Plug-and-Play Diffusion Features for Text-Driven Image-to-Image Translation

Narek Tumanyan* Michal Geyer* Shai Bagon Tali Dekel
Weizmann Institute of Science
*Indicates equal contribution.

Project webpage: https://pnp-diffusion.github.io

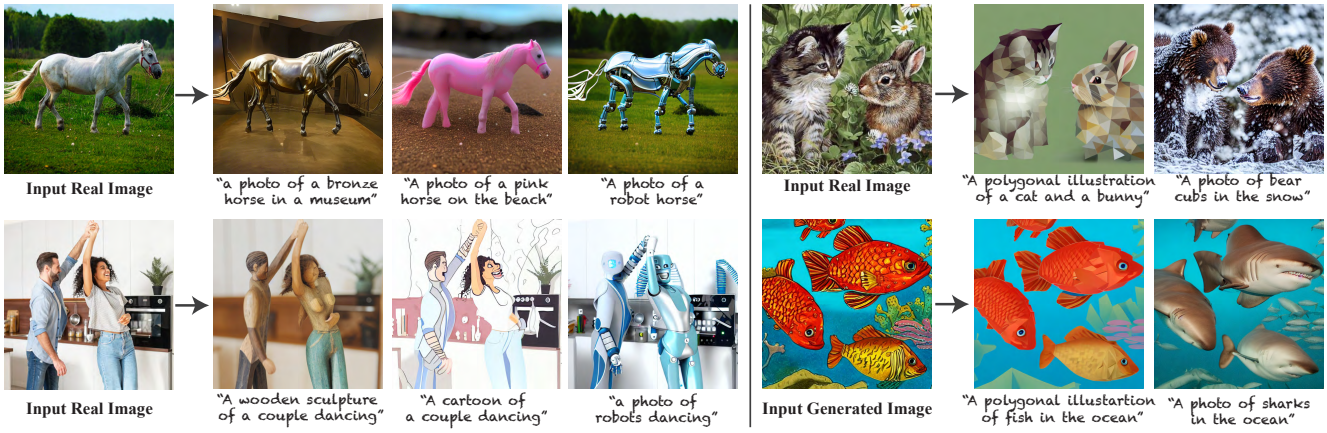

Figure 1. Given a single real-world image as input, our framework enables versatile text-guided translations of the original content. Our results exhibit high fidelity to the input structure and scene layout, while significantly changing the perceived semantic meaning of objects and their appearance. Our method does not require any training, but rather harnesses the power of a pre-trained text-to-image diffusion model through its internal representation. We present new insights about deep features encoded in such models, and an effective framework to control the generation process through simple modification of these features.

Abstract

Large-scale text-to-image generative models have been a revolutionary breakthrough in the evolution of generative AI, allowing us to synthesize diverse images that convey highly complex visual concepts. However, a pivotal challenge in leveraging such models for real-world content creation tasks is providing users with control over the generated content. In this paper, we present a new framework that takes text-to-image synthesis to the realm of image-to-image translation – given a guidance image and a target text prompt as input, our method harnesses the power of a pre-trained text-to-image diffusion model to generate a new image that complies with the target text, while preserving the semantic layout of the guidance image. Specifically, we observe and empirically demonstrate that fine-grained control over the generated structure can be achieved by manipulating spatial features and their self-attention inside the model. This results in a simple and effective approach, where features extracted from the guidance image are directly injected into the generation process of the translated image, requiring no training or fine-tuning. We demonstrate high-quality results on versatile text-guided image translation tasks, including translating sketches, rough drawings and animations into realistic images, changing of the class and appearance of objects in a given image, and modifications of global qualities such as lighting and color.

1. Introduction

With the rise of text-to-image foundation models – billion-parameter models trained on a massive amount of text-image data, it seems that we can translate our imagination into high-quality images through text [12, 34, 36, 40]. While such foundation models unlock a new world of creative processes in content creation, their power and expressivity come at the expense of user controllability, which is largely restricted to guiding the generation solely through an input text. In this paper, we focus on attaining control over the generated structure and semantic layout of the scene – an imperative component in various real-world content creation tasks, ranging from visual branding and marketing to digital art. That is, our goal is to take text-to-image generation to the realm of text-guided Image-to-Image (I2I) translation, where an input image guides the layout (e.g., the structure of the horse in Fig. 1), and the text guides the perceived semantics and appearance of the scene (e.g., "robot horse" in Fig. 1).

A possible approach for achieving control of the gen-

erated layout is to design text-to-image foundation models that explicitly incorporate additional guiding signals, such as user-provided masks [12, 28, 34]. For example, recently Make-A-Scene [12] trained a text-to-image model that is also conditioned on a label segmentation mask, defining the layout and the categories of objects in the scene. However, such an approach requires an extensive compute as well as large-scale text-guidance-image training tuples, and can be applied at test-time to these specific types of inputs. In this paper, we are interested in a unified framework that can be applied to versatile I2I translation tasks, where the structure guidance signal ranges from artistic drawings to photo-realistic images (see Fig. 1). Our method does not require any training or fine-tuning, but rather leverages a pre-trained and fixed text-to-image diffusion model [36].

Specifically, we pose the fundamental question of how structure information is internally encoded in such a model. We dive into the intermediate spatial features that are formed during the generation process, empirically analyze them, and devise a new framework that enables fine-grained control over the generated structure by applying simple manipulations to spatial features inside the model. Specifically, spatial features and their self-attentions are extracted from the guidance image, and are directly injected into the text-guided generation process of the target image. We demonstrate that our approach is not only applicable in cases where the guidance image is generated from text, but also for real-world images that are inverted into the model.

Our approach of operating in the space of diffusion features is related to Prompt-to-Prompt (P2P) [16], which recently observed that by manipulating the cross-attention layers, it is possible to control the relation between the spatial layout of the image to each word in the text. We demonstrate that fine-grained control over the generated layout is difficult to achieve solely from the interaction with a text. Intuitively, since the cross attention is formed by the association of spatial features to *words*, it allows to capture rough regions at the *object level*, yet localized spatial information that is not expressed in the source text prompt (e.g., object parts) is not guaranteed to be preserved by P2P. Instead, our method focuses only on *spatial features* and their self-affinities – we show that such features exhibit high granularity of spatial information, allowing us to control the generated structure, while not restricting the interaction with the text. Our method outperforms P2P in terms of structure preservation and is superior in working with real guidance images.

To summarize, we make the following key contributions:
(i) We provide new empirical insights about internal spatial features formed during the diffusion process.

(ii) We introduce an effective framework that leverages the power of pre-trained and fixed guided diffusion, allowing to perform high-quality text-guided I2I translation without any training or fine-tuning.

(iii) We show, both quantitatively and qualitatively that our method outperforms existing state-of-the-art baselines, achieving significantly better balance between preserving the guidance layout and deviating from its appearance.

2. Related Work

Image-to-image translation. Image-to-Image (I2I) translation is aimed at estimating a mapping of an image from a source domain to a target domain, while preserving the domain-invariant characteristics of the input image, e.g., objects' structure or scene layout. From classical to modern data-driven methods, numerous visual problems have been formulated and tackled as an I2I task (e.g., [7, 10, 17, 32, 42]). Seminal deep-learning-based methods have proposed various GAN-based frameworks to encourage the output image to comply with the distribution of the target domain [23, 29, 30, 49]. Nevertheless, these methods require datasets of example images from both source and target domains, and often require training from scratch for each translation task (e.g., horse-to-zebra, day-to-night, summer-to-winter). Other works utilize pre-trained GANs by performing the translation in its latent space [1, 35, 45]. Several methods have also considered the task of zero-shot I2I by training a generator on a single source-target image pair example [46, 47]. With the advent of unconditional image diffusion models, several methods have been proposed to adopt or extend them for various I2I tasks [39, 48]. In this paper, we consider the task of *text-guided image-to-image translation* where the target domain is not specified through a dataset of images but rather via a target text prompt. Our method is zero-shot, does not require training and is applicable to versatile I2I tasks.

Text-guided image manipulation. With the tremendous progress in language-vision models, a surge of methods have been proposed to perform various types of text-driven image edits. Various methods have proposed to combine CLIP [33], which provides a rich and powerful joint image-text embedding space, with a pre-trained unconditional image generator, e.g., a GAN [6, 14, 26, 31] or a diffusion model [2, 22, 25]. For example, DiffusionCLIP [22] uses CLIP to fine-tune a diffusion model to perform text guided manipulations. Concurrent to our work, [25] uses CLIP and semantic losses of [46] to guide a diffusion process to perform I2I translation. Aiming to edit the appearance of objects in real-world images, Text2LIVE [4] trains a generator on a single image-text pair, without additional training data; thus, avoiding the trade-off, inherent to pre-trained generators, between satisfying the target edit and maintaining high-fidelity to the original content. While these methods have demonstrated impressive text-guided semantic edits, there is still a gap between the generative prior that is learned solely from visual data (typically on specific domains or ImageNet data), and the rich CLIP text-image guiding signal that has been learned from much broader and

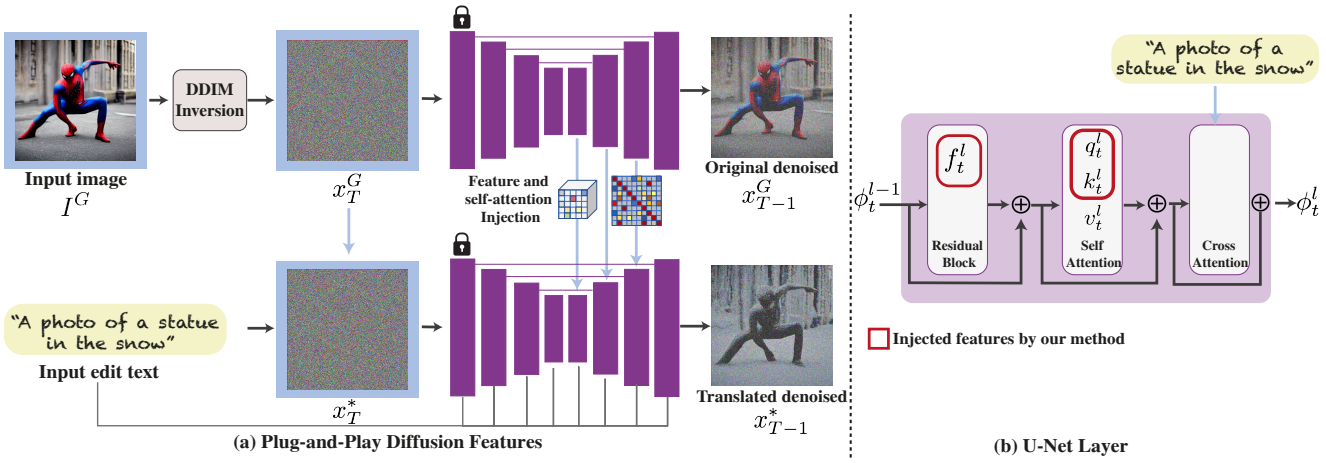

Figure 2. *Plug-and-play Diffusion Features.* (a) Our framework takes as input a guidance image and a text prompt describing the desired translation; the guidance image is inverted to initial noise $\boldsymbol{x}_T^G$, which is then progressively denoised using DDIM sampling. During this process, we extract $(\boldsymbol{f}_t^l, \boldsymbol{q}_t^l, \boldsymbol{k}_t^l)$ – spatial features from the decoder layers and their self-attention, as illustrated in (b). To generate our text-guided translated image, we fix $\boldsymbol{x}_T^* = \boldsymbol{x}_T^G$ and inject the guidance features $(\boldsymbol{f}_t^l, \boldsymbol{q}_t^l, \boldsymbol{k}_t^l)$ at certain layers, as discussed in Sec. 4.

richer data. Recently, text-to-image generative models have closed this gap by directly conditioning image generation on text during training [12, 28, 34, 36, 40]. These models have demonstrated unprecedented capabilities in generating high-quality and diverse images from text, capturing complex visual concepts (e.g., object interactions, geometry, or composition). Nevertheless, such models offer little control over the generated content. This creates a great interest in developing methods to adopt such unconstrained text-to-image models for controlled content creation.

Several *concurrent* methods have taken first steps in this direction, aiming to influence different properties of the generated content [13, 21, 38, 48]. DreamBooth [38] and Textual Inversion [13] share the same high-level goal of "personalizing" a pre-trained text-to-image diffusion model given a few user-provided images. Our method also leverages a pre-trained text-to-image diffusion model to achieve our goal, yet does not involve any training or fine-tuning. Instead, we devise a simple framework that intervenes in the generation process by directly manipulating the spatial features.

As discussed in Sec. 1, our methodological approach is related to Prompt-to-Prompt [16], yet our method offers several key advantages: (i) enables fine-grained control over the generated shape and layout, (ii) allows to use arbitrary text-prompts to express the target translation; in contrast to P2P that requires word-to-word alignment between a source and target text prompts, (iii) demonstrates superior performance of real-world guidance images.

Lastly, SDEdit [27] is another method that applies edits on user provided images using free text prompts. Their method noises the guidance image to an intermediate diffusion step, and then denoises it conditioned on the input prompt. This simple approach leads to impressive results, yet exhibit a tradeoff between preserving the guidance layout and fulfilling the target text. We demonstrate that our method significantly outperforms SDEdit, providing better balance between these two ends.

3. Preliminary

Diffusion models [11, 18, 36, 43] are probabilistic generative models in which an image is generated by progressively removing noise from an initial Gaussian noise image, $\boldsymbol{x}_T \sim \mathcal{N}(0, \mathbf{I})$. These models are founded on two complementary random processes. the *forward* process, in which Gaussian noise is progressively added to a clean image, $\boldsymbol{x}_0$:

$$\boldsymbol{x}_t = \sqrt{\alpha_t} \cdot \boldsymbol{x}_0 + \sqrt{1-\alpha_t} \cdot \boldsymbol{z} \tag{1}$$

where $\boldsymbol{z} \sim \mathcal{N}(0, \mathbf{I})$ and $\{\alpha_t\}$ are the noise schedule.

The *backward* process is aimed at gradually denoising $\boldsymbol{x}_T$, where at each step a cleaner image is obtained. This process is achieved by a neural network $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$ that predicts the added noise $\boldsymbol{z}$. Once trained, each step of the backward process consists of applying $\boldsymbol{\epsilon}_\theta$ to the current $\boldsymbol{x}_t$, and adding a Gaussian noise perturbation to obtain a cleaner $\boldsymbol{x}_{t-1}$.

Diffusion models are rapidly evolving and have been extended and trained to progressively generate images *conditioned* on a guiding signal $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, \boldsymbol{y}, t)$, e.g., conditioning the generation on another image [39], class label [19], or text [22, 28, 34, 36].

In this work, we leverage a pre-trained text-conditioned Latent Diffusion Model (LDM), a.k.a Stable Diffusion [36], in which the diffusion process is applied in the latent space of a pre-trained image autoencoder. The model is based on a U-Net architecture [37] conditioned on the guiding prompt P. Layers of the U-Net comprise a residual block, a self-attention block, and a cross-attention block, as illustrated in Fig. 2 (b). The residual block convolve image features $\boldsymbol{\phi}_t^{l-1}$

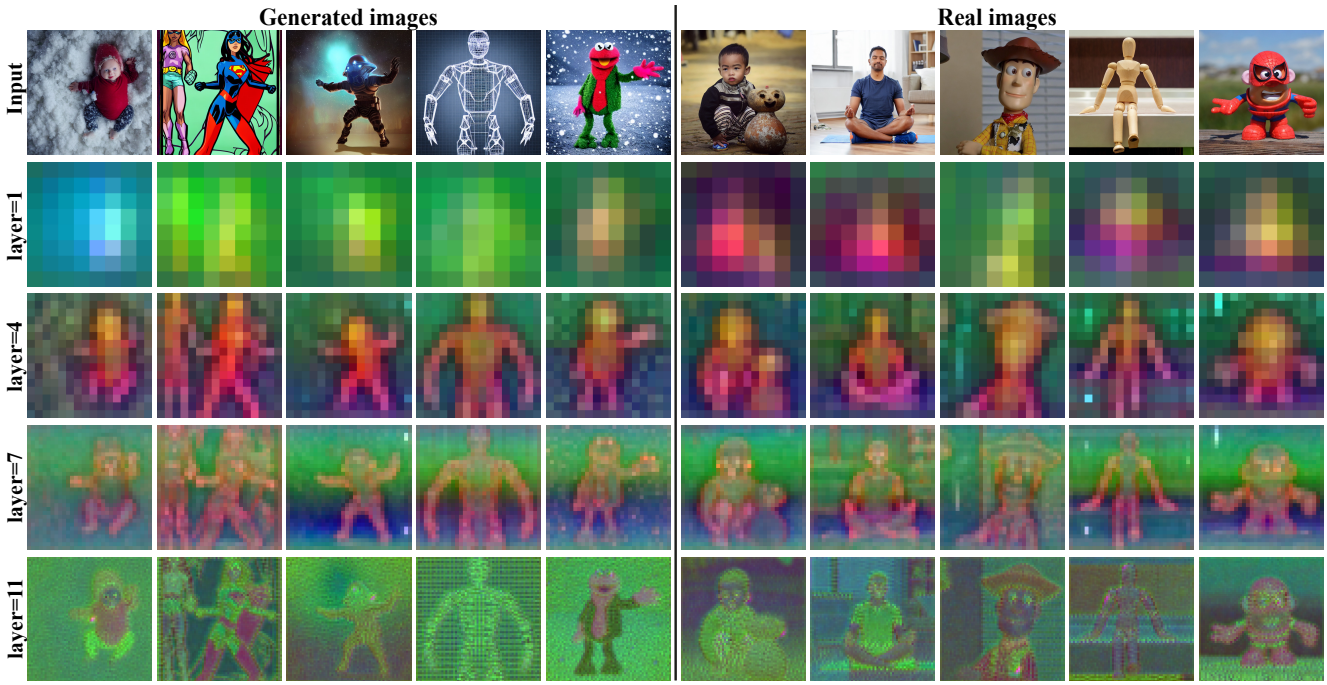

Figure 3. *Visualising diffusion features.* We used a collection of 20 humanoid images (real and generated), and extracted spatial features from different decoder layers, at roughly 50% of the generation process ($t = 540$). For each block, we applied PCA on the extracted features across *all* images and visualized the top three leading components. Intermediate features (layer 4) reveal semantic regions (e.g., legs or torso) that are shared across all images, under large variations in object appearance and the domain of images. Deeper features capture more high-frequency information, which eventually forms the output noise predicted by the model. See SM for additional visualizations.

from the previous layer l–1 to produce intermediate features $\boldsymbol{f}_t^l$. In the self-attention block, features are projected into queries, $\boldsymbol{q}_t^l$, keys, $\boldsymbol{k}_t^l$, and values, $\boldsymbol{v}_t^l$, and the output of the block is given by:

$$\hat{\boldsymbol{f}}_t^l = \boldsymbol{A}_t^l \boldsymbol{v}_t^l \quad \text{where} \quad \boldsymbol{A}_t^l = \text{Softmax}\left(\boldsymbol{q}_t^l {\boldsymbol{k}_t^l}^T\right) \tag{2}$$

This operation allows for long-range interactions between image features. Finally, cross-attention is computed between the spatial image features and the token embedding of the text prompt P.

4. Method

Given an input guidance image I^G and a target prompt P, our goal is to generate a new image I^* that complies with P and preserves the structure and semantic layout of I^G. We consider StableDiffusion [36], a state-of-the-art pre-trained and fixed text-to-image LDM model, denoted by $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, P, t)$. This model is based on a U-Net architecture, as illustrated in Fig. 2 and discussed in Sec. 3.

Our key finding is that fine-grained control over the generated structure can be achieved by manipulating spatial features inside the model during the generation process. Specifically, we observe and empirically demonstrate that: (i) spatial features extracted from intermediate decoder layers encode localized semantic information and are less affected by appearance information, and (ii) the self-attention,

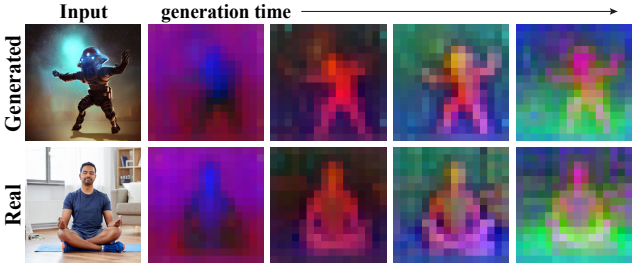

Figure 4. *Diffusion features over generation time-steps.* Visualizing PCA of spatial features of layer $l = 4$ for the humanoid images (Fig. 3). Semantic parts are shared (have similar colors) across images at each time step.

representing the affinities between the spatial features, allows to retain fine layout and shape details.

Based on our findings, we devise a simple framework that extracts features from the generation process of the guidance image I^G and directly injects them along with P into the generation process of I^*, requiring no training or fine-tuning (Fig. 2). Our approach is applicable for both text-generated and real-world guidance images, for which we apply DDIM inversion [44] to get the initial $\boldsymbol{x}_T^G$.

Spatial features. In text-to-image generation, one can use descriptive text prompts to specify various scene and object proprieties, including those related to their shape, pose and

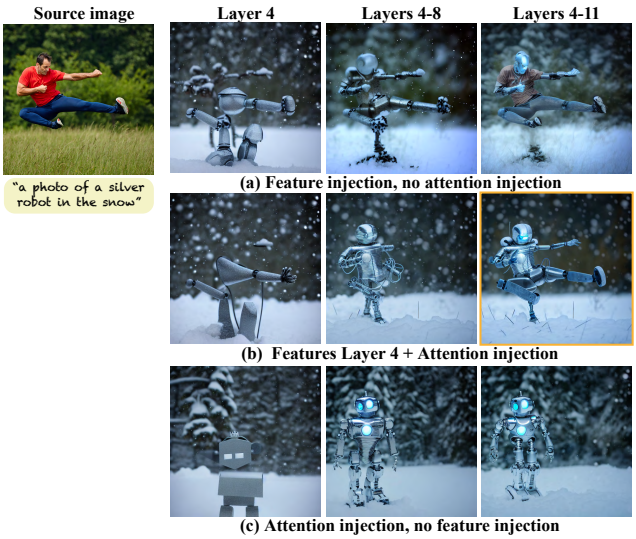

Figure 5. *Ablating features and attention injection.* (a) Features extracted from the guidance image (left) are injected into the generation process of the translated image (guided by a given text prompt). While features at intermediate layers (*Layer 4*) exhibit localized semantic information (Fig. 3), solely injecting these features is insufficient for retaining the guidance structure. Incorporating deeper (and higher resolution) features leads to better structure preservation, but results in appearance leakage from the guidance image to the generated one (*Layers 4-11*). (b) Injecting features only at layer 4 and self-attention maps at higher-resolution layers alleviates this issue. (c) Injecting only self-attention maps restricts the affinities between the features, yet there is no semantic association between the guidance features and the generated ones, resulting in misaligned structure. *The result of our final configuration is highlighted in orange.*

scene layout, e.g., "*a photo of a horse galloping in the forest*". However, the exact scene layout, the shape of the object and its fine-grained pose often significantly vary across generated images from the same prompt under different initial noise $\boldsymbol{x}_T$. This suggests that the diffusion process itself and the resulting *spatial* features have a role in forming such fine-grained spatial information. This hypothesis is strengthened by [5], which demonstrated that semantic part segments can be estimated from spatial features in an unconditional diffusion model.

We opt to gain a better understanding of how such semantic spatial information is internally encoded in $\boldsymbol{\epsilon}_\theta$. To this end, we perform a simple PCA analysis which allows us to reason about the visual properties dominating the high-dimensional features in $\boldsymbol{\epsilon}_\theta$. Specifically, we generated a diverse set of images containing various humanoids in different styles, including both real and text-generated images; sample images are shown in Fig. 3. For each image, we extract features $\boldsymbol{f}_t^l$ from each layer of the decoder at each time step t, as illustrated in Fig. 2(b). We then apply PCA on $\boldsymbol{f}_t^l$ across all images.

Fig. 3 shows the first three principal components for a representative subset of the images across different layers

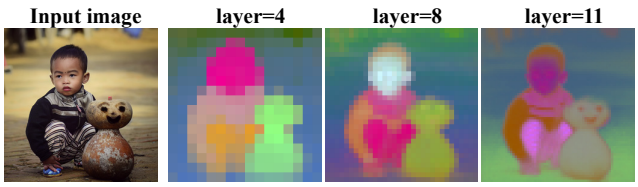

Figure 6. *Self-attention visualization.* Showing 3 leading components of the self-attention matrix $\boldsymbol{A}_t^l$ computed for the input image for three different layers. The principal components are aligned with the layout of the image: similar regions share similar colors. Note how all pixels of the pants share similar color, despite their different appearance in the input image.

and a single time step. As seen, the coarsest and shallowest layer is mostly dominated by foreground-background separation, depicting only a crude blob in the location of the foreground object. Interestingly, we can observe that the intermediate features (layer 4) encode localized semantic information shared across objects from different domains and under significant appearance variations – similar object parts (e.g., legs, torso, head) are depicted in similar colors *across* all images (*layer=4* row in Fig. 3). These properties are consistent across the generation process as shown in Fig. 4. As we go deeper into the network, the features gradually capture more high-frequency low-level information which eventually forms the output noise predicted by the network. Extended feature visualizations can be found in the Supplementary Materials (SM) on our website.

Feature injection. Based on these observations, we now turn to the translation task. Let $\boldsymbol{x}_T^G$ be the initial noise, obtained by inverting I^G using DDIM [44].

Given the target prompt P, the generation of the translated image I^* is carried with the same initial noise, i.e., $\boldsymbol{x}_T^* = \boldsymbol{x}_T^G$; we refer the reader to Appendix B for an analysis and justification of this design choice.

At each step t of the backward process, we extract the guidance features $\{\boldsymbol{f}_t^l\}$ from the denoising step: $\boldsymbol{z}_{t-1}^G = \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t^G, \varnothing, t)$.[1] These features are then injected into the generation of I^*, i.e., in the denoising step of $\boldsymbol{x}_t^*$, we override the resulting features $\{\boldsymbol{f}_t^{*l}\}$ with $\{\boldsymbol{f}_t^l\}$. This operation is expressed by:

$$\boldsymbol{z}_{t-1}^* = \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t^*, P, t\ ;\ \{\boldsymbol{f}_t^l\}) \tag{3}$$

where we use $\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}}(\cdot\ ;\ \{\boldsymbol{f}_t^l\})$ to denote the modified denoising step with the injected features $\{\boldsymbol{f}_t^l\}$. In case of no injection, $\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, P, t\ ;\ \varnothing) = \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, P, t)$.

Fig. 5(a) shows the effect of injecting spatial features $\boldsymbol{f}_t^l$ at increasing layers l. As seen, injecting features only at layer $l = 4$ is insufficient for preserving the structure of the guidance image. As we inject features in deeper layers, the structure is better preserved, yet appearance information is

[1] In the case of a *generated* guidance image, $\boldsymbol{z}_{t-1}^G = \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t^G, P_G, t)$, where P_G is the text used to generate I^G.

leaked into the generated image (e.g., shades of the red t-shirt and blue jeans are apparent in *Layer 4-11*). To achieve a better balance between preserving the structure of I^G and deviating from its appearance, we do not modify spatial features at deep layers, but rather leverage the self-attention layers as discussed below.

Self-attention. Self-attention modules compute the *affinities* $\boldsymbol{A}_t^l$ between the spatial features after linearly projecting them into queries and keys. These affinities have a tight connection to the established concept of self-similarly, which has been used to design structure descriptors by both classical and modern works [3,24,41,46]. This motivates us to consider the attention matrices $\boldsymbol{A}_t^l$ to achieve fine-grained control over the generated content.

Fig. 6, shows the leading principal components of a matrix $\boldsymbol{A}_t^l$ for a given image. As seen, in early layers, the attention is aligned with the semantic layout of the image, grouping regions according to semantic parts. Gradually, higher-frequency information is captured.

Practically, injecting the self-attention matrix is done by replacing the matrix $\boldsymbol{A}_t^l$ in Eq. 2. Intuitively, this operation pulls features close together, according to the affinities encoded in $\boldsymbol{A}_t^l$. We denote this additional operation by modifying Eq. (3) as follows:

$$\boldsymbol{z}_{t-1}^* = \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, P, t; \boldsymbol{f}_t^4, \{\boldsymbol{A}_t^l\}) \tag{4}$$

Fig. 5(b) shows the effect of Eq. (4) for increasing injection layers; the maximal injection layer of $\boldsymbol{A}_t^l$ controls the level of fidelity to the original structure, while mitigating the issue of appearance leakage. Fig. 5(c) demonstrates the pivotal role of the features $\boldsymbol{f}_t^4$. As seen, with only self-attention, i.e., $\boldsymbol{z}_{t-1}^* = \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, P, t; \{\boldsymbol{A}_t^l\})$, there is no semantic association between the original content and the translated one, resulting in large deviations in structure.

Our *plug-and-play diffusion features* framework is summarized in Alg. 1, and is controlled by two parameters: (i) τ_f defines the sampling step t until which $\boldsymbol{f}_t^4$ are injected. (ii) τ_A is the sampling step until which $\boldsymbol{A}_t^l$ are injected. In all our results, we use a default setting where self-attention is injected into all the decoder layers. The exact settings of the parameters are discussed in Sec. 5.

Negative-prompting. In classifier-free guidance [20], the predicted noise $\boldsymbol{\epsilon}$ at each sampling step is given by:

$$\boldsymbol{\epsilon} = w\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, P, t) + (1-w)\,\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, \varnothing, t) \tag{5}$$

where $w > 1$ is the guidance strength. That is, $\boldsymbol{\epsilon}$ is being extrapolated towards the conditional prediction $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, P, t)$ and pushed away from the unconditional one $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, \varnothing, t)$. This increases the fidelity of the denoised image to the prompt P, while allowing to deviate from $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, \varnothing, t)$. Similarly, by replacing the empty prompt in Eq. (5) with a "negative" prompt P_n, we can push away $\boldsymbol{\epsilon}$ from $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, P_n, t)$. For example, using P_n that describes the guidance image, we can steer the denoised image away from the original content.

We use a parameter $\alpha \in [0, 1]$ to balance between neutral and negative prompting:

$$\tilde{\boldsymbol{\epsilon}} = \alpha\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, \varnothing, t) + (1-\alpha)\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, P_n, t) \tag{6}$$

We plug $\tilde{\boldsymbol{\epsilon}}$ instead of $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, \varnothing, t)$ in Eq. (5). That is, $\epsilon = w\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, P, t) + (1-w)\tilde{\boldsymbol{\epsilon}}$.

In practice, we find negative-prompting to be beneficial for handling textureless "primitives" guidance images (e.g., silhouette images). For natural-looking guidance images, it plays a minor role. See Appendix A.1 for more details.

Algorithm 1 Plug-and-Play Diffusion Features

Inputs:
I^G ▷ real guidance image
P ▷ target text prompt
τ_f, τ_A ▷ injection thresholds

$\boldsymbol{x}_T^G \leftarrow$ DDIM-inv(I^G)
$\boldsymbol{x}_T^* \leftarrow \boldsymbol{x}_T^G$ ▷ Starting from same seed
for $t \leftarrow T \dots 1$ **do**
  $\boldsymbol{z}_{t-1}^G, \boldsymbol{f}_t^4, \left\{\boldsymbol{A}_t^l\right\} \leftarrow \boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t^G, \varnothing, t\right)$
  $\boldsymbol{x}_{t-1}^G \leftarrow$ DDIM-samp$\left(\boldsymbol{x}_t^G, \boldsymbol{z}_{t-1}^G\right)$
  if $t > \tau_f$ **then** $\boldsymbol{f}_t^{*4} \leftarrow \boldsymbol{f}_t^4$ **else** $\boldsymbol{f}_t^{*4} \leftarrow \varnothing$
  if $t > \tau_A$ **then** $\boldsymbol{A}_t^{*l} \leftarrow \boldsymbol{A}_t^l$ **else** $\boldsymbol{A}_t^{*l} \leftarrow \varnothing$
  $\boldsymbol{z}_{t-1}^* \leftarrow \hat{\boldsymbol{\epsilon}}_\theta\left(\boldsymbol{x}_t^*, P, t\,;\, \boldsymbol{f}_t^{*4}, \left\{\boldsymbol{A}_t^{*l}\right\}\right)$
  $\boldsymbol{x}_{t-1}^* \leftarrow$ DDIM-samp$\left(\boldsymbol{x}_t^*, \boldsymbol{z}_{t-1}^*\right)$
end for
Output: $I^* \leftarrow \boldsymbol{x}_0^*$

5. Results

We thoroughly evaluate our method both quantitatively and qualitatively on diverse guidance image domains, both real and generated ones, as discussed below. Please see Appendix C for full implementation details of our method.

Datasets. Our method supports versatile text-guided image-to-image translation tasks and can be applied to arbitrary image domains. Since there is no existing benchmark for such diverse settings, we created two new datasets: (i) *Wild-TI2I*, comprises of 148 diverse text-image pairs, 53% of which consists of real guidance images that we gathered from the Web; (ii) *ImageNet-R-TI2I*, a benchmark we derived from the ImageNet-R dataset [15], which comprises of various renditions (e.g., paintings, embroidery, etc.) of ImageNet object classes. To adopt this dataset for our purpose, we manually selected 3 high-quality images from 10 different classes. To generate our image-text examples, we created a list of text templates by defining for each source

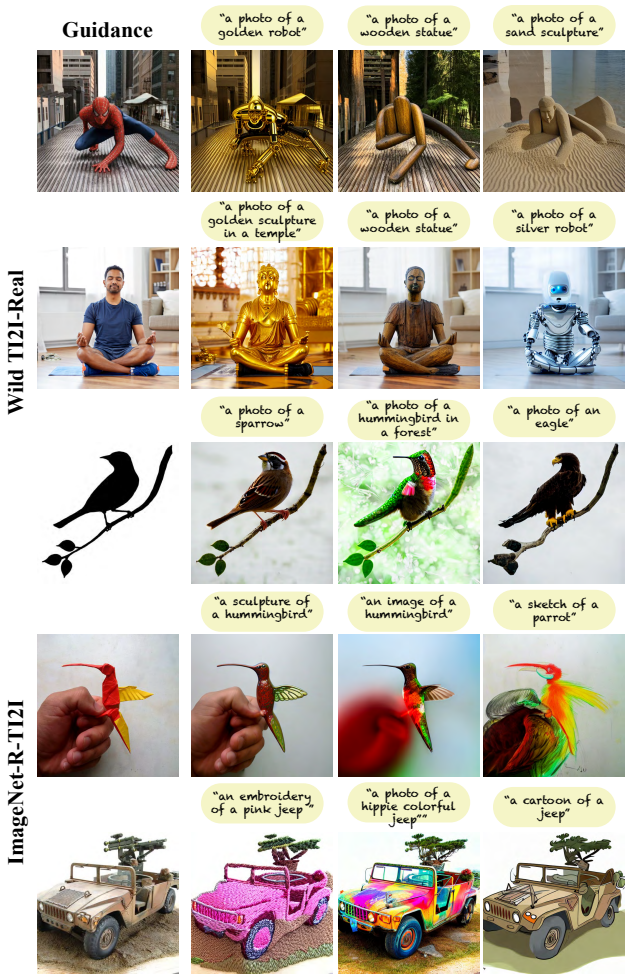

Figure 7. Sample results of our method on image-text pairs from *Wild-TI2I* and *ImageNet-R-TI2I* benchmarks.

class target categories and styles, and automatically sampled their combinations. This results in total of 150 image-text pairs. See Appendix D for full details.

Figs. 1 and 7 show a sample of our results on both real and generated guidance images. Our results show both adherence to the guidance shape and compliance with different target prompts. Our method successfully handles both naturally looking as well as artistic and textureless guidance images.

5.1. Comparison to Prior/Concurrent Work

We focus our comparisons on state-of-the-art baselines that can be applied to diverse text-guided I2I tasks, including: (i) SDEdit [27] under three different noising levels, (ii) P2P [16], (iii) DiffuseIT [25], and (iv) VQGAN-CLIP [9]. We further provide qualitative comparisons to Text2LIVE [4], FlexIT [8] and DiffusionCLIP [22].

We note that P2P requires a source prompt that is word-aligned to the target prompt. Thus, we include a qualitative and quantitative comparison to P2P on our *ImageNet-R-TI2I* benchmark, for which we automatically created aligned source-target prompts using the labels provided for the renditions and object categories. We further include qualitative comparison to a subset of *Wild-TI2I* for which the source and target prompts are aligned. For evaluating P2P on real guidance images, we applied DDIM inversion with the source text as in [16].

Fig. 8 shows sample results of our method compared with the baselines. As seen, our method successfully translates diverse inputs, and works well for both real and generated guidance images. In all cases, our results exhibit both high preservation to the guidance layout and high fidelity to the target prompt. This is in contrast to SDEdit that suffers from an inherent tradeoff between the two – with low noise level, the guidance structure is well preserved but in the expanse of hardly changing the appearance; larger deviation in appearances can be achieved with higher noise level, yet the structure is damaged. VQGAN+CLIP exhibits the same behavior, with overall lower image quality. Similarly, DiffuseIT shows high fidelity to the guiding shape, with little changes to the appearance.

In comparison to P2P, it can be seen that their results on generated guidance images (first 3 rows) depict high fidelity to the target prompt, yet only rough preservation of layout, e.g., results in different number ducks (first row), or deviation from the mouse shape (second row). Furthermore, their method struggles to deviate from the guidance appearance and satisfy the target edit when it is applied to real images (4-8 rows). We speculate that the reason is that DDIM inversion in their case is applied with a source text, requiring using low guidance scale at sampling. In contrast, our method performs DDIM inversion with an empty prompt, allowing us to use arbitrary guidance scale or prompts at generation.

We numerically evaluate these results using two complementary metrics: text-image CLIP cosine similarity to quantify how well the generated images comply with the text prompt (higher is better), and distance between DINO-ViT self-similarity [46], to quantify the extent of structure preservation (lower is better).

As seen in Fig. 9, our method outperforms the baselines by achieving both high preservation of structure (in par with SDEdit w/ very low noising level), and high fidelity to the target text (in par with SDEdit w/ very high noising level). We note that VQGAN-CLIP and DiffuseIT directly use the evaluation metrics as their objective (CLIP loss in [9] and DINO self-similarity in [25]), which explains their respective scores in these metrics.

Extended comparison to P2P [16] To factor out the effect of DDIM inversion, we expand our comparison to P2P on *generated* guidance images. Specifically, we created a *generated-ImageNet-R-TI2I* benchmark by using text prompts expressing the same object classes and renditions described in Appendix D.1.

As seen in Fig. 10, both our method and P2P comply

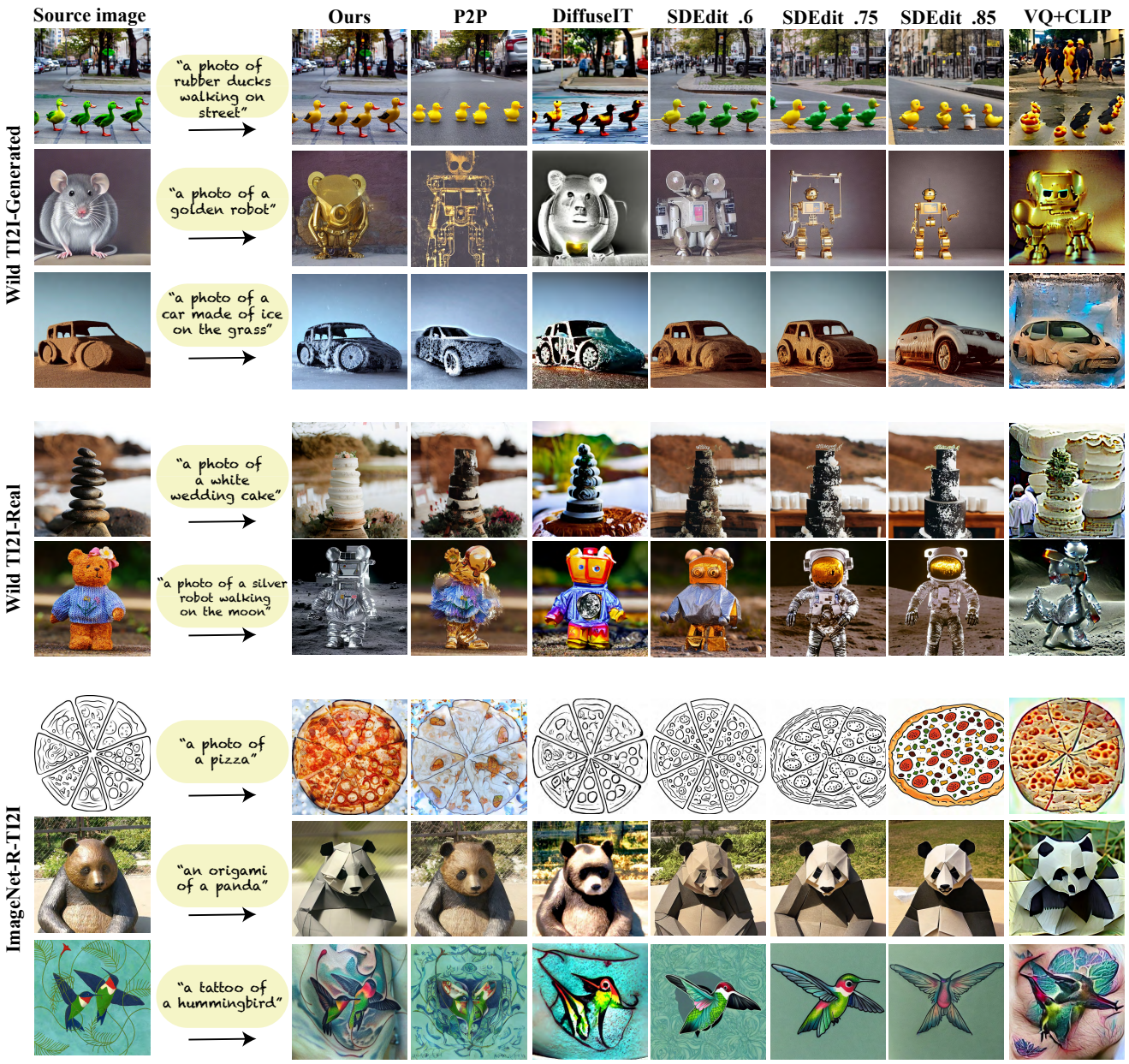

Figure 8. *Comparisons.* Sample results are shown for each of the two benchmarks: *ImageNet-R-TI2I* and *Wild-TI2I*, which includes both real and generated guidance images. From left to right: the guidance image and text prompt, our results, P2P [16], DiffuseIT [25], SDedit [27] with 3 different noising levels, and VQ+CLIP [9].

with the target text. However, P2P often results in large deviations from the guidance structure, especially in cases where multiple prompts edits are applied (last two rows in Fig. 10). Our method demonstrates fine-grained structure preservation across all these examples, while successfully translating multiple traits (e.g., both category and style). These properties are strongly evident by Fig. 9, where our method results in a significantly lower self-similarity distance, even compared to P2P with injecting cross-attention at all timesteps ($t = 1000$).

Additional baselines. Fig. 11 shows qualitative comparisons with: (i) Text2LIVE [4], (ii) DiffusionCLIP [22], and (iii) FlexIT [8]. These methods either fail to deviate from the guidance image or result in noticeable visual artifacts. Since Text2LIVE is designed for layered textural editing, thus it can only "paint" over the guidance image and cannot apply any structural changes necessary to convey the target edit (e.g. dog to venom, church to Asian tower). Moreover, Text2LIVE does not leverage a strong generative prior, hence often results in low visual quality. FlexIT often fails

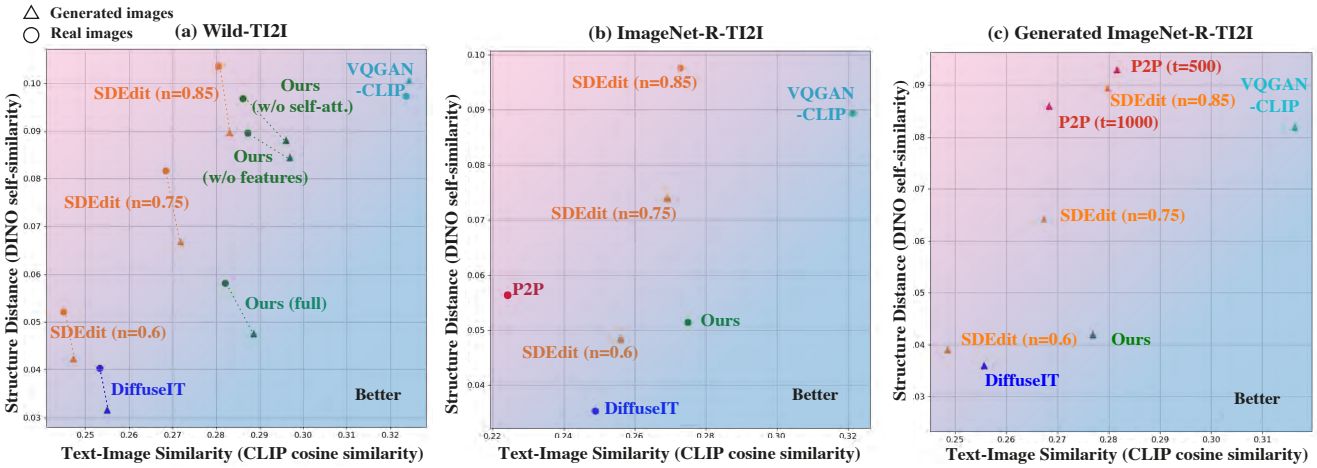

Figure 9. *Quantitative evaluation*. We measure CLIP cosine similarity (higher is better) and DINO-ViT self-similarity distance (lower is better) to quantify the fidelity to text and preservation of structure, respectively. We report these metrics on three benchmarks: (a) *Wild-TI2I* for which an ablation of our method is included, (b) *ImageNet-R-TI2I*, and (c) *Generated-ImageNet-R-TI2I*. Note that we could compare to P2P only for (b) and (c) due to their prompts restriction. All baselines struggle to achieve both low structure distance and a high CLIP score. Our method exhibit a better balance between these two ends across all benchmarks.

Figure 10. *Comparison to P2P on generated* ImageNet-R-TI2I *benchmark.* While P2P results demonstrate high fidelity to the the target text, there are noticeable deviation from the guidance structure, especially in cases of multiple word swaps (last two rows). Across all examples, our results adhere to the target edit while preserving the guidance scene layout and object pose.

to deviate from the guidance content, which may be caused to their regularization that encourages the guidance and output images to match in LPIPS sense. We also note that DiffusionCLIP requires fine-tuning an unconditional diffusion model for each target edit on a set of 30+ images from a single domain (e.g. dog faces, churches).

5.2. Ablation

We ablate our key design choices by evaluating our performance for the following cases: (i) w/o spatial features injection (w/o features), (ii) w/o self-attention injection. The metrics are reported in Fig. 9(a) and a representative example is shown in Fig. 5. The results demonstrate that both features and self-attention are critical for structure preservation – the features provide a semantic association between the original and translated content, while self-attention is essential for maintaining this association and capturing finer structural information. Further ablations can be found in Appendix A and Tab. 1.

6. Discussion and Conclusion

We presented a new framework for diverse text-guided image-to-image translation, founded on new insights about the internal representation of a pre-trained text-to-image diffusion model. Our method, based on simple manipulation of features, outperforms existing baselines, achieving a significantly better balance between preserving the guidance layout and deviating from its appearance. As for limitations, our method relies on the semantic association between the original and translated content in the diffusion feature space. Thus, it does not work well on detailed label segmentation masks where regions are colored arbitrarily (Fig. 12). In addition, we are relying on DDIM inversion, which we found to work well in most of our examples. Nevertheless, we observed that for textureless "minimal" images, DDIM may occasionally result in a latent that encodes dominant low-frequency appearance information, in

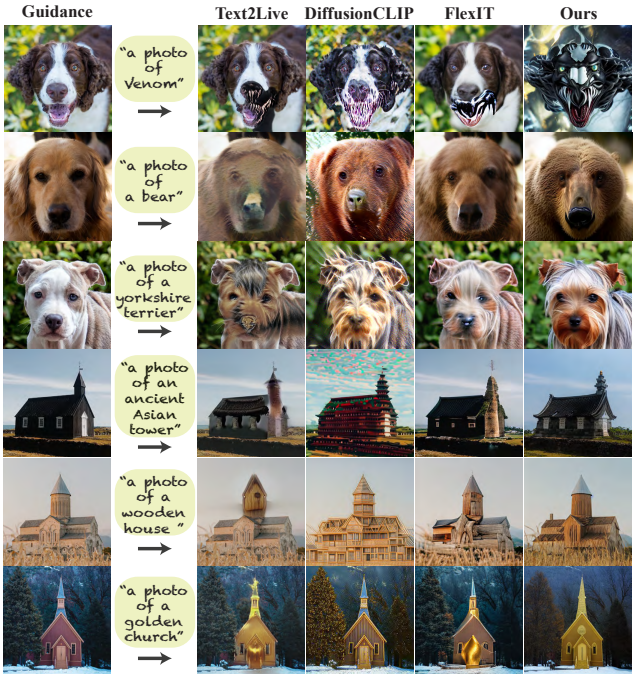

Figure 11. Qualitative comparisons to additional baselines: Text2LIVE [4], DiffusionCLIP [22], FlexIT [8]. These methods fail to deviate from the structure for matching the target prompt, or create undesired artifacts.

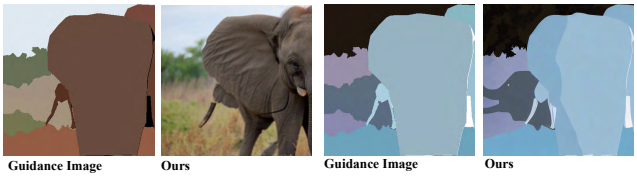

Figure 12. *Limitations.* Our method fails when there is no semantic association between the guidance content and the target text. Thus, it does not perform well on solid segmentation masks with arbitrary colors.

which case some appearance information would leak into our results. We believe that our work demonstrates the yet unrealized potential of the rich and powerful feature space spanned by pre-trained text-to-image diffusion models. We hope it will motivate future research in this direction.

Acknowledgments: We thank Omer Bar-Tal for his insightful comments and discussion. This project received funding from the Israeli Science Foundation (grant 2303/20), the Carolito Stiftung, and the NVIDIA Applied Research Accelerator Program. Dr. Bagon is a Robin Chemers Neustein AI Fellow.

References

[1] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Restyle: A residual-based stylegan encoder via iterative refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2

[2] 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*, 2022. 2

[3] Shai Bagon, Ori Brostovski, Meirav Galun, and Michal Irani. Detecting and sketching the common. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010. 6

[4] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2live: Text-driven layered image and video editing. In *European Conference on Computer Vision*. Springer, 2022. 2, 7, 8, 10

[5] Dmitry Baranchuk, Andrey Voynov, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. In *International Conference on Learning Representations*, 2022. 5

[6] David Bau, Alex Andonian, Audrey Cui, YeonHwan Park, Ali Jahanian, Aude Oliva, and Antonio Torralba. Paint by word. *arXiv preprint arXiv:2103.10951*, 2021. 2

[7] Tao Chen, Ming-Ming Cheng, Ping Tan, Ariel Shamir, and Shi-Min Hu. Sketch2photo: internet image montage. *ACM Trans. Graph.*, 2009. 2

[8] Guillaume Couairon, Asya Grechka, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. FlexIT: Towards flexible semantic image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 7, 8, 10

[9] 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. In *European Conference on Computer Vision*, pages 88–105. Springer, 2022. 7, 8

[10] Tali Dekel, Chuang Gan, Dilip Krishnan, Ce Liu, and William T Freeman. Sparse, smart contours to represent and edit images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2

[11] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 2021. 3

[12] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. In *European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 3

[13] 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. 3

[14] Rinon Gal, Or Patashnik, Haggai Maron, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 2022. 2

[15] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *ICCV*, 2021. 6

[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. 2, 3, 7, 8

[17] Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless, and David Salesin. Image analogies. In Lynn Pocock, editor, *ACM Trans. on Graphics (Proceedings of ACM SIGGRAPH)*, 2001. 2

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 2020. 3

[19] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23:47–1, 2022. 3

[20] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 6

[21] 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. *arXiv preprint arXiv:2210.09276*, 2022. 3

[22] 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*, 2022. 2, 3, 7, 8, 10

[23] Kunhee Kim, Sanghun Park, Eunyeong Jeon, Taehun Kim, and Daijin Kim. A style-aware discriminator for controllable image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[24] Nicholas Kolkin, Jason Salavon, and Gregory Shakhnarovich. Style transfer by relaxed optimal transport and self-similarity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. 6

[25] Gihyun Kwon and Jong Chul Ye. Diffusion-based image translation using disentangled style and content representation. *arXiv preprint arXiv:2209.15264*, 2022. 2, 7, 8

[26] Xingchao Liu, Chengyue Gong, Lemeng Wu, Shujian Zhang, Hao Su, and Qiang Liu. Fusedream: Training-free text-to-image generation with improved clip+ gan space optimization. *arXiv preprint arXiv:2112.01573*, 2021. 2

[27] 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. 3, 7, 8

[28] 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, 3

[29] Taesung Park, Alexei A. Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive learning for unpaired image-to-image translation. In *European Conference on Computer Vision*, 2020. 2

[30] Taesung Park, Jun-Yan Zhu, Oliver Wang, Jingwan Lu, Eli Shechtman, Alexei Efros, and Richard Zhang. Swapping autoencoder for deep image manipulation. *Advances in Neural Information Processing Systems*, 2020. 2

[31] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2

[32] Lara Raad and Bruno Galerne. Efros and freeman image quilting algorithm for texture synthesis. *Image Process. Line*, 2017. 2

[33] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings the International Conference on Machine Learning (ICML)*, 2021. 2

[34] 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, 3

[35] 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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021. 2

[36] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 1, 2, 3, 4

[37] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 2015. 3

[38] 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. 3

[39] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, 2022. 2, 3

[40] 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, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 1, 3

[41] Eli Shechtman and Michal Irani. Matching local self-similarities across images and videos. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007. 6

[42] Yichang Shih, Sylvain Paris, Frédo Durand, and William T. Freeman. Data-driven hallucination of different times of day from a single outdoor photo. *ACM Trans. Graph.*, 2013. 2

[43] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*. PMLR, 2015. 3

[44] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2020. 4, 5

[45] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics (TOG)*, 40(4):1–14, 2021. 2

[46] Narek Tumanyan, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Splicing vit features for semantic appearance transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 2, 6, 7

[47] Yael Vinker, Eliahu Horwitz, Nir Zabari, and Yedid Hoshen. Image shape manipulation from a single augmented training sample. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13769–13778, 2021. 2

[48] Tengfei Wang, Ting Zhang, Bo Zhang, Hao Ouyang, Dong Chen, Qifeng Chen, and Fang Wen. Pretraining is all you need for image-to-image translation. In *arXiv*, 2022. 2, 3

[49] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, 2017. 2

A. Ablations

A.1. Negative-prompting.

We qualitatively and quantitatively ablate the effect of negative prompting (see Sec. 4 of the main paper). Tab. 1 compares our metrics w/ and w/o negative prompting (bottom row and second row), using our *Wild-TI2I* and *ImageNet-R-TI2I* benchmarks. The results indicate that the usage of negative prompting (bottom row) leads to slightly larger deviation from the guidance image (higher LPIPS distance between I^G and I^*), while introducing only a minor reduction in structure preservation. Sample results of this ablation are shown in Fig. 13, where we can also notice that negative prompting has a larger effect for "primitive images", i.e, simple "textureless" images such as silhouettes (top two rows) than natural guidance images.

A.2. Injected features.

Our method injects features to the *decoder* block, in a specific layer which we observed to capture localized semantic information. To complete our analysis, we extend our PCA feature visualization to include both the decoder and encoder features. As seen in Fig. 15, the encoder resemble a mirrored trend to the decoder: the encoder features start with high frequency noise (layer 1), which is gradually transformed into cleaner features that depict lower-frequency content throughout the layers. Nevertheless, localized semantic information is overall less apparent in the encoder's features. To numerically evaluate this, we consider a modified version of our method where the encoder features from layer 7, which resemble some semantic information, are additionally injected. As seen in Tab. 1, this combination results in worse CLIP score in all data-sets and smaller LPIPS deviation from the guidance image on most sets (first row).

B. Initial noise $\boldsymbol{x}_T$ and spatial features

We observed that in order for our method to work, the initial noise used to generate the translated image $\boldsymbol{x}_T^*$ has to match the initial guidance noise $\boldsymbol{x}_T^G$. Since we inject features into the decoder from the very first step of the backward process, this dependency on the random seed can only be explained by the encoder features at $t = T$, denoted by $\boldsymbol{f}_t^{e_l}$. Recall that these features depend on both $\boldsymbol{x}_T^*$ and the target prompt P. This raises the question: why $\boldsymbol{f}_T^{e_l}$ originated from $\boldsymbol{x}_T^* = \boldsymbol{x}_T^G$ and an arbitrary text prompt P, allow our method to work? We hypothesize that in $t = T$ the target prompt has little effect on the encoder features $\boldsymbol{f}_T^{e_l}$, thus the injected decoder features $\boldsymbol{f}_T^4$ comply with the encoder features. In contrast, changing the seed results in a mismatch between $\boldsymbol{f}_T^4$ and $\boldsymbol{f}_T^{e_l}$. This may be surprising since images generated from the same seed under different text prompts may dramatically differ from one another (see Fig. 14 bottom).

To validate this hypothesis, we performed an analysis that shows that features formed from the same $\boldsymbol{x}_T$ under arbitrary prompts $\{P_i\}$ are significantly more correlated than

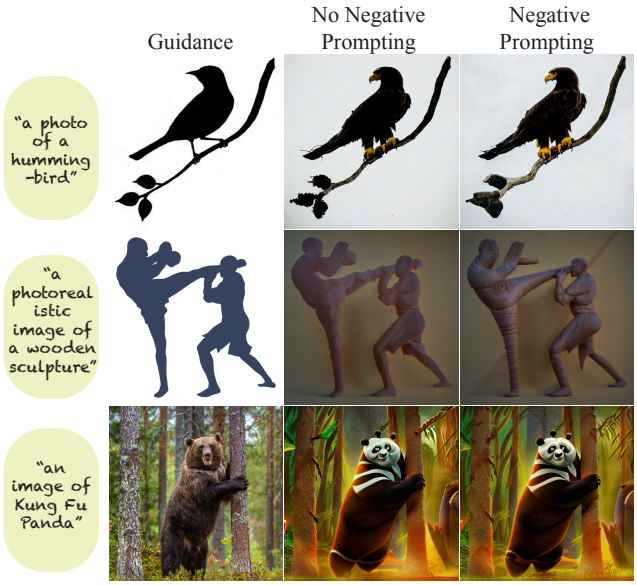

Figure 13. *Qualitative ablation of negative prompting.* The effect of negative prompting is most meaningful on textureless guidance images. In the case of realistic images (row 3) it has a minor effect.

those generated under the same prompt P with different seeds $\{\boldsymbol{x}_T^i\}$. Specifically, we used 10 different prompts and 10 different seeds to generate 100 images, using all combinations. We considered the images generated under: (i) the same prompt across different seeds, and (ii) the same seed across different prompts. Total of 20 sets of 10 images each. We then measured the variance between the feature maps within each of these sets. In Fig. 14 (top), we report these variances (averaged across spatial location) as a function of the encoder layer l. As seen, changing the initial seed, for any fixed prompt, results in significantly higher variance across features for all layers l compared to fixing the seed and changing only the prompt. These findings validate our hypothesis and support our method's dependency on the initial seed.

C. Implementation Details

We use Stable Diffusion as our pre-trained text-to-image model; we use the *StableDiffusion-v-1-4* checkpoint provided via official HuggingFace webpage.

In all of our experiments, we use DDIM deterministic sampling with 50 steps. In the case of real guidance images, we perform deterministic DDIM inversion with 1000 forward steps and then perform deterministic DDIM sampling with 1000 backward steps. Our translation results are performed with 50 sampling steps, thus we extract features only at these steps. We set our default injection thresholds to: $\tau_A = 25$, $\tau_f = 40$ out of the 50 sampling steps; for primitive guidance image, we found that $\tau_A = \tau_f = 25$ to work better.

During translation, we set the classifier-free guidance scale for real and generated guidance images to 15.0 and 7.5, respectively. The use of negative prompting is con-

	Wild-TI2I Real			Wild-TI2I Generated			ImageNet-R-TI2I		
	Self-Sim ↓	CLIP ↑	LPIPS ↑	Self-Sim ↓	CLIP ↑	LPIPS ↑	Self-Sim ↓	CLIP ↑	LPIPS ↑
w/ encoder-feat-7	0.058	0.280	0.527	0.035	0.264	0.453	0.05	0.273	0.458
w/o neg. prompt	0.052	0.281	0.490	0.033	0.275	0.441	0.048	0.274	0.451
w/o feat.	0.090	0.288	0.584	0.084	0.297	0.633	0.076	0.281	0.534
w/o self-attn.	0.097	0.286	0.597	0.090	0.295	0.657	0.089	0.278	0.564
Our method	0.058	0.282	0.521	0.048	0.289	0.542	0.051	0.275	0.462

Table 1. **Quantitative evaluation on WILD-Real benchmark**. We evaluate the distance in DINO-ViT self-similarity for structure preservation, CLIP score for target text faithfulness and LPIPS distance for deviation from the guidance image. We ablate the features injection, self-attention injection, negative prompting, and additional feature injection in the encoder blocks. We report these scores on our three text guided I2I benchmarks. The configuration reported in the main paper (encoder features + self-attention injection and negative prompting) is the best balance between the three metrics across the datasets.

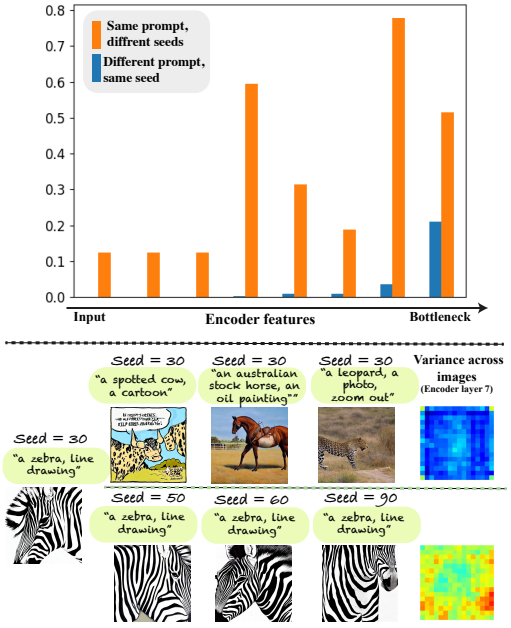

Figure 14. *Measuring features' variance: different prompts vs. different seeds.* We consider 10 different seeds and 10 different prompts, and generate 100 images using all combinations. We extract features from the encoder (which are skipped to the decoder) for all generated images at $t = T$, and compute the variance over features originated from: (i) the same prompt across different seeds, and (ii) the same seed across different prompts. The estimated variance under each of these settings is plotted in orange and blue bars, respectively. We observe that although the generated images using (ii) do not exhibit shared visual properties, their features are correlated (low variance). In contrast, images generated using (i) are more visually similar, yet their features are significantly less correlated.

trolled via a hyperparameter α used to interpolate between the predicted noise $\epsilon_\theta(N)$ and the predicted noise $\epsilon_\theta(\varnothing)$, as described in Sec. 4. We set an initial value α_0 in the first sampling step, and then gradually decrease it. For real guidance images, we set $\alpha_0 = 1.0$ and use a linear scheduler $\alpha(t) = \alpha_0 - t$. For generated guidance images, we set $\alpha_0 = 0.75$. We use the linear scheduler for realistic generated images, and for generated images that are primitive/textureless, we use an exponential scheduler $\alpha(t) = e^{-6 \cdot t}$.

For running the competitors, we use their official implementations: Prompt-to-Prompt, DiffuseIT, DiffusionCLIP, Text2LIVE, FlexIT. For running SDEdit on StableDiffusion, we use the implementation available in the Stable Diffusion official repo. For running VQGAN-CLIP, we used the publicly available repo.

D. Benchmarks

D.1. ImageNet-R-TI2I benchmark.

To test our method on a wide range of guidance images, we turn to Image-Net-R [15], a dataset that contains various renditions of 200 classes from ImageNet. We manually select 10 classes: "`Castle`", "`Cat`", "`Goldfish`", "`Hummingbird`", "`Husky`", "`Jeep`", "`Panda`", "`Penguin`", "`Pizza`", "`Violin`". To avoid low-quality images, we manually selected 3 images per class, totaling 30 guidance images.

Additionally, we automatically created 5 target prompts per image. All the prompts share the same template: "≪rendition≫ of a ≪class≫", e.g. "a painting of a jeep". ≪rendition≫ is one of the existing renditions in the real ImageNet-R data-set: "`an art`", "`a cartoon`", "`a graphic`", "`a deviantart`", "`a painting`", "`a sketch`", "`a graffiti`", "`an embroidery`", "`an origami`", "`a pattern`", "`a sculpture`", "`a tattoo`", "`a toy`", "`a video-game`", "`a photo`", "`an image`". For two (out of five) target prompts per image, we changed the correct ≪class≫ to another object class randomly sampled from 5 related classes (to avoid completely unreasonable translations such as penguin → jeep).

Overall, our *ImageNet-R-TI2I* benchmark contains 150 image-text pairs: 30 guidance images with 5 target prompts each.

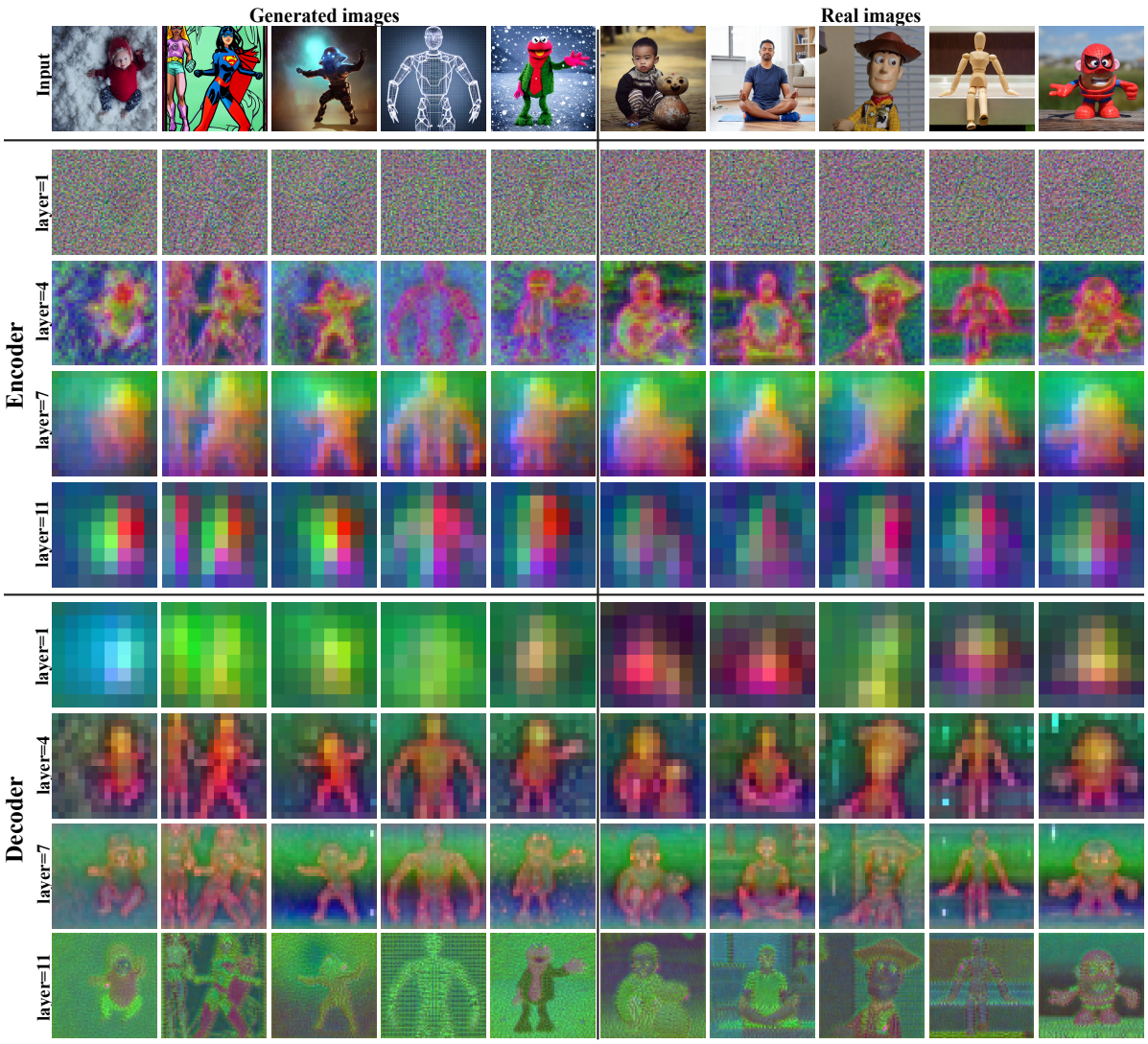

Figure 15. *Visualizing diffusion features for both encoder and decoder.* Extending the visualization of Fig. 3 in the main paper to include features from encoder blocks of the U-Net at time $t = 540$ (top part).

D.2. Wild TI2I benchmark.

We collected a diverse dataset of 148 text-image pairs, containing different object classes (people, animals, food, landscapes) in different renditions (realistic images, drawings, solid masks, sketches and illustrations) with different levels of semantic details. 53% of the examples consists of real guidance images that we gathered from the Web, and the rest are generated from text.

We will publicly release our benchmarks and code for academic use.