Image-Editing Specialists: A Multi-Objective Approach for Diffusion Models

Elior Benarous
Harvard University
ETH Zürich
ebenarous@ethz.ch

Yilun Du
Harvard University
Google DeepMind

Heng Yang
Harvard University

(a) Input Image (b) *"Add snow on the road"* (c) *"Change the road into wood"* (d) *"Add rain on the road"* (e) *"Change the time to nighttime"*

Abstract

[Pending abstract] We present a method for training image-editing diffusion models based of textual instructions and visual prompts to capture subtle user preferences without the need for long descriptive prompts. It achieves so while requiring neither a large and curated training data or human feedback. Our method largely improves the alignment with instructions in two ways. First, it localizes the region to edit far better, largely reducing the modification of regions in the image that are unrelated to the instruction. Second, it captures fine nuances in the desired edit by leveraging a visual prompt, which considerably simplifies the user's efforts to attain a highly specific edit. We show that our model can perform intricate edits in complex image like busy road scenes, and can also perform sim2real transfer that greatly helps applications where real data is scarce.

1. Introduction

Text-to-image (T2I) generative models have achieved remarkable success in creating visually appealing images based on text prompts [24, 43, 46], largely due to advancements in aligning captions with images [1, 41].

Beyond the remarkable generative capabilities of T2I models, instructional image editing has become one of the most practical approach to semantic modification [11, 53]. Unlike traditional image editing [12, 21, 27, 32, 37, 62], which necessitates descriptive captions for both the input and modified images, instruction-based image editing requires only natural-language instructions. This method is more straightforward, as it requires specifying only the elements that need alteration in the original image, without involving the remaining unrelated attributes.

Although instructional editing has gained popularity in consumer applications, it has yet to show effective reusability of its samples as data augmentation in downstream model training. Simple-to-generate synthetic samples have shown to serve as pretraining data to teach early-training features [7, 8, 26]. Recently, some studies focused on leveraging T2I models as sources of high quality data augmentation. Some works generate diverse samples by applying naive perturbation to the noisy latents [61, 77], the conditioning embeddings [79] or throughout the denoising process [19]. Others do so by designing, optimizing, or reformulating prompts [6, 16, 50, 55, 56, 73]. Further, Azizi et al. [3] finetune a T2I model to reduce the gap between real data from a specific domain and class-conditional generated samples. Still, none leverage instructional editing for data augmentations that are closely aligned with the user's intentions. In our work, we focus on a method to train instruction-based diffusion models to generate samples that are of high precision and follow instructions with high fidelity. We show that such specialized models can serve as highly useful training data generators, particularly in data-scarce domains, where the collection might be costly or timely.

To become truly useful for downstream applications, we identify two key criteria that an image-editing model should satisfy.

Structural Alignment: It should ensure high-fidelity preservation of content in the initial image that is unrelated to the editing request. Current state-of-the-art image-editing models without segmentation or semantic masks in the forward pass struggle to perform precise edits on the subject specified by the text prompt, while leaving unrelated areas of the image intact. Notably, the training data for InstructPix2Pix was created with the Prompt-to-Prompt method [21]. Hence, it suffers from the same limitations as Prompt-to-Prompt, including imperfect preservation of background pixels.

Semantic Alignment: It should enable fine-grained control over the visual aspects of desired modifications. One effective approach is to utilize both textual and image prompts to express nuanced stylistic preferences. This strategy captures subtleties that are difficult to articulate with text alone and may elude language models. While a text prompt can convey the broad theme of an edit, specifying the precise nuances of its interpretation proves more challenging and impractical. Consequently, it is preferable to enhance user alignment with their envisioned edits without resorting to lengthy prompts, which contradicts the essence of instruction-based systems. Most works on visual prompting for image generation have been targeted at style transfer, where an image's style is edited across the whole frame [22, 57, 64]. Other recent studies have focused on finetuning pre-trained T2I models for subject-driven editing from a set of reference images [20, 48]. However their methods require unique identifiers to capture the edit nuance. In our work, we bring our attention to localized edits conditioned on a visual prompt, and limit the complexity of our text prompts to simple instructions.

The standard diffusion model training objective presents inherent challenges for image editing because it enforces uniformly reconstructing the entire image, without distinguishing between regions that require preservation and those that need modification. Achieving this balance with a standard reconstruction loss necessitates an oracle to generate precise input-output pairs. However, such oracles are often biased, lack scalability, or produce low-quality samples that require further pruning using various filtering techniques [11, 13, 53, 74]. To address these limitations, we frame the problem as an alignment issue, leveraging the capabilities of the InstructPix2Pix model [11]. Although this model can generate edits, it requires refined training to better align with human semantic and structural preferences, especially in complex scene edits. Our reinforcement learning (RL) framework enhances this alignment and circumvents the need for an oracle, as the method is based on self-play. By achieving closer alignment with human preferences, the model is more likely to generate natural-looking edits with better visual appeal, making it a suitable method for high-quality data augmentation.

Accordingly, we introduce a novel RL approach designed to specialize image editing models into producing edits with styles highly aligned with visual prompts while accurately preserving the original structure in non-pertinent areas of the image. Unlike traditional reinforcement learning with human feedback (RLHF) methods, our approach bypasses the need for human annotators by employing AI models to provide feedback that accurately simulates human preferences in terms of structure and semantics. We demonstrate our method's effectiveness in editing complex scenes, such as outdoor city street environments for autonomous driving applications, and its utility in data augmentation for downstream robot manipulation policy training. Our experiments show that it surpasses current state-of-the-art methods in structural preservation and instruction alignment. Our contributions are summarized as follow:

1. We design a framework to train image-editing models with RLAIF
2. We craft a multi-objective approach tailored to learning visual nuances beyond simple textual instructions
3. We showcase the quality of the samples by using them as effective training data for data-scarce applications

2. Related Works

Text-guided Image Editing. In prior work, text-guided image editing methods fall in three categories: architectural tricks that require no training, few-steps optimization for each sample, and large scale finetuning. In the first category, Prompt-to-Prompt (P2P) [21] manipulates attention maps in the diffusion model to control the layout and content of the editing. Plug-and-Play (PNP) [62] injects self-attention maps and spatial features to improve the structure. Still, these remain limited in their ability to edit complex scenes. Others leverage a segmentation mask [2, 12, 33, 39, 54, 65, 68] or semantic mask [36, 38, 72, 75] during the forward pass, which imposes a greater inconvenience on users by requiring them to supply these additional masks.

In the second category, Null-Text Inversion [37] optimizes the null-text embedding during the inversion of an input image. Imagic [27] finetunes model weights and embeddings to align with the input image and the edit text prompt. RB-Modulation [47] leverages stochastic optimal controller to align content and style with visual prompts. Still, these are slow at inference because optimization is requires for each generation.

Lastly, other methods adopt the standard reconstruction training. like [11, 13, 52, 53, 74]. InstructPix2Pix [11] generates a large synthetic dataset of image pairs with P2P, coupled with instruction-based text prompts. Similarly, Emu Edit [53] expands the dataset with P2P, screens training samples with semantic and structural filters techniques, and leverages multi-task training. SuTI [13] leverages finetuned experts version of Imagen [49] to create high quality samples for the editing model to learn from. MagicBrush [74] assembles an image editing dataset synthesized with

DALL-E 2 [43] and manually prune samples. Alchemist [52] uses a rendering tool targeted to modifying material attributes. However, these methods require an oracle to generate images, which requires some curation, and may still infuse its limitations.

Reinforcement Learning for Diffusion. Aligning model outputs with human preferences has seen a wide success in the field of language modeling. For objectives that are complex to define explicitly, a popular strategy is reinforcement learning with human feedback (RLHF) [4, 15, 40, 58], where we first teach a reward function to capture output preferences, and leverage reinforcement learning algorithms like proximal policy optimization [51] to finetune models with such rewards.

In the field of diffusion models, several works study the use of human feedback for T2I generation. [30] collects human annotations, and perform maximum likelihood training where the reward is naively used as a weight. [67] designs a reward model that better captures finegrained human preferences. [9, 17] show that diffusion models can be trained with RL using a reward model judging images' aesthetics [69]. Specific to instructional image-editing, HIVE [76] extends large-dataset supervised training by collecting human feedback on edits and performing off-policy RLHF training. However, these methods need a reward model trained on large-scale human annotation. Not only is this process cumbersome, but it also provides supervision of limited quality. First, it quickly becomes hard for humans to evaluate at a finegrained level the preservation faithfulness of pixels unrelated to the edit instruction. Second, semantic alignment remains vague as it is only compared to the short instruction prompts, where different individuals could easily disagree on the specific interpretation.

Inspired by advances with RLAIF [5, 29], we alleviate the need for humans-in-the-loop and opt for a method where AI models provide the preference supervision tailored to solving the two issues above. Further, we train on-policy by leveraging the framework established by D3PO [70], positing that using online samples would lead to better results.

3. Method

In this section, we describe the custom objective designed to obtain parallel supervision for the semantic and structural alignment. In Sec. 3.1, we describe how to alleviate the need for a reward model. Then, we explain in Sec. 3.2 how we design our two separate objectives. Finally, in Sec. 3.3, we present the modified architecture to intake the additional visual prompt conditioning and its modified score estimate formulation for classifier-free guidance with three conditionings.

3.1. Reinforcement Learning Training of Diffusion Models

Our model should learn from a reward that captures the structural and semantic alignment. For such, the reward function must intake the input image to be edited I_{og}, the instruction prompt c_T, the target style image I_{sty}, and compare those to the generated edit I_{gen}.

Most RLHF methods train a reward model to then train a downstream model. However, Direct Preference Optimization (DPO) [42] showed that preference ranking can be used to train language models and circumvent reward models. [63] showed that this could be extended to diffusion models. In our work, we leverage the framework introduced by D3PO [70], which expands that of DPO into a multi-step Markov Decision Process (MDP).

Given a pair of outputs $(y_1, y_2) \sim \pi_{\text{ref}}(y|x)$ generated from a reference pre-trained model π_{ref}, we denote the preference as $y_w \succ y_l|x$ and store the raking tuple (x, y_w, y_l) in dataset $\mathcal{D}$, where y_w and y_l are the prefered and dispreferred samples respectively. Following the Bradley-Terry model [10], the human preference distribution p^* can be expressed by using a reward function r^* as:

$$p^*(y_1 \succ y_2 \mid x) = \frac{\exp(r^*(x, y_1))}{\exp(r^*(x, y_1)) + \exp(r^*(x, y_2))} \quad (1)$$

A parametrized reward model r_ϕ can then be trained via maximum likelihood estimation to approximate r^* with:

$$\mathcal{L}_R(r_\phi, \mathcal{D}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \left[\log \rho(r_\phi(x, y_w) - r_\phi(x, y_l))\right] \quad (2)$$

where ρ is the logistic function. Prior works in RL have for objective to optimize a distribution such that its associated reward is maximized, allthewhile regularizing this distribution with the KL divergence to remain similar to its initial reference distribution:

$$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi_\theta(y|x)} \left[r_\phi(x, y)\right] - \beta \mathbb{D}_{\text{KL}} \left[\pi_\theta(y|x) \parallel \pi_{\text{ref}}(y|x)\right] \quad (3)$$

where β controls the deviation between π_θ and π_{ref}. This distribution takes the following for optimal solution:

$$\pi_r(y \mid x) = \frac{1}{Z(x)} \pi_{\text{ref}}(y \mid x) \exp\left(\frac{1}{\beta} r(x, y)\right) \quad (4)$$

where $Z(x) = \sum_y \pi_{\text{ref}}(y|x) \exp\left(\frac{1}{\beta} r(x, y)\right)$ is the partition function. Reorganizing Eq. 4, we obtain the expression for the reward as a function of its associated optimal policy.

$$r(x, y) = \beta \log \frac{\pi_r(y \mid x)}{\pi_{\text{ref}}(y \mid x)} + \beta \log Z(x) \quad (5)$$

Substituting the parametrized reward function and policy for their optimal counterparts, we reintegrate that expression into Eq. 2. With the change of variables, the loss function is now expressed over policies rather than over reward functions. This closed form avoids having to train a reward model, but rather allows us to directly optimize the model.

$$\mathcal{L}_{\text{DPO}}(\pi_\theta;\pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\rho\left(\beta\log\frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta\log\frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}\right)\right] \tag{6}$$

Extending this to diffusion models, we note a key difference to the described framework. The output is not generated from a single forward pass, but rather a sequential process. To address this, we pose the T-horizon MDP formulation, adapted from [59], for the T-timesteps long denoising process.

$$\begin{aligned}
&\boldsymbol{s}_t = (\boldsymbol{x}_{T-t}, \boldsymbol{c}, t) \qquad P_0(s_0) = (\mathcal{N}(\mathbf{0},\boldsymbol{I}), p(\boldsymbol{c}), \delta_0)\\
&\boldsymbol{a}_t = \boldsymbol{x}_{T-t-1} \qquad\quad P(\boldsymbol{s}_{t+1}\mid \boldsymbol{s}_t, \boldsymbol{a}_t) = (\delta_{\boldsymbol{x}_{T-t-1}}, \delta_c, \delta_{t+1})\\
&r(\boldsymbol{s}_t,\boldsymbol{a}_t) = r((\boldsymbol{x}_{T-t}, \boldsymbol{c}, t), \boldsymbol{x}_{T-t-1})\\
&\pi(\boldsymbol{a}_t \mid s_t) = p_\theta(\boldsymbol{x}_{T-t-1}\mid \boldsymbol{x}_{T-t}, \boldsymbol{c}, t)
\end{aligned}$$

where $p_\theta(\boldsymbol{x}_{0:T}|\cdot)$ is a T2I diffusion model, δ is the Dirac delta distribution, and $\boldsymbol{c}$ is the conditioning distributed according to $p(\boldsymbol{c})$. Note that we disregard r as our method circumvents it. With such, we treat the denoising process as a sequence of observations and actions: $\sigma = \{s_0, a_0, s_1, a_1, ..., s_{T-1}, a_{T-1}\}$. Since we can only judge the denoised output, we would need to update $\pi_\theta(\sigma) = \prod_t^T \pi_\theta(s_t, a_t)$, which is intractable. Following [70], we assume that if the final output of a sequence is preferred over that of another sequence, then any state-action pair of the winning sequence is preferred over that of the losing sequence. Hence, we determine the preferred sequence by sampling an initial state $s_0 = s_0^w = s_0^l$, generating two independent sequences, and ranking their output. Accordingly, we express the objective at a certain timestep as:

$$\mathcal{L}_t(\pi_\theta) = -\mathbb{E}_{(s_t,\sigma_w,\sigma_l)}\left[\log\rho\left(\beta\log\frac{\pi_\theta(a_t^w|s_t^w)}{\pi_{\text{ref}}(a_t^w|s_t^w)} - \beta\log\frac{\pi_\theta(a_t^l|s_t^l)}{\pi_{\text{ref}}(a_t^l|s_t^l)}\right)\right] \tag{7}$$

3.2. Multi-objective Joint Training

Now that we know how a preference ranking is converted into a loss that can train our diffusion model, we explain how we rank two generated sample based off their structural alignment with the input image, and their semantic alignment with the text and visual prompts. Our approach splits each objective into two separate scores.

Structural score. To obtain a score for how well the structure of the input image has been preserved, we leverage monocular depth estimation models [71]. Given a pair of input and edited images, we compute the depth map of each image independently, and define our structural score as the L_1 distance between the two resulting depth maps.

$$\mathcal{L}_{struct} = \frac{1}{h\cdot w}\sum_{i,j}^{h,w} |f_\phi(I_{og})_{i,j} - f_\phi(I_{gen})_{i,j}|$$

where f_ϕ is the depth model. With this metric, we capture any missing, additional or deformed elements in the edit compared to the input.

Semantic score. Contrary to the structural score, the semantic alignment should only be measured inside the region of the generated image where the edit is expected. Accordingly, we isolate the relevant regions with a text-conditioned segmentation model to locate the element to be edited. Here, we use grounded-SAM2 [34, 44, 45]. Same goes for the visual prompt, whose style may not be covering the entire frame. Subsequently, we determine the semantic alignment score by computing the distance between the embeddings of relevant patches in the generated and target style images. Additionally, we find that adding the standard pixel-space reconstruction objective on the instruction-irrelevant pixels, which should remain identical, enforces better localization and sharper bounds of the region of the image that must be edited. This serves as a regularizer to prevent the style from spreading over the whole generated frame to unrelated regions. Accordingly, our semantic score is:

$$\begin{aligned}
\mathcal{L}_{sem} = &D(m_{og}\odot I_{og},\ m_{sty}\odot I_{sty},\ f_\theta)\\
&+\lambda\cdot(1-m_{og})\odot\|I_{og}-I_{gen}\|_2
\end{aligned}$$

where $D(\cdot)$ is a distance metric, m_{og} and m_{sty} are the binary segmentation map obtained from the input and style images respectively, $\odot$ defines the element-wise multiplication, here the cosine distance, f_θ is an encoder, and λ is a hyperparameter to weigh the influence of the pixel reconstruction objective relative to the semantic one. We find empirically that $\lambda = 0.05$ achieves the ideal balance.

Similar to [78], we obtain *advantages* [60] by normalizing the scores on a per-batch basis using the mean and variance of each training batch. We then combine the distinct advantages into a unique score, with their relative contribution weighed by a hyperparameter α.

$$\mathcal{L}_{total} = \hat{A}_{struct} + \alpha\cdot\hat{A}_{sem}\ \text{, where }\ \hat{A} = \frac{\mathcal{L}-\mu_{\mathcal{L}}}{\sqrt{\sigma_{\mathcal{L}}^2+\epsilon}}$$

Finally, we rank two sequences according to the score of their respective output.

3.3. Architecture for Multiple Conditionings

Our architecture is based on that of InstructPix2Pix [11], itself is adapted from Stable Diffusion [46] with one main architecture modification. That is, they add input channels to the first convolutional layer to intake the encoded input image. We extend this and add more input channels to the first convolutional layer to intake both the input and style images. The weights that operate on the newly added input channels are initialized to zero. In practice, we find that applying a cross-attention layer before feeding the visual prompt improves performance, as it helps identify regions of both images relevant to the instruction. Here, the query is the linear projection of the concatenation of $\mathcal{E}(I_{og})$ and $\mathcal{E}(I_{sty})$, and the key and query are obtained by projecting the CLIP encoding of the instruction prompt.

Accordingly, we adapt the score estimate formulation as follows. Classifier-free guidance (CFG) [23] shifts probability mass where an implicit classifier assigns high likelihood to the conditioning, improving visual quality and faithfulness of samples. The standard CFG estimate is:

$$\tilde{e}_\theta(z_t, c) = e_\theta(z_t, \varnothing) + s \cdot (e_\theta(z_t, c) - e_\theta(z_t, \varnothing))$$

InstructPix2Pix adds a conditioning on the input image to be edited on top of the text instruction, further disentangling the score estimate yields:

$$\begin{aligned}\tilde{e}_\theta(z_t, c_I, c_T) = {} & e_\theta(z_t, \varnothing, \varnothing) \\ & + s_I \cdot (e_\theta(z_t, c_I, \varnothing) - e_\theta(z_t, \varnothing, \varnothing)) \\ & + s_T \cdot (e_\theta(z_t, c_I, c_T) - e_\theta(z_t, c_I, \varnothing))\end{aligned}$$

In our case, we find it beneficial to extend CFG with respect to the additional visual prompt, and use the following (see Appendix 6.1 for the complete derivation):

$$\begin{aligned}\tilde{e}_\theta(z_t, c_{I_{og}}, c_{I_{tgt}}, c_T) = {} & e_\theta(z_t, \varnothing, \varnothing, \varnothing) \\ & + s_{I_{og}} \cdot (e_\theta(z_t, c_{I_{og}}, \varnothing, \varnothing) - e_\theta(z_t, \varnothing, \varnothing, \varnothing)) \\ & + s_{I_{tgt}} \cdot (e_\theta(z_t, c_{I_{og}}, c_{I_{tgt}}, \varnothing) - e_\theta(z_t, c_{I_{og}}, \varnothing, \varnothing)) \\ & + s_T \cdot (e_\theta(z_t, c_{I_{og}}, c_{I_{tgt}}, c_T) - e_\theta(z_t, c_{I_{og}}, c_{I_{tgt}}, \varnothing))\end{aligned}$$

4. Experiments

incomplete This section presents the experimental results and ablation studies of our technical choices, demonstrating the effectiveness of our method. We apply the default guidance scale parameters ($s_{I_{\text{in}}}$ and s_T) from InstructPix2Pix for a fair comparison.

We evaluate our method focusing on the ability to perform located edits in complex scenes. Accordingly, we choose crowded street images from the Oxford RobotCar dataset [35] as our input images to be edited. We train our method to learn 7 types of edits, all related to modifying the road, as it is guaranteed to remain visible on every image without requiring complex curation of the input images. The edit requests range from realistic weather modification like adding snow on the road, to stylistic material modification like turning the road into wood. We also showcase the ability to capture subtle instruction nuances beyond the text prompt by training two models with the same text instruction but two different styles of visual conditionings.

Additionally, we show the ability of our model to produce edits that serve as training data in a data-scarce domain.

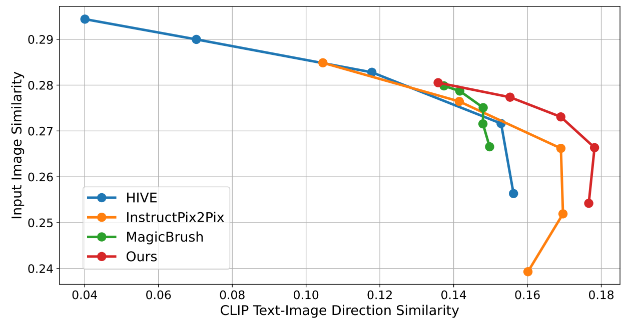

Figure 2. Comparison between instructional editing models. We plot the trade-off between consistency with the input image (Y-axis) and consistency with the edit (X-axis). For both metrics, higher is better. For all methods, we fix the same parameters as in [11] and vary the $s_{I_{in}} \in [1.0, 2.0]$

4.1. Baseline Comparisons

althought he research community generaly measures the alignment with clip or dino, we find that leveragign dreamsim is best, as it is trained to be aligned with humans, and even offer an ensemble method based on finetuned versions of clip and dino architectures. dino captures finegrained, clip does not capture enough detail, dreamsim is inbetween sweet spot

First, models trained in supervised learning methods are most likely to have biases, snow could be falling, rain could be falling, snow and rain might look a certain way, thickness, sparseness, color etc.

We train over 20 steps (or hours, nb samples) with a batch size of 512.

We report quantitative results of our semantic alignment cosine distance, and comparison with InstructPix2Pix [11], and MagicBrush [74] on these weather domains. We argue that this is a better loss to evaluate image edits, as it exclusively considers the region of interest, rather than what other metrics do when they encode the entire image. We argue that disentangling the reported metrics between the pixels that should be edited and the ones that should not, better highlights the precision of the edit. For instance, EmuEdit and MagicBrush report the L1 and L2 scores for the whole image, stating that a lower score is better. This makes sense to quantify how well regions in the image that are unrelated to the instruction prompt have been preserved. However, it is counterintuitive to measure the quality of the pixels that are expected to have been edited with such metric, as a lower score would indicate a lower strength of editing. As in the presented method, we rely on grounded SAM2 to produce those masks. We find that it produces high quality masks that often surpass the contour precision of objects compared to human-made masks provided in public benchmarks [74].

Method	$\text{Depth}_{\text{out}}\downarrow$	$\text{Depth}_{\text{all}}\downarrow$	$\text{L}_{2_{\text{out}}}\downarrow$
NTI-P2P	21.06	29.18	**0.014**
IP2P	35.93	50.52	0.064
MBrush	46.06	60.69	0.054
HIVE	21.56	29.47	0.036
Ours	**14.16**	**18.39**	0.062

Table 1. Pixel Structure Preservation Metrics (Averaged over 8 datasets x 3000 generated samples)

We then evaluate the alignment between our visual prompt and the generated edit. As expected, other methods that do not have the ability to leverage a visual prompt cannot mimick the nuance found in the conditioning. We further show that leveraging the visual prompt improves the alignment with the text prompt.

Also, we show that our model can interpret nuances beyond a text prompt, Specifically, we train two separate models to add a thick and a sparse layer of snow on the road, respectively, without being told this information in the text prompt.

Method	$\text{STY}_{\text{in}}\uparrow$	$\text{CLIP}_{\text{txt}}\uparrow$
NTI-P2P	0.244	0.216
IP2P	0.264	0.238
MBrush	0.250	0.234
HIVE	0.248	0.224
Ours	**0.282**	**0.251**

Table 2. Semantic Alignment Metrics Meaned across 3 encoders (Averaged over 8 datasets x 3000 generated samples)

Method	HPSv2 ↑	Image Reward ↑	Pick Score ↑
NTI-P2P	20.29	-0.950	18.93
IP2P	21.85	-0.314	18.92
MBrush	21.92	-0.454	18.84
HIVE	21.61	-0.567	18.89
Ours	**22.12**	**-0.134**	**18.94**

Table 3. Image Aesthetics Metrics (Averaged over 8 datasets x 3000 generated samples)

We then compare the score obtained from aesthetics judging models, namely HPSv2 [66], Image Reward [69], and Pick Score [28], which are all trained to mimick human preferences. We find that our model outperforms all other baselines across the different edits.

4.2. Ablation Study

Classifier-free-guidance score How does the parameter s_{sty} influence the generated samples

Information Encoders Different encoders capture different information in their representations, guiding the learning process with distinct models leads to different outputs. Other works commonly use Dino and CLIP. However, recently, Dreamsim [18] showed to outperform those encoders in alignment with human preferences. Similarly, we find that this encoder captures the best balance between high and low level features compared to Dino and CLIP.

4.3. Sim2Real Edits

Finally, showcase the ability for our model to produce complex edits that are hard to explain with a text prompt, and that effectively serve as downstream model training data. Robotics policies are known to be very brittle and perform poorly out-of-distribution. Further, due to the complexity of collecting real-world data in those robot learning tasks, many works use a simulation environment to collect large-scale data to pre-train their robot policy [31]. Still, the gap between the simulation domain and real domain is generally large. Hence, models often require a layer of finetuning on real-world data to perform correctly, which still requires costly and lengthy data collection in real-world. Here, we aim to train our diffusion model to perform image editing to modify said simulation images into samples closely resemble the real-world domain. This requires only a *few images* from the real world to serve as target style conditioning for the editing process, rather than *tens of thousands*. We select a robot manipulation task referred to as Push-T, first introduced in [14], where a T-shaped block is placed randomly on a surface and a autonomous arm must push it to align its pose with that of a target, drawn somewhere else on that same surface.

5. Conclusion

in this section, i tell you that my results are great and that my method is unique because it has not been done before In our paper, we introduce a novel method for instructional editing with neither a large dataset of curated images or human feedback. Rather, we leverage other AI models to provide a balanced supervision to align generated images with what humans would like. Specifically, we showcase its ability to produce sharp edits and preserve instruction-irrelevant regions in an image. Further, we show that our model can interpret nuances beyond a text prompt, largely alleviating the need for complex and highly descriptive textual prompts. Lastly, we present an example application in a robot manipulation setting, where the T2I generative model serves as data augmentation, and significantly reduces the required amount of real-world data.

While our method demonstrates impressive performance, we have also identified limitations inherent to its training method. Types of edits are limited to local ones, where a clear segmentation can be drawn around the shape of the region to be edited. Further, the edits are limited to ones that do not modify the global shape of the element to be modified, but rather modify its texture or surface. Some interesting future work could include adding some more flexible bounds in the masking and depth alignment operations (by ignoring the top quantile for depth, or voluntarily enlarging the segmentation mask for semantics) to allow for edits that involve adding, removing, or modifying the shape of objects a bit. another limitation is that the semantic alignment is dependent on what the encoder can capture. Also, the method relies on the zero-shot performance of the base model. Hence, the biases of such encoder and diffusion model might be infused in the final outputs. Nevertheless, these limitations can be addressed by substituting the encoder or generative prior with suitable alternatives in a plug-and-play fashion.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning, 2022. 1

[2] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, page 18187–18197. IEEE, 2022. 2

[3] Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. Synthetic data from diffusion models improves imagenet classification, 2023. 1

[4] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback, 2022. 3

[5] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional ai: Harmlessness from ai feedback, 2022. 3

[6] Hritik Bansal and Aditya Grover. Leaving reality to imagination: Robust classification via generated datasets, 2023. 1

[7] Manel Baradad, Jonas Wulff, Tongzhou Wang, Phillip Isola, and Antonio Torralba. Learning to see by looking at noise, 2022. 1

[8] Elior Benarous, Sotiris Anagnostidis, Luca Biggio, and Thomas Hofmann. Harnessing synthetic datasets: The role of shape bias in deep neural network generalization, 2023. 1

[9] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning, 2024. 3

[10] Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. 3

[11] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions, 2023. 1, 2, 4, 5

[12] 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, 2023. 1, 2

[13] Wenhu Chen, Hexiang Hu, Yandong Li, Nataniel Ruiz, Xuhui Jia, Ming-Wei Chang, and William W. Cohen. Subject-driven text-to-image generation via apprenticeship learning, 2023. 2

[14] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 2024. 6

[15] Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences, 2023. 3

[16] Lisa Dunlap, Alyssa Umino, Han Zhang, Jiezhi Yang, Joseph E. Gonzalez, and Trevor Darrell. Diversify your vision datasets with automatic diffusion-based augmentation, 2023. 1

[17] Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models, 2023. 3

[18] Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. Dreamsim: Learning new dimensions of human visual similarity using synthetic data, 2023. 6

[19] Yunxiang Fu, Chaoqi Chen, Yu Qiao, and Yizhou Yu. Dreamda: Generative data augmentation with diffusion models, 2024. 1

[20] 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, 2022. 2

[21] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control, 2022. 1, 2

[22] Amir Hertz, Andrey Voynov, Shlomi Fruchter, and Daniel Cohen-Or. Style aligned image generation via shared attention, 2024. 2

[23] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. 5

[24] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. 1

[25] Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005. 11

[26] Hirokatsu Kataoka, Kazushige Okayasu, Asato Matsumoto, Eisuke Yamagata, Ryosuke Yamada, Nakamasa Inoue, Akio Nakamura, and Yutaka Satoh. Pre-training without natural images, 2021. 1

[27] 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, 2023. 1, 2

[28] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation, 2023. 6

[29] Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. Rlaif vs. rlhf: Scaling reinforcement learning from human feedback with ai feedback, 2024. 3

[30] Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. Aligning text-to-image models using human feedback, 2023. 3

[31] Xuanlin Li, Kyle Hsu, Jiayuan Gu, Karl Pertsch, Oier Mees, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, Sergey Levine, Jiajun Wu, Chelsea Finn, Hao Su, Quan Vuong, and Ted Xiao. Evaluating real-world robot manipulation policies in simulation, 2024. 6

[32] Jun Hao Liew, Hanshu Yan, Daquan Zhou, and Jiashi Feng. Magicmix: Semantic mixing with diffusion models, 2022. 1

[33] Huan Ling, Karsten Kreis, Daiqing Li, Seung Wook Kim, Antonio Torralba, and Sanja Fidler. Editgan: High-precision semantic image editing, 2021. 2

[34] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 4

[35] Will Maddern, Geoff Pascoe, Chris Linegar, and Paul Newman. 1 Year, 1000km: The Oxford RobotCar Dataset. *The International Journal of Robotics Research (IJRR)*, 36(1):3–15, 2017. 5

[36] 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, 2022. 2

[37] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models, 2022. 1, 2

[38] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, 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, 2023. 2

[39] 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, 2022. 2

[40] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022. 3

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 1

[42] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2024. 3

[43] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022. 1, 3

[44] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos, 2024. 4

[45] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks, 2024. 4

[46] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. 1, 4

[47] Litu Rout, Yujia Chen, Nataniel Ruiz, Abhishek Kumar, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. Rb-modulation: Training-free personalization of diffusion models using stochastic optimal control, 2024. 2

[48] 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, 2023. 2

[49] 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, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022. 2

[50] Mert Bulent Sariyildiz, Karteek Alahari, Diane Larlus, and Yannis Kalantidis. Fake it till you make it: Learning transferable representations from synthetic imagenet clones, 2023. 1

[51] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. 3

[52] Prafull Sharma, Varun Jampani, Yuanzhen Li, Xuhui Jia, Dmitry Lagun, Fredo Durand, William T. Freeman, and Mark Matthews. Alchemist: Parametric control of material properties with diffusion models, 2023. 2, 3

[53] Shelly Sheynin, Adam Polyak, Uriel Singer, Yuval Kirstain, Amit Zohar, Oron Ashual, Devi Parikh, and Yaniv Taigman. Emu edit: Precise image editing via recognition and generation tasks, 2023. 1, 2

[54] Yichun Shi, Xiao Yang, Yangyue Wan, and Xiaohui Shen. Semanticstylegan: Learning compositional generative priors for controllable image synthesis and editing, 2022. 2

[55] Joonghyuk Shin, Minguk Kang, and Jaesik Park. Fill-up: Balancing long-tailed data with generative models, 2023. 1

[56] Jordan Shipard, Arnold Wiliem, Kien Nguyen Thanh, Wei Xiang, and Clinton Fookes. Diversity is definitely needed: Improving model-agnostic zero-shot classification via stable diffusion, 2023. 1

[57] Kihyuk Sohn, Nataniel Ruiz, Kimin Lee, Daniel Castro Chin, Irina Blok, Huiwen Chang, Jarred Barber, Lu Jiang, Glenn Entis, Yuanzhen Li, Yuan Hao, Irfan Essa, Michael Rubinstein, and Dilip Krishnan. Styledrop: Text-to-image generation in any style, 2023. 2

[58] Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback, 2022. 3

[59] R.S. Sutton and A.G. Barto. Reinforcement learning: An introduction. *IEEE Transactions on Neural Networks*, 9(5): 1054–1054, 1998. 4

[60] Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems*. MIT Press, 1999. 4

[61] Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models, 2023. 1

[62] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation, 2022. 1, 2

[63] Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization, 2023. 3

[64] Haofan Wang, Matteo Spinelli, Qixun Wang, Xu Bai, Zekui Qin, and Anthony Chen. Instantstyle: Free lunch towards style-preserving in text-to-image generation, 2024. 2

[65] Su Wang, Chitwan Saharia, Ceslee Montgomery, Jordi Pont-Tuset, Shai Noy, Stefano Pellegrini, Yasumasa Onoe, Sarah Laszlo, David J. Fleet, Radu Soricut, Jason Baldridge, Mohammad Norouzi, Peter Anderson, and William Chan. Imagen editor and editbench: Advancing and evaluating text-guided image inpainting, 2023. 2

[66] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis, 2023. 6

[67] Xiaoshi Wu, Keqiang Sun, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score: Better aligning text-to-image models with human preference, 2023. 3

[68] Shaoan Xie, Zhifei Zhang, Zhe Lin, Tobias Hinz, and Kun Zhang. Smartbrush: Text and shape guided object inpainting with diffusion model, 2022. 2

[69] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation, 2023. 3, 6

[70] Kai Yang, Jian Tao, Jiafei Lyu, Chunjiang Ge, Jiaxin Chen, Qimai Li, Weihan Shen, Xiaolong Zhu, and Xiu Li. Using human feedback to fine-tune diffusion models without any reward model, 2024. 3, 4

[71] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2, 2024. 4

[72] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models, 2023. 2

[73] Jianhao Yuan, Francesco Pinto, Adam Davies, and Philip Torr. Not just pretty pictures: Toward interventional data augmentation using text-to-image generators, 2024. 1

[74] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing, 2024. 2, 5

[75] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 2

[76] Shu Zhang, Xinyi Yang, Yihao Feng, Can Qin, Chia-Chih Chen, Ning Yu, Zeyuan Chen, Huan Wang, Silvio Savarese, Stefano Ermon, Caiming Xiong, and Ran Xu. Hive: Harnessing human feedback for instructional visual editing, 2024. 3

[77] Yifan Zhang, Daquan Zhou, Bryan Hooi, Kai Wang, and Jiashi Feng. Expanding small-scale datasets with guided imagination, 2023. 1

[78] Yinan Zhang, Eric Tzeng, Yilun Du, and Dmitry Kislyuk. Large-scale reinforcement learning for diffusion models, 2024. 4

[79] Yongchao Zhou, Hshmat Sahak, and Jimmy Ba. Training on thin air: Improve image classification with generated data, 2023. 1

6. Appendix

6.1. Derivation for Classifier-free Guidance with Three Conditionings

We introduce separate guidance scales like ip2p to enable separately trading off the strength of each conditioning. The modified score estimate for our model is derived as follows. Our generative model learns $P(z|c_T,\ c_{I_{tgt}},\ c_{I_{og}})$, which corresponds to the probability distribution of the image latents $z = \mathcal{E}(x)$ conditioned on an input original image $c_{I_{og}}$, a target style image $c_{I_{tgt}}$, and a text instruction c_T. We arrive at our particular classifier-free guidance formulation by expressing the conditional probability as follows:

$$\begin{aligned} P(z|c_T,\ c_{I_{tgt}},\ c_{I_{og}}) &= \frac{P(z, c_T,\ c_{I_{tgt}},\ c_{I_{og}})}{P(c_T,\ c_{I_{tgt}},\ c_{I_{og}})} \\ &= \frac{P(c_T|c_{I_{tgt}}, c_{I_{og}}, z)P(c_{I_{tgt}}|c_{I_{og}}, z)P(c_{I_{og}}|z)P(z)}{P(c_T,\ c_{I_{tgt}},\ c_{I_{og}})} \end{aligned} \tag{8}$$

Diffusion models estimate the score [25] of the data distribution, i.e. the derivative of the log probability. Taking the logarithm of the expression above yields the following:

$$\begin{aligned} \log(P(z|c_T,\ c_{I_{tgt}},\ c_{I_{og}})) = &\log(P(c_T|c_{I_{tgt}}, c_{I_{og}}, z)) \\ &+ \log(P(c_{I_{tgt}}|c_{I_{og}}, z)) \\ &+ \log(P(c_{I_{og}}|z)) \\ &+ \log(P(z)) \\ &- \log(P(c_T,\ c_{I_{tgt}},\ c_{I_{og}})) \end{aligned} \tag{9}$$

Taking the derivative and rearranging, we obtain:

$$\begin{aligned} \nabla_z \log(P(z|c_T,\ c_{I_{tgt}},\ c_{I_{og}})) = &\nabla_z \log(P(z)) \\ &+ \nabla z \log(P(c_{I_{og}}|z)) \\ &+ \nabla z \log(P(c_{I_{tgt}}|c_{I_{og}}, z)) \\ &+ \nabla_z \log(P(c_T|c_{I_{tgt}}, c_{I_{og}}, z)) \end{aligned} \tag{10}$$