Difference Inversion: Interpolate and Isolate the Difference with Token Consistency for Image Analogy Generation

Hyunsoo Kim[1,2], Donghyun Kim[†1], Suhyun Kim[†3]
[1] Korea University [2] Korea Institute of Science and Technology [3] Kyung Hee University
{climba,d_kim}@korea.ac.kr,dr.suhyun.kim@gmail.com

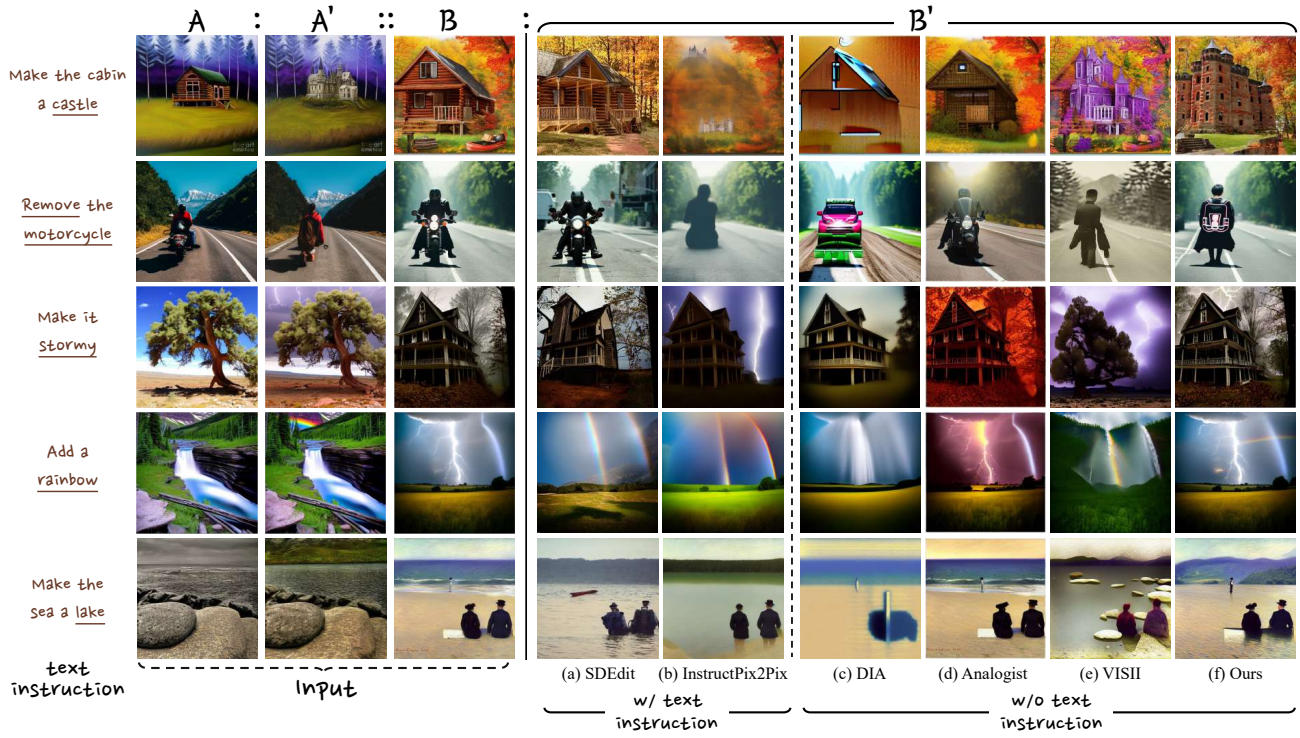


Figure 1. We propose ***Difference Inversion***, a method that generates B' conditioned on an image triplet $\{A, A', B\}$ that satisfies $A : A' :: B : B'$. Our method produces a significantly more plausible B' than other baselines. Note that SDEdit and InstructPix2Pix take the text instruction as input, whereas DIA, Analogist, VISII, and our Difference Inversion only use the image triplet A, A', B as input.

Abstract

How can we generate an image B' that satisfies $A : A' :: B : B'$, given the input images A,A' and B? Recent works have tackled this challenge through approaches like visual in-context learning or visual instruction. However, these methods are typically limited to specific models (e.g. *InstructPix2Pix. Inpainting models) rather than general diffusion models* (e.g. *Stable Diffusion, SDXL). This dependency may lead to inherited biases or lower editing capabilities. In this paper, we propose Difference Inversion, a method that isolates only the difference from A and A' and applies it to B to generate a plausible B'. To address model dependency, it is crucial to structure prompts in the form of a "Full Prompt" suitable for input to stable diffusion models, rather than using an "Instruction Prompt". To this end, we accurately extract the Difference between A and A' and combine it with the prompt of B, enabling a plug-and-play application of the difference. To extract a precise difference, we first identify it through 1) Delta Interpolation. Additionally, to ensure accurate training, we propose the 2) Token Consistency Loss and 3) Zero Initialization of Token Embeddings. Our extensive experiments demonstrate that Difference Inversion outperforms existing baselines both quantitatively and qualitatively, indicating its ability to generate more feasible B' in a model-agnostic manner.*

[†]Co-corresponding author.

1. Introduction

The goal of image analogy generation [39] is to create the target image B' based on the image triplet $\{A, A', B\}$, ensuring that it satisfies the image analogy [12, 19] formulation $A : A' :: B : B'$. With the huge advancement of large-scale diffusion models [5, 7, 16, 29, 32, 34], it is natural to consider leveraging these models for solving image analogy generation from the perspective of conditional image generation. Unlike existing conditional diffusion models [1, 2, 13, 22, 26, 31] that usually take text prompts as conditions, image analogy generation assumes that only an image triplet is provided as input. For example, consider the top example in Fig. 1, where the transformation is changing a cabin into a castle. Previous conditional diffusion models ((a) and (b)) use a text instruction (*i.e.* "Make the cabin a castle") as well as B to generate the target image B'. On the other hand, in image analogy generation, B' is generated using only the image triplet without any text prompt.

Pioneered by DIA [39], several studies [10, 23, 25, 39, 45] have explored diffusion-based image analogy generation in terms of visual in-context learning or visual instruction. From the perspective of visual in-context learning, [10, 23, 45] treat the generation of B' as an image inpainting task, where A, A', B and initial noise are concatenated into a grid-based input image, and B' is denoised from noise (see Fig. 2 (b)). This approach is limited to pretrained inpainting models and often requires manual prompt engineering by human to obtain additional text prompts. Visual Instruction Inversion (VISII, Fig. 2 (c)) [25] inverts visual instructions from A to A' using instruction tokens but is also limited to InstructPix2Pix [2]. Moreover, to efficiently optimize the visual instruction, VISII initializes the instruction tokens with image A', which leaves residual information from A' in the instruction, leading to unintended artifacts in B'. For instance, in the fourth example of Fig. 1 (e), the transformation from A to A' involves adding a rainbow, so the desired transformation for B to B' would similarly involve adding only the rainbow. However, in the images generated by VISII, residual information from A' (*i.e.* mountains) remains, leading to unintended elements in the output.

In this paper, we introduce Difference Inversion, which leverages "Full Prompt" with both prompt tokens representing the input image and Difference Tokens that exclusively encode difference information. Difference Tokens can focus on only the difference (Delta) between A and A' and the combined tokens are more appropriate as an input of Stable Diffusion models. This contrasts with VISII, which relies on an image-based "Instruction Prompt" that can only be applied as input to InstructPix2Pix. To accurately disentangle the difference, we propose Delta Interpolation to define a more precise Delta. Additionally, we introduce Token Consistency Loss, which ensures that the difference

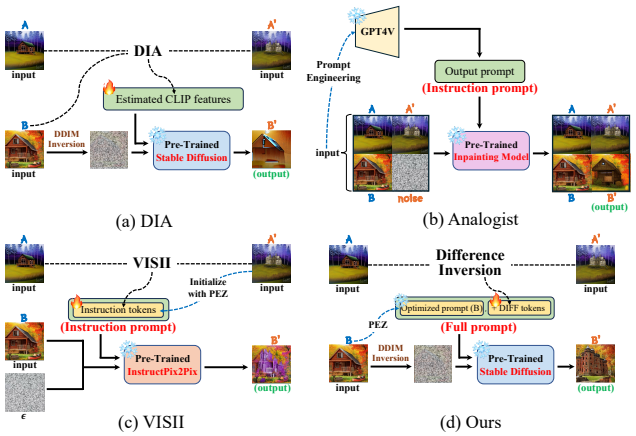


Figure 2. **Architectural comparison of image analogy generation baselines.** We compare our approach with three baseline methods: (a) DIA, (b) Analogist, and (c) VISII. Detailed descriptions can be found in Sec. 2. It is noteworthy that, unlike (b) and (c), which each depend on specific models, our method can be applied to general stable diffusion models.

information emerges when the token is present and disappears when it is absent. By using Token Consistency Loss, the Difference Tokens trained on the Interpolated Delta can be seamlessly concatenated with any query image prompt, enabling their use without introducing unwanted artifacts. Finally, we zero-initialize the Difference Tokens to ensure stable optimization of the differences and to reduce bias by preventing any initial preference from being introduced.

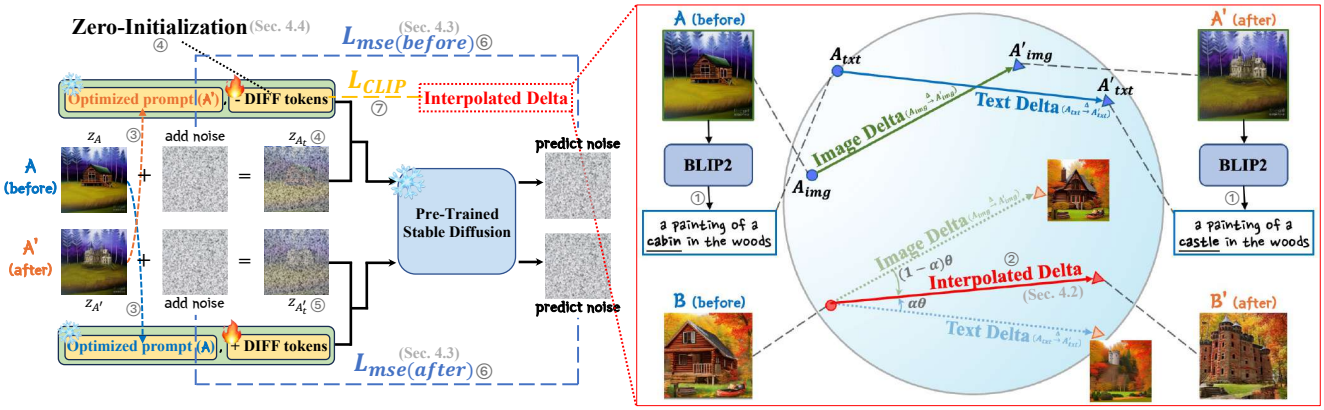
The main contributions of our paper are as follows:

- We propose Difference Inversion, which inverts precise differences into Difference Tokens and applies them to any query B to generate B' within general stable diffusion frameworks.
- To extract the exact difference between images A and A', we additionally introduce a Delta Interpolation, Token Consistency Loss and Zero-Initialization of Token Embeddings. These ensures that the extracted differences contain only the intended modifications, free from unwanted artifacts.
- Our extensive experiments demonstrate that Difference Inversion outperforms existing baselines both quantitatively and qualitatively, as well as in human and large-scale Vision Language Models (VLMs) evaluations.

2. Related works

2.1. Image Analogy Generation via Diffusion Models

The goal of image analogy generation [10, 23, 25, 39, 45] is to produce B' by conditioning on A, A' and B in the form of $A : A' :: B : B'$. DIA [39] attempts to invert images to obtain conditioning matrices for diffusion, enabling high-



(a) Overall Pipeline of Difference Inversion (b) Visualization of Interpolated Delta

Figure 3. **Overall pipeline of *Difference Inversion*.** Given an image triplet $\{A, A', B\}$, Difference Inversion extracts the difference between A and A' using DIFF tokens. The Difference is estimated as the Interpolated Delta, computed between A and A' by applying Spherical Linear Interpolation (Slerp) on the Image Delta and Text Delta, effectively capturing both visual and semantic information. The circled numbers in the figure correspond to the steps outlined in Algorithm 1.

level analogy through semantic vector operations. However it fails to capture not only the desired editing changes but also the detailed image information accurately. Starting with ImageBrush [45], several studies combined A,A',B, and B' into a single grid-formatted image and used a pretrained inpainting model to denoise the B' section from noise, generating a plausible image for B'. However, relying solely on a pretrained inpainting model makes it challenging to generate a plausible B'. Leveraging the knowledge of large language models (LLMs) or large-scale vision language model (VLMs) is often necessary, which consequently requires cumbersome prompt engineering. In a different approach, VISII [25] achieves $A \rightarrow A'$ transformations by inverting visual instructions into text tokens and applying them to query image (*e.g.* B) without relying on an VLMs. However, this process leaves unwanted information from A and A' in B'. As a result, B' may be generated in an unintended direction as shown in Fig. 1 (c). Additionally, since the inverted instruction can only be used with InstructPix2Pix [2], it is challenging to apply this method to various Stable Diffusion models, resulting in low generalizability. In this paper, we introduce Difference Inversion that precisely extracts only the difference between A and A' and applies it to B, avoiding unintended artifacts. Furthermore, since it forms a Full Prompt rather than an Instruction Prompt, it can be applied to a wide range of Stable Diffusion models.

2.2. Diffusion based Inversion

In the wide range of diffusion-based conditional generation methods, DDIM Inversion [24, 37], which identifies the initial noise corresponding to a given image, serves as a core component. Once the initial noise is identified by inversion, it is well-known that better results can be generated from the inverted noise when applied to personalized editing [9, 17, 40] or image-to-image translation [4, 11]. Alongside this trend, DreamBooth [33] and Textual Inversion [8] focus on optimizing special text tokens to learn representations that capture personalized information. ReVersion [15] also optimizes specific text prompts to capture and represent the relationship between two images. These inversion methods utilize a soft prompt approach, optimizing continuous embeddings. In contrast, PEZ [42] proposes hard prompt optimization to generate interpretable prompts that effectively capture the concept of a specific image. Similarly, our work optimizes Difference Tokens in a hard prompt manner, making them plug-and-play with other prompts, thereby enhancing their suitability for general Stable Diffusion models.

3. Preliminaries

Denoising diffusion models [14, 36] are generative models that learn the reversal of the forward process using denoising autoencoders, enabling the iterative sampling of images from noise. In the early stages of diffusion models [5, 14, 38], the forward and reverse diffusion process operates directly in the pixel space. Latent diffusion models [32], also well known as Stable Diffusion, performs this process in latent space, enabling more efficient image generation. Given an input image x and a text condition y, the image encoder transforms x into z_0. The denoising model ϵ_θ is then trained to predict a denoised version of its input z_t, as shown in the following equation.

$$\mathcal{L} = \mathbb{E}_{\mathcal{E}(x),y,\epsilon\sim\mathcal{N}(0,1),t}\|\epsilon - \epsilon_\theta(z_t, t, \tau_\theta(y))\|_2^2 \quad (1)$$

where z_t is a noisy variant of the original input z_0 in time t and τ_θ is text encoder. Since Stable Diffusion accepts text prompts as input, there have been attempts to add special tokens to text prompts for tasks such as personalized editing. Textual Inversion [8], for example, learns to capture specific concepts or objects using special token (s_*).

Similarly, VISII (Fig. 2 (c)) also utilizes instruction tokens C_T to learn the instruction information necessary for generating B'. Since VISII needs to learn the transformation instruction from image A to A' and apply them to B, it leverages InstructPix2Pix [2]. Furthermore, to better capture the transformation, C_T is initialized via discrete token optimization (PEZ [42]) on the image A'. Since VISII relies on InstructPix2Pix, it learns C_T in the form of an instruction prompt. In contrast, our Difference Inversion isolates only the difference information, enabling a plug-and-play approach with general stable diffusion prompts. Further details will be provided in the following section.

4. Method

Given an image triplet $\{A, A', B\}$, our goal is to extract the difference between (A, A') and apply it to B. To capture the difference, we optimize DIFF tokens ($\tilde{D}$) to effectively encode the extracted differences. We first introduce the basic concept of Difference Inversion in Sec. 4.1. In Sec. 4.2, we explain how we optimize the DIFF token using the Interpolated Delta. Then, in Sec. 4.3, we present the token consistency loss to enhance image consistency, followed by an introduction to token embedding initialization in Sec. 4.4.

4.1. Difference Inversion

We start from the key observation in VISII, which is the similar concept with our work. As shown in Fig. 1 (e), generating B' with VISII results in unwanted artifacts. This occurs because the optimized visual instruction contains not only the difference (*e.g.* rainbow) but also information specific to A and A' (*e.g.* mountains). Therefore, the key is to enable the inversion of only the precise difference.

In this paper, we define the difference as "Delta," which can be obtained as the difference between embedding vectors from the CLIP [30] encoder $\mathcal{E}_I$. Letting the CLIP image encoder be represented by $\mathcal{E}_I$, the Image Delta $\mathcal{D}_{img}$ can be expressed as follows:

$$\mathcal{D}_{img} = \mathcal{E}_I(A'_{img}) - \mathcal{E}_I(A_{img}) \tag{2}$$

To optimize DIFF tokens, we first initialize the text prompts of A and A' using PEZ. This step aims to fully anchoring the prompt of A (or A') and isolate the only difference, enabling more precise extraction of the difference with the DIFF token in place. As shown in Fig. 3 (a), we only optimize DIFF tokens while optimized prompt embedding is frozen. Note that while optimization with PEZ can be time-consuming, it can be replaced with text captions generated by BLIP2 [18], trading off a slight performance drop for faster processing (Sec. 4.3)

However, simply optimizing the DIFF token only with the Image Delta did not sufficiently capture the Difference information. As shown in Fig. 3 (b), B' with the Image Delta retains much of the original image's cabin features rather than fully transforming into the intended castle. We hypothesize that this is because the CLIP embedding space may not fully represent the intricate details of the images. In the following section, we introduce Delta Interpolation to extract a more refined difference.

Algorithm 1 Difference Inversion

Require: pretrained denoising model ϵ_θ; Image encoder $\mathcal{E}$; CLIP Image, Text Encoder $\mathcal{E}_I$, $\mathcal{E}_T$; Captioning model BLIP2; interpolation rate α; Learning rate γ; $t_{before}, t_{after} \sim \mathcal{U}(0, T)$; $\epsilon_{before}, \epsilon_{after} \sim \mathcal{N}(0, 1)$;
Input: Image pair $\{A, A'\}$
Output: Difference Tokens $\tilde{D} = \{D_1^*, ..., D_n^*\}$
// Extract the caption of A and A' ...①
cap_A = BLIP2(A); $cap_{A'}$ = BLIP2(A')
// Calculate Image Delta $\mathcal{D}_I$ and Text Delta $\mathcal{D}_T$
$\mathcal{D}_I = \mathcal{E}_I(A') - \mathcal{E}_I(A)$; $\mathcal{D}_T = \mathcal{E}_T(cap_{A'}) - \mathcal{E}_{txt}(cap_A)$
// Calculate Interpolated Delta $\mathcal{D}_{inter}$ ...②
$\mathcal{D}_{inter} = Slerp(\mathcal{D}_I, \mathcal{D}_T, \alpha)$
// Optimize the prompt w.r.t. A and A' ...③
$prompt_A$ = PEZ(A); $prompt_{A'}$ = PEZ(A')
Encode $z_A = \mathcal{E}(A)$; $z_{A'} = \mathcal{E}(A')$
// Zero-Initialization token embeddings ...④
Initialize $\tilde{D}$ with Zero embeddings
for $i = 1$ **to** N **do**
  // Prepare noisy latents z_A and $z_{A'}$ ...⑤
  $z_{A_t} \leftarrow$ add ϵ_{before} to z_A at timestep t_{before}
  $z_{A'_t} \leftarrow$ add ϵ_{after} to $z_{A'}$ at timestep t_{after}
  // Predict each noise $\hat{\epsilon}_{before}, \hat{\epsilon}_{after}$
  $\hat{\epsilon}_{before} = \epsilon_\theta(z_{A_t}, t_{before}, \{prompt_{A'}, -\tilde{D}\})$
  $\hat{\epsilon}_{after} = \epsilon_\theta(z_{A'_t}, t_{after}, \{prompt_A, \tilde{D}\})$
  // Compute Token Consistency loss ...⑥
  $\mathcal{L}_{tc} = \|\epsilon_{before} - \hat{\epsilon}_{before}\|_2 + \|\epsilon_{after} - \hat{\epsilon}_{after}\|_2$
  // Compute Clip loss $\mathcal{L}_{clip}$ ...⑦
  $\mathcal{L}_{clip} = \text{cosine}(\tilde{D}, \mathcal{D}_{inter})$
  $\mathcal{L} = \lambda_{tc} * \mathcal{L}_{tc} + \lambda_{clip} * \mathcal{L}_{clip}$
  Update $\tilde{D} = \tilde{D} - \gamma \nabla \mathcal{L}$
end for
Return: $\tilde{D}$

4.2. Delta Interpolation

Although the Image Delta effectively captures the visual difference, it still fails to fully reflect the specific semantic differences. To obtain more detailed semantic information, we use an image captioning model to extract cap-

tions for each of the images A and A'. We utilize BLIP2, but for more detailed captions, BLIP3 [44] or large-scale VLMs [3, 20, 21] can be also be considered. The extracted captions contain more specific details, such as "cabin" and "castle," which were used to derive the Text Delta D_{txt} that contains more semantic information than Image Delta.

$$\mathcal{D}_{txt} = \mathcal{E}_T(A'_{txt}) - \mathcal{E}_T(A_{txt}) \tag{3}$$

where $\mathcal{E}_T$ is CLIP text encoder.

After obtaining both Image and Text Delta of A and A' with CLIP, we apply Spherical Linear Interpolation (Slerp) [35] to integrate both embeddings. Since CLIP learns image-text pairs to maximize their cosine similarity, we hypothesize that it would be possible to find a more refined Delta within the joint hypersphere of the CLIP space. Therefore, to identify the Interpolated Delta $\mathcal{D}_{inter}$ on this hypersphere, we apply Slerp shown below.

$$\mathcal{D}_{inter} = Slerp(\mathcal{D}_{img}, \mathcal{D}_{txt}; \alpha) \tag{4}$$

$$= \frac{sin((1-\alpha)\theta)}{sin(\theta)}\mathcal{D}_{img} + \frac{sin(\alpha\theta)}{sin(\theta)}\mathcal{D}_{txt} \tag{5}$$

where α is interpolation ratio between Image and Text Delta. We set α to 0.8, and the ablation results are in Fig. 7.

Finally, we optimize the DIFF token to minimize cosine distance with this Interpolated Delta as:

$$\mathcal{L}_{clip} = \text{cosine}(\tilde{D}, \mathcal{D}_{inter}) \tag{6}$$

4.3. Token Consistency Loss

Using only the Interpolated Delta $\mathcal{D}_{inter}$ may resemble obtaining a visual instruction from A to A'. However, in Difference Inversion, it is also important to preserve aspects that should remain unchanged when generating B'. To achieve this, it is essential to accurately extract only the bidirectional difference between A and A'. Visual instruction considers only the transformation from $A \rightarrow A'$, whereas the difference must account for both $A \rightarrow A'$ and $A' \rightarrow A$. For DIFF tokens ($\tilde{D}$), this implies that when $\tilde{D}$ is present, the result should correspond to A'; otherwise, it should yield A.

Additionally, maintaining consistency in elements we do not intend to modify during this process is necessary. Inspired by Cycle Consistency Loss [43, 47], we introduce Token Consistency Loss, which ensures that A' is generated when $\tilde{D}$ is present and A when $\tilde{D}$ is absent. Specifically, the prompt concatenated with prompt A and $\tilde{D}$ should accurately reconstruct image A, while the prompt concatenated with prompt A' and $-\tilde{D}$ (reverse direction of $\tilde{D}$) should accurately reconstruct image A'. This can be expressed as follows:

$$\mathcal{L}_{mse_{before}} = \|\epsilon_{before} - \epsilon_\theta(z_{A_t}, t_{before}, \{prompt_{A'}, -\tilde{D}\})\|_2 \tag{7}$$

$$\mathcal{L}_{mse_{after}} = \|\epsilon_{after} - \epsilon_\theta(z_{A'_t}, t_{after}, \{prompt_A, \tilde{D}\})\|_2 \tag{8}$$

where $t_{before}, t_{after} \sim N(0, 1)$

The Token Consistency loss can be written as

$$\mathcal{L}_{tc} = \mathcal{L}_{mse_{before}} + \mathcal{L}_{mse_{after}} \tag{9}$$

Finally, to optimize the DIFF tokens $\tilde{D}$, we utilized the following objective:

$$\mathcal{L} = \lambda_{tc} * \mathcal{L}_{tc} + \lambda_{clip} * \mathcal{L}_{clip} \tag{10}$$

where λ_{tc} and λ_{tc} denotes weight parameter of each loss, respectively. We empirically set λ_{tc} to 0.01 and λ_{clip} to 6.

4.4. Zero-Initialization of Token Embeddings

Thus far, we have discussed the optimization process for the DIFF tokens ($\tilde{D}$). Finally, we address the approach for initializing $\tilde{D}$. It is crucial for the $\tilde{D}$ to capture accurate differences, making it essential to effectively disentangle the original image information from the difference. From this perspective, randomly initializing the $\tilde{D}$ can lead to unintended information being embedded within $\tilde{D}$ (see Config. C in Fig. 5). Motivated by ContorlNet [46], we initialize the token embeddings with zero to ensure that only the intended differences are captured within the $\tilde{D}$, minimizing unwanted artifacts. By initializing all $\tilde{D}$ to zero, we ensure that each token robustly encapsulates only the distinguishing differences from the original image. Detailed experimental results are presented in Sec. 5.

5. Experiments

5.1. Experimental and Implementation Detail

We used Stable Diffusion 2.1 as our baseline. All experiments were conducted on a single NVIDIA GeForce RTX 4090 GPU. For further details on the experimental setup, please refer to the supplementary materials.

Baselines. We compared our methodology with four baselines on image analogy generation. These methods all generate B' using only an image triplet $\{A, A', B\}$. Note that, some grid-image-based approaches (ImageBrush [45] and InstructGIE [23]) are also close with our works but since their code is not publicly available, we did not include them in the comparison. The implementation details for each baseline are provided in the supplementary materials.

Dataset. We randomly sample 300 instruct pairs from the InstructPix2Pix dataset to measure quantitative and qualitative results. The InstructPix2Pix dataset is structured as

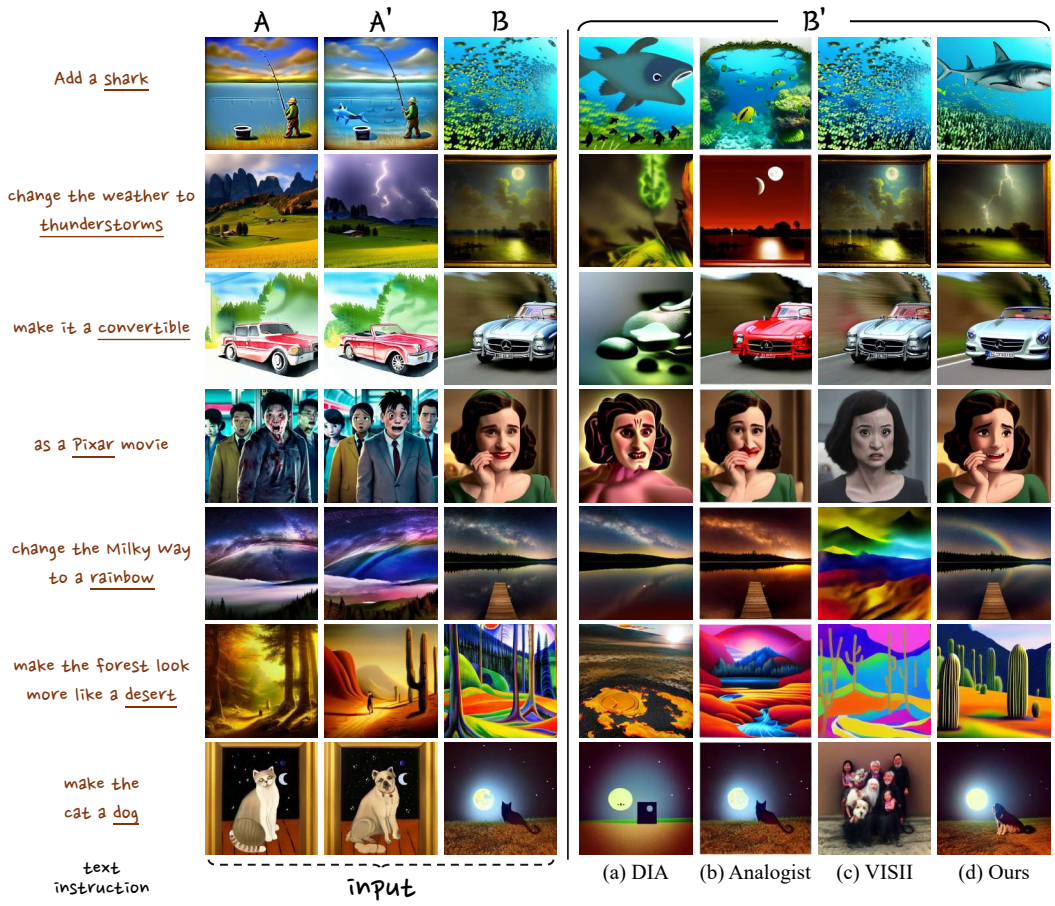


Figure 4. **Qualitative comparison to baseline methods.** Given inputs A, A' and B, we generate B' without utilizing any text instructions.

triplets in the form of {before image, after image, text instruction}. We utilized the before image and after image as A and A', respectively. B was randomly selected from other before images with the same text instruction. Note that we do not use the text instruction included in the dataset.

Evaluation Metric. We evaluate how well the difference is reflected using the Directional score using CLIP. We also measure the directional score using DINO-v2, which offers a more refined embedding space than CLIP. Additionally, we conduct a user study to verify the perceptual suitability of our results from a human perspective. And finally, leveraging the significant advancements in large-scale vision language models (VLMs), we perform prompt engineering to enable VLMs to evaluate our tasks, as shown in the figure 5. We adopt GPT-4o [27], Qwen2-72B [41] and Llama3.2-90B [6] for VLM evaluation, and the question prompts are provided in the supplementary materials. For human evaluation, we use a four-option multiple-choice format with randomized order of the options. On the other hand, for VLM evaluation, we employ a two-option format to minimize the influence of numbering and enable a more precise comparison. Further details about human and LLM evaluations are in the supplementary materials.

5.2. Qualitative Results

As shown in Fig. 4, our approach qualitatively generates significantly superior images compared to existing baselines. It not only accurately reflects the transformation from $A \rightarrow A'$ but also preserves as much information from B as possible, resulting in a plausible B'. In Fig. 1, (a), (b), and (c) depict results generated using only the images A, A', and B without text instructions, while (d) and (e) utilize text instructions. Across all baselines, our results produce a B' that best captures the before-after differences.

5.3. Quantitative Results

Directional score. As shown in Table 1, our model achieves the best performance among image-only baselines in terms of the directional score using CLIP. Note that VISII's CLIP score is nearly comparable to ours, which is attributable to the fact that the dataset used for evalua-

Metric	Image + Text		Image only			
	SDEdit	InstructPix2Pix	DIA	Analogist	VISII	Ours
CLIP (↑)	0.1105	**0.2000**	0.0294	0.0398	0.1007	**0.1024**
DINO-v2 (↑)	0.5374	**0.5723**	0.5114	0.5098	0.5414	**0.5732**

Table 1. **Directional Score of CLIP and DINO-v2.** We conducted quantitative evaluation by measuring the directional score using CLIP and DINO-v2. The directional score represents the embedding similarity between $A \rightarrow A'$ and $B \rightarrow B'$.

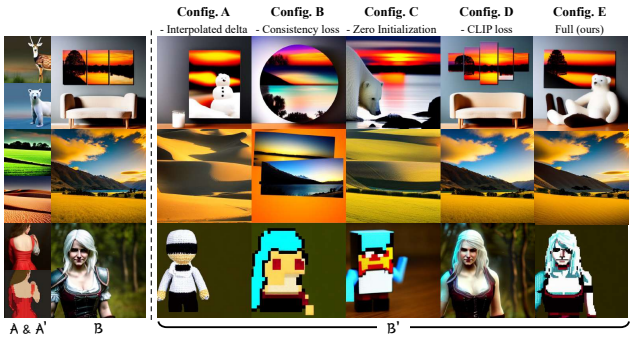


Figure 5. **Qualitative comparison with ablation studies for each proposed method.** Each column shows the results when the corresponding method ***is not applied.***

Model	DIA	Analogist	VISII	Ours
Percentage (%)	4%	15%	9%	72%

Table 2. **Human evaluation for image analogy generation**. In each question, participants were asked which B' would best complete the analogy $A : A' :: B : B'$. We report the percentage preference for Difference Inversion over 50 images evaluated by 60 people.

tion was also utilized in training VISII's backbone, InstructPix2Pix. In other words, VISII operates on a dataset it has already been trained on, whereas our model is evaluated on previously unseen data. Despite using Stable Diffusion as backbone, our model still performed exceptionally well. This gap becomes even clearer when measured with DINO-v2 [28] rather than CLIP. Furthermore, our DINO-v2 score surpasses even SDEdit and InstructPix2Pix, both of which utilize text descriptions.

Human evaluation. Although DINO-v2 offers a better directional score compared to CLIP, it still does not fully capture how plausible B' is in the $A : A' :: B : B'$ relationship. To address this, we conduct human evaluations to assess which B' generated by the various baselines was the most appropriate. We asked 60 participants to select the most suitable image analogy generation result (B') from four baselines, including our method. The order of models is randomly assigned for each evaluation, and the images

Model (vs Ours)	DIA	Analogist	VISII
GPT4o [27]	90%	76%	74%
Qwen2-VL-72B [41]	92%	70%	80%
Llama-3.2-90B [6]	72%	60%	62%

Table 3. **VLM evaluation for image analogy generation.** We provide the large-scale VLM with a prompt explaining the image analogy generation task and ask it to select the most suitable image for B', similar to the human evaluation (see in Sec. 5.1). The results above reflect the win rate of Difference Inversion compared to each baseline in pairwise comparisons, evaluated using 50 examples for each model (a total of 150 evaluations each).

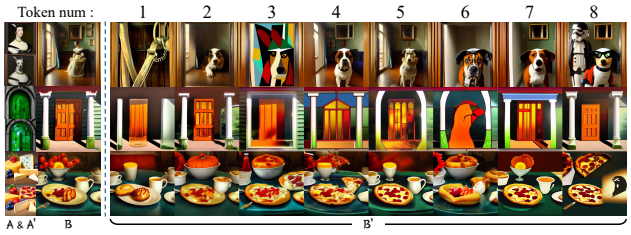


Figure 6. **Ablation results on the number of Difference tokens ($\tilde{D}$).** The fewer $\tilde{D}$ used, the less effectively they capture the difference information; however, using too many $\tilde{D}$ can lead to capturing additional, unrelated information. Experimentally, we found that using a total of 5 tokens strikes the right balance.

used in the survey are available in the supplementary materials. As shown in Table 2, our model significantly outperforms the other three baselines in the human evaluation.

VLM evaluation. We also conduct evaluation with large-scale VLMs known for their strong reasoning capabilities. We leverage GPT-4o *, Qwen2-VL-72B-Instruct †, and Llama-3.2-90B-Vision ‡ for evaluation, and unlike human evaluation, we employed a two-option format to compare our model against the baselines. As shown in Table 3, the preference for our model is significantly higher compared to existing baselines. This indicates that our method not only extracts the exact difference but also can generates plausible B', fulfilling the primary goal of image analogy generation.

5.4. Further Analysis

Ablation study. Lastly, we performed ablation experiments on each of our proposed methods. Fig. 5 presents the ablation results for Token Consistency Loss, Delta Interpolation, Zero Initialization and CLIP loss. When Token Consistency Loss is not used, we observed significant degra-

*https://chatgpt.com/
†https://huggingface.co/Qwen/Qwen2.5-72B-Instruct
‡https://huggingface.co/meta-llama/Llama-3.2-90B-Vision

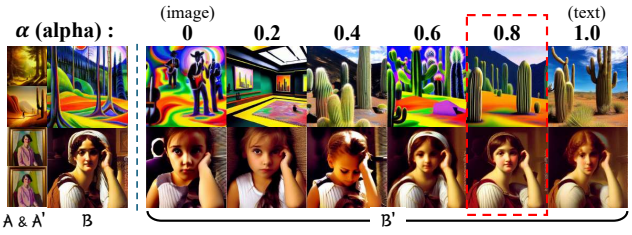


Figure 7. **Ablation results of α.** Since our delta ultimately needs to be optimized into text tokens, assigning a higher weight to α tends to better capture the desired differences. We finally set α to 0.8 for all experiments.

Figure 8. **Transferability to other Stable Diffusion models.** Our Difference Inversion can also be applied to other general Stable Diffusion models, such as SDXL.

dation in consistency with the original image (B). Without Delta Interpolation and using only Image Delta, the method fails to capture the difference effectively. Similarly, Zero Initialization of Token Embedding shows that when initialized randomly, the difference information is not properly reflected. The ablation results for the number of difference tokens and the interpolation ratio alpha are shown in Fig. 6 and 7, respectively. Additional hyperparameter search results for each method, including lambda tc and lambda clip, can be found in the supplementary materials.

Transferability. Since our Difference Inversion is an inference-based method that does not require access to the pretrained diffusion model weights, it can be applied even if the diffusion model is provided in an API form. Though instruction inversion models such as VISII are also inference-based, they depend solely on specific models (*e.g.* InstructPix2Pix). We test our method on SDXL, which possesses a stronger image prior and is capable of generating more complex and refined images. Fig. 8 demonstrates the application of Difference Inversion to SDXL.

6. Limitations and Discussions

Failure cases. Despite the strong performance of our model across various editing tasks compared to various baselines, there are still some failure cases. In Fig. 9, it is evident that Difference Inversion struggles to capture global differences (a) or detailed differences (b). For example,

(a) Global Difference (b) Detailed Difference

Figure 9. **Failure cases.** When the difference is complex or difficult to define precisely, Difference Inversion may struggle to capture it effectively.

global differences (a) such as weather shifts or transformations like image-to-sketch cannot be effectively applied to B' when optimized through text tokens. Similarly, for detailed differences (b), as shown in the top example of changing a boy to a crying boy, the approach may result in changing a girl to another girl or fail to accurately capture the intended crying expression altogether. In cases like adding cherries to a cake (bottom example), such changes may also be poorly reflected in B', highlighting its limitations in capturing intended variations.

Inherent limitations of the CLIP space. We hypothesize that the primary reason for the aforementioned failure cases stems from the inherent limitations of the CLIP space. Although we attempt to derive finer Delta using Image and Text Delta within the CLIP space, it does not always align well with human perception, potentially making it unstable for providing precise directional guidance. In the supplementary materials, we illustrate this using a colorization analogy on MNIST data, showing the CLIP similarity for transitions from ground truth $A \rightarrow A'$ and $B \rightarrow B'$. Notably, the CLIP similarity yields only around 0.5, indicating that optimization based on this directional signal is challenging. We anticipate that future approaches may involve using more refined embedding spaces from foundation models such as DINO-v2 or learning additional mapping networks to enhance similarity and stability.

7. Conclusion

In this paper, we propose Difference Inversion, a model-agnostic inversion method that applies only the differences to B' without introducing unwanted artifacts. Difference Inversion captures precise differences from A and A' and can be applied to any query image B. This approach aligns well with the goal of image analogy generation, resulting in the creation of feasible B'. Additionally, our method can be applied to various diffusion models, paving the way for new opportunities not only in image editing but also across diverse computer vision tasks using analogy formulations.

8. Acknowledgements

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2024-RS-2023-00258649, 50%) supervised by the IITP(Institute for Information Communications Technology Planning Evaluation). This research was also supported by the Institute of Information &communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. RS-2019-II190079, Artificial Intelligence Graduate School Program(Korea University), 1%) and the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT)(No. RS-2025-00562437 40%, No. RS-2024-00341514, 9%)

References

[1] Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schönlieb, and Christian Etmann. Conditional image generation with score-based diffusion models. *arXiv preprint arXiv:2111.13606*, 2021. 2

[2] 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, 3, 4

[3] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198, 2024. 5

[4] Bin Cheng, Zuhao Liu, Yunbo Peng, and Yue Lin. General image-to-image translation with one-shot image guidance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22736–22746, 2023. 3

[5] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2, 3

[6] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 6, 7

[7] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*. 2

[8] 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, 4

[9] Jing Gu, Yilin Wang, Nanxuan Zhao, Tsu-Jui Fu, Wei Xiong, Qing Liu, Zhifei Zhang, He Zhang, Jianming Zhang, Hyun-Joon Jung, et al. Photoswap: Personalized subject swapping in images. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[10] Zheng Gu, Shiyuan Yang, Jing Liao, Jing Huo, and Yang Gao. Analogist: Out-of-the-box visual in-context learning with image diffusion model. *ACM Transactions on Graphics (TOG)*, 43(4):1–15, 2024. 2

[11] 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. 3

[12] Aaron Hertzmann, Charles E Jacobs, Nuria Oliver, Brian Curless, and David H Salesin. Image analogies. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 557–570. 2023. 2

[13] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 2

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3

[15] Ziqi Huang, Tianxing Wu, Yuming Jiang, Kelvin CK Chan, and Ziwei Liu. Reversion: Diffusion-based relation inversion from images. *arXiv preprint arXiv:2303.13495*, 2023. 3

[16] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6007–6017, 2023. 2

[17] 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. 3

[18] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 4

[19] Jing Liao, Yuan Yao, Lu Yuan, Gang Hua, and Sing Bing Kang. Visual attribute transfer through deep image analogy. *arXiv preprint arXiv:1705.01088*, 2017. 2

[20] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024. 5

[21] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024. 5

[22] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021. 2

[23] Zichong Meng, Changdi Yang, Jun Liu, Hao Tang, Pu Zhao, and Yanzhi Wang. Instructgie: Towards generalizable image editing. In *European Conference on Computer Vision*, pages 18–34. Springer, 2025. 2, 5

[24] 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. 3

[25] Thao Nguyen, Yuheng Li, Utkarsh Ojha, and Yong Jae Lee. Visual instruction inversion: Image editing via image prompting. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3

[26] 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

[27] OpenAI. Openai: Introducing chatgpt, 2022. 6, 7

[28] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 7

[29] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 2

[30] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 4

[31] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 2

[32] 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. 2, 3

[33] 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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023. 3

[34] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 2

[35] Ken Shoemake. Animating rotation with quaternion curves. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pages 245–254, 1985. 5

[36] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015. 3

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

[38] 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. 3

[39] Adéla Šubrtová, Michal Lukáč, Jan Čech, David Futschik, Eli Shechtman, and Daniel Sỳkora. Diffusion image analogies. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–10, 2023. 2

[40] Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. p+: Extended textual conditioning in text-to-image generation. *arXiv preprint arXiv:2303.09522*, 2023. 3

[41] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 6, 7

[42] Yuxin Wen, Neel Jain, John Kirchenbauer, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Hard prompts made easy: Gradient-based discrete optimization for prompt tuning and discovery. *Advances in Neural Information Processing Systems*, 36, 2024. 3, 4

[43] Sihan Xu, Ziqiao Ma, Yidong Huang, Honglak Lee, and Joyce Chai. Cyclenet: Rethinking cycle consistency in text-guided diffusion for image manipulation. *Advances in Neural Information Processing Systems*, 36, 2024. 5

[44] Le Xue, Manli Shu, Anas Awadalla, Jun Wang, An Yan, Senthil Purushwalkam, Honglu Zhou, Viraj Prabhu, Yutong Dai, Michael S Ryoo, et al. xgen-mm (blip-3): A family of open large multimodal models. *arXiv preprint arXiv:2408.08872*, 2024. 5

[45] Yifan Yang, Houwen Peng, Yifei Shen, Yuqing Yang, Han Hu, Lili Qiu, Hideki Koike, et al. Imagebrush: Learning visual in-context instructions for exemplar-based image manipulation. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3, 5

[46] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 5

[47] 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*, pages 2223–2232, 2017. 5