

Null-text Inversion for Editing Real Images using Guided Diffusion Models

Ron Mokady*^{† 1,2}, Amir Hertz*^{† 1,2}, Kfir Aberman¹, Yael Pritch¹, and Daniel Cohen-Or^{† 1,2}

¹Google Research, ²The Blavatnik School of Computer Science, Tel Aviv University

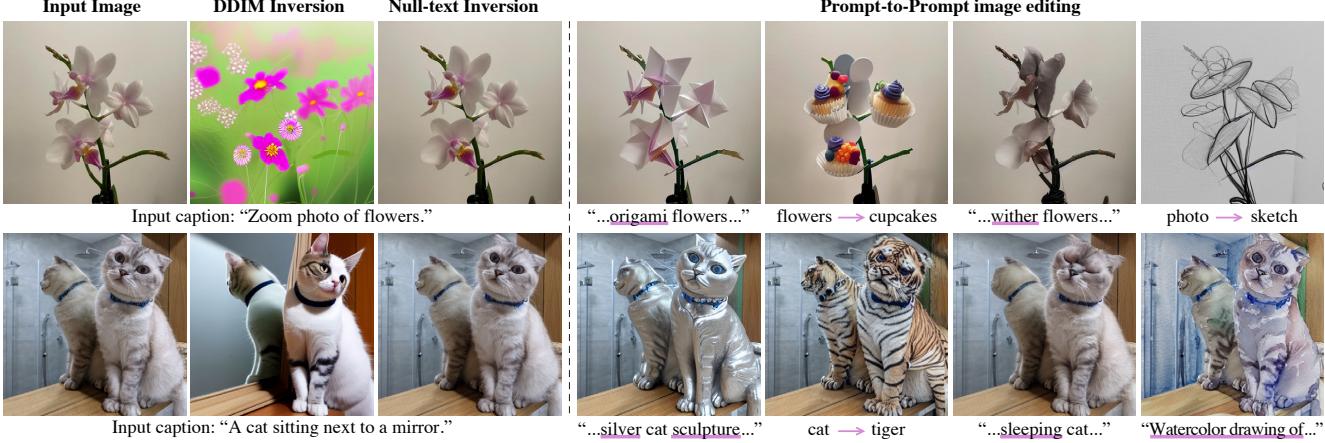


Figure 1. **Null-text inversion for real image editing.** Our method takes as input a real image (leftmost column) and an associated caption. The image is inverted with a DDIM diffusion model to yield a diffusion trajectory (second column to the left). Once inverted, we use the initial trajectory as a pivot for null-text optimization that accurately reconstructs the input image (third column to the left). Then, we can edit the inverted image by modifying only the input caption using the editing technique of Prompt-to-Prompt [16].

Abstract

Recent text-guided diffusion models provide powerful image generation capabilities. Currently, a massive effort is given to enable the modification of these images using text only as means to offer intuitive and versatile editing. To edit a real image using these state-of-the-art tools, one must first invert the image with a meaningful text prompt into the pretrained model’s domain. In this paper, we introduce an accurate inversion technique and thus facilitate an intuitive text-based modification of the image. Our proposed inversion consists of two novel key components: (i) Pivotal inversion for diffusion models. While current methods aim at mapping random noise samples to a single input image, we use a single pivotal noise vector for each timestamp and optimize around it. We demonstrate that a direct inversion is inadequate on its own, but does provide a good anchor for our optimization. (ii) null-text optimization, where we only modify the unconditional textual embedding that is used for classifier-free guidance, rather than the input text embedding. This allows for keeping both the model weights and the conditional embedding intact and hence enables applying prompt-based editing while avoiding the cumbersome tuning of the model’s weights. Our null-text inversion, based on the publicly available Stable Diffusion model, is extensively evaluated on a variety of images and prompt editing, showing high-fidelity editing of real images.

1. Introduction

The progress in image synthesis using text-guided diffusion models has attracted much attention due to their exceptional realism and diversity. Large-scale models [27, 30, 32] have ignited the imagination of multitudes of users, enabling image generation with unprecedented creative freedom. Naturally, this has initiated ongoing research efforts, investigating how to harness these powerful models for image editing. Most recently, intuitive text-based editing was demonstrated over synthesized images, allowing the user to easily manipulate an image using text only [16].

However, text-guided editing of a real image with these state-of-the-art tools requires *inverting* the given image and textual prompt. That is, finding an initial noise vector that produces the input image when fed with the prompt into the diffusion process while preserving the editing capabilities of the model. The inversion process has recently drawn considerable attention for GANs [7, 41], but has not yet been fully addressed for text-guided diffusion models. Although an effective DDIM inversion [13, 35] scheme was suggested for unconditional diffusion models, it is found lacking for text-guided diffusion models when classifier-free guidance [18], which is necessary for meaningful editing, is applied.

In this paper, we introduce an effective inversion scheme, achieving near-perfect reconstruction, while retaining the rich text-guided editing capabilities of the original model (see Fig. 1). Our approach is built upon the analysis of two key aspects of guided diffusion models: classifier-free guidance and DDIM inversion.

* Equal contribution.

[†]Performed this work while working at Google.

In the widely used classifier-free guidance, in each diffusion step, the prediction is performed twice: once unconditionally and once with the text condition. These predictions are then extrapolated to amplify the effect of the text guidance. While all works concentrate on the conditional prediction, we recognize the substantial effect induced by the unconditional part. Hence, we optimize the embedding used in the unconditional part in order to invert the input image and prompt. We refer to it as *null-text optimization*, as we replace the embedding of the empty text string with our optimized embedding.

DDIM Inversion consists of performing DDIM sampling in reverse order. Although a slight error is introduced in each step, this works well in the unconditional case. However, in practice, it breaks for text-guided synthesis, since classifier-free guidance magnifies its accumulated error. We observe that it can still offer a promising starting point for the inversion. Inspired by GAN literature, we use the sequence of noised latent codes, obtained from an initial DDIM inversion, as pivot [29]. We then perform our optimization around this pivot to yield an improved and more accurate inversion. We refer to this highly efficient optimization as *Diffusion Pivotal Inversion*, which stands in contrast to existing works that aim to map all possible noise vectors to a single image.

To the best of our knowledge, our approach is the first to enable the text editing technique of Prompt-to-Prompt [16] on real images. Moreover, unlike recent approaches [19, 39], we do not tune the model weights, thus avoiding damaging the prior of the trained model and duplicating the entire model for each image. Throughout comprehensive ablation study and comparisons, we demonstrate the contribution of our key components to achieving a high-fidelity reconstruction of the given real image, while allowing meaningful and intuitive editing abilities. For our code, built upon the publicly available Stable Diffusion model, please visit our project page <https://null-text-inversion.github.io/>.

2. Related Work

Large-scale diffusion models, such as Imagen [32], DALL-E 2 [27], and Stable Diffusion [30], have recently raised the bar for the task of generating images conditioned on plain text, known as text-to-image synthesis. Exploiting the powerful architecture of diffusion models [17, 17, 30, 34–36], these models can generate practically any image by simply feeding a corresponding text, and so have changed the landscape of artistic applications.

However, synthesizing very specific or personal objects which are not widespread in the training data has been challenging. This requires an *inversion* process that given input images would enable regenerating the depicted object using a text-guided diffusion model. Inversion has been studied extensively for GANs [7, 10, 22, 41, 42, 44], ranging

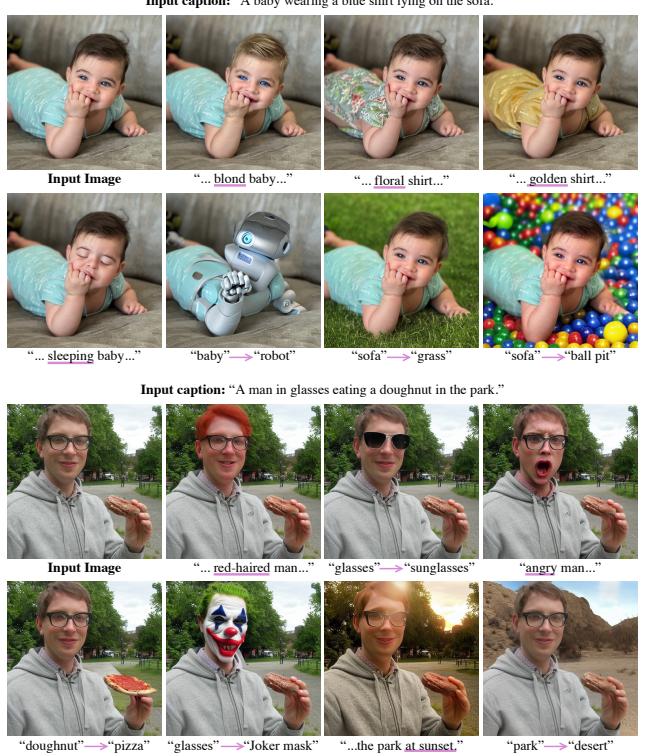


Figure 2. Real image editing using our method. We first apply a single null-text inversion over the real input image, achieving high-fidelity reconstruction. Then, various Prompt-to-Prompt text-based editing operations are applied. As can be seen, our inversion scheme provides high fidelity while retaining high editability. See additional examples in Appendix C (Fig. 10).

from latent-based optimization [1, 2] and encoders [28, 38] to feature space encoders [40] and fine-tuning of the model [3, 29]. Motivated by this, Gal et al. [15] suggest a textual inversion scheme for diffusion models that enables regenerating a user-provided concept out of 3 – 5 images. Concurrently, Ruiz et al. [31] tackled the same task with model-tuning. However, these works struggle to edit a given real image while accurately reproducing the unedited parts.

Naturally, recent works have attempted to adapt text-guided diffusion models to the fundamental challenge of single-image editing, aiming to exploit their rich and diverse semantic knowledge. Meng et al. [23] add noise to the input image and then perform a text-guided denoising process from a predefined step. Yet, they struggle to accurately preserve the input image details. To overcome this, several works [4, 5, 25] assume that the user provides a mask to restrict the region in which the changes are applied, achieving both meaningful editing and background preservation.

However, requiring that users provide a precise mask is burdensome. Furthermore, masking the image content removes important information, which is mostly ignored in the inpainting process. While some text-only editing approaches are bound to global editing [12, 20, 21], Bar-Tal

et al. [6] propose a text-based localized editing technique without using any mask. Their technique allows high-quality texture editing, but not modifying complex structures, since only CLIP [26] is employed as guidance instead of a generative diffusion model.

Hertz et al. [16] suggest an intuitive editing technique, called Prompt-to-Prompt, of manipulating local or global details by modifying only the text prompt when using text-guided diffusion models. By injecting internal cross-attention maps, they preserve the spatial layout and geometry which enable the regeneration of an image while modifying it through prompt editing. Still, without an inversion technique, their approach is limited to synthesized images. Sheynin et al. [33] suggest training the model for local editing without the inversion requirement, but their expressiveness and quality are inferior compared to current large-scale diffusion models. Concurrent to our work, DiffEdit [9] uses DDIM inversion for image editing, but avoids the emerged distortion by automatically producing a mask that allows background preservation.

Also concurrent, Imagic [19] and UniTune [39] have demonstrated impressive editing results using the powerful Imagen model [32]. Yet, they both require the restrictive fine-tuning of the model. Moreover, Imagic requires a new tuning for each editing, while UniTune involves a parameter search for each image. Our method enables us to apply the text-only intuitive editing of Prompt-to-Prompt [16] on real images. We do not require any fine-tuning and provide highly-quality local and global modifications using the publicly available Stable Diffusion [30] model.

3. Method

Let \mathcal{I} be a real image. Our goal is to edit \mathcal{I} , using only text guidance, to get an edited image \mathcal{I}^* . We use the setting defined by Prompt-to-Prompt [16], where the editing is guided by source prompt \mathcal{P} and edited prompt \mathcal{P}^* . This requires the user to provide a source prompt. Yet, we found that automatically producing the source prompt using an off-the-shelf captioning model [24] works well (see Sec. 4). For example, see Fig. 2, given an image and a source prompt "A baby wearing...", we replace the baby with a robot by providing the edited prompt "A robot wearing...".

Such editing operations first require inverting \mathcal{I} to the model's output domain. Namely, the main challenge is faithfully reconstructing \mathcal{I} by feeding the source prompt \mathcal{P} to the model, while still retaining the intuitive text-based editing abilities.

Our approach is based on two main observations. First, DDIM inversion produces unsatisfying reconstruction when classifier-free guidance is applied, but provides a good starting point for the optimization, enabling us to efficiently achieve high-fidelity inversion. Second, optimizing the unconditional null embedding, which is used in classifier-free guidance, allows an accurate reconstruction while avoid-

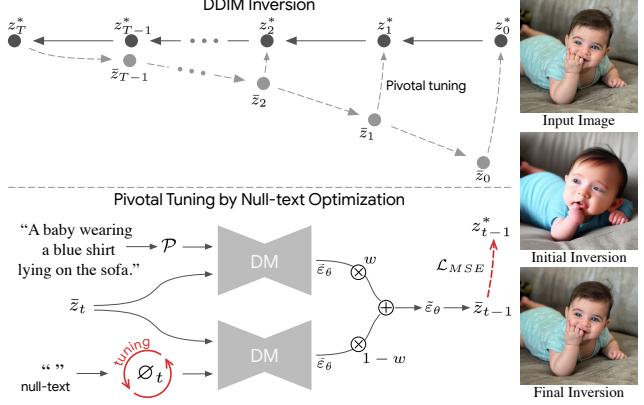


Figure 3. Null-text Inversion overview. Top: pivotal inversion. We first apply an initial DDIM inversion on the input image which estimates a diffusion trajectory $\{z_t^*\}_{0}^T$. Starting the diffusion process from the last latent z_T^* results in unsatisfying reconstruction as the latent codes become farther away from the original trajectory. We use the initial trajectory as a pivot for our optimization which brings the diffusion backward trajectory $\{\bar{z}_t\}_1^T$ closer to the original image encoding z_0^* . Bottom: null-text optimization for timestamp t . Recall that classifier-free guidance consists of performing the prediction ϵ_θ twice – using text condition embedding and unconditionally using null-text embedding \emptyset (bottom-left). Then, these are extrapolated with guidance scale w (middle). We optimize only the unconditional embeddings \emptyset_t by employing a reconstruction MSE loss (in red) between the predicated latent code z_{t-1} to the pivot z_{t-1}^* .

ing the tuning of the model and the conditional embedding. Thereby preserving the desired editing capabilities.

Next, we provide a short background, followed by a detailed description of our approach in Sec. 3.2 and Sec. 3.3. A general overview is provided in Fig. 3.

3.1. Background and Preliminaries

Text-guided diffusion models aim to map a random noise vector z_t and textual condition \mathcal{P} to an output image z_0 , which corresponds to the given conditioning prompt. In order to perform sequential denoising, the network ϵ_θ is trained to predict artificial noise, following the objective:

$$\min_{\theta} E_{z_0, \varepsilon \sim N(0, I), t \sim \text{Uniform}(1, T)} \|\varepsilon - \epsilon_\theta(z_t, t, \mathcal{C})\|_2^2. \quad (1)$$

Note that $\mathcal{C} = \psi(\mathcal{P})$ is the embedding of the text condition and z_t is a noised sample, where noise is added to the sampled data z_0 according to timestamp t . At inference, given a noise vector z_T , The noise is gradually removed by sequentially predicting it using our trained network for T steps.

Since we aim to accurately reconstruct a given real image, we employ the deterministic DDIM sampling [35]:

$$z_{t-1} = \sqrt{\frac{\alpha_{t-1}}{\alpha_t}} z_t + \left(\sqrt{\frac{1}{\alpha_{t-1}} - 1} - \sqrt{\frac{1}{\alpha_t} - 1} \right) \cdot \epsilon_\theta(z_t, t, \mathcal{C}).$$

For the definition of α_t and additional details, please refer to Appendix E. Diffusion models often operate in the image pixel space where z_0 is a sample of a real image. In

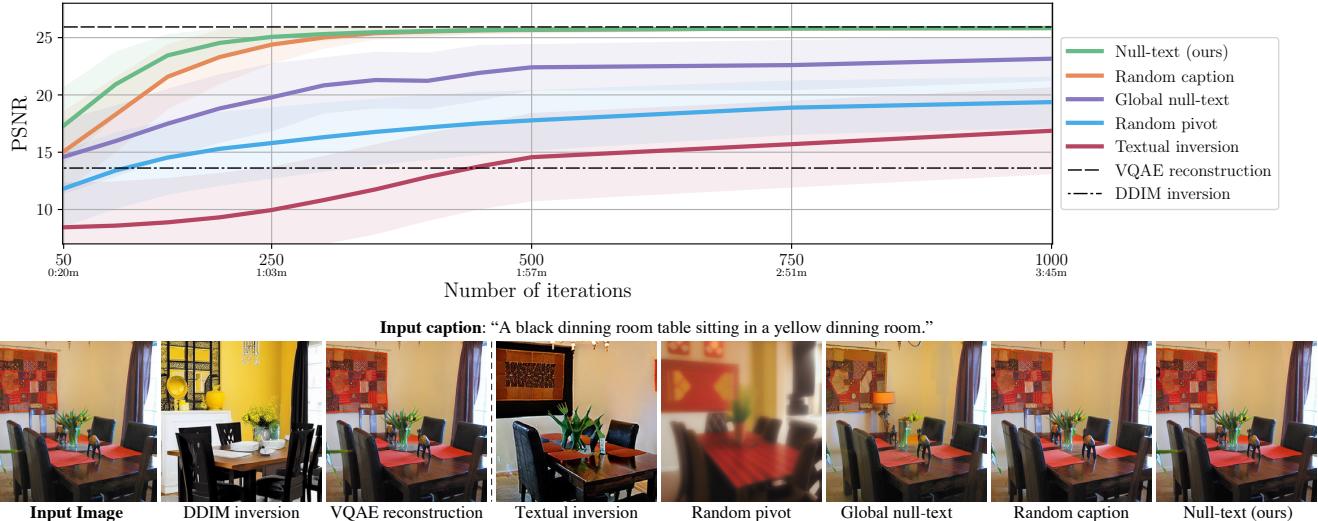


Figure 4. **Ablation Study.** Top: we compare the performance of our full algorithm (green line) to different variations, evaluating the reconstruction quality by measuring the PSNR score as a function of number optimization iterations and running time in minutes. Bottom: we visually show the inversion results after 200 iterations of our full algorithm (on right) compared to other baselines. Results for all iterations are shown in Appendix B (Figs. 13 and 14).

our case, we use the popular and publicly available Stable Diffusion model [30] where the diffusion forward process is applied on a latent image encoding $z_0 = E(x_0)$ and an image decoder is employed at the end of the diffusion backward process $x_0 = D(z_0)$.

Classifier-free guidance. One of the key challenges in text-guided generation is the amplification of the effect induced by the conditioned text. To this end, Ho et al. [18] have presented the classifier-free guidance technique, where the prediction is also performed unconditionally, which is then extrapolated with the conditioned prediction. More formally, let $\emptyset = \psi("")$ be the *embedding* of a null text and let w be the guidance scale parameter, then the classifier-free guidance prediction is defined by:

$$\tilde{\varepsilon}_\theta(z_t, t, \mathcal{C}, \emptyset) = w \cdot \varepsilon_\theta(z_t, t, \mathcal{C}) + (1 - w) \cdot \varepsilon_\theta(z_t, t, \emptyset).$$

E.g., $w = 7.5$ is the default parameter for Stable Diffusion.

DDIM inversion. A simple inversion technique was suggested for the DDIM sampling [13, 35], based on the assumption that the ODE process can be reversed in the limit of small steps:

$$z_{t+1} = \sqrt{\frac{\alpha_{t+1}}{\alpha_t}} z_t + \left(\sqrt{\frac{1}{\alpha_{t+1}} - 1} - \sqrt{\frac{1}{\alpha_t} - 1} \right) \cdot \varepsilon_\theta(z_t, t, \mathcal{C}).$$

In other words, the diffusion process is performed in the reverse direction, that is $z_0 \rightarrow z_T$ instead of $z_T \rightarrow z_0$, where z_0 is set to be the encoding of the given real image.

3.2. Pivotal Inversion

Recent inversion works use random noise vectors for each iteration of their optimization, aiming at mapping every noise vector to a single image. We observe that this

is inefficient as inference requires only a single noise vector. Instead, inspired by GAN literature [29], we seek to perform a more “local” optimization, ideally using only a single noise vector. In particular, we aim to perform our optimization around a pivotal noise vector which is a good approximation and thus allows a more efficient inversion.

We start by studying the DDIM inversion. In practice, a slight error is incorporated in every step. For unconditional diffusion models, the accumulated error is negligible and the DDIM inversion succeeds. However, recall that meaningful editing using the Stable Diffusion model [30] requires applying classifier-free guidance with a large guidance scale $w > 1$. We observe that such a guidance scale amplifies the accumulated error. Therefore, performing the DDIM inversion procedure with classifier-free guidance results not only in visual artifacts, but the obtained noise vector might be out of the Gaussian distribution. The latter decreases the editability, i.e., the ability to edit using the particular noise vector.

We do recognize that using DDIM inversion with guidance scale $w = 1$ provides a rough approximation of the original image which is highly editable but far from accurate. More specifically, the reversed DDIM produces a T steps trajectory between the image encoding z_0 to a Gaussian noise vector z_T^* . Again, a large guidance scale is essential for editing. Hence, we focus on feeding z_T^* to the diffusion process with classifier-free guidance ($w > 1$). This results in high editability but inaccurate reconstruction, since the intermediate latent codes deviate from the trajectory, as illustrated in Fig. 3. Analysis of different guidance scale values for the DDIM inversion is provided in Appendix B (Fig. 9).

Motivated by the high editability, we refer to this initial DDIM inversion with $w = 1$ as our *pivot* trajectory and



Figure 5. Fine control editing using attention re-weighting. We can use attention re-weighting to further control the level of dryness over the field or create a denser zebra pattern over the couch.

perform our optimization around it with the standard guidance scale, $w > 1$. That is, our optimization maximizes the similarity to the original image while maintaining our ability to perform meaningful editing. In practice, we execute a separate optimization for each timestamp t in the order of the diffusion process $t = T \rightarrow t = 1$ with the objective of getting close as possible to the initial trajectory z_T^*, \dots, z_0^* :

$$\min \|z_{t-1}^* - z_{t-1}\|_2^2, \quad (2)$$

where z_{t-1} is the intermediate result of the optimization. Since our pivotal DDIM inversion provides a rather good starting point, this optimization is highly efficient compared to using random noise vectors, as demonstrated in Sec. 4.

Note that for every $t < T$, the optimization should start from the endpoint of the previous step ($t + 1$) optimization, otherwise our optimized trajectory would not hold at inference. Therefore, after the optimization of step t , we compute the current noisy latent \bar{z}_t , which is then used in the optimization of the next step to ensure our new trajectory would end near z_0 (see Eq. (3) for more details).

3.3. Null-text optimization

To successfully invert real images into the model’s domain, recent works optimize the textual encoding [15], the network’s weights [31, 39], or both [19]. Fine-tuning the model’s weight for each image involves duplicating the entire model which is highly inefficient in terms of memory consumption. Moreover, unless fine-tuning is applied for each and every edit, it necessarily hurts the learned prior of the model and therefore the semantics of the edits. Direct optimization of the textual embedding results in a non-interpretable representation since the optimized tokens does not necessarily match pre-existing words. Therefore, an intuitive prompt-to-prompt edit becomes more challenging.

Instead, we exploit the key feature of the classifier-free guidance — the result is highly affected by the unconditional prediction. Therefore, we replace the default null-text embedding with an optimized one, referred to as *null-text*

optimization. Namely, for each input image, we optimize only the unconditional embedding \emptyset , initialized with the null-text embedding. The model and the conditional textual embedding are kept unchanged.

This results in high-quality reconstruction while still allowing intuitive editing with Prompt-to-Prompt [16] by simply using the optimized unconditional embedding. Moreover, after a single inversion process, the same unconditional embedding can be used for multiple editing operations over the input image. Since null-text optimization is naturally less expressive than fine-tuning the entire model, it requires the more efficient pivotal inversion scheme.

We refer to optimizing a single unconditional embedding \emptyset as a *Global* null-text optimization. During our experiments, as shown in Fig. 4, we have observed that optimizing a different “null embedding” \emptyset_t for each timestamp t significantly improves the reconstruction quality while this is well suited for our pivotal inversion. And so, we use per-timestamp unconditional embeddings $\{\emptyset_t\}_{t=1}^T$, and initialize \emptyset_t with the embedding of the previous step \emptyset_{t+1} .

Putting the two components together, our full algorithm is presented in algorithm 1. The DDIM inversion with $w = 1$ outputs a sequence of noisy latent codes z_T^*, \dots, z_0^* where $z_0^* = z_0$. We initialize $\bar{z}_T = z_T$, and perform the following optimization with the default guidance scale $w = 7.5$ for the timestamps $t = T, \dots, 1$, each for N iterations:

$$\min_{\emptyset_t} \|z_{t-1}^* - z_{t-1}(\bar{z}_t, \emptyset_t, \mathcal{C})\|_2^2. \quad (3)$$

For simplicity, $z_{t-1}(\bar{z}_t, \emptyset_t, \mathcal{C})$ denotes applying DDIM sampling step using \bar{z}_t , the unconditional embedding \emptyset_t , and the conditional embedding \mathcal{C} . At the end of each step, we update

$$\bar{z}_{t-1} = z_{t-1}(\bar{z}_t, \emptyset_t, \mathcal{C}).$$

We find that early stopping reduces time consumption, resulting in ~ 1 minute using a single A100 GPU.

Finally, we can edit the real input image by using the noise $\bar{z}_T = z_T^*$ and the optimized unconditional embeddings $\{\emptyset_t\}_{t=1}^T$. Please refer to Appendix D for additional implementation details.

4. Ablation Study

In this section, we validate the contribution of our main components, thoroughly analyzing the effectiveness of our design choices by conducting an ablation study. We focus on the fidelity to the input image which is an essential evaluation for image editing. In Sec. 5 we demonstrate that our method performs high-quality and meaningful manipulations.

Experimental setting. Evaluation is provided in Fig. 4. We have used a subset of 100 images and captions pairs, randomly selected from the COCO [8] validation set. We then applied our approach on each image-caption pair using the default Stable Diffusion hyper-parameters for an increasing number of iterations per diffusion step, $N =$

Algorithm 1: Null-text inversion

- 1 Input:** A source prompt embedding $\mathcal{C} = \psi(\mathcal{P})$ and input image \mathcal{I} .
- 2 Output:** Noise vector z_T and optimized embeddings $\{\emptyset_t\}_{t=1}^T$.
- 3** Set guidance scale $w = 1$;
- 4** Compute the intermediate results z_T^*, \dots, z_0^* using DDIM inversion over \mathcal{I} ;
- 5** Set guidance scale $w = 7.5$;
- 6** Initialize $\bar{z}_T \leftarrow z_T^*$, $\emptyset_T \leftarrow \psi("")$;
- 7 for** $t = T, T - 1, \dots, 1$ **do**
- 8 for** $j = 0, \dots, N - 1$ **do**
- 9** $\emptyset_t \leftarrow \emptyset_t - \eta \nabla_{\emptyset} \|z_{t-1}^* - z_{t-1}(\bar{z}_t, \emptyset_t, \mathcal{C})\|_2^2$;
- 10 end**
- 11** Set $\bar{z}_{t-1} \leftarrow z_{t-1}(\bar{z}_t, \emptyset_t, \mathcal{C})$, $\emptyset_{t-1} \leftarrow \emptyset_t$;
- 12 end**
- 13 Return** $\bar{z}_T, \{\emptyset_t\}_{t=1}^T$

1, …, 20 (see algorithm 1). The reconstruction quality was measured in terms of mean PSNR. We now turn to analyze different variations of our algorithm.

DDIM inversion. We mark the DDIM inversion as a lower bound for our algorithm, as it is the starting point of our optimization, producing unsatisfying reconstruction when classifier-free guidance is applied (see Sec. 3.2).

VQAE. For an upper bound, we consider the reconstruction using the VQ auto-encoder [14], denoted *VQAE*, which is used by the Stable Diffusion model. Although the latent code or the VQ-decoder can be further optimized according to the input image, this is out of our scope, since it would be only applicable to this specific model [30] while we aim for a general algorithm. Therefore, our optimization treats its encoding z_0 as ground truth, as the obtained reconstruction is quite accurate in most cases.

Our method. As can be seen in Fig. 4, our method converges to a near-optimal reconstruction with respect to the VQAE upper bound after a total number of 500 iterations ($N = 10$) and even after 250 iterations (~ 1 minute on an A100 GPU) we achieve high-quality inversion.

Random Pivot. We validate the importance of the DDIM initialization by replacing the DDIM-based trajectory with a *single random trajectory* of latent codes, sharing the same starting point z_0 — the input image encoding. In other words, we randomly sample a single Gaussian noise $\sim N(0, I)$ for each image and use it to noise the corresponding encoding z_0 from $t = 1$ to $t = T$ using the diffusion scheduler. As presented in Fig. 4, the DDIM initialization is crucial for fast convergence, since the initial error becomes significantly larger when the pivot is random.

Robustness to different input captions. Since we require an input caption, it is only natural to ask whether

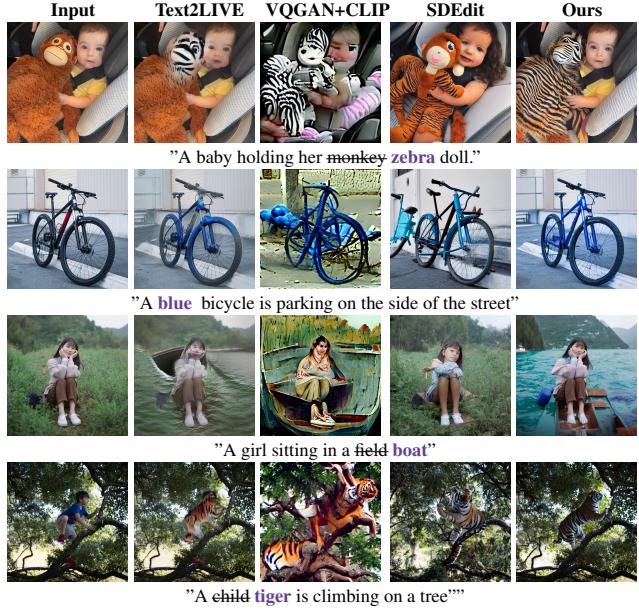


Figure 6. Comparison. Text2LIVE [6] excels at replacing textures locally but struggles to perform more structured editing, such as replacing a kid with a tiger. VQGAN+CLIP [11] obtains inferior realism. SDEdit [23] fails to faithfully reconstruct the original image, resulting in identity drift when humans are involved. Our method achieves realistic editing of both textures and structured objects while retaining high fidelity to the original image. Additional examples provided in Appendix C (Fig. 16).

our method is highly sensitive to the chosen caption. We take this to the extreme by sampling a *random caption* from the dataset for each image. Even with unaligned captions, the optimization converges to an optimal reconstitution with respect to the VQAE. Therefore, we conclude that our inversion is robust to the input caption. Clearly, choosing a random caption is undesired for text-based editing. But, providing any reasonable and editable prompt would work, including using an off-the-shelf captioning model [24,37]. This is illustrated in Appendix B (Fig. 12). We invert an image using multiple captions, demonstrating that the edited parts should be included in the source caption in order to produce semantic attention maps for editing. For example, to edit the print on the shirt, the source caption should include a "shirt with a drawing" or a similar phrase.

Global null-text embedding. We refer to optimizing a single embedding \emptyset for all timestamps as a *Global* embedding. As can be seen, such optimization struggles to converge, since it is less expressive than our final approach, which uses embedding per-timestamp $\{\emptyset_t\}_{t=1}^T$. See additional implementation details in Appendix D.

Textual inversion. We compare our method to *textual inversion*, similar to the proposed method by Gal et al. [15]. We optimize the textual embedding $\mathcal{C} = \psi(\mathcal{P})$ using random noise samples instead of pivotal inversion. That is, we randomly sample a different Gaussian noise for each



Figure 7. **Comparison to mask-based methods.** As can be seen, mask-based methods do not require inversion as the region outside the mask is kept. However, unlike our approach, such methods often struggle to preserve details that are found inside the masked region. For example, basket size is not preserved.

optimization step and obtain z_t by adding it to z_0 according to the diffusion scheduler. Intuitively, this objective aims to map all noise vectors to a single image, in contrast to our pivotal tuning inversion which focuses on a single trajectory of noisy vectors. The optimization objective is then defined:

$$\min_{\mathcal{C}} E_{z_0, \varepsilon \sim N(0, I), t} \|\varepsilon - \varepsilon_{\theta}(z_t, t, \mathcal{C})\|_2^2. \quad (4)$$

Note that Gal et al. [15] have attempted to regenerate a specific object rather than achieve an accurate inversion. As presented in Fig. 4, the convergence is much slower than ours and results in poor reconstruction quality.

Textual inversion with a pivot. We observe that employing our pivotal inversion with the mentioned textual inversion improves the reconstruction quality significantly, results in a comparable reconstruction to ours. This further demonstrates the power of performing the optimization using a pivot. However, we do observe that editability is reduced compared to the null-text optimization. In particular, as demonstrated in Appendix B (Fig. 15), the attention maps are less accurate which decreases the performance of Prompt-to-prompt editing.

Null-text optimization without pivotal inversion. We observe that optimizing the unconditional null-text embedding using random noise vectors, instead of pivotal inversion as described in previous paragraphs, completely breaks the null-text optimization. The results are inferior even to the DDIM inversion baseline as presented in Appendix B (Figs. 13 and 14). We hypothesize that null-text optimization is less expressive than model-tuning and thus depends on the efficient pivotal inversion, as it struggles to map all noise vectors to a single image.

5. Results

Real image editing is presented in Figs. 1, 2 and 5, showing our method not only reaches remarkable reconstruction quality but also retains high editability. In particular, we use the intuitive approach of Prompt-to-Prompt [16] and demonstrate that the editing capabilities which previously

Table 1. **User study results.** The participants were asked to select the best editing result in terms of fidelity to both the input image and the textual edit instruction.

VQGAN+CLIP	Text2Live	SDEDIT	Ours
3.8%	16.6%	14.5%	65.1%

were constrained to synthesized images are now applied to real images using our inversion technique.

As can be seen in Fig. 2, our method effectively modifies both textures ("floral shirt") and structured objects ("baby" to "robot"). Since we support the local editing of Prompt-to-Prompt and achieve high-fidelity reconstruction, the original identity is well preserved, even in the challenging case of a baby face. Fig. 2 also illustrates that our method requires only a single inversion process to perform multiple editing operations. Using a single inversion procedure, we can modify hair color, glasses, expression, background, and lighting and even replace objects or put on a joker make-up (bottom rows). Using Prompt-to-Prompt, we can also attenuate or amplify the effect of a specific word over real images, as appeared in Fig. 5. For additional examples, please refer to Appendix C.

Visual results for our high-fidelity reconstruction are presented in Figs. 1 and 8, and Appendix C (Fig. 16), supporting our quantitative measures in Sec. 4.

5.1. Comparisons

Our method aims attention at intuitive editing using only text, and so we compare our results to other text-only editing methods: (1) VQGAN+CLIP [11], (2) Text2Live [6], and (3) SDEDIT [23]. We evaluated these on the images used by Bar-Tal et al. [6] and photos that include more structured objects, such as humans and animals, which we gathered from the internet. In total, we use 100 samples of images, input captions, and edited captions.

We also compare our method to the mask-based methods of (4) Glide [25], (5) Blended-Diffusion [5], and (6) Stable Diffusion Inpaint [30]. The latter fine-tunes the diffusion model using an inpainting objective, allowing simple editing by inpainting a masked region using a target prompt.

Lastly, we consider the concurrent work of (7) Imagic [19], which employs model-tuning per editing operation and has been designed for the Imagen model [32]. We refrain from comparing to the concurrent works of Unitune [39] and DiffEdit [9] as there are no available implementations.

Qualitative Comparison. As presented in Fig. 6, VQGAN+CLIP [11] mostly produces unrealistic results. Text2LIVE [6] handles texture modification well but fails to manipulate more structured objects, e.g., placing a boat (3rd row). Both struggle due to the use of CLIP [26] which lacks a generative ability. In SDEDIT [23], the noisy image is fed to an intermediate step in the diffusion process, and therefore, it struggles to faithfully reconstruct the original

details. This results in severe artifacts when fine details are involved, such as human faces. For instance, identity drifts in the top row, and the background is not well preserved in the 2nd row. Contrarily, our method successfully preserves the original details, while allowing a wide range of realistic and meaningful editing, from simple textures to replacing well-structured objects.

Fig. 7 presents a comparison to mask-based methods, showing these struggles to preserve details that are found inside the masked region. This is due to the masking procedure that removes important structural information, and therefore, some capabilities are out of the inpainting reach.

A comparison to Imagic [19], which operates in a different setting – requiring model-tuning for each editing operation, is provided in Appendix B (Fig. 17). We first employ the unofficial Imagic implementation for Stable Diffusion and present the results for different values of the interpolation parameter $\alpha = 0.6, 0.7, 0.8, 0.9$. This parameter is used to interpolate between the target text embedding and the optimized one [19]. In addition, the Imagic authors applied their method using the Imagen model over the same images, using the following parameters $\alpha = 0.93, 0.86, 1.08$. As can be seen, Imagic produces highly meaningful editing, especially when the Imagen model is involved. However, Imagic struggles to preserve the original details, such as the identity of the baby (1st row) or cups in the background (2nd row). Furthermore, we observe that Imagic is quite sensitive to the interpolation parameter α , as a high value reduces the fidelity to the image and a low value reduces the fidelity to the text guidance, while a single value cannot be applied to all examples. Lastly, Imagic takes a longer inference time, as shown in Appendix C (Tab. 2).

Quantitative Comparison. Since ground truth is not available for text-based editing of real images, quantitative evaluation remains an open challenge. Similar to [6, 16], we present a user study in Tab. 1. 50 participants have rated a total of 48 images for each baseline. The participants were recruited using *Prolific* ([prolific.co](https://www.prolific.co)). We presented side-by-side images produced by: VQGAN+CLIP, Text2LIVE, SDEdit, and our method (in random order). We focus on methods that share a similar setting to ours – no model tuning and mask requirement. The participants were asked to choose the method that better applies the requested edit while preserving most of the original details. A print screen is provided in Appendix F (Fig. 18). As shown in Tab. 1, most participants favored our method.

Quantitative comparison to Imagic is presented in Appendix B (Fig. 11), using the unofficial Stable Diffusion implementation. According to these measures, our method achieves better scores for LPIPS perceptual distance, indicating a better fidelity to the input image.

5.2. Evaluating Additional Editing Technique

Most of the presented results consist of applying our method with the editing technique of Prompt-to-Prompt [16]. However, we demonstrate that our method is not confined to a specific editing approach, by showing it improves the results of the SDEdit [23] editing technique.

In Fig. 8 (top), we measure the fidelity to the original image using LPIPS perceptual distance [43] (lower is better), and the fidelity to the target text using CLIP similarity [26] (higher is better) over 100 examples. We use different values of the SDEdit parameter t_0 (marked on the curve), i.e., we start the diffusion process from different $t = t_0 \cdot T$ using a correspondingly noised input image. This parameter controls the trade-off between fidelity to the input image (low t_0) and alignment to the text (high t_0). We compare the standard SDEdit to first applying our inversion and then performing SDEdit while replacing the null-text embedding with our optimized embeddings. As shown, our inversion significantly improves the fidelity to the input image.

This is visually demonstrated in Fig. 8 (bottom). Since the parameter t_0 controls a reconstruction-editability trade-off, we have used a different parameter for each method (SDEdit with and without our inversion) such that both achieve the same CLIP score. As can be seen, when using our method, the true identity of the baby is well preserved.

6. Limitations

While our method works well in most scenarios, it still faces some limitations. The most notable one is inference time. Our approach requires approximately one minute on GPU for inverting a single image. Then, infinite editing operations can be made, each takes only ten seconds. This is not enough for real-time applications. Other limitations come from using Stable Diffusion [30] and Prompt-to-Prompt editing [16]. First, the VQ auto-encoder produces artifacts in some cases, especially when human faces are involved. We consider the optimization of the VQ decoder as out of scope here, since this is specific to Stable Diffusion and we aim for a general framework. Second, we observe that the generated attentions maps of Stable Diffusion are less accurate compared to the attention maps of Imagen [32], i.e., words might not relate to the correct region, indicating inferior text-based editing capabilities. Lastly, complicated structure modifications are out of reach for Prompt-to-Prompt, such as changing a seating dog to a standing one as in [19]. Our inversion approach is orthogonal to the specific model and editing techniques, and we believe that these will be improved in the near future.

7. Conclusions

We have presented an approach to invert real images with corresponding captions into the latent space of a text-guided diffusion model while maintaining its powerful editing ca-

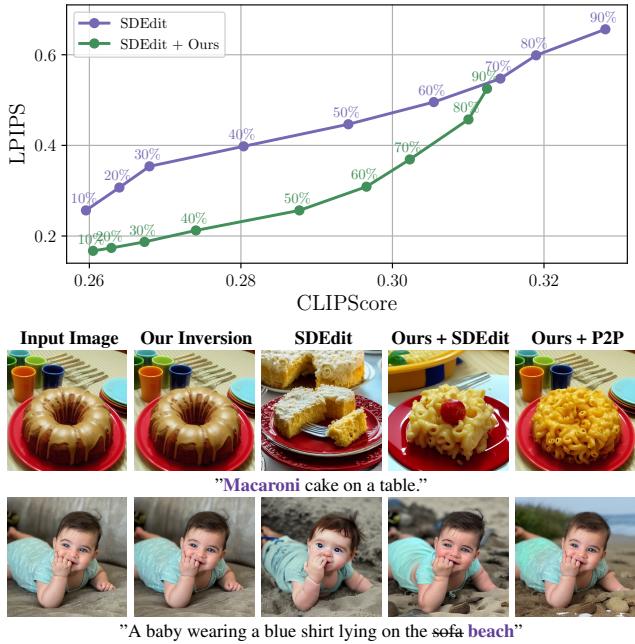


Figure 8. Our method improves SDEdit results. Top: we evaluate SDEdit with and without applying null-text inversion. In each measure, a different SDEdit parameter is used, i.e., different percent of diffusion steps are applied over the noisy image (marked on the curve). We measure both fidelity to the original image (via LPIPS, low is better) and fidelity to the target text (via CLIP, high is better). Bottom, from left to right: input image, null-text inversion, SDEdit, applying SDEdit after null-text inversion, and applying Prompt-to-Prompt after null-text inversion. As can be seen, our inversion significantly improves the fidelity to the original image when applied before SDEdit.

pabilities. Our two-step approach first uses DDIM inversion to compute a sequence of noisy codes, which roughly approximate the original image (with the given caption), then uses this sequence as a fixed pivot to optimize the input null-text embedding. Its fine optimization compensates for the inevitable reconstruction error caused by the classifier-free guidance component. Once the image-caption pair is accurately embedded in the output domain of the model, prompt-to-prompt editing can be instantly applied at inference time. By introducing two new technical concepts to text-guided diffusion models – pivotal inversion and null-text optimization, we were able to bridge the gap between reconstruction and editability. Our approach offers a surprisingly simple and compact means to reconstruct an arbitrary image, avoiding the computationally intensive model-tuning. We believe that null-text inversion paves the way for real-world use case scenarios for intuitive, text-based, image editing.

8. Acknowledgments

We thank Yuval Alaluf, Rinon Gal, Aleksander Holynski, Bryan Eric Feldman, Shlomi Fruchter and David Salesin for their valuable inputs that helped improve this work, and to Bahjat Kawar, Shiran Zada and Oran Lang for

providing us with their support for the Imagic [19] comparison. We also thank Jay Tenenbaum for the help with writing the background.

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. [2](#)
- [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. [2](#)
- [3] Yuval Alaluf, Omer Tov, Ron Mokady, Rinon Gal, and Amit H. Bermano. Hyperstyle: Stylegan inversion with hypernetworks for real image editing, 2021. [2](#)
- [4] Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *arXiv preprint arXiv:2206.02779*, 2022. [2](#)
- [5] 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. [2, 7](#)
- [6] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kassten, and Tali Dekel. Text2live: Text-driven layered image and video editing. *arXiv preprint arXiv:2204.02491*, 2022. [3, 6, 7, 8, 11, 12](#)
- [7] Amit H Bermano, Rinon Gal, Yuval Alaluf, Ron Mokady, Yotam Nitzan, Omer Tov, Oren Patashnik, and Daniel Cohen-Or. State-of-the-art in the architecture, methods and applications of stylegan. In *Computer Graphics Forum*, volume 41, pages 591–611. Wiley Online Library, 2022. [1, 2](#)
- [8] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. [5](#)
- [9] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022. [3, 7](#)
- [10] Antonia Creswell and Anil Anthony Bharath. Inverting the generator of a generative adversarial network. *IEEE transactions on neural networks and learning systems*, 30(7):1967–1974, 2018. [2](#)
- [11] Katherine Crowson. Vqgan + clip, 2021. <https://colab.research.google.com/drive/1L8oL-vLJXVcRzCFbPwOoMkPKJ8-aYdPN>. [6, 7, 12](#)
- [12] Katherine Crowson, Stella Biderman, Daniel Kornis, Dashiell Stander, Eric Hallahan, Louis Castricato, and Edward Raff. Vqgan-clip: Open domain image generation and editing with natural language guidance. *arXiv preprint arXiv:2204.08583*, 2022. [2, 11](#)
- [13] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. [1, 4](#)
- [14] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis, 2020. [6](#)
- [15] 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. 2, 5, 6, 7
- [16] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 1, 2, 3, 5, 7, 8
- [17] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2, 13
- [18] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 1, 4
- [19] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Hui-Tang Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. *ArXiv*, abs/2210.09276, 2022. 2, 3, 5, 7, 8, 9, 11, 12, 19
- [20] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2426–2435, 2022. 2
- [21] Gihyun Kwon and Jong Chul Ye. Clipstyler: Image style transfer with a single text condition. *arXiv preprint arXiv:2112.00374*, 2021. 2
- [22] Zachary C Lipton and Subarna Tripathi. Precise recovery of latent vectors from generative adversarial networks. *arXiv preprint arXiv:1702.04782*, 2017. 2
- [23] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021. 2, 6, 7, 8, 12
- [24] Ron Mokady, Amir Hertz, and Amit H Bermano. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*, 2021. 3, 6
- [25] Alex 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. *arXiv preprint arXiv:2112.10741*, 2021. 2, 7
- [26] 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. *arXiv preprint arXiv:2103.00020*, 2021. 3, 7, 8
- [27] 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
- [28] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. *arXiv preprint arXiv:2008.00951*, 2020. 2
- [29] Daniel Roich, Ron Mokady, Amit H. Bermano, and Daniel Cohen-Or. Pivotal tuning for latent-based editing of real images. *ACM Transactions on Graphics (TOG)*, 2022. 2, 4
- [30] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 1, 2, 3, 4, 6, 7, 8, 12
- [31] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022. 2, 5
- [32] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamvar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 1, 2, 3, 7, 8
- [33] Shelly Sheynin, Oron Ashual, Adam Polyak, Uriel Singer, Oran Gafni, Eliya Nachmani, and Yaniv Taigman. Knndiffusion: Image generation via large-scale retrieval. *arXiv preprint arXiv:2204.02849*, 2022. 3
- [34] 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, 13
- [35] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2020. 1, 2, 3, 4, 13
- [36] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019. 2
- [37] Matteo Stefanini, Marcella Cornia, Lorenzo Baraldi, Silvia Cascianelli, Giuseppe Fiameni, and Rita Cucchiara. From show to tell: a survey on deep learning-based image captioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 6
- [38] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *arXiv preprint arXiv:2102.02766*, 2021. 2
- [39] 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 preprint arXiv:2210.09477*, 2022. 2, 3, 5, 7
- [40] Tengfei Wang, Yong Zhang, Yanbo Fan, Jue Wang, and Qifeng Chen. High-fidelity gan inversion for image attribute editing. *ArXiv*, abs/2109.06590, 2021. 2
- [41] Weihao Xia, Yulun Zhang, Yujiu Yang, Jing-Hao Xue, Bolei Zhou, and Ming-Hsuan Yang. Gan inversion: A survey, 2021. 1, 2
- [42] Raymond A. Yeh, Chen Chen, Teck Yian Lim, Alexander G. Schwing, Mark Hasegawa-Johnson, and Minh N. Do. Semantic image inpainting with deep generative models, 2017. 2
- [43] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 8
- [44] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A Efros. Generative visual manipulation on the natural image manifold. In *European conference on computer vision*, pages 597–613. Springer, 2016. 2

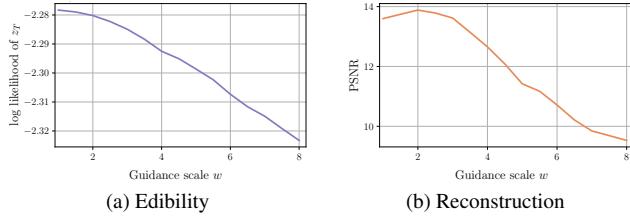


Figure 9. Setting the guidance scale for DDIM. We evaluate the DDIM inversion with different values of the guidance scale. On left, we measure the log-likelihood of the latent vector z_T with respect to multivariate normal distribution. This estimates the editability as z_T should ideally distribute normally and deviation from this distribution reduces our ability to edit the image. On right, we measure the reconstruction quality using PSNR. As can be seen, using a small guidance scale, such as $w = 1$, results in better editability and reconstruction.

Appendix

A. Societal Impact

Our work suggests a new editing technique for manipulating real images using state-of-the-art text-to-image diffusion models. This modification of real photos might be exploited by malicious parties to produce fake content in order to spread disinformation. This is a known problem, common to all image editing techniques. However, research in identifying and preventing malicious editing is already making significant progress. We believe our work would contribute to this line of work, since we provide an analysis of the inversion and editing procedures using text-to-image diffusion models.

B. Ablation Study

DDIM Inversion. To validate our selection of the guidance scale parameter of $w = 1$ during the DDIM Inversion (see Algorithm 1, line 3, in the main text), we conduct the DDIM inversion with different values of w from 1 to 8 using the same data as in Section 4. For each inversion, we measure the log-likelihood of the result latent image $z_T^* \in \mathcal{R}^{64 \times 64 \times 4}$ under the standard multivariate normal distribution. Intuitively, to achieve high editability we would like to maximize this term since during training z_T^* distributes normally. The mean log-likelihood as a function of w is plotted in Fig. 9a. In addition, we measure the reconstruction with respect to the ground truth input image using the PSNR metric. As can be seen in Fig. 9b, increasing the value of w results in less editable latent vector z_T^* and poorer initial reconstruction for our optimization, and therefore we use $w = 1$.

Robustness to different input captions. In Fig. 12 (top), we demonstrate our robustness to different input captions by successfully inverting an image using multiple captions. Yet, the edited parts should be included in the source cap-

Table 2. Inference time comparison. We measure both inversion and editing time for different methods. SDEdit is faster than ours, as an inversion is not employed by default, but fails to preserve the unedited parts. Our method is more efficient than the rest of the baselines, as it provides accurate reconstruction with faster inversion time, while also allowing multiple editing operations after a single inversion.

Method	Inversion	Editing	Multiple edits
VQGAN + CLIP	—	~ 1m	No
Text2Live	—	~ 9m	No
SDEdit	—	10s	Yes
Imagic	~ 5m	10s	No
Ours	~ 1m	10s	Yes

tion in order to produce semantic attention maps for these (Fig. 12 bottom). For example, to edit the print on the shirt, the source caption should include a "shirt with a drawing" term or a similar one.

Null-text optimization without pivotal inversion. Optimizing the null-text embedding fails without the efficient pivotal inversion. This is demonstrated in Fig. 13 and 14, where the non-pivotal null-text optimization produces low-quality reconstruction (2nd row).

Textual inversion with a pivot. Fig. 15 illustrate performing textual inversion around a pivot, i.e., similar to our pivotal inversion but optimizing the conditioned embedding. This results in a comparable reconstruction to ours, as demonstrated in Fig. 15 (bottom), but editability is reduced. By analyzing the attention maps (Fig. 15, top), observing that these are less accurate than ours. For example, using our null-text optimization, the attention referring to "goats" is much more local, and attention referring to "desert" is more accurate. Consequently, editing the "desert" results in artifacts over the goats (Fig. 15, bottom).

C. Additional results

Additional editing results of our method are provided in Fig. 10 and additional comparisons are provided in Fig. 16.

Inference time comparison. As can be seen in Tab. 2, SDEdit is the fastest since an inversion is not employed, but as a result, it fails to preserve the details of the original image. Our method is more efficient than Text2Live [6], VQGAN+CLIP [12] and Imagic [19], as it provides an accurate reconstruction in ~ 1 minute, while also allowing multiple editing operations after a single inversion.

Comparison to Imagic Quantitative comparison to Imagic is presented in Fig. 11, using the unofficial Stable Diffusion implementation. According to these measures,

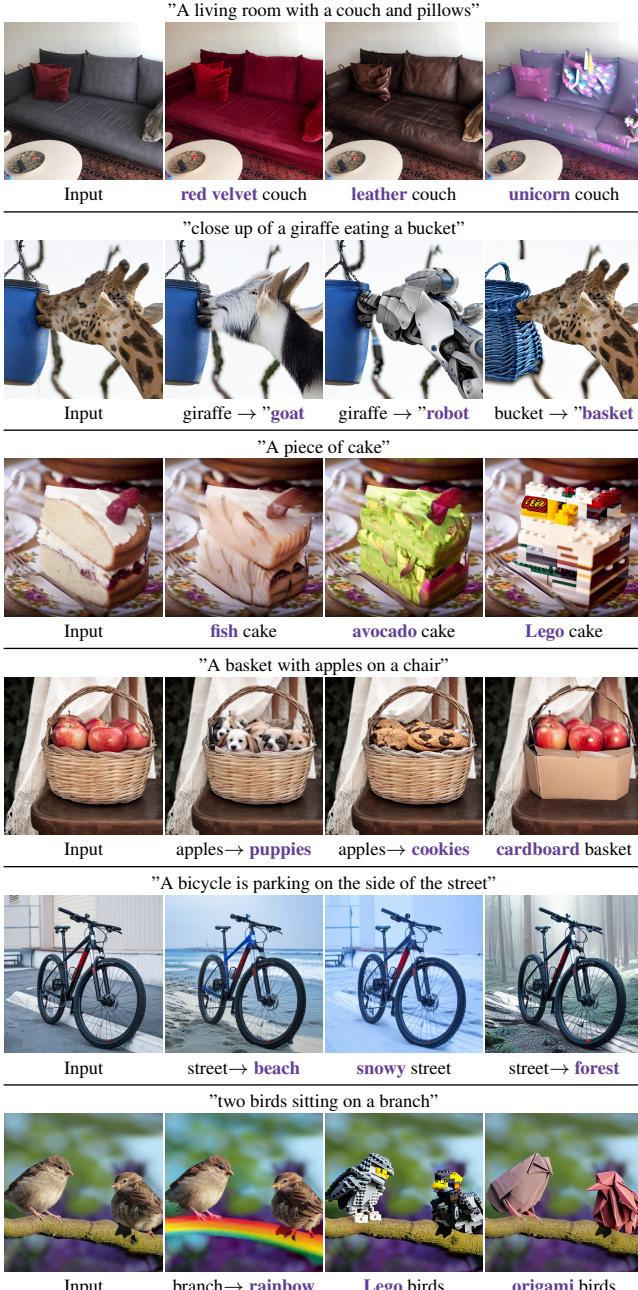


Figure 10. Additional editing results for our method.

our method achieves better preservation of the original details (lower LPIPS). This is also supported by the visual results in Fig. 17, as Imagic struggles to accurately retain the background. Furthermore, we observe that Imagic is quite sensitive to the interpolation parameter α , as a high value reduces the fidelity to the image and a low value reduces the fidelity to the text, while a single value cannot be applied to all examples. In addition, the authors of Imagic applied their method on the same three images, presented in Fig. 17, using $\alpha = 0.93, 0.86, 1.08$. This results in much better quality, however, still the background is not preserved.

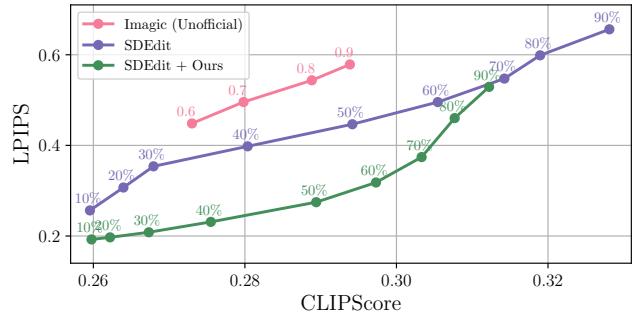


Figure 11. Comparison to Imagic We quantitatively evaluate Imagic using the unofficial implementation for Stable Diffusion. We measure both fidelity to the original image (via LPIPS, low is better) and fidelity to the target text (via CLIP, high is better). We use different values of the text embedding interpolation parameter α , marked on the curve. The high LPIPS perceptual distance indicates that Imagic fails to retain high fidelity to the original image.

D. Implementation details

In all of our experiments, we employ the Stable Diffusion [30] using a DDIM sampler with the default hyperparameters: number of diffusion steps $T = 50$ and guidance scale $w = 7.5$. Stable diffusion utilizes a pre-trained CLIP network as the language model ψ . The null-text is tokenized into *start-token*, *end-token*, and 75 non-text padding tokens. Notice that the padding tokens are also used in CLIP and the diffusion model since both models do not use masking.

All inversion results except the ones in the ablation study were obtained using $N = 10$ (See Algorithm 1 in the main paper) and a learning rate of 0.01. We have used an early stop parameter of $\epsilon = 1e - 5$ such that the total inversion for an input image and caption took $40s - 120s$ on a single A100 GPU. Namely, for each timestamp t , we stop the optimization when the loss function value reaches $\epsilon = 1e - 5$.

Baseline Implementations. For the comparisons in section 5, we use the official implementation of Text2Live* [6] and VQGAN+CLIP† [11]. We have implemented the SDEdit [23] method over Stable Diffusion based on the official implementation‡. We also compare our method to Imagic [19] using an unofficial implementation§ (see Appendix C).

Global null-text Inversion. The algorithm for optimizing only a single null-text embedding \emptyset for all timestamps is presented in algorithm 2. In this case, since the optimization of \emptyset in a single timestamp affects all other timestamps, we change the order of the iterations in Algorithm 1. That is, we perform N iterations in each we optimize \emptyset for all the diffusion timestamps by iterating over t . As shown in Sec-

*<https://github.com/omerbt/Text2LIVE>

†<https://github.com/nerdyrodent/VQGAN-CLIP>

‡<https://github.com/ermongroup/SDEdit>

§<https://github.com/ShivamShirao/diffusers/tree/main/examples/imagic>

tion 4, the convergence of this optimization is much slower than our final method. More specifically, we found that only after 7500 optimization steps (about 30 minutes) the global null-text inversion accurately reconstruct the input image.

Algorithm 2: Global null-text inversion

```

1 Input: A source prompt  $\mathcal{P}$  and input image  $\mathcal{I}$ .
2 Output: Noise vector  $z_T$  and an optimized
   embedding  $\emptyset$ .
3 Set guidance scale  $w = 1$ ;
4 Compute the intermediate results  $z_T^*, \dots, z_0^*$  of
   DDIM inversion for image  $\mathcal{I}$ ;
5 Set guidance scale  $w = 7.5$ ;
6 Initialize  $\emptyset \leftarrow \psi("")$ ;
7 for  $j = 0, \dots, N - 1$  do
8   Set  $\bar{z}_T \leftarrow z_T^*$ ;
9   for  $t = T, T - 1, \dots, 1$  do
10    |  $\emptyset \leftarrow \emptyset - \eta \nabla_{\emptyset} \|z_{t-1}^* - z_{t-1}(\bar{z}_t, \emptyset, \mathcal{C})\|_2^2$ ;
11    | Set  $\bar{z}_{t-1} \leftarrow z_{t-1}(\bar{z}_t, \emptyset, \mathcal{C})$ ;
12 end
13 Return  $\bar{z}_T, \emptyset$ 

```

E. Additional Background - Diffusion Models

Diffusion Denoising Probabilistic Models (DDPM) [17, 34] are generative latent variable models that aim to model a distribution $p_\theta(x_0)$ that approximates the data distribution $q(x_0)$ and easy to sample from. DDPMs model a “forward process” in the space of x_0 from data to noise. This is called “forward” due to its procedure progressing from x_0 to x_T . Note that this process is a Markov chain starting from x_0 , where we gradually add noise to the data to generate the latent variables $x_1, \dots, x_T \in X$. The sequence of latent variables, therefore, follows $q(x_1, \dots, x_t \mid x_0) = \prod_{i=1}^t q(x_t \mid x_{t-1})$, where a step in the forward process is defined as a Gaussian transition $q(x_t \mid x_{t-1}) := N(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t I)$ parameterized by a schedule $\beta_0, \dots, \beta_T \in (0, 1)$. When T is large enough, the last noise vector x_T nearly follows an isotropic Gaussian distribution.

An interesting property of the forward process is that one can express the latent variable x_t directly as the following linear combination of noise and x_0 without sampling intermediate latent vectors:

$$x_t = \sqrt{\alpha_t}x_0 + \sqrt{1 - \alpha_t}w, \quad w \sim N(0, I), \quad (5)$$

where $\alpha_t := \prod_{i=1}^t (1 - \beta_i)$.

To sample from the distribution $q(x_0)$, we define the dual “reverse process” $p(x_{t-1} \mid x_t)$ from isotropic Gaussian noise x_T to data by sampling the posteriors $q(x_{t-1} \mid x_t)$. Since the intractable reverse process $q(x_{t-1} \mid x_t)$ depends on the unknown data distribution $q(x_0)$, we approximate it

with a parameterized Gaussian transition network $p_\theta(x_{t-1} \mid x_t) := N(x_{t-1} \mid \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$. The $\mu_\theta(x_t, t)$ can be replaced [17] by predicting the noise $\varepsilon_\theta(x_t, t)$ added to x_0 using equation 5.

We use Bayes’ theorem to approximate

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1 - \alpha_t}} \varepsilon_\theta(x_t, t) \right). \quad (6)$$

Once we have a trained $\varepsilon_\theta(x_t, t)$, we can use the following sample method

$$x_{t-1} = \mu_\theta(x_t, t) + \sigma_t z, \quad z \sim N(0, I). \quad (7)$$

We can control σ_t of each sample stage, and in DDIMs [35] the sampling process can be made deterministic using $\sigma_t = 0$ in all the steps. The reverse process can finally be trained by solving the following optimization problem:

$$\min_{\theta} L(\theta) := \min_{\theta} E_{x_0 \sim q(x_0), w \sim N(0, I), t} \|w - \varepsilon_\theta(x_t, t)\|_2^2,$$

teaching the parameters θ to fit $q(x_0)$ by maximizing a variational lower bound.

F. User-Study

An illustration of our user study is provided in Fig. 18

G. Image Attribution

Girl in a field: https://unsplash.com/photos/1pCpWipo_jM
 Birds on a branch: <https://pixabay.com/photos/sparrows-birds-perched-sperlings-3434123/>
 Basket with apples: <https://unsplash.com/photos/4Bj27zMqNSE>
 Bicycle: https://unsplash.com/photos/vZAk_n9Plfc
 Child climbing: <https://unsplash.com/photos/oLZViCDG-dk>
 Mountains: <https://pixabay.com/photos/desert-mountains-sky-clouds-peru-4842264/>
 Giraffe: <https://www.flickr.com/photos/tambako/30850708538/>
 Blue-haired woman in the forest: <https://unsplash.com/photos/I3oRtzyBIFg>
 Dining table: <https://cocodataset.org/#explore?id=360849>
 Elephants: <https://cocodataset.org/#explore?id=345520>
 Man with a doughnut: <https://cocodataset.org/#explore?id=360849>
 Cake on a table: <https://cocodataset.org/#explore?id=413699>
 Piece of cake: <https://cocodataset.org/#explore?id=133063>



Input Image

Input caption: "A woman with a blue hair."



Our Inversion

"...smiling woman..." "...sad woman..." "...curly blue hair..." "...green hair..." woman → squirrel woman → storm trooper

Input caption: "A woman in the forest."



Our Inversion

"...forest at fall..." "...forest at winter..." forest → city forest → beach forest → water park forest → magic kingdom

Input caption: "A woman wearing a shirt with a drawing."



Our Inversion

"...long sleeves shirt..." "...turtle neck shirt..." "...red shirt..." "... drawing of kermit." "...of cookie monster." "...of inspector gadget."

Cross-attention maps

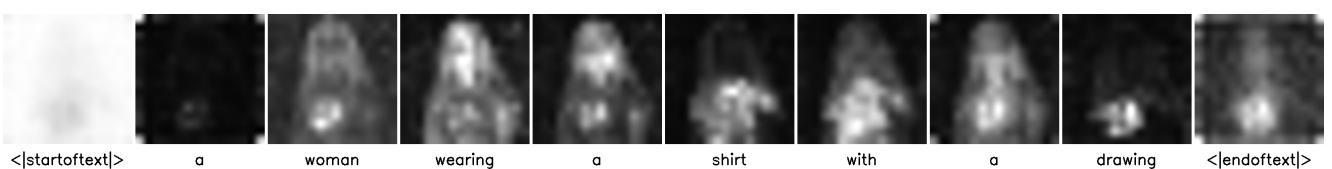
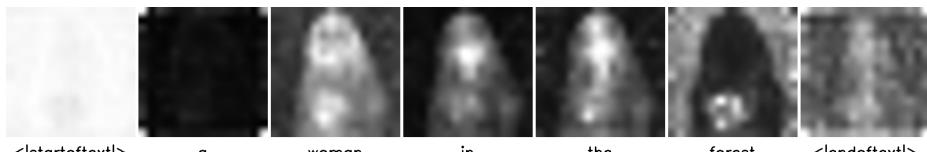
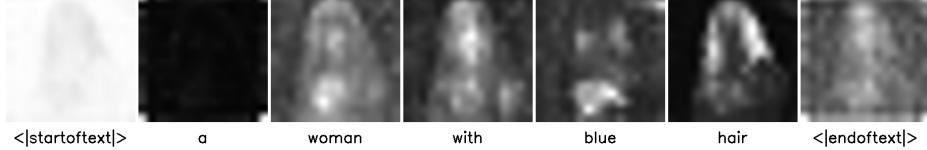


Figure 12. **Robustness to the input caption.** We can invert an input image (top) using different input captions (first column). Naturally, the selection of the caption effects the editing abilities with Prompt-to-Prompt, as can be seen in the visualization of the cross-attention map (bottom). Yet, our method is not particularly sensitive to the exact wording of the prompt.

Input caption: “A black dinning room table sitting in a yellow dinning room.”

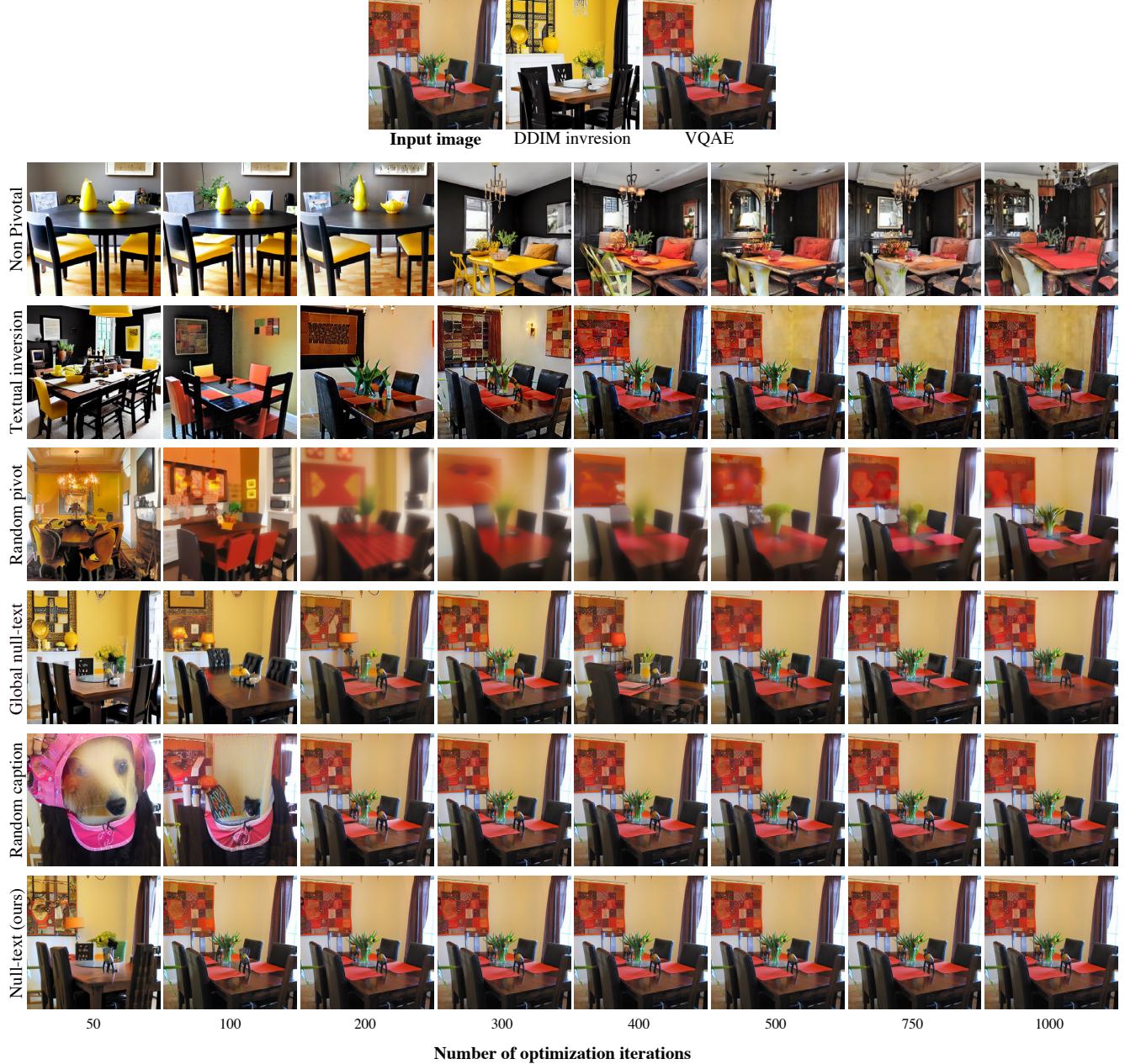


Figure 13. Ablation study. We show the inversion results for an increasing number of optimization iterations. Our method achieves high-quality reconstruction with fewer optimization steps.

Input caption: “Two people riding elephants in dirty deep water.”

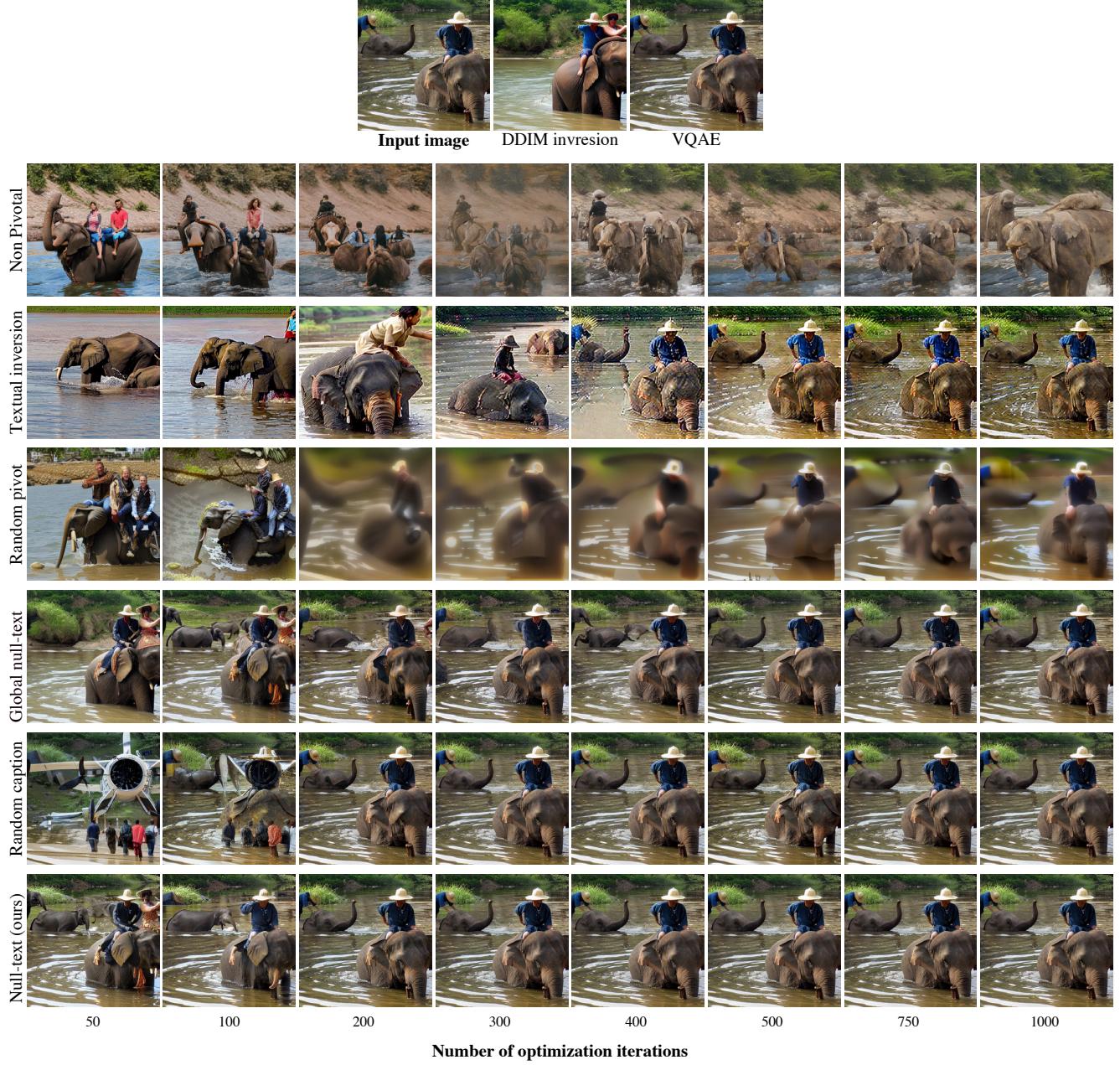


Figure 14. Ablation study. We show the inversion results for an increasing number of optimization iterations. Our method achieves high-quality reconstruction with fewer optimization steps.

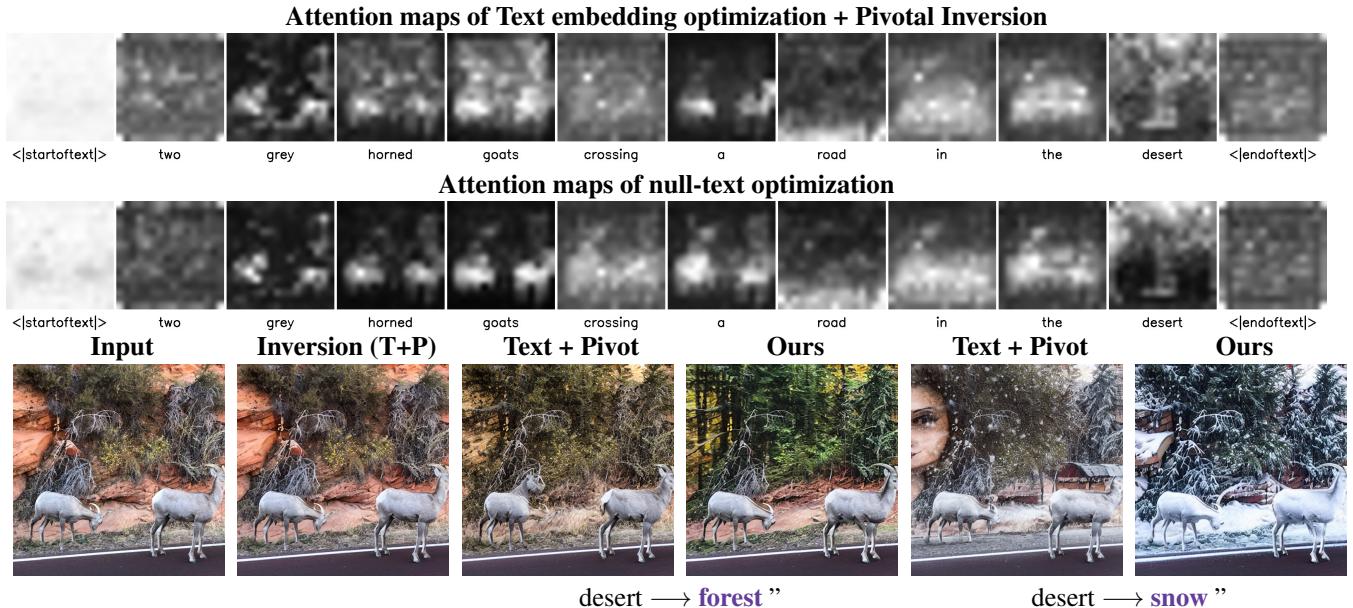


Figure 15. Ablation study - Textual inversion with a pivot. We compare our method to replacing the null-text optimization with optimizing the conditional (textual) embedding while still applying pivotal inversion. As can be seen (top), this results in less accurate attention maps, and thus, in less accurate editing capabilities. In particular, textual inversion with a pivot achieves high-fidelity reconstruction ("Inversion (T+P)"), but goat heads distort (bottom) when editing is applied due to the inaccurate attention maps.

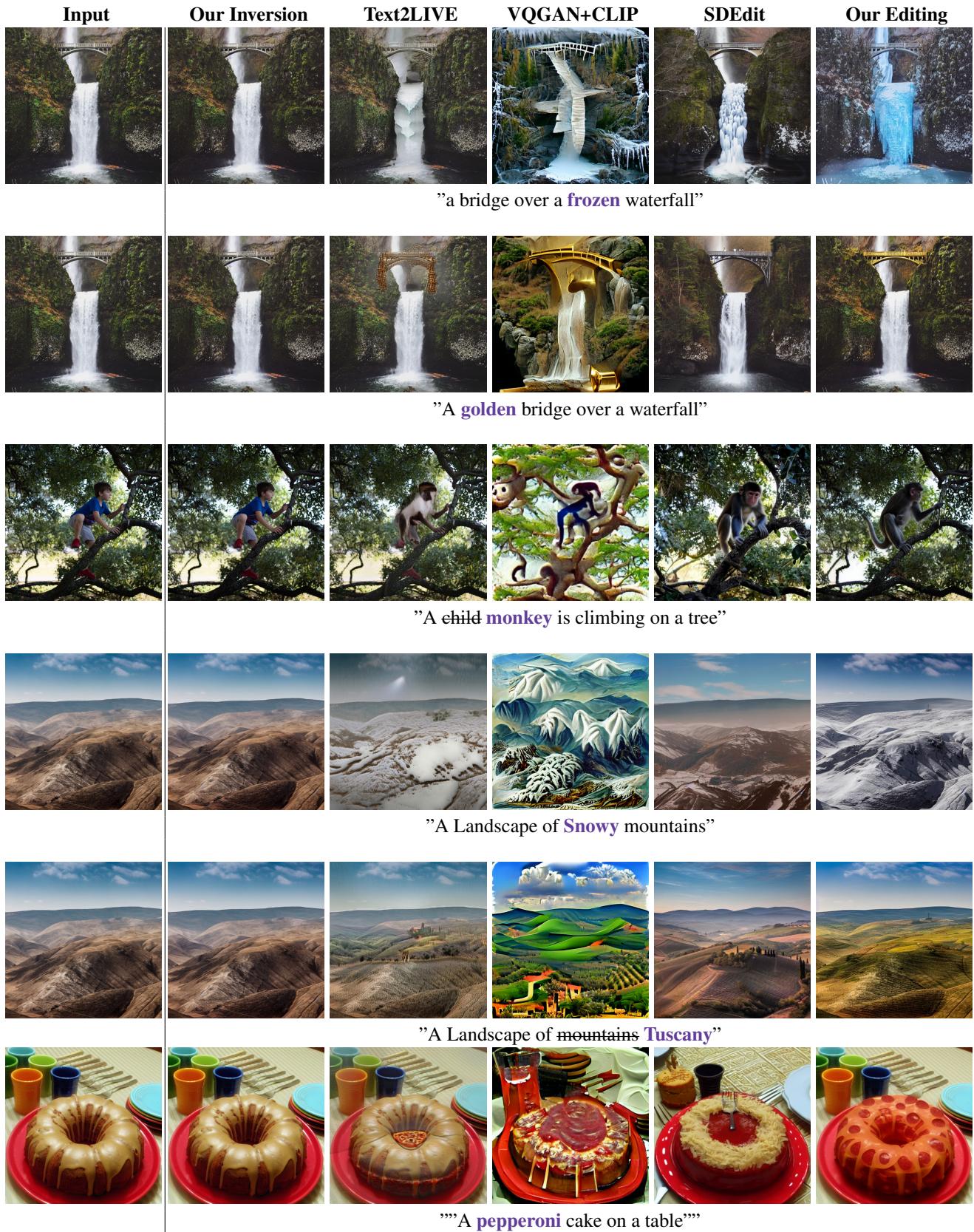


Figure 16. Additional comparison results.

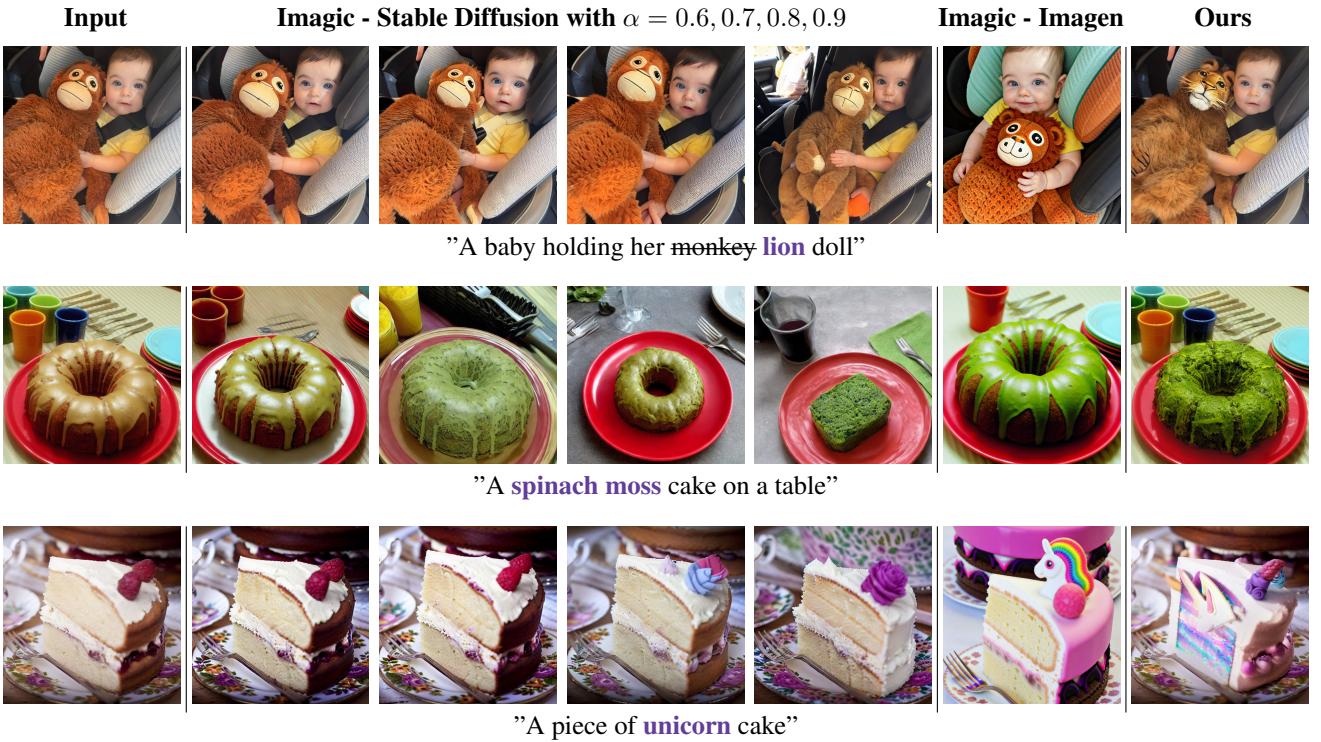


Figure 17. **Comparison to Imagic [19].** We first employ the unofficial Imagic implementation for Stable Diffusion and present the results for different values of the interpolation parameter $\alpha = 0.6, 0.7, 0.8, 0.9$ (left to right). In addition, Imagic authors applied their method using the Imagen model over the same images, using the parameters $\alpha = 0.93, 0.86, 1.08$ (from top to bottom row). As can be seen, Imagic produces highly meaningful editing, especially when the Imagen model is involved. However, Imagic struggles to preserve the original details, such as the identity of the baby (1st row) or cups in the background (2nd row). Furthermore, we observe that each example requires a separate tuning of the α parameter. Lastly, recall that each Imagic editing requires a separate tuning of the model.

4 / 24

Which image below better applies the requested edit to the input image on top, * while preserving most of the details from the input image?



Input Image

Edit instruction: couch → unicorn pattern couch



image 1



image 2



image 3



image 4

Image 1

Image 2

Image 3

Image 4

Figure 18. User study print screen.