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

Manuel Brack[1,2*†] **Linoy Tsaban**[3*] **Katharina Kornmeier**[2*] **Apolinário Passos**[3*]
Felix Friedrich[2,4*] **Patrick Schramowski**[1,2,4,5] **Kristian Kersting**[1,2,4,6]
[1]German Research Center for Artificial Intelligence (DFKI),
[2]Computer Science Department, TU Darmstadt, [3]HuggingFace, [4]Hessian.AI,
[5]LAION, [6]Centre for Cognitive Science, TU Darmstadt
brack@cs.tu-darmstadt.de

Abstract

Text-to-image diffusion models have recently received a lot of interest for their astonishing ability to produce high-fidelity images from text only. Subsequent research efforts are aiming to exploit the capabilities of these models and leverage them for intuitive, textual image editing. However, existing methods often require time-consuming fine-tuning and lack native support for performing multiple edits simultaneously. To address these issues, we introduce LEDITS++, an efficient yet versatile technique for image editing using text-to-image models. LEDITS++ requires no tuning nor optimization, runs in a few diffusion steps, natively supports multiple simultaneous edits, inherently limits changes to relevant image regions, and is architecture agnostic.

1 Introduction

Recently, text-to-image models have gained increasing popularity for their ability to generate high-quality images from text alone. A growing body of research is dedicated to utilizing these models for intuitive, textual image editing. Unfortunately, current methods often require costly fine-tuning or optimization to ensure reasonable reconstructions of the input image [12, 16, 3] or trade-in efficiency improvements for unnecessarily strong image alterations [11, 15]. Moreover, none of the existing approaches offer native support for performing multiple arbitrary edits simultaneously and in isolation.

To ease textual image editing, we present LEDITS++[3], a novel method for efficient and versatile image editing using text-to-image diffusion models. Firstly, LEDITS++ sets itself apart as a parameter-free solution requiring no fine-tuning nor any optimization. We derive characteristics of an edit-friendly noise space with a perfect input reconstruction, which were previously proposed for the DDPM sampling scheme [11], for a significantly faster multistep stochastic differential-equation (SDE) solver [14]. This novel invertibility of the DPM-solver++ facilitates editing with LEDITS++ in as little as 20 total diffusion steps for inversion and inference combined.

Moreover, LEDITS++ places a strong emphasis on semantic grounding to enhance the visual and contextual coherence of the edits. This ensures that changes are limited to the relevant regions in the image, preserving the original image's fidelity as much as possible. LEDITS++ also provides users with the flexibility to combine multiple edits seamlessly, opening up new creative possibilities for intricate image manipulations. Finally, the approach is architecture-agnostic and compatible with any diffusion model, whether latent or pixel-based.

[*] Equal contribution

[†] Work conducted as Research Intern at Adobe

[3] LEDITS++ stands for *Limitless Edits* with sde-dpm-solver++.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

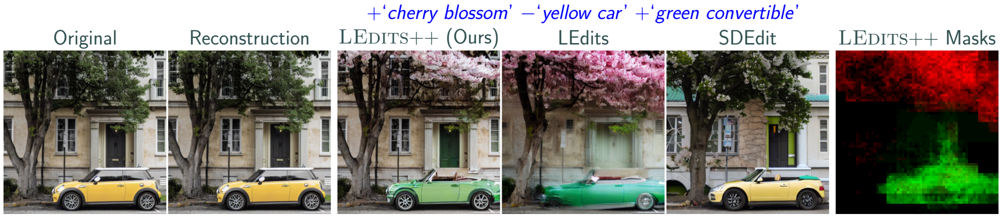


Figure 1: Exemplary edit performed using LEdits++ in only 25 diffusion steps with Stable Diffusion 1.5. The method perfectly reconstructs the input image, applies a complex, compounded edit, and grounds each change to a semantically reasonable image region. LEdits++ significantly outperforms previous methods in edit fidelity and faithfulness to the input image.

We make the implementation of LEdits++ as well as an interactive demo publicly available[4] to facilitate easy accessibility and experimentation.

2 LEdits++: Efficient and Versatile Textual Image Editing

The methodology of LEdits++ can be broken down into three components: (1) efficient image inversion, (2) versatile textual editing, and (3) semantic grounding of image changes. More in-depth details and mathematical derivations of each component can be found in App. A.

Component 1: Image Inversion. Utilizing text-to-image models for editing real images requires conditioning the generation of the input image. Recently, Huberman-Spiegelglas *et al.* proposed an inversion technique [11] for the DDPM sampler [9] that addresses key limitations of the prevalent DDIM inversion [16]. Specifically, their inversion perfectly reconstructs the input image, only needs to be calculated for the number of timesteps used at inference, and requires no optimization for error correction. We here demonstrate the same properties for the significantly faster SDE version of the multistep dpm-solver++ [14]. Our results indicate that LEdits++ with this second-order sde-dpm-solver++ produces high-quality edits in 10-30 timesteps, depending on the complexity of the changes. Fig 1 illustrates the reconstruction and the benefits of the improved scheduler. These improvements are highlighted by the comparison with LEdits whose DDIM scheduler is incapable of producing a high-fidelity image with this low number of diffusion steps.

Component 2: Textual Editing. Recently, Brack *et al.* proposed Semantic Guidance (SEGA) [2] to control the image generation of diffusion models with arbitrary, textual concepts. We employ a similar technique for editing real images with LEdits++. More precisely, after inverting the input image as described above, we calculate a dedicated guidance vector for each of the editing prompts at all diffusion steps. The formulation of this guidance term ensures that concepts are largely isolated and consequently do not interfere with each other. Additionally, this design choice lets the user control the edit strength for each applied concept individually. As shown in Fig 1, these properties allow for versatile and complex edits using intuitive instructions. The direct comparison with SDEdit [15] further illustrates the benefits of dedicated guidance terms for each concept, as SDEdit is incapable of faithfully executing multiple edit instructions from a single prompt.

Component 3: Semantic Grounding. Lastly, a capable image editing method should ensure a balance between faithfully executing the edit instruction and minimal deviation from the input image. LEdits++'s properties of perfect reconstruction and sophisticated control over dedicated guidance terms for each edit already contribute to achieving this balance. Additionally, we limit any change to the specific image regions relevant to each edit. To this end, LEdits++ demonstrates that attention maps for the edit prompts can be extracted from the U-Net's cross-attention layers to restrict edits semantically. We intersect these coarse regions of interest with fine-granular implicit masks obtained from the model's noise estimates. Fig. 1 shows semantic grounding of the edits '*cherry blossom*' and '*green convertible*' to respectively relevant image regions, although both concepts were not present in the original image. In comparison, LEdits and SDEdit both make substantial changes to irrelevant parts of the image.

[4] code & demo at `https://huggingface.co/editing-images/ledtisplusplus`

Societal Impact

With LEDITS++, we aim to provide an easy-to-use image editing framework. It lowers the barrier of entry for experienced artists and novices alike, allowing them to unlock the full potential of generative AI in the pursuit of creative expression. Moreover, it puts the user in control for fruitful human-machine collaboration. Crucially, current text-to-image models [18, 17, 20] hold the potential to wield a profound influence on society. When applied in creative and design domains, their dual use offers both promise and peril, as highlighted by prior research [1, 6]. The models are trained on large amounts of data from the web [22], granting them the inherent capacity to generate content that may contravene societal norms, including the creation of inappropriate material like pornography [21]. More alarmingly, the inadvertent generation of inappropriate content is precipitated by spurious correlations within these models. Harmless prompts can lead to the creation of decidedly objectionable content [1]. A prime example of this phenomenon lies in the correlation between specific phrases and the perpetuation of stereotypes, such as the connection between mentions of ethnicity and economic status. For example, an increase of the concept '*black person*' may inadvertently amplify the appearance of the concept '*poverty*.' Conversely, methods like LEDITS++ also possess the potential to mitigate bias and inappropriateness, a prospect highlighted by prior reserach [6, 5], e.g. through dataset augmentation [19]. Furthermore, established strategies offer means to mitigate the generation of inappropriate content [21, 7] that could deployed in tandem with LEDITS++. In summary, we advocate for a cautious approach to the utilization of these models, recognizing both the risks and promises they bring to the realm of AI-powered image editing.

References

[1] 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.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

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

[11] Inbar Huberman-Spiegelglas, Vladimir Kulikov, and Tomer Michaeli. An edit friendly ddpm noise space: Inversion and manipulations. *arXiv preprint arXiv:2304.06140*, 2023.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] 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.

[18] 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.

[19] 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.

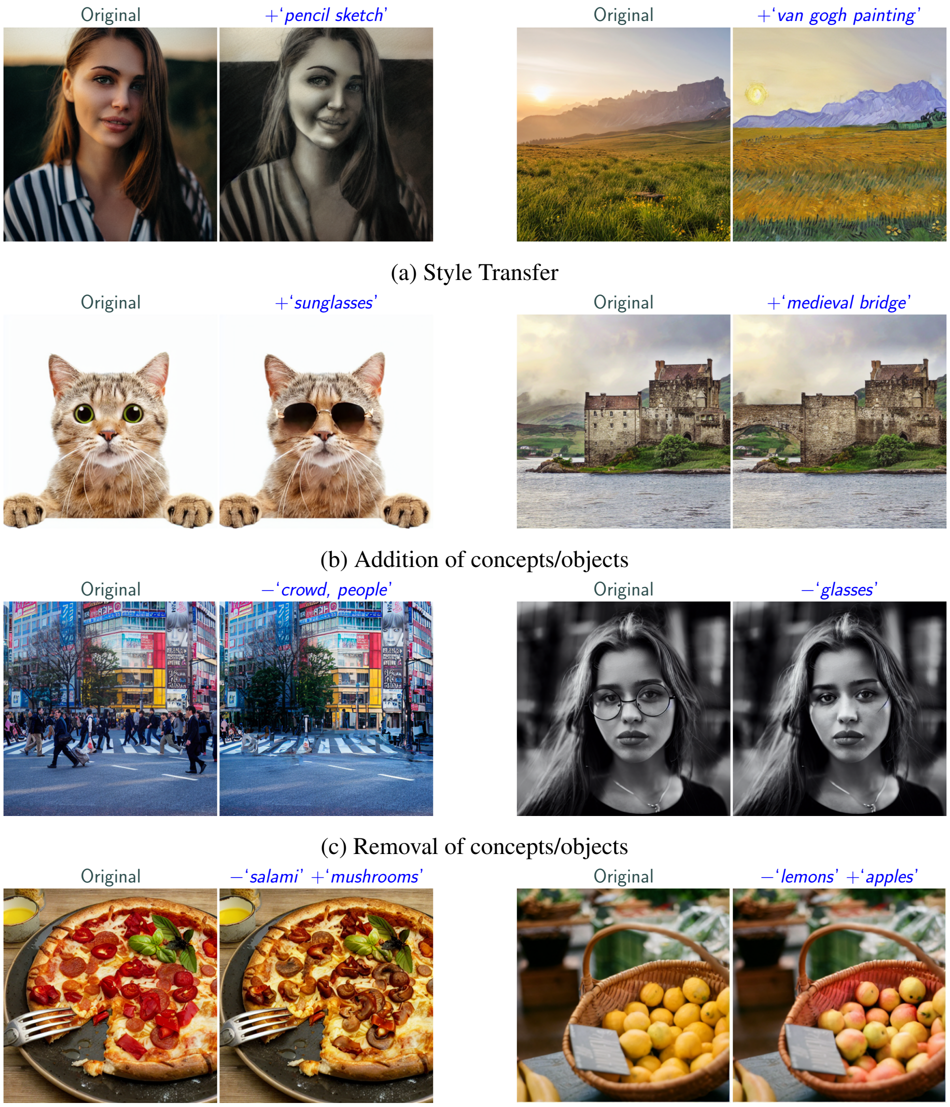
[20] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language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

[21] 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.

[22] 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.

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

[24] 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.



(a) Style Transfer

(b) Addition of concepts/objects

(c) Removal of concepts/objects

(d) Complex replacements of objects

Figure 2: Further examples highlighting the versatility of LEDITS++ on 4 different editing tasks. All examples were generated with the implementation based on Stable Diffusion v1.5. The ‘Orginal’ image shows the respective VAE reconstruction of the input.

A Detailed Methodology

Subsequently, we derive the inner workings and intuition of LEDITS++ in detail.

A.1 Guided Diffusion Models

Let us first define some background on diffusion models (DM) in general. 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.

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

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

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

where $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \Pi_{s=1}^{t}\alpha_s$ and $\epsilon_\mathbf{t} \sim \mathcal{N}(0, \mathbf{I})$. 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

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

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 [23] and $\eta = 1$ to the DDPM scheme [9]. Lastly, $\hat{\mu}_t(x_t)$ corresponds to

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

Here $\hat{\epsilon}_\theta(x_t)$ is an estimate of ϵ_t produced by our neural network DM with learned parameters θ . For text-to-image generation, the model is conditioned on a text prompt p to produce images faithful to that prompt. The training objective of a DM $\hat{x}_\theta$ can be written as

$$\mathbb{E}_{\mathbf{x},\mathbf{c}_p,\epsilon,t}\left[w_t||\hat{\mathbf{x}}_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t, \mathbf{c}_p) - x_0||_2^2\right] \tag{4}$$

where $(\mathbf{x}, \mathbf{c}_p)$ is conditioned on text prompt p, and w_t influences the image fidelity depending on t. Consequently, 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 some guidance technique.

Most DMs rely on classifier-free guidance [10], a conditioning method using a purely generative diffusion model, eliminating the need for an additional pre-trained classifier. During training, the text conditioning $\mathbf{c}_p$ is randomly dropped with a fixed probability, resulting in a joint model for unconditional and conditional objectives. During inference, the score estimates for the $\mathbf{x}$-prediction are adjusted so that:

$$\hat{\epsilon}_\theta(x_t, \mathbf{c}_p) := \hat{\epsilon}_\theta(x_t) + s_g(\hat{\epsilon}_\theta(x_t, \mathbf{c}_p) - \hat{\epsilon}_\theta(x_t)) \tag{5}$$

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.

A.2 Inversion

Utilizing text-to-image models for editing real images requires conditioning the generation on the input image. One of the first emerging approaches simply added noise to the image (Eq. 2) for an intermediate step in the diffusion process, e.g., $t = 0.5T$ [15]. Subsequently, the remaining diffusion steps can be performed using this intermediate x_t and an edit prompt p (Eq. 3). However, the resulting image is likely to diverge significantly from the input since it is partially regenerated. Performing prior tuning of the model to reproduce the input image is feasible [12] but not computationally efficient. Consequently, recent works have largely relied on inverting the deterministic DDIM sampling process to identify a x_T that will be denoised to the input image x_0. However, faithful reconstructions are only obtained in the limit of small steps, thus requiring large numbers of inversion steps. And even when using T inversion steps, a small error will incur at each timestep, often accumulating into meaningful deviations, specifically when using classifier-free guidance.

Recently, Huberman-Spiegelglas *et al.* proposed an inversion technique [11] for the DDPM sampler [9] that addresses key limitations of the prevalent DDIM inversion [16]. In addition to the edit-friendly properties of this noise space demonstrated by Huberman-Spiegelglas *et al.*, the method only requires the same amount of timesteps during inversion, that will be used in the generative process. A perfect reconstruction of x_0 is achieved for any number of timesteps.

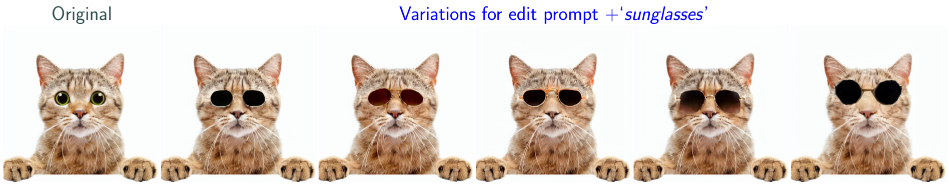


Figure 3: LEDITS++ inherently provides meaningful variations of an edit instruction.

However, there exist more efficient schemes than DDPM for sampling diffusion models 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.*[24], DDPM can viewed as a first-order stochastic differential equation (SDE) solver when formulating the reverse diffusion process as an SDE. In fact, this SDE can be solved more efficiently—i.e. in fewer steps—using a higher-order differential equation solver, here *dpm-solver++* [14]. The reverse diffusion process from Eq. 3 for the second-order sde-dpm-solver++ can be written as

$$x_{t-1} = \hat{\mu}_t(x_t, x_{t+1}) + \sigma_t z_t, \qquad t = T, ..., 1 \tag{6}$$

where now

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

and $\hat{\mu}_t$ not only depends on x_t, but also x_{t+1}

$$\begin{aligned}\hat{\mu}_t(x_t, x_{t+1}) = &\frac{\sqrt{1-\bar{\alpha}_{t-1}}}{\sqrt{1-\bar{\alpha}_t}}e^{-h_{t-1}}x_t + \sqrt{\bar{\alpha}_{t-1}}(1-e^{-2h_{t-1}})\hat{\epsilon}_\theta(x_t)\\ &+ 0.5\sqrt{\bar{\alpha}_{t-1}}(1-e^{-2h_{t-1}})\frac{h_t}{h_{t-1}}(\hat{\epsilon}_\theta(x_t) - \hat{\epsilon}_\theta(x_{t+1}))\end{aligned}$$

with

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

For the detailed derivation of the solver and proof of faster convergence, we refer the reader to the relevant literature [14, 13]. Now we can devise our inversion process. Given an input image x_0 we construct an auxiliary reconstruction sequence of noise images $x_1, ..., x_T$ as

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\tilde{\epsilon}_t \tag{7}$$

where $\tilde{\epsilon}_t \sim \mathcal{N}(0, \mathbf{I})$. The key difference to Eq. 2 is that $\tilde{\epsilon}_t$ are statistically independent, which is a desirable property for image editing [11]. Lastly, the respective z_t for the inversion can be derived from Eq. 6 as

$$z_t = \frac{x_{t-1} - \hat{\mu}_t(x_t, x_{t+1})}{\sigma_t}, \qquad t = T, .., 1 \tag{8}$$

Importantly, 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 required number of timesteps can be reduced further by stopping the inversion at an intermediate step $t < T$ and beginning the generative process at that step. Empirically, we observed that $t \in [0.9T, 0.8T]$ usually produces edits of the same fidelity as $t = T$, indicating that earlier timesteps are less relevant to the edit. Furthermore, the stochastic nature of our non-deterministic inversion easily provides meaningful variations of an edit by resampling $\tilde{\epsilon}_t$. As shown in Fig. 3 this provides even more versatility to the user.

A.3 Editing

After constructing 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. In line with the mathematical derivations from prior research [2, 10], we devise a dedicated guidance term for each concept e based on conditioned and unconditioned estimates. Let us formally define LEDITS++'s guidance by starting with a single editing prompt. We compute

$$\hat{\epsilon}_\theta(x_t, \mathbf{c}_e) := \hat{\epsilon}_\theta(x_t) + \gamma(x_t, \mathbf{c}_e) \tag{9}$$

with guidance term γ. Consequently, setting $\gamma = 0$ will reconstruct the input image x_0. Intuitively 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 $\hat{\epsilon}_\theta(x_t, \mathbf{c}_e)$, depending on the guidance direction.

$$\gamma(x_t, \mathbf{c}_e) = \phi(\psi; s_e, \lambda)\psi(x_t, \mathbf{c}_e) \tag{10}$$

where ϕ applies an edit guidance scale s_e element-wise, and ψ depends on the edit direction:

$$\psi(x_t, \mathbf{c}_e) = \begin{cases} \hat{\epsilon}_\theta(x_t, \mathbf{c}_e) - \hat{\epsilon}_\theta(x_t) & \text{if pos. guidance} \\ -\big(\hat{\epsilon}_\theta(x_t, \mathbf{c}_e) - \hat{\epsilon}_\theta(x_t)\big) & \text{if neg. guidance} \end{cases} \tag{11}$$

Thus, changing the guidance direction is reflected by the direction between $\hat{\epsilon}_\theta(x_t,\mathbf{c}_e)$ and $\hat{\epsilon}_\theta(x_t)$. The term ϕ identifies those dimensions of the image and respective $\hat{\epsilon}$ 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 App A.4. 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. 9 generalizes to the classifier-free guidance term in Eq. 5.

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 weighted sum of all γ_t^i results in

$$\hat{\gamma}_t(x_t, \mathbf{c}_p; \mathbf{e}) = \sum\nolimits_{i \in I} \gamma_t^i(x_t, \mathbf{c}_{e_i}) \tag{12}$$

A.4 Masking

The masking term ϕ can be derived as the intersection (pointwise product) of binary masks M_1 and M_2 combined with scaling factor s_e

$$\phi(\psi; s_{e_i}, \lambda) = s_{e_i} M_i^1 M_i^2 \tag{13}$$

where M_i^1 is a binary mask automatically generated from the U-Net's cross-attention layers and M_i^2 is a binary mask derived from the edit-conditioned noise estimate. Intuitively, 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. While attribution maps derived from the cross-attention layers of the U-Net have been utilized in prior work [4, 8], LEDITS++ demonstrates that they are also able to capture regions of an image relevant to an editing concept that is not already present. The calculation of a dedicated mask for each edit prompt ensures that the corresponding guidance terms remain largely isolated, limiting interference between multiple edits.

Formally, at each time step t, a U-Net 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., 16x16 for SD) are averaged over all heads and layers, and the resulting maps are summed over all editing tokens, resulting in a single map $A_t^{e_i} \in R^{16x16}$. Importantly, we utilize the same U-Net evaluation $\hat{\epsilon}_\theta(x_t, \mathbf{c}_e)$ already performed in Eq. 11. 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

$$M_1 = \begin{cases} 1 & \text{if } |A_t^{e_i}| \geq \eta_\lambda(|A_t^{e_i}|) \\ 0 & \text{else} \end{cases} \tag{14}$$

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 the size of the selected image region is determined by λ.

Similar to prior work [2], the fine-grained mask M_2 is calculated based on the guidance vector ψ of noise estimates calculated in Eq. 11. 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

$$M_2 = \begin{cases} 1 & \text{if } |\psi| \geq \eta_\lambda(|\psi|) \\ 0 & \text{else} \end{cases} \tag{15}$$

In general, the threshold λ should be chosen to reflect the type of performed edit. Changes affecting the entire image, such as style transfers, should choose smaller $\lambda \rightarrow 0$, whereas edits targeting specific objects or regions should use λ proportional to the regions prominence in the image.