Supplementary Material

Zero-shot Inversion Process for Image Attribute Editing with Diffusion Models

A Useful Lemmas

Theorem 1 (Kwon et al. (2022, Theorem 1)). *Let ϵ_t^θ be a predicted noise during the original reverse process at t and $\widetilde{\epsilon}_t^\theta$ be its shifted counterpart. Then, $\widetilde{x}_{t-1} \approx x_{t-1}$ where $\widetilde{x}_{t-1} = \sqrt{\alpha_{t-1}} P_t(\widetilde{\epsilon}_t^\theta(x_t)) + D_t(\widetilde{\epsilon}_t^\theta(x_t))$. I.e., the shifted terms of $\widetilde{\epsilon}_t^\theta$ in P_t and D_t destruct each other in the reverse process.*

B Detail of Our Framework

Text Encoder: To extract textual features, the text prompt t derived from the attribute t_{attr} is encoded into a vector. This vector is subsequently utilized to calculate the CLIP loss in tandem with the target image. As exemplified by the work of Brown et al. (2020), the evolution of substantial language models has significantly drawn attention to the text prompt, especially in the realm of multimodal applications, e.g., GPT-4 (OpenAI, 2023) and PaLM2 (Google, 2023). These language models have been extensively trained on a vast array of linguistic data and have proven to be adept in few-shot learning scenarios. In this study, the CLIP model (Radford et al., 2021) is employed as the Text Encoder.

In our work, the text prompt t contains two components: t_{source} and t_{target}. These components are generated through the following procedure. Let E_T denote the text encoder equipped with vocabulary V. The attribute t_{attr} constitutes a sequence of phrases denoted as $t_{\text{attr}} = (s_1, \ldots, s_k)$, where each $s_i \in V$. To illustrate with an example, when the attribute t_{attr} is "glasses," the value of k is 1. Drawing parallels with prompts in natural language processing (NLP) (Schick and Schütze, 2020), we establish the notion of a pattern as a function P, which takes the attribute t_{attr} as input and generates two phrases or sentences, t_{source} and t_{target}, in the vocabulary V. This dynamic yields the text prompt $t = (t_{\text{source}}, t_{\text{target}})$.

For instance, when the attribute t_{attr} pertains to a person, a conceivable pattern could be $P(t_{\text{attr}}) =$ ("a person", "a person with t_{attr}"). Consequently, if the input attribute is $t_{\text{attr}} =$ "glasses," the derived text prompt would be $P(t_{\text{attr}}) =$ ("a person", "a person with glasses"), from which $t_{\text{source}} =$ "a person" and $t_{\text{target}} =$ "a person with glasses." Collectively, they constitute the comprehensive text prompt t. After that, the Text Encoder processes this prompt to encode t_{source} as $E_T(t_{\text{source}}) \in \mathbb{R}^d$ and t_{target} as $E_T(t_{\text{target}}) \in \mathbb{R}^d$.

Visual Generator: To acquire the visual attributes corresponding to the specified attribute, we employ a text-image model known as the Visual Generator. Notably, large scale generative models have demonstrated pronounced robustness and adaptability in conditional generation tasks (Saharia et al., 2022; Rombach et al., 2022). While these models may not possess the capacity to pinpoint or precisely modify attributes, they are proficient in generating attribute-associated features within the visual domain. In this study, we adopt UniDiffuser (Bao et al., 2023) as the designated Visual Generator. UniDiffuser uniquely combines text and image generation within a single model, leveraging the concurrent utilization of marginal, conditional, and joint distributions derived from multimodal data.

To acquire the reference image, denoted as $i_{\text{ref}} = G_V(t_{\text{target}})$, images are extracted from the conditional distribution $p(x_0|t_{\text{target}})$ through the following process:

$$p_\theta(x_0|t_{\text{target}}) = \int p_\theta(x_{0:T}|t_{\text{target}}) dx_{1:T},\ p_\theta(x_{0:T}|t_{\text{target}}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t, t_{\text{target}}) \quad (7)$$

where $x_T \sim \mathcal{N}(0, \boldsymbol{I})$, $p_\theta(x_{t-1}|x_t, t_{\text{target}}) = \mathcal{N}(x_{t-1}|\mu_t(x_t, t_{\text{target}}), \sigma_t^2 \boldsymbol{I})$, and the mean $\mu_t = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{\beta_t}{\sqrt{1-\overline{\alpha_t}}} \mathbb{E}[\epsilon^x|x_t, t_{\text{traget}}])$ is predicted by a noise predictor ϵ_θ (Bao et al., 2023). Upon completion of the denoising procedure, we obtain the reference image i_{ref}.

Attribute Encoder: The Attribute Encoder E_A functions as a downsampling network, encompassing attention blocks and residual blocks. This type of downsampling network has found wide application in both detection (Ronneberger et al., 2015) and generation (Ho et al., 2020). Through down-sampling, the original image undergoes encoding into a latent space, thereby resulting in a transformed representation. Upon input of the reference image, the Attribute Encoder yields the latent embedding of visual features, i.e., $E_A(i_{\text{ref}}) \in \mathbb{R}^D$. This embedding, represented by $E_A(i_{\text{ref}})$, efficiently facilitates attribute manipulation.

Edit Generator: The role of the Edit Generator revolves around the generation of the target image. The efficacy of diffusion models has been extensively showcased in prior research (Ho et al., 2020; Choi et al., 2021). In this study, the DDIM model (Song et al., 2020a) is harnessed to generate images, as defined by Equation 2.

C Experiment Details

C.1 Parameters

The parameters of our ZIP in different datasets are shown in Table 4. The code is also attached to the supplementary materials.

Table 4: The parameters of ZIP

Parameter	CelebA-HQ	LSUN-church	LSUN-bedroom
Resolution of images	256×256	256×256	256×256
Time step T	1000	1000	1000
Batch size of training	9	9	9
Inversion time step	40	40	40
Learning rate of training	0.5	0.5	0.5
The weight of visual features Δh	0.3	0.1	0.1
The weight of CLIP loss λ_{clip}	0.8	2	0.8
The weight of reconstruction loss λ_{recon}	3	3	3

C.2 Evaluation

Inception Score: Inception Score (ISC) (Salimans et al., 2016) is used to access how realistic generated images are. The computation of ISC is:

$$ISC = exp(\mathbb{E}_{\text{x}} KL(p(y|x)||p(y))) \tag{8}$$

where $KL(p(y|x)||p(y))$ is the KL divergence between the conditional distribution $p(y|x)$ and the margianl distribution $p(y)$. Both the conditional and marginal distribution is calculated from features extracted from the images. The score is calculated on random splits of the images such that both a mean and standard deviation of the score are returned. In this paper, ISC is computed based on the original weights from (Heusel et al., 2017a).

Fréchet Inception Distance: Fréchet Inception Distance (FID) (Szegedy et al., 2016) also is used to access the quality of generated images. Different from ISC, which is computed only by the generation images ignoring the real images, FID uses the features from Inception Network to evaluate the similarity of real images and generated images. FID is computed by:

$$FID = |\mu - \mu_\omega| + tr(\Sigma + \Sigma_\omega - 2(\Sigma\Sigma_\omega)^{\frac{1}{2}}) \tag{9}$$

where $\mathcal{N}(\mu, \Sigma)$ is the multivariate normal distribution estimated from Inception v3 (Szegedy et al., 2016) features calculated on real life images and $\mathcal{N}(\mu_w, \Sigma_w)$ is the multivariate normal distribution

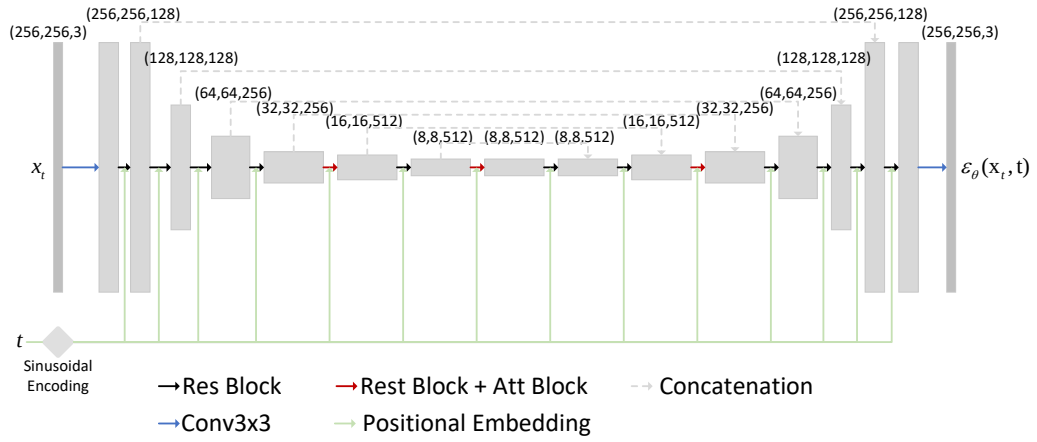

Figure 9: **The Architecture of Editing Generator**. The U-Net architecture of Editing Generator outputs 256×256 images. The Δh is inserted into the middle layer.

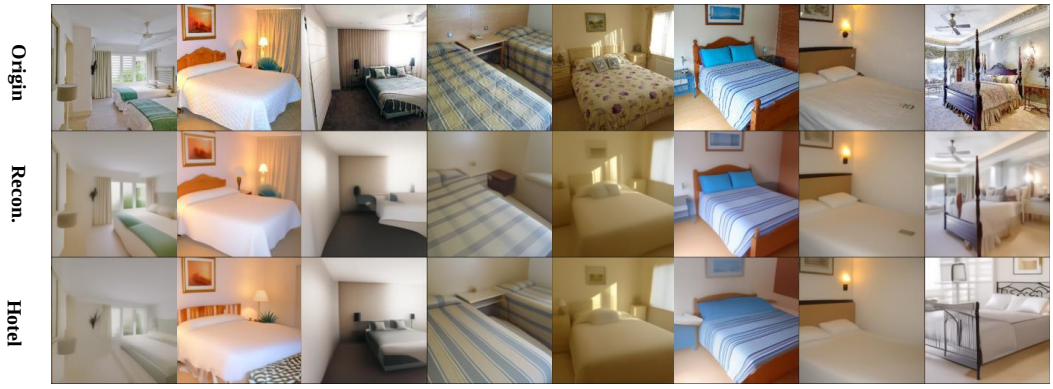

Figure 10: **The results in LSUN-bedroom**.

estimated from Inception v3 features calculated on generated images. In this paper, FID is computed based on the original weights from (Heusel et al., 2017b).

CLIP Score: CLIP Score (Radford et al., 2021) is a reference-free metric that can be used to evaluate the correlation between a generated caption for an image and the actual content of the image. It has been found to be highly correlated with human judgment. The metric is defined as:

$$extCLIPScore(i, c) = \max(100 * \cos(E_i, E_c), 0) \tag{10}$$

which corresponds to the cosine similarity between visual CLIP embedding E_i for an image i and textual CLIP embedding E_c for an caption c. In the following experiments, the model "clip-vit-base-patch16" from official checkpoints of CLIP model (Radford et al., 2021) is used to evaluate the attribute changes.

In our experiments, we use the code of torch-fidelity [1] to compute ISC and FID. CLIP Score is computed based on TorchMetrics [2].

[1] https://github.com/toshas/torch-fidelity.

[2] https://torchmetrics.readthedocs.io/en/stable/multimodal/clip_score.html.

Table 5: The results of LSUN dataset

	gothic			night			department			factory			bedroom-hotel		
Method	ISC	FID	CLIP	ISC	FID	CLIP	ISC	FID	CLIP	ISC	FID	CLIP	ISC	FID	CLIP
Asyrp (Kwon et al., 2022)	2.769	93.13	21.98	3.198	111.4	22.05	3.290	118.3	**23.49**	3.175	109.7	22.15	2.033	102.5	24.64
Ours	2.958	118.0	**24.24**	3.486	209.9	**23.24**	2.631	112.0	23.21	3.346	133.7	**22.92**	2.599	115.5	**26.52**

D Network Architecture

D.1 Attribute Encoder and Editing Generator

We use a U-net backbone as our Editing Generator as shown in Figure 9. The Attribute Encoder shares parameters with the Editing Generator in the down-sampling parts (the first eight layers). In the training process, we only update the parameters of the Attribute Encoder.

D.2 Others

The other parts of our framework are frozen with the official checkpoints. We use Unidiffuser (Bao et al., 2023) and CLIP model (Radford et al., 2021) as the Visual Generator and the Text Encoder with the official checkpoints, respectively.

E More Results

We supply more results for different datasets, such as LSUN-church in Figure 11 LSUN-bedroom in Figure 10, and CelebA-HQ in in Figure 15. The other attributes for different datasets, such as "Department" and "Factory", are shown in a bigger size.

In Figure 12 and Figure 13, we show more results in test data for the in-domain attribute "gender" and "smile". For visual features, ZIP can generate extra features of gender. For the deformation of facial features, ZIP can also change the original features well. Moreover, in Figure 14, the out-of-domain attribute is shown.

Also, as Figure 16 shows, the top half is the editing process for the attribute "makeup" without the reference image while the bottom has the reference image. Though both of them generate the details of the face, the results with the reference image are more $concentrated$ on the features of the attribute "makeup", such as the part of lips.

Figure 11: **The results in LSUN-church**.

Figure 12: **The results in CelebA-HQ for the attribute "gender"**.

Figure 13: **The results in CelebA-HQ for the attribute "smile"**.

Figure 14: **The results in CelebA-HQ for the attribute "makeup"**.

Figure 15: **The results in CelebA-HQ**.

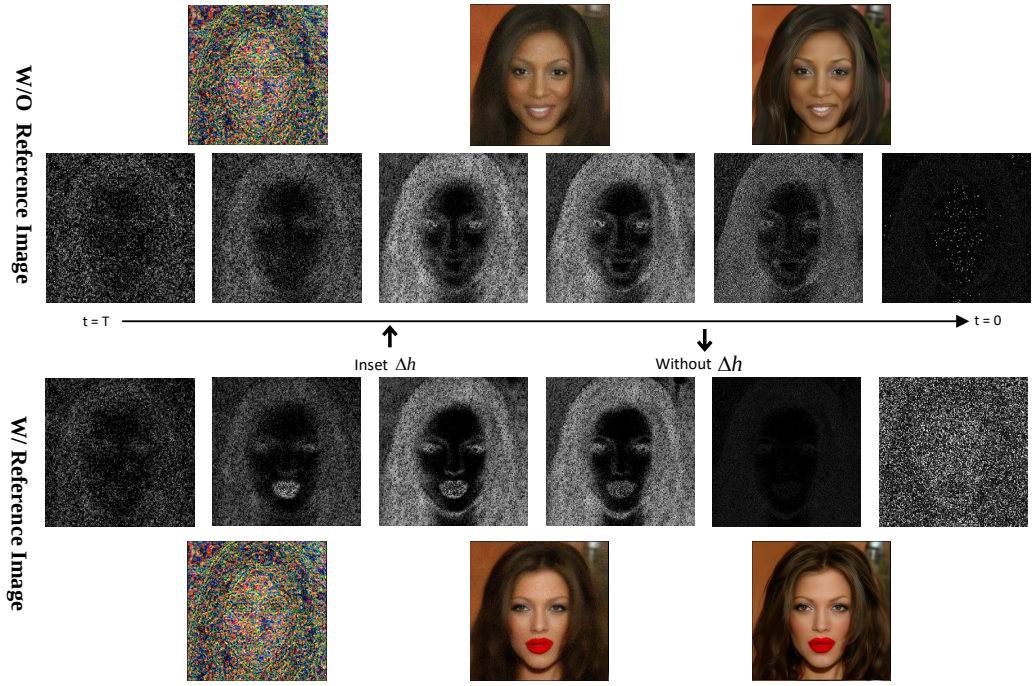

Figure 16: The visualization of noises in ZIP at different time steps t. The image is edited for the attribute of "makeup."