and appearance replacing tasks. The results are presented in Fig. 8. As can be seen, although the specially trained Paint-by-example can naturally integrate objects into an image, it is difficult to maintain the original object identity. Our method performs better in object identity and has richer texture details than DragonDiff. In object moving and appearance replacing tasks, text-guided Self-Guidance lacks consistent constraints, making editing results deviate from the original image. The editing accuracy of DragonDiff still has room for improvement, *e.g.*, color and details. In comparison, our method has better content consistency and editing accuracy. For quantization, we compute the CLIP [31] distance between the edited results and the target description. We select 16 editing samples for each task. The results in Tab. 2 demonstrate the promising performance of our method.

Discussion between our image prompt and IP-Adapter.

As mentioned above, there are several methods proposed to use images as prompts to provide more accurate and customized descriptions for the generated results, such as IP-Adapter [51]. However, most of these methods focus on ob-



Figure 11. Effectiveness of the regional gradient guidance. In experiment, we remove the content consistency gradient $\mathcal{E}_{content}$.

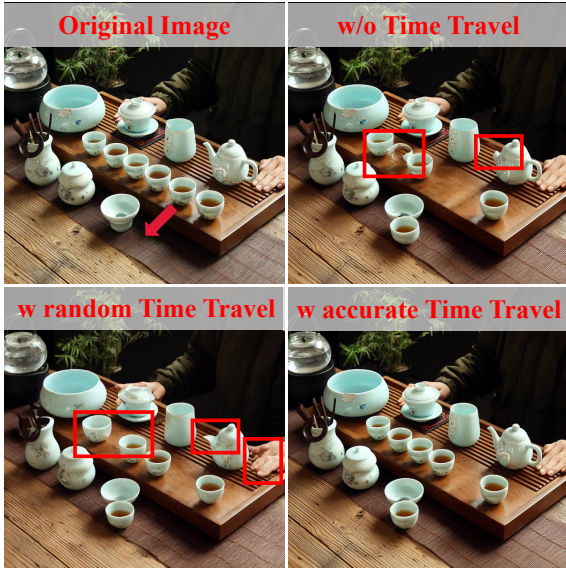


Figure 12. Visualization of editing without time travel, with random time travel, and with our accurate time travel.

ject customization, and overemphasis on detail description will compromise their performance. Therefore, IP-Adapter compresses the image into a small number of tokens to avoid detail descriptions. This paper studies introducing image prompts into fine-grained image editing. We use the Q-Former structure to map the image into 64 tokens to enhance the detail expression ability. Fig. 9 shows that IP-Adapter not ideal for direct insertion into fine-grained image editing tasks due to the lack of detail description. Our method can enhance the consistency of texture details between the edited result and the original image.

4.3. Ablation Study

Image prompt. Image prompt provides a detailed description of the editing content in our editing pipeline. We conduct an experiment in Fig. 10 to demonstrate its effectiveness. First, we apply it in the pure DDIM inversion and then reconstruction. The result in the first row presents that DDIM inversion based on the text prompt exhibits noticeable distortions and is unstable. After using our image prompt, DDIM inversion can reconstruct stable and high-fidelity results. In the second row of Fig. 10, we show the editing with and without the image prompt. It can be seen

that the image prompt provides a better generation prior for editing content, reducing the probability of distortion.

Regional gradient guidance. To rectify the interference between different gradient guidance, we use Eq. 8 to produce the guidance. We demonstrate its effectiveness in Fig. 11 by only using $m_{edit} \cdot \mathcal{E}_{edit}$ and \mathcal{E}_{edit} . Note that we remove the content consistency guidance $\mathcal{E}_{content}$ to highlight the interference. The results show that if the acting range of \mathcal{E}_{edit} is not constrained, the editing gradient will have an impact on the consistency of some unrelated areas, e.g., distortion of the fingers in the background. After applying regional constraints, the content of the background part has better consistency even without $\mathcal{E}_{content}$.

Time travel. Time travel is used to build recurrent guidance in a single diffusion time step, thereby refining the editing effect. We present its effectiveness in Fig. 12. As can be seen, the editing result in the complex scenario has distortions without the time travel strategy. If using the random time travel (i.e., $\mathbf{z}_t \sim \mathcal{N}(\sqrt{1 - \beta_{t-1}}\mathbf{z}_{t-1}, \beta_{t-1}\mathbf{I})$), the randomness will affect the consistency between the editing result and the original image. After adopting our accurate time travel, the editing quality is improved.

5. Conclusion

In this paper, we aim to rectify two issues in diffusion-based fine-grained image editing: (1) in complex scenarios, editing results often lack accuracy and exhibit unexpected artifacts; (2) lack of flexibility to harmonize editing operation, e.g., imagine new content. In our solution, we introduce image prompt into fine-grained image editing, providing a more detailed content description for the edited image. This method can be plugged into various fine-grained image editing tasks without task-specific training. To improve the editing flexibility, we propose a regional SDE strategy to inject randomness into the editing area while maintaining content consistency in other areas. Furthermore, we introduce regional score-based gradient guidance and a time travel strategy to improve the editing quality further. Extensive experiments demonstrate that our method can achieve promising performance in various fine-grained image editing tasks, i.e., object moving, resizing, pasting, appearance replacing, and content dragging. The complexity is also reduced compared with existing diffusion-based methods.

Limitations Although our method improves the flexibility of diffusion-based image editing and reduces distortion probability, editing difficulties still exist in some scenarios that require a large amount of content imagination, such as rotating a car by dragging its front. We think that this is due to the base model SD. It has a diverse generation space but lacks 3D perception of individual objects. In our future work, we will enhance the editing capabilities of diffusion models in this regard.

References

- [1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4432–4441, 2019. 3
- [2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan++: How to edit the embedded images? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8296–8305, 2020.
- [3] Yuval Alaluf, Omer Tov, Ron Mokady, Rinon Gal, and Amit Bermano. Hyperstyle: Stylegan inversion with hypernetworks for real image editing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18511–18521, 2022. 3
- [4] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18208–18218, 2022. 3
- [5] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 1
- [6] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 1, 3
- [7] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22560–22570, 2023. 3
- [8] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1):53–65, 2018. 2, 3
- [9] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 3
- [10] Yuki Endo. User-controllable latent transformer for stylegan image layout editing. In *Computer Graphics Forum*, pages 395–406. Wiley Online Library, 2022. 6
- [11] Dave Epstein, Allan Jabri, Ben Poole, Alexei Efros, and Aleksander Holynski. Diffusion self-guidance for controllable image generation. *Advances in Neural Information Processing Systems*, 36, 2024. 1, 3, 7
- [12] Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Reddy Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. In *Proceedings of the International Conference on Learning Representations*, 2022.
- [13] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. In *Proceedings of the International Conference on Learning Representations*, 2022. 1, 3
- [14] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 4
- [15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2
- [16] Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Direct inversion: Boosting diffusion-based editing with 3 lines of code. *arXiv preprint arXiv:2310.01506*, 2023. 1, 3
- [17] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *Proceedings of the International Conference on Learning Representations*, 2018. 7
- [18] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6007–6017, 2023. 3
- [19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6
- [20] Jing Yu Koh, Daniel Fried, and Russ R Salakhutdinov. Generating images with multimodal language models. *Advances in Neural Information Processing Systems*, 36, 2024. 2
- [21] Gihyun Kwon and Jong Chul Ye. Diffusion-based image translation using disentangled style and content representation. In *Proceedings of the International Conference on Learning Representations*, 2022. 3
- [22] Dongxu Li, Junnan Li, and Steven Hoi. Blip-diffusion: Pre-trained subject representation for controllable text-to-image generation and editing. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3, 4
- [23] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the International Conference on Machine Learning*, pages 19730–19742, 2023. 4
- [24] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*, 2023. 2
- [25] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024. 2
- [26] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022. 5
- [27] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *Proceedings of the International Conference on Learning Representations*, 2021. 3

- [28] Chong Mou, Xintao Wang, Jiechong Song, Ying Shan, and Jian Zhang. Dragdiffusion: Enabling drag-style manipulation on diffusion models. In *The Twelfth International Conference on Learning Representations*, 2023. 2, 3, 4, 5, 6
- [29] Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *Proceedings of the International Conference on Machine Learning*, pages 16784–16804, 2022. 1, 2
- [30] Xingang Pan, Ayush Tewari, Thomas Leimkühler, Lingjie Liu, Abhimitra Meka, and Christian Theobalt. Drag your gan: Interactive point-based manipulation on the generative image manifold. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 1, 3, 6, 7
- [31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763, 2021. 4, 6, 7
- [32] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 1, 2, 4
- [33] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1, 2, 3, 5, 6
- [34] Simo Ryu. Low-rank adaptation for fast text-to-image diffusion fine-tuning, 2023. 2, 3, 4
- [35] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 1
- [36] Christoph Schuhmann. Laion-400m. 6
- [37] Eyal Segalis, Dani Valevski, Danny Lumen, Yossi Matias, and Yaniv Leviathan. A picture is worth a thousand words: Principled recaptioning improves image generation. *arXiv preprint arXiv:2310.16656*, 2023. 1, 2
- [38] Maximilian Seitzer. pytorch-fid: FID Score for PyTorch. <https://github.com/mseitzer/pytorch-fid>, 2020. Version 0.3.0. 6, 7
- [39] Yujun Shi, Chuhui Xue, Jiachun Pan, Wenqing Zhang, Vincent YF Tan, and Song Bai. Dragdiffusion: Harnessing diffusion models for interactive point-based image editing. *arXiv preprint arXiv:2306.14435*, 2023. 2, 3, 4, 5, 6, 7
- [40] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 2
- [41] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2, 3, 5
- [42] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 2
- [43] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020. 3
- [44] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *Proceedings of the International Conference on Learning Representations*, 2020. 3, 5
- [45] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. *Advances in Neural Information Processing Systems*, 36:1363–1389, 2023. 3
- [46] Dani Valevski, Matan Kalman, Eyal Molad, Eyal Segalis, Yossi Matias, and Yaniv Leviathan. Unitune: Text-driven image editing by fine tuning a diffusion model on a single image. *ACM Transactions on Graphics (TOG)*, 42(4):1–10, 2023. 3
- [47] Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15943–15953, 2023. 2, 3
- [48] Chen Henry Wu and Fernando De la Torre. Unifying diffusion models’ latent space, with applications to cyclediffusion and guidance. *arXiv preprint arXiv:2210.05559*, 2022. 2
- [49] Chen Henry Wu and Fernando De la Torre. A latent space of stochastic diffusion models for zero-shot image editing and guidance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7378–7387, 2023. 2
- [50] Binxin Yang, Shuyang Gu, Bo Zhang, Ting Zhang, Xuejin Chen, Xiaoyan Sun, Dong Chen, and Fang Wen. Paint by example: Exemplar-based image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18381–18391, 2023. 7
- [51] Hu Ye, Jun Zhang, Sibio Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023. 2, 3, 4, 7
- [52] Jiwen Yu, Yinhuai Wang, Chen Zhao, Bernard Ghanem, and Jian Zhang. Freedom: Training-free energy-guided conditional diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23174–23184, 2023. 5