Training-free Content Injection using h-space in Diffusion Models

Jaeseok Jeong* Mingi Kwon* Youngjung Uh†
Yonsei University
Seoul, Rej
{jete_jeong,kwor

Abstract

Diffusion models (DMs) synthesize high-quality images in various domains. However, controlling their generative process is still hazy because the intermediate variables in the process are not rigorously studied. Recently, the bottleneck feature of the U-Net, namely $\boldsymbol{h}$-space, is found to convey the semantics of the resulting image. It enables StyleCLIP-like latent editing within DMs. In this paper, we explore further usage of $\boldsymbol{h}$-space beyond attribute editing, and introduce a method to inject the content of one image into another image by combining their features in the generative processes. Briefly, given the original generative process of the other image, 1) we gradually blend the bottleneck feature of the content with proper normalization, and 2) we calibrate the skip connections to match the injected content. Unlike custom-diffusion approaches, our method does not require time-consuming optimization or fine-tuning. Instead, our method manipulates intermediate features within a feed-forward generative process. Furthermore, our method does not require supervision from external networks. Project page: https://curryjung.github.io/DiffStyle/

1. Introduction

Diffusion models (DMs) have gained recognition in various domains due to their remarkable performance in random generation [29, 68]. Naturally, researchers and practitioners seek ways to control the generative process. In this sense, text-to-image DMs provide a way to reflect a given text for generating diverse images using classifier-free guidance [3, 20, 53, 59, 60, 65]. In the same context, image guidance synthesizes random images that resemble the reference images that are given for the guidance [1, 8, 13, 48, 49]. On the other hand, deterministic DMs, such as ODE samplers, have been used to edit real images while preserving most of the original image [32, 45, 47, 68, 69]. DiffusionCLIP [39] and Imagic [38] first embed an input image into noise and finetune DMs for editing. While these approaches provide some control for DMs, the intermediate variables in the process are not rigorously studied, as opposed to the latent space of generative adversarial networks (GANs). Critically, previous studies do not provide insight into the intermediate features of DMs.

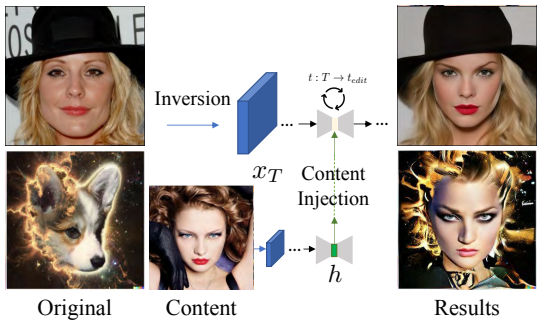


Figure 1. **Overview of InjectFusion.** During the content injection, the bottleneck feature map is recursively injected during the sampling process started from the inverted $\boldsymbol{x}_T$ of images. The target content is reflected in the result images while preserving the original images.

Recently, Asyrp [43] discovered a hidden latent space of pretrained DMs located at the bottleneck of the U-Net, named $\boldsymbol{h}$-*space*. Shifting the latent feature maps along a certain direction enables semantic attribute changes, such as adding a smile. When combined with deterministic inversion, it allows real image manipulation using a pretrained frozen DM. However, its application is limited to changing certain attributes, and it does not provide as explicit operations as in GANs, such as replacing feature maps.

In this paper, we explore further usage of $\boldsymbol{h}$-*space* beyond attribute editing and introduce a method that injects the content of one image into another image. Figure 1 overviews our new generative process for content injection. It starts by inverting two images into noises. Instead of running generative processes from them individually, we set one generative process as an original and inject the bottleneck features of the other generative process. As the bottleneck features convey the semantics of the resulting image, it is equivalent to injecting the content. The injection happens

*These authors contributed equally to this work
†corresponding author

recursively along the timesteps.

However, unlike GAN, DMs are usually designed with U-Net which has skip connections. If one directly changes the bottleneck only, it distorts the relation between the skip connection and the bottleneck. Our method, named InjectFusion, treats this problem with two methods. 1) InjectFusion blends the content bottleneck to the original bottleneck gradually along the generative process. The blended feature is properly normalized to keep the correlation with the skip connections. 2) InjectFusion calibrates the latent $\boldsymbol{x}_t$ directly to preserve the correlation between $\boldsymbol{h}$-*space* and skip connections. This calibration is not only able to be used for InjectFusion but also for any other feature manipulation methods.

InjectFusion enables content injection using pretrained unconditional diffusion models without any training. To the best of our knowledge, our method is the first to tackle these applications without additional training or extra networks. It provides convenience for users to experiment with existing pretrained DMs. In the experiments, we analyze the effect of individual components and demonstrate diverse use cases. Although there is no comparable method with a perfect fit, we compare InjectFusion against closely related methods, including DiffuseIT [42].

2. Background

In this section, we review various approaches for controlling the results of DMs and cover preliminaries.

2.1. Diffusion models and controllability

After DDPMs [29] provide a universal approach for DMs, Song et al. [69] unify DMs with score-based models in SDEs. Subsequent works have focused on improving generative performance of DMs [9, 34, 54, 68, 74]. Other works attempt to manipulate the resulting images by replacing latent variables in DMs and generating random images with the color or strokes of the desired images [8, 49] but they fall short of content injection.

Recently, some works have proposed to control DMs by manipulating latent features in DMs. Asyrp [43] considers the bottleneck of U-Net as a semantic latent space ($\boldsymbol{h}$-*space*) through the asymmetric reverse process. However, it focuses only on semantic editing, e.g., making a person smile. Plug-and-Play [71] injects an intermediate feature in DMs to provide structural guidance. However, it does not consider the correlation between the skip connection and the feature. Similarly, injecting self-attention features enables semantic image editing by retaining structure or objects/characters [6, 71]. However, they should rely on text prompts to determine the destinations, which is often vague and insufficient in describing abstract and fine-grained visual concepts.

ADM [18] introduces gradient-guidance to control generative process [1, 46, 53, 66], but it does not allow detailed manipulation. The guidance controls the reverse process of DMs and can be extended to image-guided image translation without extra training but it depends on the external model (e.g. DINO ViT [7]) and struggles to overcome a huge disparity in color distribution. [42]

2.2. Injecting contents from exemplar images

For given exemplar images with an object, Dreambooth variants [41, 61] fine-tune pretrained DMs to generate different images containing the object. Instead of fine-tuning the whole model, LoRA variants [44, 64, 80] introduce auxiliary networks or fine-tune a tiny subset of the model. As opposed to modifying models, textual inversion variants [21, 26] embed visual concepts into text embeddings for the same task. However, these methods require extra training or optimization steps to reflect the exemplars. On the other hand, our method does not require training or optimization but works on frozen pretrained models. In addition, while these methods rely on the form of text to reflect the exemplars, our method directly works on the intermediate features in the model.

ControlNet variants [44, 52, 80] can inject structural contents as a condition in the form of an edge map, segmentation mask, pose, and depth map. However, the control is limited to structure and shape. Our method preserves most of the content in the exemplar.

Some works utilize the inversion capability of DMs [5, 6, 27, 51, 71], which enables injecting contents during the reconstruction process. However, most of them rely on language to insert the contents.

2.3. Style transfer

Recently, neural style transfer [22] has evolved with the advancement of DMs and neural network architecture [19]. Some style transfer methods leverage a style encoder [62] to enable pretrained DMs to be conditioned on the visual embedding from style reference images [63, 70]. StyleDrop [67] achieves outstanding performance in extracting style features from visual examples but how to control content and shape has not been provided. Since it is vision transformer [19], universal spatial control approach of DMs [80] cannot be adapted

Exploiting external segmentation mask models and explicit appearance encoder enables decomposing the structure and appearance in [24] for style transfer, but it requires training DMs and the encoder from scratch.

2.4. Denoising Diffusion Implicit Model (DDIM)

Diffusion models learn the distribution of data by estimating denoising score matching with $\boldsymbol{\epsilon}_t^\theta$. In the denoising diffusion probabilistic model (DDPM) [29], the forward

process is defined as a Markov process that diffuses the data through parameterized Gaussian transitions. DDIM [68] redefines DDPM as $q_\sigma(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}(\sqrt{\alpha_{t-1}}\boldsymbol{x}_0 + \sqrt{1-\alpha_{t-1}-\sigma_t^2} \cdot \frac{\boldsymbol{x}_t - \sqrt{\alpha_t}\boldsymbol{x}_0}{\sqrt{1-\alpha_t}}, \sigma_t^2 \boldsymbol{I})$, where $\{\beta_t\}_{t=1}^T$ is the variance schedule and $\alpha_t = \prod_{s=1}^{t}(1-\beta_s)$. Accordingly, the reverse process becomes:

$$\boldsymbol{x}_{t-1} = \sqrt{\alpha_{t-1}} \underbrace{\left(\frac{\boldsymbol{x}_t - \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_t^\theta(\boldsymbol{x}_t)}{\sqrt{\alpha_t}} \right)}_{\text{"predicted } \boldsymbol{x}_0\text{"}} + \underbrace{\sqrt{1-\alpha_{t-1}-\sigma_t^2} \cdot \boldsymbol{\epsilon}_t^\theta(\boldsymbol{x}_t)}_{\text{"direction pointing to } \boldsymbol{x}_t\text{"}} + \underbrace{\sigma_t \boldsymbol{z}_t}_{\text{random noise}}, \tag{1}$$

where $\sigma_t = \eta\sqrt{(1-\alpha_{t-1})/(1-\alpha_t)}\sqrt{1-\alpha_t/\alpha_{t-1}}$. When $\eta = 0$, the process becomes deterministic.

2.5. Asymmetric reverse process (Asyrp)

Asyrp [43] introduces the asymmetric reverse process for using ***h-space*** as a semantic latent space. ***h-space*** is the bottleneck of U-Net, which is distinguished from the latent variable $\boldsymbol{x}_t$. For real image editing, they invert $\boldsymbol{x}_0 \sim p_{real}(\boldsymbol{x})$ into $\boldsymbol{x}_T$ through the DDIM forward process, and generate $\tilde{\boldsymbol{x}}_0$ using the new $\tilde{\boldsymbol{h}}_t$ in the modified DDIM reverse process. They use an abbreviated version of Eq. (1). We follow the notation of Asyrp throughout this paper:

$$\boldsymbol{x}_{t-1} = \sqrt{\alpha_{t-1}}\,\mathbf{P}_t(\boldsymbol{\epsilon}_t^\theta(\boldsymbol{x}_t)) + \mathbf{D}_t(\boldsymbol{\epsilon}_t^\theta(\boldsymbol{x}_t)) + \sigma_t \boldsymbol{z}_t, \tag{2}$$

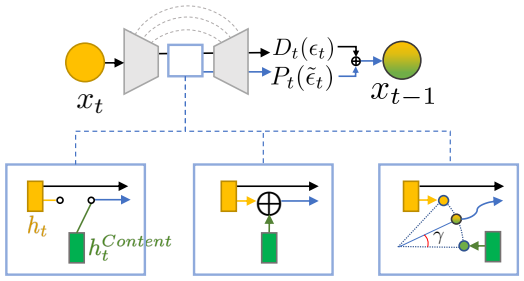
where $\mathbf{P}_t(\boldsymbol{\epsilon}_t^\theta(\boldsymbol{x}_t))$ denotes the predicted $\boldsymbol{x}_0$ and $\mathbf{D}_t(\boldsymbol{\epsilon}_t^\theta(\boldsymbol{x}_t))$ denotes the direction pointing to $\boldsymbol{x}_t$. We abbreviate $\mathbf{P}_t(\boldsymbol{\epsilon}_t^\theta(\boldsymbol{x}_t))$ as $\mathbf{P}_t$ and $\mathbf{D}_t(\boldsymbol{\epsilon}_t^\theta(\boldsymbol{x}_t))$ as $\mathbf{D}_t$ when the context clearly specifies the arguments. Following Asyrp, we omit $\sigma_t \boldsymbol{z}_t$ when $\eta = 0$. Then, Asyrp becomes:

$$\tilde{\boldsymbol{x}}_{t-1} = \sqrt{\alpha_{t-1}}\,\mathbf{P}_t(\boldsymbol{\epsilon}_t^\theta(\tilde{\boldsymbol{x}}_t|\tilde{\boldsymbol{h}}_t)) + \mathbf{D}_t(\boldsymbol{\epsilon}_t^\theta(\tilde{\boldsymbol{x}}_t|\boldsymbol{h}_t)) + \sigma_t \boldsymbol{z}_t, \tag{3}$$

where $\tilde{\boldsymbol{x}}_T = \tilde{\boldsymbol{x}}_T$ and then $\boldsymbol{\epsilon}_t^\theta(\tilde{\boldsymbol{x}}_t|\tilde{\boldsymbol{h}}_t)$ replaces the original U-Net feature maps $\boldsymbol{h}_t$ with $\tilde{\boldsymbol{h}}_t$. They show that the modification of ***h-space*** in both $\mathbf{P}_t$ and $\mathbf{D}_t$ brings a negligible change in the results. Therefore, the key idea of Asyrp is to modify only ***h-space*** of $\mathbf{P}_t$ while preserving $\mathbf{D}_t$.

Quality boosting, introduced by Asyrp, is a stochastic noise injection when the image is almost determined. It enhances fine details and reduces the noise of images while preserving the identity of the image. The whole process of Asyrp is as follows.

$$\tilde{\boldsymbol{x}}_{t-1} = \begin{cases} \sqrt{\alpha_{t-1}}\,\mathbf{P}_t(\boldsymbol{\epsilon}_t^\theta(\tilde{\boldsymbol{x}}_t|\tilde{\boldsymbol{h}}_t)) + \mathbf{D}_t & \text{if } T \geq t \geq t_{\text{edit}} \\ \sqrt{\alpha_{t-1}}\,\mathbf{P}_t(\boldsymbol{\epsilon}_t^\theta(\tilde{\boldsymbol{x}}_t|\boldsymbol{h}_t)) + \mathbf{D}_t & \text{if } t_{\text{edit}} > t \geq t_{\text{boost}} \\ \sqrt{\alpha_{t-1}}\,\mathbf{P}_t(\boldsymbol{\epsilon}_t^\theta(\tilde{\boldsymbol{x}}_t|\boldsymbol{h}_t)) + \mathbf{D}_t + \sigma_t^2 \boldsymbol{z} & \text{if } t_{\text{boost}} > t \end{cases} \tag{4}$$



(a) Replacement (b) $h_t + h_t^{Content}$ (c) Slerp

Figure 2. **Illustration of content injection methods**. (a) and (b) provide content injection but suffer quality degradation. Compared to them, (c) allows successful content injection by preserving statistics in DMs and gradually increasing the ratio of the target content.

which consists of editing, denoising, and quality boosting intervals where the hyperparameter t_{edit} determines the editing interval and t_{boost} determines the quality boosting interval. Following Asyrp, we apply quality boosting to all figures except for ablation studies.

3. Method

In this section, we explore the interesting properties of ***h-space*** with Asyrp [43] and design a method for content injection. We start by simply replacing $\boldsymbol{h}_t$ of one sample with that of another sample and observe its drawbacks in § 3.1. Then we introduce an important requirement for mixing two $\boldsymbol{h}_t$'s in § 3.2. Furthermore, we propose latent calibration to retain the crucial elements in § 3.3.

3.1. Role of *h-space*

h-space, the deepest bottleneck of the U-Net in the diffusion models (DMs), contains the semantics of the resulting images to some extent. In other words, a change in ***h-space*** with Asyrp [43] leads to editing the resulting image. Formally, setting $\tilde{\boldsymbol{h}}_t = \boldsymbol{h}_t + \Delta\boldsymbol{h}_t$ for $t \in [T, t_{\text{edit}}]$ modifies the semantics, where $\Delta\boldsymbol{h}_t$ is the direction of desired attribute. The reverse process becomes $\tilde{\boldsymbol{x}}_{t-1} = \sqrt{\alpha_{t-1}}\,\mathbf{P}_t(\boldsymbol{\epsilon}_t^\theta(\tilde{\boldsymbol{x}}_t|\tilde{\boldsymbol{h}}_t)) + \mathbf{D}_t(\boldsymbol{\epsilon}_t^\theta(\tilde{\boldsymbol{x}}_t|\boldsymbol{h}_t))$, where $\tilde{\boldsymbol{h}}_t = \boldsymbol{h}_t + \Delta\boldsymbol{h}_t^{\text{attr}}$.

We start with a question: Does $\boldsymbol{h}$ solely specify the semantics of the resulting image as in the latent codes in GANs? I.e., would replacing $\boldsymbol{h}$ totally change the output?

To answer the question, we invert two images $I^{(1)}$ and $I^{(2)}$ to noises $\boldsymbol{x}_T^{(1)}$ and $\boldsymbol{x}_T^{(2)}$ via forward process, respectively. Then we replace $\{\boldsymbol{h}_t\}^\dagger$ from $\boldsymbol{x}_T^{(1)}$ with $\{\boldsymbol{h}_t^{(2)}\}$ from $\boldsymbol{x}_T^{(2)}$ during the reconstruction (i.e., reverse process). Formally, $\tilde{\boldsymbol{x}}_{t-1} = \sqrt{\alpha_{t-1}}\,\mathbf{P}_t(\boldsymbol{\epsilon}_t^\theta(\tilde{\boldsymbol{x}}_t|\boldsymbol{h}_t^{(2)})) + \mathbf{D}_t(\boldsymbol{\epsilon}_t^\theta(\tilde{\boldsymbol{x}}_t|\boldsymbol{h}_t))$,

†Note that the reverse process is recursive. The reason we denote $\{\boldsymbol{h}_t\}$ instead of $\boldsymbol{h}_t{}^{(1)}$ is that it differs from $\boldsymbol{h}_t{}^{(1)}$ after the first replacement.

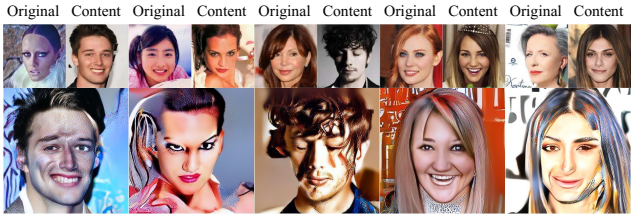


Figure 3. **Preliminary experiment.** Naïve replacement of $\boldsymbol{h}$ somehow combines the content and the original image. However, it severely degrades image quality.

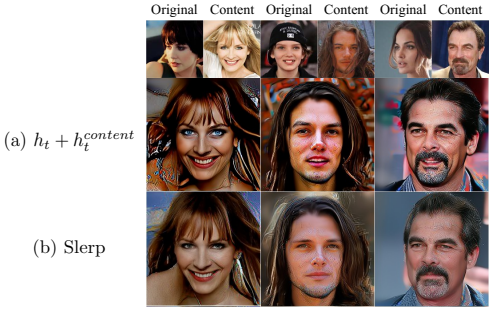


Figure 4. **Improvement in quality with Slerp.** (a) shows the result of $\boldsymbol{h}_t + \boldsymbol{h}_t^{content}$. It has some artifacts. (b) shows the result of Slerp with $\gamma = 0.5$ brings better quality. Techniques described later are not applied here for fair comparison.

$\tilde{\boldsymbol{x}}_T = \boldsymbol{x}_T^{(1)}$; which is illustrated in Figure 2a.

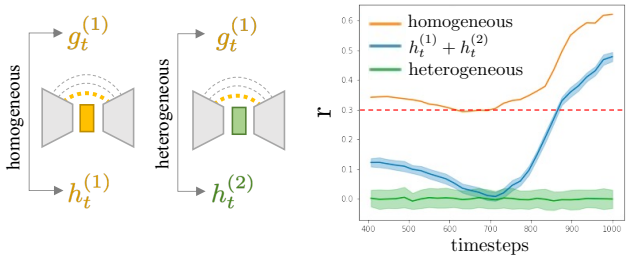
Interestingly, the resulting images with the replacement contain the people in $I^{(2)}$ with some elements of $I^{(1)}$ such as color distributions and backgrounds as shown in Figure 3. This phenomenon suggests that the main content is specified by $\boldsymbol{h}$ and the other aspects come from the other components, e.g., features in the skip connections. Henceforth, we name $\boldsymbol{h}_t^{(2)}$ as $\boldsymbol{h}_t^{content}$.

However, the replacement causes severe distortion in the images. We raise another question: how do we prevent the distortion? Note that Asyrp slightly adjusts $\boldsymbol{h}_t$ with a small change $\Delta\boldsymbol{h}_t$. On the other hand, replacing $\boldsymbol{h}_t$ as $\boldsymbol{h}_t^{content}$ completely removes $\boldsymbol{h}_t$. Assuming that the maintenance of $\boldsymbol{h}_t$ might be the key factor, we try an alternative in-between: adding $\boldsymbol{h}_t^{content}$ to $\boldsymbol{h}_t$; which is illustrated in Figure 2b. We observe far less distortion in Figure 4a.

With these preliminary experiments, we hypothesize that the replacement and the addition drive the disruption of the inherent correlations in the feature map. The subsequent sections provide grounding analyses and methods to address the problem.

3.2. Preserving statistics with Slerp

In DMs, $\boldsymbol{h}$-*space* is concatenated with skip connections and fed into the next layer. However, Asyrp [43] does not take into account the relationship between them. We observe an interesting relationship between $\boldsymbol{h}_t$ and its matching skip connections $\boldsymbol{g}_t$ (illustrated in Figure 5a) within a



(a) Matching skip connection (b) Correlation

Figure 5. **Correlation between $\boldsymbol{h}_t$ and skip connection.** $\boldsymbol{h}_t$ is highly correlated with the matching skip connection. (a) illustrates examples of matching and non-matching skip connections. (b) shows correlation between each $\tilde{\boldsymbol{h}}_t$ and skip connection. $\mathbf{r}$ is Pearson correlation coefficient and p-values of $\mathbf{r}$ are less than 1e-15. Non-matching skip connections seriously distort the correlation.

generative process and introduce requirements for replacing $\boldsymbol{h}_t$. We compute two versions of the correlation between the norms, $|\boldsymbol{h}_t|$ and $|\boldsymbol{g}_t|$:

$$r_{\text{homo}} = \frac{\sum_i \left(|\boldsymbol{h}^{(i)}| - |\bar{\boldsymbol{h}}|\right)\left(|\boldsymbol{g}^{(i)}| - |\bar{\boldsymbol{g}}|\right)}{(n-1)s_{|\boldsymbol{h}|}s_{|\boldsymbol{g}|}} \tag{5}$$

$$r_{\text{hetero}} = \frac{\sum_{j\neq i} \left(|\boldsymbol{h}^{(j)}| - |\bar{\boldsymbol{h}}|\right)\left(|\boldsymbol{g}^{(i)}| - |\bar{\boldsymbol{g}}|\right)}{(n-1)s_{|\boldsymbol{h}|}s_{|\boldsymbol{g}|}} \tag{6}$$

where n is the number of samples and s_* denotes standard deviation of $*$. We omit t for brevity.

Figure 5b shows that r_{homo}, the correlation between $\boldsymbol{h}_t$ and its matching skip connections, is roughly larger than 0.3 and is strongly positive when the timestep is close to T. On the other hand, r_{hetero}, the correlations between $\boldsymbol{h}_t$ and the skip connections in different samples, lie around zero. We try an alternative $\tilde{\boldsymbol{h}} = \boldsymbol{h}^{(i)} + \boldsymbol{h}^{(j)}$ and find its correlation is closer to r_{homo} than r_{hetero} and it produces less distortion.

Hence, we hypothesize that the correlation between $|\boldsymbol{h}|$ and $|\boldsymbol{g}|$ should remain consistent after the modification to preserve the quality of the generated images. To ensure the correlation of $\tilde{\boldsymbol{h}}_t$ equals to r_{homo}, we introduce normalized spherical interpolation (Slerp) between $\boldsymbol{h}_t$ and $\boldsymbol{h}_t^{content}$:

$$\tilde{\boldsymbol{h}}_t = f(\boldsymbol{h}_t, \boldsymbol{h}_t^{content}, \gamma) = \text{Slerp}(\boldsymbol{h}_t, \frac{\boldsymbol{h}_t^{content}}{\|\boldsymbol{h}_t^{content}\|} \cdot \|\boldsymbol{h}_t\|, \gamma), \tag{7}$$

where $\gamma \in [0, 1]$ is a coefficient of $\boldsymbol{h}_t^{content}$. (See Figure 2c.) We note that Slerp requires the inputs to have the same norm. Normalizing $\boldsymbol{h}_t^{content}$ to match the norm of $\boldsymbol{h}_t$ ensures a consistent correlation between $|\text{Slerp}(\cdot)|$ and $|\boldsymbol{g}_t^{(1)}|$ to be the same with the correlation between $|\boldsymbol{h}_t|$ and $|\boldsymbol{g}_t^{(1)}|$. Replacing $\boldsymbol{h}_t$ with $\tilde{\boldsymbol{h}}_t$ using Slerp exhibits fewer artifacts and better content preservation, as shown in Figure 4b. Besides the improvement, we can control how much content will be injected by adjusting the $\boldsymbol{h}_t$-to-$\boldsymbol{h}_t^{content}$ ratio

through parameter γ_t of Slerp. We provide an approximation of the total amount of injected content in § E.2.

3.3. Latent calibration

So far, we have revealed that mixing features in *h-space* injects the content. Although Slerp preserves the correlation between *h-space* and skip connection, altering only $\boldsymbol{h}_t$ with fixed skip connection may arrive at $\tilde{\boldsymbol{x}}_{t-1}$ that could not be reached from $\tilde{\boldsymbol{x}}_t$. Hence, we propose *latent calibration* that achieves the similar change due to $\tilde{\boldsymbol{h}}_t$ by modifying $\tilde{\boldsymbol{x}}_t$.

Specifically, after we compute $\tilde{\boldsymbol{x}}_{t-1}$, we define a slack variable $\mathbf{v} = \tilde{\boldsymbol{x}}_t + d\mathbf{v}$ and find $d\mathbf{v}$ such that $\mathbf{P}_t(\epsilon_t^\theta(\mathbf{v})) \approx \mathbf{P}_t(\epsilon_t^\theta(\tilde{\boldsymbol{x}}_t|\tilde{\boldsymbol{h}}_t))$. It ensures $\tilde{\boldsymbol{x}}'_0$ predicted from $\mathbf{v}$ is as similar as possible to $\tilde{\boldsymbol{x}}_0$ predicted from injecting $\tilde{\boldsymbol{h}}_t$ to $\tilde{\boldsymbol{x}}_t$. We model the implicit change from $\tilde{\boldsymbol{x}}_t$ to $\tilde{\boldsymbol{x}}'_t$ that brings similar change by the injection and introduce a hyperparameter ω that controls the strength of the change. To this end, we define a slack variable $\mathbf{v} = \tilde{\boldsymbol{x}}_t + d\mathbf{v}$ and find $d\mathbf{v}$ such that $\mathbf{P}_t(\epsilon_t^\theta(\mathbf{v})) \approx \mathbf{P}_t(\epsilon_t^\theta(\tilde{\boldsymbol{x}}_t|\tilde{\boldsymbol{h}}_t))$. With the DDIM equation,

$$\sqrt{\alpha_t}\mathbf{P}_t = \tilde{\boldsymbol{x}}_t - \sqrt{1-\alpha_t}\epsilon_t^\theta(\tilde{\boldsymbol{x}}_t), \tag{8}$$

we define infinitesimal as

$$\sqrt{\alpha_t}\, d\mathbf{P}_t = d\tilde{\boldsymbol{x}}_t - \sqrt{1-\alpha_t} J(\epsilon_t^\theta)\, d\tilde{\boldsymbol{x}}_t\,. \tag{9}$$

Further letting $d\tilde{\boldsymbol{x}}_t = \omega\, d\mathbf{v}$ and $J(\epsilon_t^\theta)\, d\mathbf{v} = d\epsilon_t^\theta$ induces

$$d\tilde{\boldsymbol{x}}_t = \sqrt{\alpha_t}\, d\mathbf{P}_t + \omega\sqrt{1-\alpha_t}\, d\epsilon_t^\theta\,. \tag{10}$$

Then, we define $\tilde{\boldsymbol{x}}'_t = \tilde{\boldsymbol{x}}_t + d\tilde{\boldsymbol{x}}_t$ and obtain $\tilde{\boldsymbol{x}}'_{t-1}$ by a typical denoising step.

In addition, $\mathbf{P}_t(\epsilon_t^\theta(\tilde{\boldsymbol{x}}'_t))$ in Eq. (10) has larger standard deviation than $\mathbf{P}_t(\epsilon_t^\theta(\tilde{\boldsymbol{x}}_t))$. We regularize it to have the same standard deviation of $\mathbf{P}_t(\epsilon_t^\theta(\tilde{\boldsymbol{x}}_t))$ by

$$d\mathbf{P}_t = \frac{\mathbf{P}'_t - \bar{\mathbf{P}}'_t}{|\mathbf{P}'_t|}|\mathbf{P}_t| + \bar{\mathbf{P}}'_t - \mathbf{P}_t(\epsilon_t^\theta(\tilde{\boldsymbol{x}}_t)), \tag{11}$$

where $\mathbf{P}'_t = \mathbf{P}_t(\epsilon_t^\theta(\tilde{\boldsymbol{x}}'_t))$. Then we control $\boldsymbol{x}'_t$ with an ω.

When we further expand Eq. (10) by the definition of $\mathbf{P}_t$,

$$d\tilde{\boldsymbol{x}}_t \approx (\omega-1)\sqrt{1-\alpha_t}(\epsilon_t^\theta(\tilde{\boldsymbol{x}}_t|\tilde{\boldsymbol{h}}_t) - \epsilon_t^\theta(\tilde{\boldsymbol{x}}_t)). \tag{12}$$

Interestingly, setting $\omega = 1$ reduces $d\tilde{\boldsymbol{x}}_t$ to $\mathbf{0}$, i.e., injection does not occur. And setting $\omega \approx 0$[†] drives $\tilde{\boldsymbol{x}}'_{t-1}$ close to $\tilde{\boldsymbol{x}}_{t-1}$, i.e., latent calibration does not occur. Intuitively, by Eq. (12), $\tilde{\boldsymbol{x}}'_t$ may share the predicted $\tilde{\boldsymbol{x}}_0$ with $\mathbf{P}_t(\epsilon_t^\theta(\tilde{\boldsymbol{x}}_t|\tilde{\boldsymbol{h}}_t))$ and contains original elements. In other words, we maintain the original elements by adding $d\tilde{\boldsymbol{x}}_t$ directly in *x-space* while the content injection is conducted in *h-space*.

Latent calibration consists of four steps. First, we inject the contents as $\tilde{\boldsymbol{x}}_t \to \tilde{\boldsymbol{x}}_{t-1}$ with Slerp. Second, we regularize $\mathbf{P}_t$ to preserve the original signal distribution after injection. Third, we solve the DDIM equation $\tilde{\boldsymbol{x}}'_t = \tilde{\boldsymbol{x}}_t + d\tilde{\boldsymbol{x}}_t$ by using Eq. (10). Finally, we step through a reverse process $\tilde{\boldsymbol{x}}'_t \to \tilde{\boldsymbol{x}}'_{t-1}$. In summary, we obtain target $\tilde{\boldsymbol{x}}_{t-1}$ by Slerp and generate $\tilde{\boldsymbol{x}}'_{t-1}$ without feature injection with calculated the corresponding $\tilde{\boldsymbol{x}}'_t$. Please refer to Algorithm 2 for details.

[†] ω can not be 0 because of its definition.

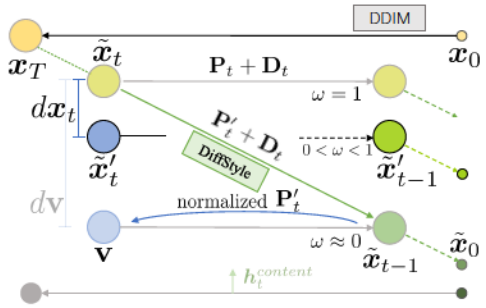


Figure 6. **Latent calibration.** The result of DDIM reverse process with given approximated $\tilde{\boldsymbol{x}}'_t$ can be similar to the result of a corresponding injected result $\tilde{\boldsymbol{x}}_{t-1}$. As ω gets close to 1, more original elements are added through $d\boldsymbol{x}_t$. Note that the effect of latent calibration is different from modifying γ because it remains predicted $\tilde{\boldsymbol{x}}_0$ by solving the DDIM equation.

3.4. Full generative process

We observe that *h-space* contains content and skip connection from $\boldsymbol{x}_T$ conveys the original elements. We utilize this phenomenon for in-domain samples and out-of-domain artistic samples. Note that it is possible to obtain inverted $\boldsymbol{x}_T$ from any arbitrary real image. Therefore, even if we use out-of-domain images such as artistic images, InjectFusion successfully retain the original elements in the images. Furthermore, local mixing of *h-space* enables injecting content into the corresponding target area as shown in Figure 12.

Algorithm 1: InjectFusion

Input: $\boldsymbol{x}_T$ (inverted latent variable from from image $I^{original}$), $\{\boldsymbol{h}_t^{content}\}_{t=t_{edit}}^{T}$ (obtained from content image $I^{content}$), ϵ_θ (pretrained model), m (feature map mask), f (Slerp)

Output: $\tilde{\boldsymbol{x}}_0$ (transferred image)

$\tilde{\boldsymbol{x}}_t \leftarrow \boldsymbol{x}_T$ **for** $t = T, ..., 1$ **do**
  if $t \geq t_{edit}$ **then**
    Extract feature map $\boldsymbol{h}_t$ from $\epsilon_\theta(\tilde{\boldsymbol{x}}_t)$;
    $\tilde{\boldsymbol{h}}_t \leftarrow f((m \otimes \boldsymbol{h}_t), (m \otimes \boldsymbol{h}_t^{content}), \gamma) \oplus (1-m) \otimes \boldsymbol{h}_t$
    $\tilde{\epsilon} \leftarrow \epsilon_\theta(\tilde{\boldsymbol{x}}_t|\tilde{\boldsymbol{h}}_t)$, $\epsilon \leftarrow \epsilon_\theta(\tilde{\boldsymbol{x}}_t)$
    Adapt Latent calibration (Algorithm 2)
  else
    $\tilde{\epsilon} = \epsilon \leftarrow \epsilon_\theta(\tilde{\boldsymbol{x}}_t)$,
  $\tilde{\boldsymbol{x}}_{t-1} \leftarrow \sqrt{\alpha_{t-1}}\left(\frac{\tilde{\boldsymbol{x}}_t - \sqrt{1-\alpha_t}\tilde{\epsilon}}{\sqrt{\alpha_t}}\right) + \sqrt{1-\alpha_{t-1}}\epsilon$

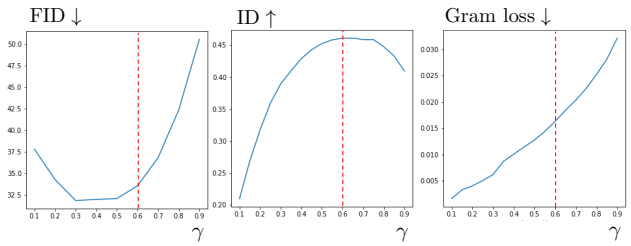


Figure 7. **Choice of γ.** (b) shows that γ should be less than 0.6 since the ID change via content injection converges at the point. If $\gamma > 0.6$, the resulting image only departs from the original image and suffers quality degradation without any advantage.

For the local mixing, each $\boldsymbol{h}_t$ is masked before Slerp and the mixed $\boldsymbol{h}_t$ is inserted into the original feature map. We provide Algorithm 1 for them and an illustration of spatial $\boldsymbol{h}_t$ mixing in Figure S1. Note that we omit latent calibration in the algorithm for simplicity. The full algorithm is provided in Appendix Algorithm 2.

4. Experiments

In this section, we present analyses on InjectFusion and showcase our applications.

Setting We use the official pretrained checkpoints of DDPM++ [49, 69] for CelebA-HQ [33] and LSUN-church/-bedroom [79], iDDPM [54] for AFHQv2-Dog [11], and ADM with P2-weighting [9, 18] for METFACES [35] and ImageNet [15]. The images have a resolution of 256×256 pixels. We freeze the model weights. We use t_{edit}=400, ω=0.3, γ=0.3, and t_{boost}=200 to produce high-quality images. For more implementation details, please refer to Appendix A.

Metrics GRAM loss (style loss) [23] indicates the style difference between the original image and the resulting image. ID computes the cosine similarity between face identity [16] of the content image and the resulting image to measure content consistency. Fréchet Inception Distance (FID) [28] provides the overall image quality. To compute FID, we compare generated 5K images from fixed 5K original-content image pairs using 50 steps of the reverse process and 25k images from the training set of CelebA-HQ without the overlap of the pairs.

4.1. Analyses

In this section, we define what elements come from the original and the content image. We provide a guideline for choosing the content injection ratio γ considering both quality and content consistency. We also show the versatility of latent calibration and propose the best interval for editing. Furthermore, we provide quantitative results that

[%]	Nose	Eyes	Jaw line	Expression	Hair color	Glasses	Skin color	Make up
Original	28.06	43.57	24.67	36.73	**95.74**	5.63	**94.15**	**90.60**
Content	**71.94**	**56.43**	**75.33**	**63.27**	4.26	**94.37**	5.85	9.30

Table
with 5
sulting

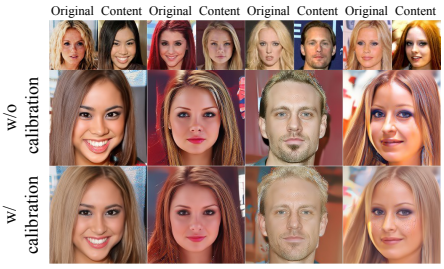


Figure 8. **Effectiveness of Latent calibration.** Latent calibration recovers elements of original images while preserving content elements. We do not use other techniques such as quality boosting for comparison.

support assumptions suggested in § 3: $\boldsymbol{h}$-*space* has content elements.

Definition of content We measured the CLIP score on CelebA attributes to reveal what information comes from the content and original images. We classify the attribute of the mixed image as closer to the original or content image with the CLIP score. In short, content includes *glasses, square jaw, young, bald, big nose, and facial expressions* and the remaining elements include *hairstyle, hair color, bang hair, accessories, beard, and makeup*. Please see the details in Appendix J. Furthermore, we conduct a user study in Table 1 to support the result of the CLIP score. It aligns with the results using CLIP score for classifying.

We define the retained elements of the original image as the color-dependent attributes and the content as the semantics and shape. Figure S22 and Figure S23 show that DMs trained on the scenes with complex layouts have different notions of content and retained elements: rough shapes of churches are considered as content and room layouts including the location of beds are considered as contents.

Content injection ratio γ We suggest that the original $\boldsymbol{h}_t$ should be partially kept in § 3.1. Figure 7 supports that the content injection ratio γ should be less than 0.6 for image quality (FID) and preservation of the original image, and $\gamma > 0.6$ does not increase ID similarity. We provide more observations on γ in Appendix B.

The effect of latent calibration Figure 8 shows that latent calibration leads to a better reflection of the original elements such as makeup and hair color. Note that, depending on the latent calibration strength ω, there is a trade-off relationship between Gram loss and ID similarity as well

	FID ↓	ID ↑	Gram loss ↓
$\boldsymbol{h}_t + \boldsymbol{h}_t^{content}$	49.94	0.3581	0.0415
Lerp	36.89	0.4040	0.0318
Slerp	**32.09**	**0.4390**	**0.0310**

Table 2. **Performance of various configurations** Slerp improves FID, ID similarity between target content images and synthesized images over other methods.

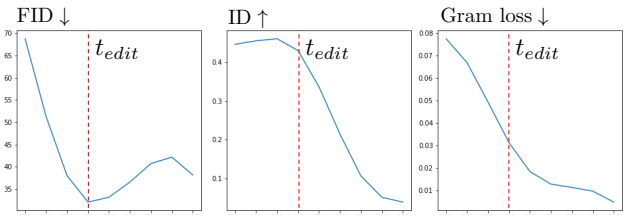


Figure 10. **Comparison with DiffuseIT** InjectFusion is effective even in situations where there is a large discrepancy between the color distributions of the original image and the content image.

as FID. We report them at various ω in Figure S4. We discover that increasing ω favors preserving the original images. More details including the efficiency of adapting latent calibration to other methods, Plug-and-Play [71] and MasaCtrl [6], can be found in Appendix C.

Quantitative comparison Table 2 shows the quantitative result of each configuration investigated in § 3. Reconstruction reports FID of the official checkpoint of DDPM++ [49] through its forward and reverse process without any modification on $\boldsymbol{h}$-*space*. We observe that $\boldsymbol{h}_t + \boldsymbol{h}_t^{content}$ harms FID with severe distortion. Slerp outperforms $\boldsymbol{h}_t + \boldsymbol{h}_t^{content}$ in all aspects.

Table 2 further shows the superiority of Slerp over linear interpolation (Lerp). It implies that the normalization for preserving the correlation between $\boldsymbol{h}_t$ and skips $\boldsymbol{g}_t$ is important. Furthermore, Figure S6 shows that Slerp resolves the remaining artifacts that reside in the resulting images by Lerp. Comparison between Slerp and Lerp will be further discussed in § E.1.

Editing interval $[T, t_{\text{edit}}]$ We observe that there is a trade-off between ID similarity and Gram loss when using a suboptimal t_{edit} and specific value of t_{edit} leads to better FID, as shown in Figure 9. We choose $t_{\text{edit}} = 400$ for its balance among the three factors. This choice also aligns with that of Asyrp [43] for editing toward unseen domains, which requires a large change, such as injecting content. Notably, we find that $t_{\text{edit}} = 400$ is also suitable for achieving content injection into artistic images.

Choice of the content injection layer Except for $\boldsymbol{h}$-*space*, the other intermediate layers in the U-Net can be candidate feature spaces for content injection. However, Figure S13a shows that content injection works well only on $\boldsymbol{h}$-*space*, while it produces artifacts and loses injected content on the other feature spaces. Injecting skip connection while content injection does not alleviate the problems as shown in Figure S13b.

4.2. Qualitative results

In-domain original images Figure 11a,b shows InjectFusion on AFHQv2-Dog [11] METFACES [35]. See Appendix D.1 for more results on various architectures and datasets.

Artistic original images In addition, we can use arbitrary original images, even if they are out-of-domain. Figure 11c shows results with artistic images as style. For the artistic references, we do not use quality boosting [43] since they aim to improve the quality and realism of $\boldsymbol{x}_0$ which may not be desirable when transferring the elements of an out-of-domain image onto the target image. We provide more results in Appendix D.1.

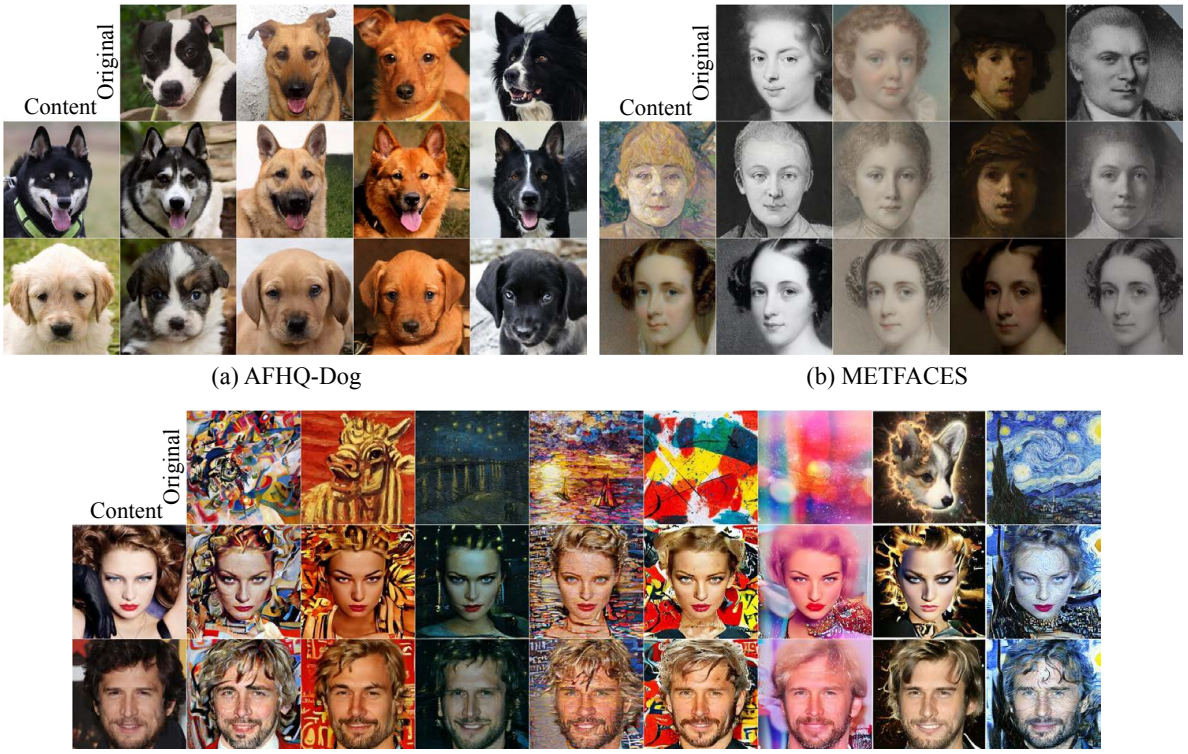
4.3. Comparison with existing methods

We first note that there is no competitor with perfect compatibility: frozen pretrained diffusion models, and no extra guidance from external off-the-shelf models. Still, we compare our content injection with DiffuseIT [42] which guides pretrained DMs using DINO ViT [7]. Figure 10 shows that DiffuseIT struggles when there is a large gap between the content image and the original image regarding color distributions. More qualitative comparisons with existing methods [11, 12, 17, 40, 56, 75] and user study are deferred to Appendix D.2.

5. Conclusion and discussion

In this paper, we have proposed a training-free content injection using pretrained DMs. The components in our method are designed to preserve the statistical properties of the original reverse process so that the resulting images are free from artifacts even when the original images are out-of-domain. We hope that our method and its analyses help the research community to harness the nice properties of DMs for various image synthesis tasks.

Although InjectFusion achieves high-quality content injection, the small resolution of the $\boldsymbol{h}$-*space* hinders fine con-



(a) AFHQ-Dog (b) METFACES

ined domain,

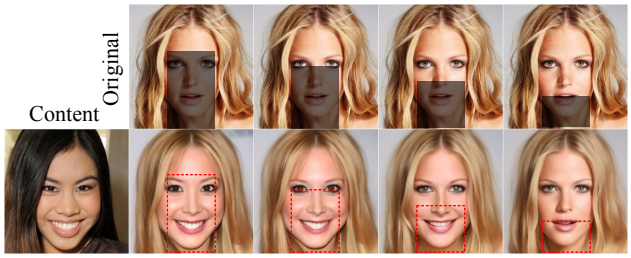


Figure 12. **Local style mixing with various feature map mask sizes.** Adjusting the size and position of the feature map mask enables to handle the area of content injection, facilitating control

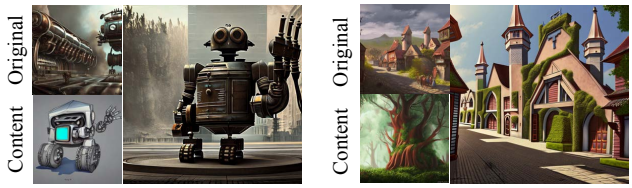


Figure 13. **InjectFusion on Stable diffusion** Although we observe similar phenomenons, the content elements of latent-level DMs is different from pixel-level DMs; More semantic elements is injected to the original image.

trol of the injecting region. We provide content injection with various masks in Figure 12.

While out-of-domain images can be used as the original image (i.e., style), injecting content-less out-of-domain images leads to meaningless results. We provide them in Figure S9. We suggest that $\boldsymbol{h}_t$ is not the universal representation for arbitrary content.

In addition, we provide pilot results of InjectFusion on Stable diffusion in Figure 13. It works somewhat similarly but the phenomenon is not as clear as in non-latent diffusion models. The bottleneck of Stable diffusion appears to be more semantically rich, possibly due to its diffusion in VAE's latent space. Unveiling the mechanisms in latent diffusion models remains our future work. Please refer to Appendix I for the details.

Lastly, we briefly discuss the effect of the scheduling strategy of the injecting ratio γ in Appendix G. Further investigation would be an interesting research direction.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) funded by the Korean government (MSIT) (RS-2023-00223062).

References

[1] 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. 1, 2, 15

[2] Kyungjune Baek, Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Hyunjung Shim. Rethinking the truly unsupervised image-to-image translation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14154–14163, 2021. 16

[3] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 1

[4] Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. *arXiv preprint arXiv:2112.03126*, 2021. 15

[5] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 2

[6] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. *arXiv preprint arXiv:2304.08465*, 2023. 2, 7, 12

[7] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 2, 7, 14

[8] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. *arXiv preprint arXiv:2108.02938*, 2021. 1, 2, 15

[9] Jooyoung Choi, Jungbeom Lee, Chaehun Shin, Sungwon Kim, Hyunwoo Kim, and Sungroh Yoon. Perception prioritized training of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11472–11481, 2022. 2, 6

[10] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8789–8797, 2018. 16

[11] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8188–8197, 2020. 6, 7, 16

[12] Min Jin Chong and David Forsyth. Jojogan: One shot face stylization. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVI*, pages 128–152. Springer, 2022. 7, 16

[13] Hyungjin Chung, Byeongsu Sim, Dohoon Ryu, and Jong Chul Ye. Improving diffusion models for inverse problems using manifold constraints. *arXiv preprint arXiv:2206.00941*, 2022. 1

[14] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022. 15

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 6

[16] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019. 6

[17] Yingying Deng, Fan Tang, Xingjia Pan, Weiming Dong, Chongyang Ma, and Changsheng Xu. Stytrˆ 2: Unbiased image style transfer with transformers. *arXiv preprint arXiv:2105.14576*, 2021. 7, 13

[18] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 2, 6

[19] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[20] 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. *arXiv preprint arXiv:2203.13131*, 2022. 1

[21] 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, 15

[22] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015. 2, 16

[23] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016. 6

[24] Vidit Goel, Elia Peruzzo, Yifan Jiang, Dejia Xu, Nicu Sebe, Trevor Darrell, Zhangyang Wang, and Humphrey Shi. Pair-diffusion: Object-level image editing with structure-and-appearance paired diffusion models. *arXiv preprint arXiv:2303.17546*, 2023. 2

[25] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 16

[26] Inhwa Han, Serin Yang, Taesung Kwon, and Jong Chul Ye. Highly personalized text embedding for image manipulation

by stable diffusion. *arXiv preprint arXiv:2303.08767*, 2023. 2

[27] 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

[28] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 6

[29] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1, 2, 15

[30] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pages 1989–1998. Pmlr, 2018. 16

[31] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 16

[32] Alexia Jolicoeur-Martineau, Ke Li, Rémi Piché-Taillefer, Tal Kachman, and Ioannis Mitliagkas. Gotta go fast when generating data with score-based models. *arXiv preprint arXiv:2105.14080*, 2021. 1

[33] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018. 6, 16

[34] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022. 2

[35] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems*, 33:12104–12114, 2020. 6, 7

[36] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 16

[37] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020. 16

[38] 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. 1

[39] Gwanghyun Kim and Jong Chul Ye. Diffusionclip: Text-guided image manipulation using diffusion models. 2021. 1, 15

[40] Hyunsu Kim, Yunjey Choi, Junho Kim, Sungjoo Yoo, and Youngjung Uh. Exploiting spatial dimensions of latent in gan for real-time image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 852–861, 2021. 7, 13, 16

[41] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1931–1941, 2023. 2, 15

[42] Gihyun Kwon and Jong Chul Ye. Diffusion-based image translation using disentangled style and content representation. *arXiv preprint arXiv:2209.15264*, 2022. 2, 7, 14

[43] Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Diffusion models already have a semantic latent space. *arXiv preprint arXiv:2210.10960*, 2022. 1, 2, 3, 4, 7, 15

[44] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. *arXiv preprint arXiv:2301.07093*, 2023. 2, 15

[45] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022. 1

[46] Xihui Liu, Dong Huk Park, Samaneh Azadi, Gong Zhang, Arman Chopikyan, Yuxiao Hu, Humphrey Shi, Anna Rohrbach, and Trevor Darrell. More control for free! image synthesis with semantic diffusion guidance. *arXiv preprint arXiv:2112.05744*, 2021. 2

[47] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *arXiv preprint arXiv:2206.00927*, 2022. 1

[48] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022. 1

[49] 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. 1, 2, 6, 7, 15

[50] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. *arXiv preprint arXiv:2211.09794*, 2022. 15

[51] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6038–6047, 2023. 2

[52] Chong Mou, Xintao Wang, Liangbin Xie, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453*, 2023. 2

[53] 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. 1, 2

[54] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 2, 6

[55] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2337–2346, 2019. 16

[56] 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*, 33:7198–7211, 2020. 7, 13

[57] Yong-Hyun Park, Mingi Kwon, Junghyo Jo, and Youngjung Uh. Unsupervised discovery of semantic latent directions in diffusion models. *arXiv preprint arXiv:2302.12469*, 2023. 15

[58] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. *arXiv preprint arXiv:2302.03027*, 2023. 15

[59] 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

[60] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1

[61] 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. 2022. 2

[62] Dan Ruta, Gemma Canet Tarres, Alex Black, Andrew Gilbert, and John Collomosse. Aladin-nst: Self-supervised disentangled representation learning of artistic style through neural style transfer. *arXiv preprint arXiv:2304.05755*, 2023. 2

[63] Dan Ruta, Gemma Canet Tarrés, Andrew Gilbert, Eli Shechtman, Nicholas Kolkin, and John Collomosse. Diff-nst: Diffusion interleaving for deformable neural style transfer. *arXiv preprint arXiv:2307.04157*, 2023. 2

[64] Simo Ryu. Low-rank adaptation for fast text-to-image diffusion fine-tuning. https://github.com/cloneofsimo/lora, 2023. 2

[65] 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

[66] Vikash Sehwag, Caner Hazirbas, Albert Gordo, Firat Ozgenel, and Cristian Canton. Generating high fidelity data from low-density regions using diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11492–11501, 2022. 2

[67] Kihyuk Sohn, Nataniel Ruiz, Kimin Lee, Daniel Castro Chin, Irina Blok, Huiwen Chang, Jarred Barber, Lu Jiang, Glenn Entis, Yuanzhen Li, et al. Styledrop: Text-to-image generation in any style. *arXiv preprint arXiv:2306.00983*, 2023. 2

[68] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 1, 2, 3

[69] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 1, 2, 6, 15

[70] Gemma Canet Tarrés, Dan Ruta, Tu Bui, and John Collomosse. Parasol: Parametric style control for diffusion image synthesis. *arXiv preprint arXiv:2303.06464*, 2023. 2

[71] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1921–1930, 2023. 2, 7, 12, 15

[72] Bram Wallace, Akash Gokul, and Nikhil Naik. Edict: Exact diffusion inversion via coupled transformations. *arXiv preprint arXiv:2211.12446*, 2022. 15

[73] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018. 16

[74] Daniel Watson, William Chan, Jonathan Ho, and Mohammad Norouzi. Learning fast samplers for diffusion models by differentiating through sample quality. In *International Conference on Learning Representations*, 2022. 2

[75] Zijie Wu, Zhen Zhu, Junping Du, and Xiang Bai. Ccpl: Contrastive coherence preserving loss for versatile style transfer. In *European Conference on Computer Vision*, pages 189–206. Springer, 2022. 7, 13

[76] Shaoan Xie, Zhifei Zhang, Zhe Lin, Tobias Hinz, and Kun Zhang. Smartbrush: Text and shape guided object inpainting with diffusion model. *arXiv preprint arXiv:2212.05034*, 2022. 15

[77] Binxin Yang, Shuyang Gu, Bo Zhang, Ting Zhang, Xuejin Chen, Xiaoyan Sun, Dong Chen, and Fang Wen. Paint by example: Exemplar-based image editing with diffusion models. *arXiv preprint arXiv:2211.13227*, 2022. 15

[78] Jaejun Yoo, Youngjung Uh, Sanghyuk Chun, Byeongkyu Kang, and Jung-Woo Ha. Photorealistic style transfer via wavelet transforms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9036–9045, 2019. 16

[79] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 6

[80] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023. 2, 15

[81] Ye Zhu, Yu Wu, Zhiwei Deng, Olga Russakovsky, and Yan Yan. Boundary guided mixing trajectory for semantic control with diffusion models. *arXiv preprint arXiv:2302.08357*, 2023. 15