LEDITS++: Limitless Image Editing using Text-to-Image Models

Manuel Brack[1,2*†] Felix Friedrich[1,3*] Katharina Kornmeier[1*] Linoy Tsaban[4]
Patrick Schramowski[1,2,3] Kristian Kersting[1,2,3] Apolinário Passos[4]
[1]TU Darmstadt, [2]DFKI, [3]hessian.AI, [4]Huggingface
{brack,friedrich}@cs.tu-darmstadt.de

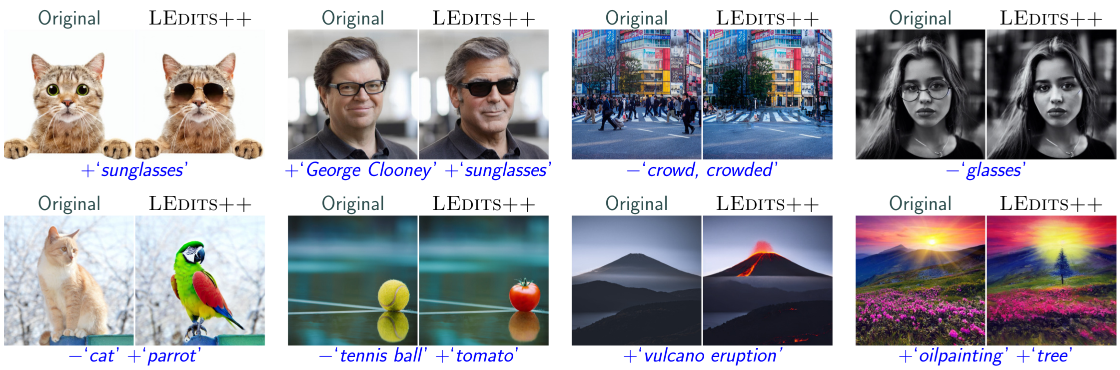

Figure 1. LEDITS++ facilitates versatile image-to-image editing. Several complex cases are available now.

Abstract

Text-to-image diffusion models have recently received increasing interest for their astonishing ability to produce high-fidelity images from solely text inputs. Subsequent research efforts aim to exploit and apply their capabilities to real image editing. However, existing image-to-image methods are often inefficient, imprecise, and of limited versatility. They either require time-consuming fine-tuning, deviate unnecessarily strongly from the input image, and/or lack support for multiple, simultaneous edits. To address these issues, we introduce LEDITS++, *an efficient yet versatile and precise textual image manipulation technique.* LEDITS++*'s novel inversion approach requires no tuning nor optimization and produces high-fidelity results with a few diffusion steps. Second, our methodology supports multiple simultaneous edits and is architecture-agnostic. Third, we use a novel implicit masking technique that limits changes to relevant image regions. We propose the novel TEdBench++ benchmark as part of our exhaustive evaluation. Our results demonstrate the capabilities of* LEDITS++ *and its improvements over previous methods.*

1. Introduction

Text-to-image diffusion models (DM) have garnered recognition for their ability to generate high-quality images from textual descriptions. A growing body of research has recently been dedicated to utilizing these models for manipulating real images. However, several barriers prevent many real-world applications of diffusion-based image editing. Current methods often entail computationally expensive model tuning or other optimization, presenting practical challenges [6, 18, 28, 30, 44]. Additionally, existing techniques have the proclivity to induce profound changes to the original image [17, 26], often resulting in completely different images. Lastly, all these approaches are inherently constrained when editing multiple (arbitrary) concepts simultaneously. We tackle these problems by introducing LEDITS++[1], a diffusion-based image editing technique addressing these limitations.

LEDITS++[2] offers a streamlined approach for textual image editing, eliminating the need for extensive parameter tuning. To this end, we derive image inversion for a more efficient diffusion sampling algorithm to a) drastically reduce computational resources and b) guarantee perfect image reconstruction. Thus, we overcome computational obstacles and avoid changes in the edited image in the first place. Furthermore, we use a novel implicit masking approach to semantically ground each edit instruction to its relevant image region. This further optimizes changes to the image by retaining the overall image composition and object identity.

*Equal contribution †Partially as research intern at Adobe
[1]LEDITS++ stands for ***Limitless Edits*** with sde-dpm-solver**++**.
[2]https://huggingface.co/spaces/leditsplusplus/project

Additionally, LEDITS++ is the only method to date to facilitate easy and versatile image editing by supporting multiple simultaneous instructions without causing undue interference. Finally, its lightweight architecture-agnostic nature ensures compatibility with both latent and pixel-based diffusion models, providing high accessibility.

In this work, we establish the methodical benefits of LEDITS++ and demonstrate that this intuitive, lightweight approach offers sophisticated semantic control for image editing. Specifically, we contribute by (i) devising a formal definition of LEDITS++ while (ii) deriving perfect inversion for a more efficient diffusion sampling method, (iii) qualitatively and empirically demonstrating its efficiency, versatility, and precision, (iv) providing an exhaustive empirical comparison to concurrent works with automatic and human user metrics, and thereby (v) introducing **T**extual **Ed**iting **B**enchmark++ (TEdBench++), a more holistic and coherent testbed for evaluating textual image manipulation.

2. Background

Recently, large-scale, text-guided DMs have enabled versatile applications in image generation [3, 33, 37]. Especially latent diffusion models [31, 34] have gained attention for their computational efficiency. Below, we discuss related work for efficient, versatile image manipulation with DMs.

Diffusion Sampling. Generating outputs with DMs requires multiple iterative denoising steps that constitute the main bottleneck at inference. Commonly used sampling methods such as DDPM [15] or DDIM [42] require tens or hundreds of steps to produce high-quality samples. Consequently, numerous works have been dedicated to speeding up the sampling process without loss in quality. Distillation efforts progressively reduce the number of required steps through further training [25, 27]. Other works focus on improving the sampling itself, e.g. using high-order ODE-solvers [22, 23, 46]. Such solvers can be readily combined with pre-trained DMs at inference to lower the number of denoising steps. With LEDITS++, we derive perfect image inversion with the DPM-Solver++, allowing image editing in as few as 20 total steps.

Semantic Control during Diffusion. While text-to-image DMs generate new, astonishing images, fine-grained control over the generative process remains challenging. Minor changes to the text prompt lead to entirely different outputs. Wu et al. [45] studied concept disentanglement using linear combinations of text embeddings to gain semantic control. Methods like Prompt-to-Prompt [14] and other works [8, 30] utilize the DM's attention layers to attribute pixels to tokens from the text prompt. Dedicated operations on the attention maps enable more control over the generated images. Other works have focused on the noise estimates of DMs [5, 20] providing semantic control over the generation process. With LEDITS++, we now enable fine-grained semantic control for manipulating real images, going beyond purely generative applications.

Real Image Editing. Since DMs' rise in popularity for text-to-image generation, they have also been explored for (real) image-to-image editing. As a first, simple approach, SDEdit added noise to the image for an intermediate step in the diffusion process [26]. While lightweight, the resulting image diverges substantially from the input as it is (partially) regenerated. Inpainting allows to keep the change small by having a user provide additional masks to restrict changes to certain image regions [2, 29]. Yet, user masks are costly or often simply unavailable. Other works have thus explored semantically grounded approaches using cross-attention instead to better control image manipulation [7, 28, 30]. In contrast, LEDITS++ leverages both attention- and noise-based masking to obtain fine-grained masks, enabling strong semantic control over real images.

Another important aspect of image manipulation methods is the required tuning and overall runtime. InstructPix2Pix continues training a DM at scale to enable image editing capabilities [6]. Finetuning instead on each individual input to constrain the generation on the real image has shown helpful [18, 44] but not computationally efficient. Consequently, recent works have largely relied on inverting the deterministic DDIM sampling process [42] to save computational resources. DDIM inversion identifies an initial noise vector that results in the input image when denoised again. However, faithful reconstructions are only obtained in the limit of small steps, thus requiring large numbers of inversion steps. Moreover, small errors will still incur at each timestep, often accumulating into meaningful deviations from the input, requiring costly error correction through optimization [28, 30]. Recently, Huberman-Spiegelglas *et al.* proposed an inversion technique [17] for the DDPM sampler [15] to address the limitations of DDIM inversion. LEDITS++ provides the same guarantees of perfect inversion with even further reduced runtime alongside an edit-friendly latent space, enabling more versatility.

3. Image Editing with Text-to-Image Models

Before devising the methodology of LEDITS++, let us first motivate the desired features and use cases. Specifically, we aim for efficiency, versatility, and precision. The goal is to provide a method that enables a fast exploratory workflow for image editing in which a user can iteratively interact with the model and explore various edits. Consequently, LEDITS++ produces outputs quickly with no tuning or optimization to not disrupt the creative process. Further, arbitrary editing instructions and combinations thereof are supported to facilitate a wide range of image manipulations (e.g., complex multi-editing). Lastly, we provide precise and sophisticated semantic control over the image editing. Each of the (potentially multiple) edit instructions can be steered individually, and changes are automatically restricted to relevant image regions. Importantly, with LEDITS++ we prioritize compositional robustness.

3.1. Guided Diffusion Models

Let us first define some general background for diffusion models. DMs iteratively denoise a Gaussian distributed variable to produce samples of a learned data distribution. Let's consider a diffusion process that gradually turns an image x_0 into Gaussian noise.

$$x_t = \sqrt{1-\beta_t}x_{t-1} + \sqrt{\beta_t}n_t, \qquad t = 1, ..., T \quad (1)$$

where n_t are iid normal distributed vectors and β_t a variance schedule. The diffusion process is equivalently expressed as

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t \quad (2)$$

where $\alpha_t = 1-\beta_t$, $\bar{\alpha}_t = \Pi_{s=1}^t \alpha_s$ and $\epsilon_t \sim \mathcal{N}(0, \mathbf{I})$. Importantly, all ϵ_t are *not* statistically independent. Instead, consecutive pairs $\epsilon_t, \epsilon_{t-1}$ are strongly dependent, which will be relevant later. To generate an (new) image $\hat{x}_0$ the reverse diffusion process starts from random noise $x_T \sim \mathcal{N}(0, \mathbf{I})$ which can be iteratively denoised as

$$x_{t-1} = \hat{\mu}_t(x_t) + \sigma_t z_t, \qquad t = T, ..., 1 \quad (3)$$

Here z_t are iid standard normal vectors, and common variance schedulers σ_t can be expressed in the general form

$$\sigma_t = \eta\beta_t\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}$$

where $\eta \in [0, 1]$. In this formulation, $\eta = 0$ corresponds to the deterministic DDIM [42] and $\eta = 1$ to the DDPM scheme [15]. Lastly, in theses cases, we have $\hat{\mu}_t(x_t) =$

$$\sqrt{\bar{\alpha}_{t-1}}\frac{x_t - \sqrt{1-\bar{\alpha}_t}\hat{\epsilon}_\theta(x_t)}{\sqrt{\bar{\alpha}_t}} + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\hat{\epsilon}_\theta(x_t)$$

Here $\hat{\epsilon}_\theta(x_t)$ is an estimate of ϵ_t produced by our neural network DM with learned parameters θ, commonly implemented as a U-Net [35]. For text-to-image generation, the model is conditioned on a text prompt p to produce images faithful to that prompt. The DM is trained to produce the noise estimate $\hat{\epsilon}_\theta(x_t)$ needed for iteratively sampling $\hat{x}_0$ (Eq. 3). For text-conditioned DMs, $\hat{\epsilon}_\theta$ is calculated using specific guidance techniques.

Most DMs rely on classifier-free guidance [16], a conditioning method using a purely generative diffusion model, eliminating the need for an additional classifier. During training, the text conditioning c_p is randomly dropped with a fixed probability, resulting in a joint model for unconditional and conditional objectives. During inference, the score estimates for the ϵ-prediction are adjusted so that:

$$\hat{\epsilon}_\theta(x_t, c_p) := \hat{\epsilon}_\theta(x_t) + s_g(\hat{\epsilon}_\theta(x_t, c_p) - \hat{\epsilon}_\theta(x_t)) \quad (4)$$

with guidance scale s_g and $\hat{\epsilon}_\theta$ defining the noise estimate with parameters θ. Intuitively, the unconditioned ϵ-prediction is pushed in the direction of the conditioned one, with s_g determining the extent of the adjustment.

3.2. LEDITS++

With the fundamentals established, the methodology of LEDITS++ can now be broken down into three components: (1) efficient image inversion, (2) versatile textual editing, and (3) semantic grounding of image changes.

Component 1: Perfect Inversion. Utilizing text-to-image models for editing real images requires conditioning the generation on the input image. Recent works have largely relied on inverting the sampling process to identify x_T that will be denoised to the input image x_0 [28, 30]. Inverting the DDPM scheduler is generally preferred over DDIM inversion since the former can be achieved in fewer timesteps and with no reconstruction error [17].

However, there exist more efficient schemes than DDPM for sampling DMs that greatly reduce the required number of steps and consequently DM evaluations. We here propose a more efficient inversion method by deriving the desired inversion properties for such a scheme. As demonstrated by Song *et al.*[43], DDPM can be viewed as a first-order stochastic differential equation (SDE) solver when formulating the reverse diffusion process as an SDE. This SDE can be solved more efficiently—in fewer steps—using a higher-order differential equation solver, here *dpm-solver++* [23]. The reverse diffusion process from Eq. 3 for the second-order sde-dpm-solver++ can be written as

$$x_{t-1} = \hat{\mu}_t(x_t, x_{t+1}) + \sigma_t z_t, \qquad t = T, ..., 1 \quad (5)$$

where now

$$\sigma_t = \sqrt{1-\bar{\alpha}_{t-1}}\sqrt{1-e^{-2h_{t-1}}}$$

and higher-order $\hat{\mu}_t$ depends now on x from two timesteps, x_t *and* x_{t+1}. Such that $\hat{\mu}_t(x_t, x_{t+1}) =$

$$\frac{\sqrt{1-\bar{\alpha}_{t-1}}}{\sqrt{1-\bar{\alpha}_t}}e^{-h_{t-1}}x_t + \sqrt{\bar{\alpha}_{t-1}}\left(1-e^{-2h_{t-1}}\right)\hat{\epsilon}_\theta(x_t)$$
$$+ 0.5\sqrt{\bar{\alpha}_{t-1}}\left(1-e^{-2h_{t-1}}\right)\frac{-h_t}{h_{t-1}}\left(\hat{\epsilon}_\theta(x_{t+1}) - \hat{\epsilon}_\theta(x_t)\right)$$

with

$$h_t = \frac{\ln(\sqrt{\bar{\alpha}_t})}{\ln(\sqrt{1-\bar{\alpha}_t})} - \frac{\ln(\sqrt{\bar{\alpha}_{t+1}})}{\ln(\sqrt{1-\bar{\alpha}_{t+1}})}$$

For the detailed derivation of the solver and proof of faster convergence, we refer the reader to the relevant literature [22, 23]. Based on the above, we now devise our inversion process. Given an input image x_0 we construct an auxiliary reconstruction sequence of noise images $x_1, ..., x_T$ as

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\tilde{\epsilon}_t \quad (6)$$

where $\tilde{\epsilon}_t \sim \mathcal{N}(0, \mathbf{I})$. Contrary to Eq. 2, the $\tilde{\epsilon}_t$ are now statistically *independent*, which is a desirable property for image editing [17]. Lastly, the respective z_t for the inversion can be derived from Eq. 5 as

$$z_t = \frac{x_{t-1} - \hat{\mu}_t(x_t, x_{t+1})}{\sigma_t}, \qquad t = T, .., 1 \tag{7}$$

with $\hat{\mu}$ and σ_t as defined above. We base our implementation on the multistep variant of sde-dpm-solver++, which only requires one evaluation of the DM at each diffusion timestep by reusing the estimates from the previous step. The number of timesteps can be reduced further by stopping the inversion at an intermediate step $t < T$ and starting the generation at that step. Empirically, we observed that $t \in [0.9T, 0.8T]$ usually produces edits of the same fidelity as $t = T$, supporting observations in previous work [17, 26] that earlier timesteps are less relevant to the edit.

Component 2: Textual Editing. After creating our reconstruction sequence $x_1, ..., x_T$ and calculating the respective z_t, we now edit the image by manipulating the noise estimate $\hat{\epsilon}_\theta$ based on a set of edit instructions $\{e_i\}_{i \in I}$. We devise a dedicated guidance term for each concept e_i based on conditioned and unconditioned estimates. Let us formally define LEDITS++'s guidance by starting with a single editing prompt e. We compute

$$\hat{\epsilon}_\theta(x_t, c_e) := \hat{\epsilon}_\theta(x_t) + \gamma(x_t, c_e) \tag{8}$$

with guidance term γ. Consequently, setting $\gamma = 0$ will reconstruct the input image x_0. We construct γ to push the unconditioned score estimate $\hat{\epsilon}_\theta(x_t)$—i.e. the input image reconstruction—away from/towards the edit concept estimate $\hat{\epsilon}_\theta(x_t, c_e)$, depending on the guidance direction:

$$\gamma(x_t, c_e) = \phi(\psi; s_e, \lambda)\psi(x_t, c_e) \tag{9}$$

where ϕ applies an edit guidance scale s_e element-wise, and ψ depends on the edit direction: $\psi(x_t, c_e) =$

$$\begin{cases} \hat{\epsilon}_\theta(x_t, c_e) - \hat{\epsilon}_\theta(x_t) & \text{if pos. guidance} \\ -\big(\hat{\epsilon}_\theta(x_t, c_e) - \hat{\epsilon}_\theta(x_t)\big) & \text{if neg. guidance} \end{cases} \tag{10}$$

Thus, changing the guidance direction is reflected by the direction between $\hat{\epsilon}_\theta(x_t,c_e)$ and $\hat{\epsilon}_\theta(x_t)$. The term ϕ identifies those dimensions of the image and respective $\hat{\epsilon}_\theta$ that are relevant to a prompt e. Consequently, ϕ returns 0 for all irrelevant dimensions and a scaling factor s_e for the others. We describe the construction of ϕ in detail below. Larger s_e will increase the effect of the edit, and $\lambda \in (0, 1)$ reflects the percentage of the pixels selected as relevant by ϕ. Notably, for a single concept e and uniform $\phi = s_e$, Eq. 8 generalizes to the classifier-free guidance term in Eq. 4.

For multiple e_i, we calculate γ_t^i as described above, with each defining their own hyperparameter values λ^i, s_e^i. The sum of all γ_t^i results in

$$\hat{\gamma}_t(x_t, c_{e_i}) = \sum\nolimits_{i \in I} \gamma_t^i(x_t, c_{e_i}) \tag{11}$$

Component 3: Semantic Grounding. The masking term ϕ (Eq. 9) is the intersection (pointwise product) of binary masks M^1 and M^2 combined with scaling factor s_e:

$$\phi(\psi; s_{e_i}, \lambda) = s_{e_i} M_i^1 M_i^2 \tag{12}$$

where M_i^1 is a binary mask generated from the U-Net's cross-attention layers and M_i^2 is a binary mask derived from the noise estimate. Intuitively, each mask is an importance map, where M_i^1 is more strongly grounded than M_i^2, but of significantly coarser granularity. Therefore, the intersection of the two yields a mask both focused on relevant image regions and of fine granularity. With LEDITS++, we empirically demonstrate that these maps can also capture regions of an image relevant to an editing concept that is not already present. Specifically for multiple edits, calculating a dedicated mask for each edit prompt ensures that the corresponding guidance terms remain largely isolated, limiting interference between them.

Formally, at each time step t, a U-Net forward pass with editing prompt e_i is performed to generate cross-attention maps for each token of the editing prompt. All cross-attention maps of the smallest resolution (e.g., 16×16 for SD) are averaged over all heads and layers, and the resulting maps are summed over all editing tokens, resulting in a single map $A_t^{e_i} \in R^{16\times16}$. Importantly, we utilize the same U-Net evaluation $\hat{\epsilon}_\theta(x_t, c_e)$ already performed in Eq. 10 to produce M^1 with minimal overhead. Each map $A_t^{e_i}$ is up-sampled to match the size of x_t. Cross-attention mask M^1 is derived by calculating the λ-th percentile of up-sampled $A_t^{e_i}$ and

$$M_i^1 = \begin{cases} 1 & \text{if } |A_t^{e_i}| \geq \eta_\lambda(|A_t^{e_i}|) \\ 0 & \text{else} \end{cases} \tag{13}$$

where $\eta_\lambda(|\cdot|)$ is the λ-th percentile. By definition, M^1 only selects image regions that correlate strongly with the editing prompt, and λ determines the size of this selected region.

The fine-grained mask M^2 is calculated based on the guidance vector ψ of noise estimates derived in Eq. 10. The difference between unconditioned and conditioned $\hat{\epsilon}_\theta$, generally captures outlines and object edges of x_t. Consequently, the largest absolute values of ψ provide meaningful segmentation information of fine granularity for M^2

$$M^2 = \begin{cases} 1 & \text{if } |\psi| \geq \eta_\lambda(|\psi|) \\ 0 & \text{else} \end{cases} \tag{14}$$

In general, threshold λ should correspond to the performed edit. Changes affecting the entire image, such as style transfer, should choose smaller λ ($\rightarrow 0$), whereas edits targeting

Figure 2. Comparison of image editing methods. (top) LEDITS++ is the only method to restrict edits to the tree leaves and position of the car. (bottom) Ours is the only approach faithfully executing all three edits and keeping changes minimal. (Best viewed in color)

Method	Reconstruction Error (RMSE) ↓	Execution Time (s) ↓	Variation/ Sampling	Semantic Grounding	Multi-Editing
SDEdit [26]	0.81 ± 0.07	2.10 ± 0.02	✓	✗	✗
Imagic [18]	0.58 ± 0.12	349.98 ± 0.45	✓	✗	✗
Vanilla DDIM Inversion	0.22 ± 0.10	37.23 ± 0.04	✗	✗	✗
Pix2Pix-Zero [30]	0.20 ± 0.09	56.78 ± 0.14	(✓)	✓	✗
DiffEdit [9]	0.13 ± 0.03	27.65 ± 0.03	✓	✓	✗
Edit-friendly DDPM [17]	**0.00**	10.36 ± 0.05	✓	✗	✗
LEDITS++ (Ours)	**0.00**	**1.78** ± 0.03	✓	✓	✓

Table 1. Comparing key properties for diffusion-based image editing techniques, with LEDITS++ offering clear methodological benefits. Due to LEDITS++'s efficient perfect inversion, it is the fastest and error-free method. At the same time, its methodology is the only enabling versatility in terms of *variation*, *semantic grounding*, and *multi-editing*. Subscript numbers indicate standard deviation.

specific objects or regions should use λ proportional to the region's prominence in the image.

4. Properties of LEDITS++

With the fundamentals of LEDITS++ established, we next showcase its unique properties and capabilities.

Efficiency. First off, LEDITS++ offers substantial performance improvements over other image editing methods. In Tab. 1, we provide a qualitative runtime comparison, with all methods being implemented for Stable Diffusion (SD) 1.5 [34]. As a parameter-free approach, LEDITS++ does not require any computationally expensive fine-tuning or optimization. Consequently, LEDITS++ is orders of magnitude faster than methods like Imagic [18] or Pix2Pix-Zero [30]. Further, we only need to invert the same number of diffusion steps used at inference, which results in significant runtime improvements over the standard DDIM inversion (21x). In addition to efficient inversion, we use a recent, fast scheduler that generally requires fewer total steps, further boosting performance. This way, LEDITS++ is six times faster than recent DDPM inversion [17] and on par with fast but poor-quality SDEdit [26].

Versatility. In addition to its efficiency, LEDITS++ remains versatile, enabling sheer limitless editing possibilities. In Fig. 1, we showcase a broad range of edit types. LEDITS++ facilitates fine-grained edits (adding/removing glasses) and holistic changes such as style transfer (painting/sketch). Furthermore, object removal and replacement facilitate even more image editing tasks. Importantly, the overall image composition is preserved in all cases. To our knowledge, LEDITS++ is the only diffusion-based image editing method inherently supporting multiple edits in isolation, which allows for more complex image manipulation. Fig. 2 highlights LEDITS++ benefits over previous methods. Our method produces the highest edit fidelity and is the only approach capable of faithfully executing multiple, simultaneous instructions. Moreover, LEDITS++ also makes the least changes to unrelated objects and the overall background and composition of the image.

Lastly, the editing versatility benefits from the stochastic nature of the perfect but non-deterministic inversion. LEDITS++ provides meaningful image variations by resampling $\tilde{\epsilon}_t$. Additionally, the visual expression of each concept in the edited image scales monotonically with s_e, and the direction and magnitude of each concept can be varied freely. We present examples of both features in App. B.

Precision. Furthermore, LEDITS++'s methodology keeps edits concise and avoids unnecessary deviations from the input image (Fig. 2). First, the perfect inversion will reconstruct the exact input image if no edit is applied (cf.

Figure 3. Exemplary edit performed with LEDITS++ in only 25 diffusion steps with SD1.5. We apply a complex, compounded edit and ground each to a semantically reasonable image region.

Sec. 3.2). Consequently, we already improve on faithfulness to the input image even before applying any edits. This benefit over other methods is highlighted by the reconstruction error in Tab. 1. Second, implicit masking will semantically ground each edit to relevant image regions. This is specifically important for editing multiple concepts at the same time. While other methods only utilize one prompt for all instructions, LEDITS++ isolates edits from each other (Eq. 11). Thus, we get dedicated masks for each concept as shown in Fig. 3. This design ensures that each instruction (e.g., red mask for 'cherry blossom') will be only applied where necessary. Subsequently, we provide further evidence for the efficacy of LEDITS++'s masking approach.

5. Semantically Grounded Image Editing

Cross-attention maps of DMs have been used extensively to ground regions of interest during image generation semantically [7, 8, 14, 30]. Nonetheless, these have not been combined with noise-based masks so far and thus lack fine granularity. Hence, we empirically evaluate the quality of implicit masks, i.e., attention maps M^1 and noise maps M^2 (Eq. 13 and 14) in the LEDITS++ setup. We use a broad segmentation task for common objects as a proxy to measure the performance of implicit masks in identifying relevant image areas from edit instructions. Specifically, we utilize segmentation masks from the COCO panoptic segmentation challenge [19]. For each unique object in an image, we retrieve the masks M^1, M^2, and their intersection per diffusion step. We use the (semantic) class label (e.g. 'person' or 'TV') as editing concept e. We consider masks at each of 50 total diffusion steps without actually editing the input image. Furthermore, we approximate mask threshold λ based on the relative size of an object's bounding box.

Concise masks with LEDITS++. Fig. 4 shows implicit masking as a reliable means to identify relevant image regions. Importantly, the intersection of both cross-attention masks M^1 and noise maps M^2 clearly outperforms each separate mask. The overall performance is even similar to a dedicated CLIPSeg model [24], despite LEDITS++ masks being implicitly calculated at inference with only minimal overhead. At the same time, LEDITS++'s masking is superior to DiffEdit's [3]. Consequently, our method's intersection of cross-attention masks and noise maps provides strong semantic grounding while being efficient during image manipulation to ensure precise editing.

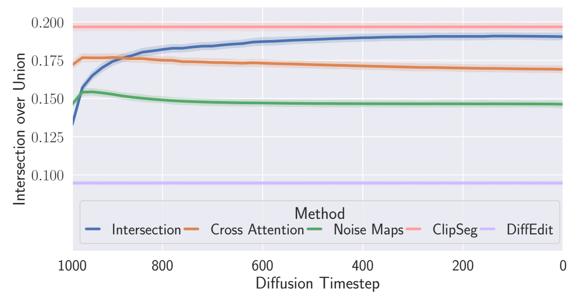

Figure 4. Semantic segmentation quality of LEDITS++. We show the intersection over union (higher is better) for COCO panoptic segmentation. The intersection masks outperform each by a clear margin, close to the CLIPSeg reference. (Best viewed in color)

6. Image Editing Evaluation

Let us now compare LEDITS++ to current SOTA methods for image manipulation on two benchmarks.

6.1. Editing Multiple Concepts

First, we investigate the complex task of performing multiple edits simultaneously. We rely on a well-established setup for semantic image manipulation [5] to evaluate multi-conditioned attribute manipulation in facial images. In our experiment, we consider 100 images from the CelebA dataset [21]. For each image, we simultaneously edit three attributes out of a set of five, leading to ten total combinations of edit concepts. Further, we perform each edit across ten different seeds, resulting in 10,000 evaluated images for each method and hyperparameter setting, over 1M images in total. As measures for comparison, we employ CLIP and LPIPS scores. CLIP measures the text-to-image similarity of the edit instruction to the edited image, and LPIPS measures the image-to-image similarity of the real to the edited image. This way, we assess the trade-off between the versatility of edits (CLIP) and the precision of those manipulations (LPIPS). We implement all methods based on SD1.5 and provide more details in App. C.

LEDITS++ outperforms competing methods. Fig. 5 shows the resulting CLIP vs. LPIPS plots for all methods. The top left corner represents the ideal editing method with maximum edit alignment without deviating from the initial image. Generally, one can observe a natural trade-off between versatility and precision for all methods, i.e., higher image-to-text alignment comes at the expense of lower similarity to the original image. LEDITS++ is closest to the ideal region and thus clearly outperforms the other methods. In particular, the outputs remain close to the original image (low LPIPS scores), thanks to the precise implicit masking.

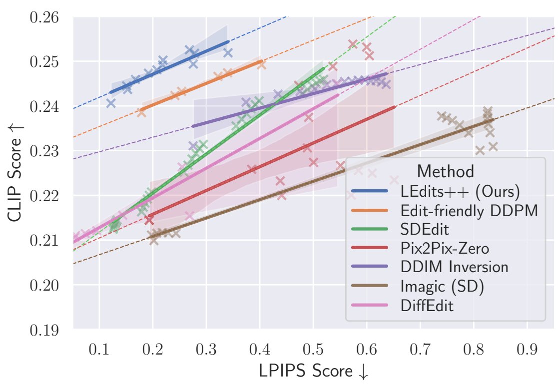

Figure 5. Comparison of instruction-alignment vs. image similarity trade-off for different editing methods. Results were reported for simultaneous manipulation of three facial attributes on CelebA. We plot CLIP scores (higher is better) of the target attributes against LPIPS similarity (lower is better). LEDITS++ clearly outperforms all competing methods. (Best viewed in color)

At the same time, it faithfully performs the edits (high CLIP scores) due to the dedicated, isolated editing for each concept. The depicted scores reflect our qualitative inspections for Pix2Pix-Zero and Imagic on such complex manipulations (cf. Fig. 2). We observed that these methods often break—either failing to perform all three edits and/or drastically altering the input image. Only edit-friendly DDPM [17] and LEDITS++ reliably achieve the maximum average CLIP score of over 0.25. This value seems to represent an upper bound according to our manual investigations, as each attribute is edited correctly for all input images, and higher scores are not observed. Despite being computationally very efficient, LEDITS++ faithfully executes each edit instruction while keeping the changes to the input low, highlighting the method's versatility and precision.

6.2. TEdBench(++)

Next, we investigate the versatility of LEDITS++'s editing capabilities by running the Textual Editing Benchmark (TEdBench [18]), a collection of 100 input images paired with textual edit instructions. However, we observed a variety of inconsistencies in TEdBench and a lack of relevant editing tasks. Therefore, we propose TEdBench++ (Fig. 6a and App. D), a more challenging revised benchmark now containing 120 entries in total.[3] We addressed misspellings and rephrased ambiguous and inconclusive instructions. In addition to resolving these issues, we added instructions targeting challenging types of image manipulations previously not included in TEdBench: multi-conditioning, object/concept removal, style transfer, and complex replacements (Fig. 6a). We provide more details in App. D.

We compare LEDITS++ on TEdBench(++) to one of the strongest editing methods, Imagic [18] with Imagen [37].

[3] https://huggingface.co/datasets/AIML-TUDA/TEdBench_plusplus

	TEdBench		TEdBench++	
	SR ↑	LPIPS ↓	SR ↑	LPIPS ↓
Imagic w/ SD1.5	0.55	0.56	0.58	0.57
LEDITS++ w/ SD1.5	**0.75**	**0.28**	**0.79**	**0.30**
Imagic w/ Imagen [18]	0.83	0.59	—	—
LEDITS++ w/ SD-XL	**0.84**	**0.33**	**0.87**	**0.34**

Table 2. Success rate (SR) and LPIPS scores on the original TedBench [18] and our revised version (TEdBench++). We compare Imagic to LEDITS++ based on different DMs and find the latter to outperform on both metrics and benchmarks.

Since both are not publicly available, we can only compare to this specific combination of DM and editing method using Kawar *et al.'s* [18] curated outputs for TEdBench. Additionally, we, therefore, cannot combine LEDITS++ with Imagen[37] and instead use a similarly advanced diffusion model, SD-XL [32]. However, to not only compete for the best fidelity outputs but focus the evaluation on methodological differences—not the pre-trained DM—we also compare both methods implemented with SD1.5. We provide further details in App. C.

LEDITS++ edits images reliably. We first asked users to assess the overall success of edits, i.e., if an edit instruction was faithfully realized for a given input image. The results in Tab. 2 show that LEDITS++ outperforms Imagic on TEdBench despite a greatly reduced runtime (Tab. 1). The difference is even stronger when comparing both methods on the same pre-trained DM, i.e., SD1.5. The high success rate on TEdBench++ (87%) and the examples shown in Fig. 1 and 6a once again highlight LEDITS++'s versatility. Overall, our proposed method can reliably perform a diverse set of editing instructions for real images.

High-quality edits with LEDITS++. While investigating both methods' performance we observed a substantial difference in edit quality. The examples in Fig. 6b particularly highlight the discrepancy in compositional robustness and object coherence. Hence, we also assessed both methods' editing quality on TEdBench(++). We focus on samples where both methods performed a successful edit, i.e., were labeled as successful by users. We show the perceptual similarity (LPIPS) to the input image in Tab. 2. One can observe that the LPIPS scores for LEDITS++ are much lower than for Imagic, empirically supporting the qualitative examples in Fig. 6b. When manually inspecting the generated images, we often found Imagic to generate a completely *new* image based on the edit instruction, entirely disregarding the input image (cf. App. Fig. 11a).

7. Discussion

Let us now discuss open research questions and limitations.

Model Dependency. While LEDITS++ achieves impressive results on a large variety of image manipulation tasks, there are external factors to consider. Since the

(a) Novel challenging examples of TEdBench++ and LEDITS++ applied, showcasing the versatility of supported edits.

(b) Qualitative comparison of LEDITS++ and Imagic on TEdBench, clearly highlighting the performance improvement.

Figure 6. Benchmark examples for LEDITS++ and Imagic on TEdBench(++). (Best viewed in color)

method is architecture-agnostic, it can be easily used with any DM. At the same time, the general editing quality strongly depends on the overall capabilities of the underlying pre-trained DM. Naturally, more capable models will also enable better edits. But, at times, specific editing instructions may fail because the used DM does not have a decent representation of the targeted concept to begin with. One example is the model failing to edit a giraffe to be *sitting* since the underlying DM generally fails to generate this pose (cf. App. F). This effect can also clearly be seen in Tab. 2, with the editing success rate of a method varying strongly between DMs. Although the same image editing method is employed (LEDITS++), the more capable SD-XL variant outperforms the weaker SD1.5 model. Nonetheless, this means that the architecture-agnostic LEDITS++ will benefit from increasingly powerful DMs.

Coherence Trade-offs. Next to the benefits of LEDITS++'s semantic grounding, there are also downsides to this approach. Overall, implicit masking limits changes to relevant portions of the image and achieves strong coherence with the original image composition. Yet, the object and its identity within the masked area may change based on various factors. Generic prompts, like "a standing cat" (cf. App. F), do not contain detailed information about this specific object ("cat"). Thus, an edit with this prompt does not guarantee to preserve object identity, particularly for strong hyperparameters. We observed that fine-tuning approaches like Imagic make the opposite trade-off, better preserving the object identity while changing the background and image composition substantially (cf. App. F). A potential remedy for a loss in object coherence with LEDITS++, is more descriptive edit prompting, e.g. using textual inversion [12].

Lastly, the automatically-inferred implicit masks allow for easy use of LEDITS++ without users tediously providing masks. Nonetheless, user intentions are diverse and cannot always be automatically inferred. Sometimes, individual user masks provide better control over the editing process. Such user masks can be easily integrated into LEDITS++ (cf. App. F), wherefore we encourage future research in this promising direction.

Societal Impact. LEDITS++ is an easy-to-use image editing technique that lowers the barrier for users and puts them in control for fruitful human-machine collaboration. Yet, the underlying text-to-image models offer both promise and peril, as highlighted by prior research [4, 11]. The (societal) biases within these models will also impact image editing applications [11]. Moreover, image manipulation can also be used adversarially to generate inappropriate [39] or fake content. Hence, we advocate for a cautious deployment of generative models together with image editing methods.

8. Conclusion

We introduced LEDITS++, an efficient yet versatile and precise method for textual image manipulation with diffusion models. It facilitates the editing of complex concepts in real images. Our approach requires no finetuning nor optimization, can be computed extremely efficiently, and is architecture agnostic. At the same time, it perfectly reconstructs an input image and uses implicit masking to limit changes to relevant image regions, thus editing precisely. Our large experimental evaluation confirms the efficiency, versatility, and precision of LEDITS++ and its components, as well as its benefits over several related methods.

Acknowledgements We gratefully acknowledge support from the BMBF (Grant No 01IS22091). This work benefited from the ICT-48 Network of AI Research Excellence Center "TAILOR" (EU Horizon 2020, GA No 952215), the Hessian research priority program LOEWE within the project WhiteBox, the HMWK cluster projects "Adaptive Mind" and "Third Wave of AI", and from the NHR4CES.

References

[1] Omri Avrahami, Kfir Aberman, Ohad Fried, Daniel Cohen-Or, and Dani Lischinski. Break-a-scene: Extracting multiple concepts from a single image. *SIGGRAPH Asia*, 2023. 16

[2] Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *arXiv:2206.02779*, 2022. 2

[3] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, Tero Karras, and Ming-Yu Liu. eDiff-I: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv:2211.01324*, 2022. 2, 6

[4] Federico Bianchi, Pratyusha Kalluri, Esin Durmus, Faisal Ladhak, Myra Cheng, Debora Nozza, Tatsunori Hashimoto, Dan Jurafsky, James Zou, and Aylin Caliskan. Easily Accessible Text-to-Image Generation Amplifies Demographic Stereotypes at Large Scale. In *Proceedings of ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, 2023. 8, 11

[5] Manuel Brack, Felix Friedrich, Dominik Hintersdorf, Lukas Struppek, Patrick Schramowski, and Kristian Kersting. Sega: Instructing text-to-image models using semantic guidance. In *Proceedings of the Advances in Neural Information Processing Systems: Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2023. 2, 6

[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 (CVPR)*, 2023. 1, 2

[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. *arXiv:2304.08465*, 2023. 2, 6

[8] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Trans. Graph.*, 42, 2023. 2, 6

[9] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023. 5

[10] Zoe De Simone, Angie Boggust, Arvind Satyanarayan, and Ashia Wilson. What is a Fair Diffusion Model? Designing Generative Text-To-Image Models to Incorporate Various Worldviews. *arXiv preprint arXiv:2309.09944*, 2023. 11

[11] Felix Friedrich, Manuel Brack, Lukas Struppek, Dominik Hintersdorf, Patrick Schramowski, Sasha Luccioni, and Kristian Kersting. Fair Diffusion: Instructing Text-to-Image Generation Models on Fairness. *arXiv preprint arXiv:2302.10893*, 2023. 8, 11

[12] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 8, 16

[13] Rohit Gandikota, Joanna Materzynska, Jaden Fiotto-Kaufman, and David Bau. Erasing concepts from diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 11

[14] 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 (ICLR)*, 2023. 2, 6

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Proceedings of the Advances in Neural Information Processing Systems: Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 2, 3

[16] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv:2207.12598*, 2022. 3

[17] Inbar Huberman-Spiegelglas, Vladimir Kulikov, and Tomer Michaeli. An edit friendly ddpm noise space: Inversion and manipulations. *arXiv preprint arXiv:2304.06140*, 2023. 1, 2, 3, 4, 5, 7, 12, 13

[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 (CVPR)*, 2023. 1, 2, 5, 7, 12, 13, 14

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014. 6

[20] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B. Tenenbaum. Compositional visual generation with composable diffusion models. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2022. 2

[21] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, December 2015. 6

[22] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In *NeurIPS*, 2022. 2, 3

[23] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint arXiv:2211.01095*, 2022. 2, 3

[24] Timo Lüddecke and Alexander Ecker. Image segmentation using text and image prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 6

[25] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv:2310.04378*, 2023. 2

[26] 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 (ICLR)*, 2022. 1, 2, 4, 5, 12, 13

[27] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik P.

Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[28] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2, 3, 12

[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 (ICML)*. PMLR, 2022. 2, 11

[30] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *SIGGRAPH*, 2023. 1, 2, 3, 5, 6, 12, 13

[31] Pablo Pernias, Dominic Rampas, Mats L. Richter, Christopher J. Pal, and Marc Aubreville. Wuerstchen: An efficient architecture for large-scale text-to-image diffusion models. *arXiv:2306.00637*, 2023. 2

[32] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv:2307.01952*, 2023. 7

[33] 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. 2, 11

[34] 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 (CVPR)*, 2022. 2, 5

[35] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241, 2015. 3

[36] Harrison Rosenberg, Shimaa Ahmed, Guruprasad V. Ramesh, Ramya Korlakai Vinayak, and Kassem Fawaz. Unbiased Face Synthesis With Diffusion Models: Are We There Yet? *arXiv preprint arXiv:2309.07277*, 2023. 11

[37] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv:2205.11487*, 2022. 2, 7, 11

[38] Dvir Samuel, Rami Ben-Ari, Simon Raviv, Nir Darshan, and Gal Chechik. Generating images of rare concepts using pre-trained diffusion models. *arXiv:2304.14530*, 2023. 17

[39] Patrick Schramowski, Manuel Brack, Björn Deiseroth, and Kristian Kersting. Safe latent diffusion: Mitigating inappropriate degeneration in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 8, 11

[40] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. Laion-5b: An open large-scale dataset for training next generation image-text models. In *Proceedings of NeurIPS Datasets and Benchmarks*, 2022. 11

[41] Norbert Schwarz. How the questions shape the answers. *American Psychologist*, 1999. 15

[42] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 2, 3

[43] 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 (ICLR)*, 2021. 3

[44] Dani Valevski, Matan Kalman, Yossi Matias, and Yaniv Leviathan. Unitune: Text-driven image editing by fine tuning an image generation model on a single image. *arXiv:2210.09477*, 2022. 1, 2

[45] Qiucheng Wu, Yujian Liu, Handong Zhao, Ajinkya Kale, Trung Bui, Tong Yu, Zhe Lin, Yang Zhang, and Shiyu Chang. Uncovering the disentanglement capability in text-to-image diffusion models. *arXiv:1212.08698*, 2022. 2

[46] Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. Unipc: A unified predictor-corrector framework for fast sampling of diffusion models. *arXiv:2302.04867*, 2023. 2