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) *"Turn the road into wood"* (d) *"Add rain on the road"* (e) *"Change the time to nighttime"*

Abstract

We present a novel approach to training specialized instruction-based image-editing diffusion models, addressing key challenges in structural preservation with input images and semantic alignment with user prompts. We introduce a reinforcement learning framework that aligns the diffusion model with human preferences without relying on extensive human annotations or curating a large dataset. Our method significantly improves the alignment with instructions in two ways. First, the proposed models achieve precise and structurally coherent modifications in complex scenes while maintaining high fidelity in instruction-irrelevant areas. Second, they capture fine nuances in the desired edit by leveraging a visual prompt, enabling detailed control over visual edits without lengthy textual prompts. This approach simplifies users' efforts to achieve highly specific edits, requiring only 5 reference images depicting a certain concept for training. Experimental results demonstrate that our models can perform intricate edits in complex images, like crowded road scenes, after just 14 training steps, exhibiting 36% improvement in structural alignment compared to SOTA baselines, on top of closer adherence to the edit directives. We explore applications in generating data for downstream model training in data-scarce domains, demonstrating our method's potential to alleviate costly dataset curation through complex sim-to-real edits.

1. Introduction

Text-to-image (T2I) generative models have achieved remarkable success in creating visually appealing images based on text prompts [26, 49, 52], largely due to advancements in aligning captions with images [1, 47].

Leveraging these impressive generative capabilities, T2I models have facilitated the development of instructional image editing, offering a highly practical approach for semantic modifications [11, 59]. Unlike conventional image editing techniques [13, 23, 30, 36, 41, 68] that necessitate detailed descriptive captions for both the input and modified images, instruction-based image editing relies only on natural-language directives. This method is more straightforward, as it requires specifying only the elements that need alteration in the original image, without involving the other unrelated attributes.

Although instructional editing has gained popularity in consumer applications, it has yet to demonstrate effective reusability of its samples in use-cases demanding precision and consistency, like data generation for downstream model training. Simple-to-generate synthetic samples have proven effective as pretraining data [7, 8, 29]. Building on this success, recent studies have explored leveraging T2I models as sources of high quality data augmentation. Some works generate diverse samples by applying naive perturbation to the noisy latents [67, 83], to the conditioning embeddings [85] or throughout the denoising process [20]. Others do so by designing, optimizing, or reformulating prompts [6, 17, 56, 61, 62, 79]. 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 of these approaches leverage instructional editing for data augmentations that closely align with user intentions. To serve effectively in downstream applications, we identify two key criteria that an image-editing model should satisfy.

Structural Alignment: The model should confine its modifications to the regions specified by the editing request, preserving the original scene elsewhere with high fidelity. 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 [11] was created with the Prompt-to-Prompt method [23]. Consequently, it suffers from the same limitations as Prompt-to-Prompt, including imperfect preservation of background elements. While diffusion models are traditionally trained for pixel-perfect reconstruction, we argue that preserving high-level structural similarity is essential for effective image editing. This global coherence is more readily noticed, and hence contributes significantly to the overall perceived realism and similarity between the original and edited images.

Semantic Alignment: The model 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. Therefore, 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 targeted style transfer, where an image's style is edited across the whole frame [24, 63, 70]. Other recent studies have focused on finetuning pre-trained T2I models for subject-driven editing from a set of reference images [22, 54]. 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, limiting the complexity of our text prompts to simple instructions. We highlight the relevance of visual prompts, which further alleviate the user's burden to convey the desired edit when combined with instruction-styled text prompts. Moreover, prior approaches often rely on the diffusion denoising objective, which can lead to reproductions of the reference style that may not correspond well with human expectations. Instead, we propose enforcing alignment in the latent space of an encoder trained to match human judgments, capturing both high and low-level semantic features.

We therefore focus on aligning the diffusion model with human preferences, specifically targeting these structural and semantic aspects. This approach necessitates moving away from the standard denoising objective, which also circumvents the need for complex training data preparation. Specifically, the denoising objective presents inherent challenges for image editing, as it enforces uniform reconstruction of the entire image without distinguishing between regions that require preservation and those that need modification. Learning to edit with the denoising 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, 14, 59, 80].

In this work, we introduce a novel self-play method designed to specialize instruction-based image editing diffusion models into producing edits with styles highly aligned with visual prompts while accurately preserving the original structure in non-pertinent areas of the image. These models build on the capabilities of the pretrained InstructPix2Pix model [11]. Unlike traditional reinforcement learning with human feedback (RLHF) methods, our approach bypasses the need for human annotators by employing AI models to provide feedback (RLAIF) that accurately simulates human preferences in terms of structure and semantics. We demonstrate our method's effectiveness by training models that perform natural-looking edits of intricate scenes with better visual appeal. Our experiments show that it surpasses current state-of-the-art methods in structural preservation and instruction alignment, resulting in samples that are favored by human preference prediction models. We also showcase these models' ability to serve as highly effective training data generators in data-scarce domains, where data collection is often costly and time-consuming. Our contributions are summarized as follows:

1. We propose a novel approach to train image-editing models with RLAIF, addressing both structural and semantic alignment challenges.
2. We adapt the architecture of our T2I diffusion model to follow subtle nuances from a few visual exemplars, while keeping simple textual instructions.
3. We conduct comprehensive quantitative and qualitative evaluations, highlighting improved precision in meticulous edits, and stronger alignment with both realistic and abstract instruction prompts.
4. We showcase the ability to perform complex edits in a data-scarce sim-to-real setting, where synthesized samples are used to train a robot manipulation policy.

2. Related Works

Text-guided Image Editing. Prior approaches to text-guided image editing can be categorized into in three distinct categories: architectural tricks that require no training, few-steps optimization for every generated sample, and

large scale finetuning. In the first category, Prompt-to-Prompt (P2P) [23] manipulates attention maps in the diffusion model to control the layout and content of the editing. Plug-and-Play [68] 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, 13, 37, 43, 60, 71, 74] or semantic mask [40, 42, 78, 81] 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 [41] optimizes the null-text embedding during the inversion of an input image. Imagic [30] finetunes model weights and embeddings to align with the input image and the edit text prompt. RB-Modulation [53] leverages a stochastic optimal controller to align content and style with visual prompts. A major drawback of these methods remains their slow speed at inference, because optimization is required for each generation.

Lastly, other methods adopt the standard denoising training. InstructPix2Pix [11] learns from a large synthetic dataset of image pairs with P2P, coupled with instruction-based text prompts. Similarly, Emu Edit [59] expands the dataset created with P2P, screens training samples with semantic and structural filters, and leverages multi-task training. SuTI [14] leverages finetuned expert versions of Imagen [55] to create high quality samples for the editing model to learn from. MagicBrush [80] assembles an image editing dataset synthesized with DALL-E 2 [49] and manually prunes samples. Alchemist [58] uses a rendering tool targeted to modifying material attributes. However, these methods need an oracle to generate images, which requires some curation, and may still infuse its limitations.

Our work builds upon InstructPix2Pix, addressing the misalignment between structural preservation and prompt adherence through few targeted finetuning steps, while maintaining the model's ability to generalize to unseen input images.

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, 16, 45, 64], where we first teach a reward function to mimic human preferences, and leverage reinforcement learning algorithms like proximal policy optimization [57] to finetune models with such rewards.

In the field of diffusion models, several works study the use of human feedback for T2I generation. [34] collects human annotations, and perform maximum likelihood training where the reward is naively used as a weight. [73] designs a reward model that better captures finegrained human preferences. [9, 18] show that diffusion models can be trained with RL using a reward model emulating human preferences [75]. Specific to instructional image-editing, HIVE [82] 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, unlike automated metrics that maintain constant vigilance, human evaluators' ability to detect structural preservation inconsistencies degrades over time due to fatigue and attention variability. 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, 33], 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 [76], 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

Most RLHF methods train a reward model to then train a downstream model. However, Direct Preference Optimization (DPO) [48] showed that preference ranking can be used to train language models and circumvent reward models, which [69] extended to diffusion models. In our work, we leverage the framework introduced by D3PO [76], 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 ranking 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, while 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.

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

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 [65], for the T-timesteps long denoising process.

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

where $p_\theta(\mathbf{x}_{0:T}|\cdot)$ 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 [76], 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:

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

3.2. Multi-objective Joint Training

Having established the mathematical framework for converting preference rankings into a trainable loss function for our diffusion model, we now elucidate the methodology for determining the relative ranking between two generated samples, I_{gen}. Our model learns from an objective that enforces structural alignment with the input image I_{in}, and semantic alignment with the text instruction and visual style prompts, c_T and I_{sty} respectively. We achieve this by decoupling the 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 a monocular depth estimation model [77]. 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.

$$\mathcal{L}_{struct} = \frac{1}{h \cdot w} \sum_{i,j}^{h,w} |f_\phi(I_{\text{in}})_{i,j} - f_\phi(I_{\text{gen}})_{i,j}| \quad (8)$$

where f_ϕ is the depth model and $h \times w$ is the image resolution. With this metric, we capture any missing, additional or deformed elements in the edit compared to the input.

Semantic Score. The semantic alignment should only be measured inside the region of interest, where the edit is expected. Accordingly, we locate the element to be edited with a text-conditioned segmentation model. Here, we use grounded-SAM2 [38, 50, 51]. Same goes for the visual prompt, whose style to be extracted may not cover the entire frame. For edits expected to span the whole frame, the mask can be defined to cover the entire image. Subsequently, we determine the semantic alignment score by computing the distance between the embeddings of instruction-relevant patches in the generated and style prompt images. Additionally, we find that incorporating the 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 instruction-irrelevant regions. Accordingly, our semantic score is:

$$\mathcal{L}_{sem} = D(m_{\text{in}} \odot I_{\text{in}},\ m_{\text{sty}} \odot I_{\text{sty}},\ f_\theta) + \lambda \cdot (1 - m_{\text{in}}) \odot \|I_{\text{in}} - I_{\text{gen}}\|_2^2 \tag{9}$$

where $D(\cdot)$ is a distance metric, m_{in} and m_{sty} are the binary segmentation masks 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, and that DreamSim [19], amongst other encoders, captures features best aligned with this task (see Sec. 4.2 for ablations).

Similar to [84], we obtain *advantages* [66] 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 adapted from Stable Diffusion [52] with one main architecture modification. That is, Brooks et al. [11] add input channels to the first convolutional layer of the U-Net 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. We posit that it helps better localize regions in both images that are relevant to the instruction. Here, the query is the linear projection of the concatenation of $\mathcal{E}(I_{\text{in}})$ and $\mathcal{E}(I_{\text{sty}})$, where $\mathcal{E}$ is the VAE encoder [31], and the key and value are obtained by projecting the CLIP encoding of the instruction prompt. The cross-attention output maintains the same spatial dimensions as the VAE-encoded image to preserve compatibility with the pre-trained U-Net architecture.

Classifier-free guidance (CFG) [25] shifts probability mass where an implicit classifier assigns high likelihood to the conditioning, improving visual quality and faithfulness of samples. In our case, we find it beneficial to extend the CFG estimate with respect to both the input image and the visual prompt (see Appendix 6.1 for the complete derivation):

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

4. Experiments

This section presents a comprehensive analysis of our experimental results and ablation studies, demonstrating the effectiveness of our proposed approach. We apply the default guidance scale parameters ($s_{I_{\text{in}}}$ and s_T) from InstructPix2Pix for a fair comparison, and set our visual conditioning score as $s_{I_{\text{sty}}} = 3$ for all quantitative experiments, a decision we discuss in Sec. 4.2.

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 [39] as our input images to be edited. We train our method to learn 7 types of edits separately (see full list in Sec. 6.2), all related to modifying the road, as it is guaranteed to remain visible on every frame 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. During training, we alternate between 5 conditioning style images that relate to a same text prompt to prevent overfitting on spurious cues. We conduct evaluations for each edit type on a set of 3500 images at resolution 256×256. 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 conditioning. Additionally, we display the ability of our model to produce edits that serve as effective training data in a data-scarce robot learning setting.

4.1. Baseline Comparisons

We compare our model's performance against that of InstructPix2Pix (IP2P) [11], HIVE [82], MagicBrush (MBrush) [80], all based on the stable diffusion v1.5 backbone.

We first compare the results qualitatively in Figure 2. We notice that InstructPix2Pix struggles to precisely locate the region to be edited while leaving the rest intact, and similar to MagicBrush, fails to generate naturalistic edits with unrealistic prompts. Conversely, HIVE often prefers maintaining high fidelity with the input image at the cost of minimalistic edits. Our model achieves a better balance between excessive and insufficient editing, resulting in edits that are both highly aligned with the prompts and remain faithful to the input image. Further, our method reduces the impact of biases associated with concepts mentioned in the text prompt. For instance, we suppress InstructPix2Pix's hal-

Figure 2. Qualitative comparison. Our method outperforms its counterparts by significantly editing the image while sharply preserving the structure of regions unrelated to the text prompt. We omit visualizing the conditioning style image as the other methods cannot leverage it. See more samples in Sec. 6

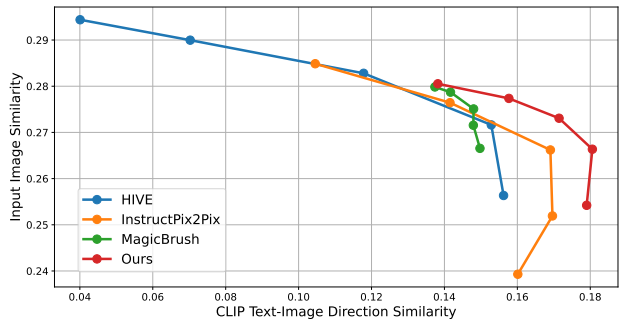

Figure 3. Comparison between instructional editing models. We plot the trade-off between consistency with the input image (Y-axis) and consistency with the edit (X-axis). For both metrics, higher is better. For all methods, we fix the same parameters as in [11] and vary the $s_{I_{in}} \in [1.0, 2.0]$

lucinations of a forest when generating an image with the word 'wood' mentioned in the instruction.

We also quantitatively analyze the tradeoff between the fidelity with the input image, and the alignment with the text instruction. The former is measured through the cosine similarity of image patch embeddings that are outside of the mask locating the element to be edited. We rely on grounded SAM2 to produce high-quality masks, which we find surpass human-made masks provided in public benchmarks [80], regarding contour precision. With such masking, our reported metrics are better targeted to the regions of interest. We capture both high and low-level visual features by taking the average score across three encoders, namely

Method	$\text{Depth}_{\text{out}}\downarrow$	$\text{Depth}_{\text{all}}\downarrow$	$\text{L2}_{\text{out}}\downarrow$	$\text{STY}_{\text{in}}\uparrow$	$\text{CLIP}_{\text{txt}}\uparrow$
IP2P	35.93	50.52	0.064	0.264	0.238
MBrush	46.06	60.69	0.054	0.250	0.234
HIVE	21.56	29.47	**0.036**	0.248	0.224
Ours	**14.67**	**18.80**	0.063	**0.283**	**0.251**

Table 1. Comparison of structure preservation through predicted monocular depth mask alignment and reconstruction metrics (left), and semantic alignment with both the text and visual prompts (right). For the text alignment, we use a descriptive prompt to capture all information on the image. We refer to the regions of the image considered in a certain metric with the indices "in", "out", and "all", which respectively correspond to: inside the mask, outside the mask, and the entire frame.

DINOv2 [44], CLIP [46], and DreamSim [19]. The text alignment is measured through the directional CLIP similarity [21], which denotes how much the change in descriptive text captions agrees with the change in the input and generated images. This evaluation operates on the entire frame, i.e. no masking operation is involved. Both metrics are antagonistic, increasing the desired edit strength will reduce the output's faithfulness to the input image. Still, we find that when comparing our method with its counterparts in Fig. 5, our results have notably higher directional similarity values for the same image consistency. We also confirm our qualitative insights, noticing that HIVE tends to prioritize preserving the original image over executing the requested edits, whereas our method closely respects the editing directive. In Table 1, we also find that HIVE better reconstructs the regions that should be kept intact. Still, our method outperforms the others on the depth mask alignment metrics, both measured on these regions and on the whole images. This suggests that, while our model has some imprecision on a finegrained level, it successfully preserves the fundamental structure, which is crucial for visual coherence and perceived realism.

We then demonstrate our model's capability to interpret subtle details beyond what is specified in a text prompt. To illustrate this, we train two distinct models to add a dense and a sparse layer of snow on the road, even when these specifics were not included in the text prompt. Fig. 4 reveals that our model excels at reproducing the visual style of the concept hinted at in the text prompt, while respecting the spatial composition of the input image during the edit. For instance, the sparse snow generated by our model mirrors the stripe-like pattern of the conditioning image and aligns with the road's direction in the input images. This underscores the benefit of leveraging a visual prompt to infuse fine-grained nuances without requiring an extensive text prompt.

We quantitatively validate these observations in Table 1. Our evaluation focuses on two key aspects: visual seman-

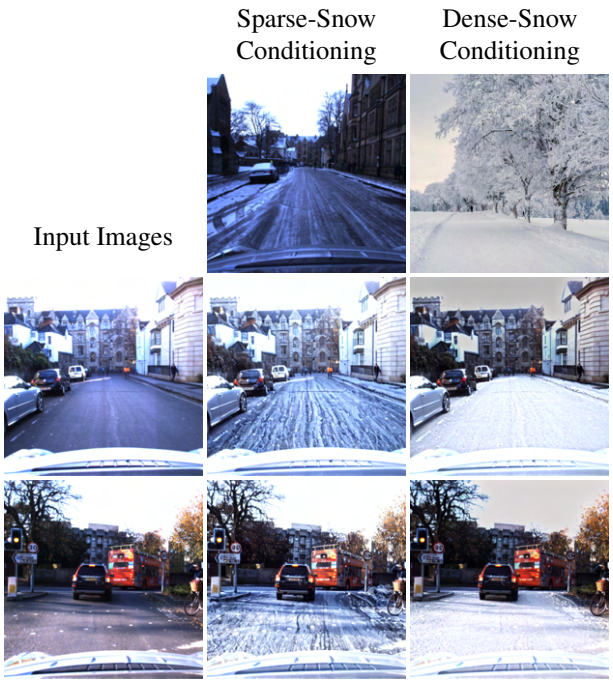

Figure 4. Visualization of the impact of the visual prompt on generated samples when provided the text instruction "add snow on the road". Our method effectively captures semantic nuances beyond that described in the text prompt.

Method	HPSv2 ↑	ImageReward ↑	PickScore ↑
IP2P	21.85	-0.314	18.92
MBrush	21.92	-0.454	18.84
HIVE	21.61	-0.567	18.89
Ours	**22.12**	**-0.137**	**18.95**

Table 2. Comparison between instructional editing models on human-preference-aligned visual scoring metrics.

tic alignment between regions-of-interest, and text-image alignment scores. For the visual alignment, we compute the cosine similarity between masked regions in the generated and conditioning images (STY_{in}), averaging across DINOv2, CLIP, and DreamSim embeddings. The text-image alignment is measured through CLIP similarity (CLIP_{txt}) between the edited image and output caption. We find that our model outperforms its counterparts, both in the visual and text-image alignments, further confirming our method's efficacy to increase instruction fidelity.

Finally, in Table 2, we compare the scores obtained from T2I synthesis preference prediction models, namely HPSv2 [72], ImageReward [75], and PickScore [32], which are all trained to emulate human preferences. We find that our model surpasses all other baselines across the different edits. This confirms our model's superior ability to preserve

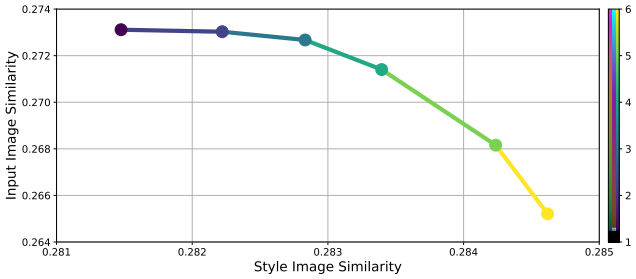

Figure 5. We plot the trade-off between consistency with the input image (Y-axis) and consistency with the visual prompt (X-axis). For both metrics, higher is better. We fix the same parameters as in [11] and vary the $s_{I_{\text{sty}}} \in [1.0, 6.0]$

the essential structural features of the input image, which are crucial for perceived realism, while also aligning effectively with the specified editing instructions.

4.2. Ablation Study

Classifier-free-guidance scale. In Figure 5, we provide an analysis of the effect of the visual prompt's classifier-free-guidance scale $s_{I_{\text{sty}}}$. We find that increasing its value results in an edit more strongly aligned with the visual prompt on which the generation was conditioned. However, this comes at the cost of a lesser similarity with the input image. We find that values of $s_{I_{\text{sty}}}$ in the range $1-5$ typically produce the best results, and hence present quantitative results in Sec. 4.1 with $s_{I_{\text{sty}}} = 3$. In practice, and for qualitative results shown in the paper, we find it beneficial to adjust this guidance weight for each edit type to obtain the best balance between faithfulness to the input and alignment with the visual prompt.

Encoder Choice. Different encoders capture different information in their representations, guiding the learning process with distinct models leads to different outputs. Other works commonly use DINOv2 and CLIP as evaluation metrics for generated samples quality [11, 59, 80]. However, recently, DreamSim showed to outperform those encoders in alignment with human preferences. We therefore compare the impact of the selected encoder by training three versions of our method to edit sparse snow, and replacing the encoder in Eq. 9. We only analyze qualitative results, since all semantic visual alignment metrics are based on these encoders. We find in Fig. 6 that the images generated by the DINOv2-guided model possess a grainy texture that is not present in the visual prompt. Also, the CLIP-guided model reproduces excessively smooth and vaguely defined snow lanes compared to the visual prompt, and that a purple tint overlays across the frame. Contrastingly, DreamSim better enforces the color of the visual prompt, and does not lead to learning spurious cues like an unrelated tint or saturated

Figure 6. Qualitative comparison of the reproduction of the visual prompt induced by different encoders. Best viewed zoomed-in.

colors. Further, it best reproduces the structure of the snow stripes. This results in more realistic samples with stronger alignment to the prompt.

4.3. Sim-to-Real Editing

Finally, we 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 and pre-train the policy [35]. 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 resembling the real-world domain. This process requires only *5 images* from the real world to serve as visual style conditioning for the generation, rather than *thousands* when using the data to directly train the policy. We select a robot manipulation task referred to as Push-T, first introduced in [15], where a T-shaped block is placed randomly on a surface and an autonomous arm with a cylinder-shaped end-effector must push it to align its pose with that of a target, drawn somewhere else on that same surface.

We show in Fig. 7 that our model can perform highly realistic edits, far beyond what the base model can produce. Our method is capable of iterative edits, firstly modifying the background to match the table surface and then the T-block into its orange color. We find that the input images are too unrealistic for the depth prediction model to produce coherent depth maps, hindering the training process. Accordingly, we enforce structure through the alignment of Canny edge maps [12] rather than depth maps. We find that our model is capable of interpreting the texture of the table in our scenario and its subtle reflections of the light. We also find that our method correctly locates the foreground element to be edited, where InstructPix2Pix undesirably applies the edit across the frame. We also find that the visual prompt helps teach our T2I generative model to add some slight shading around the T-block, although it was not mentioned in either the foreground or background edit instruction.

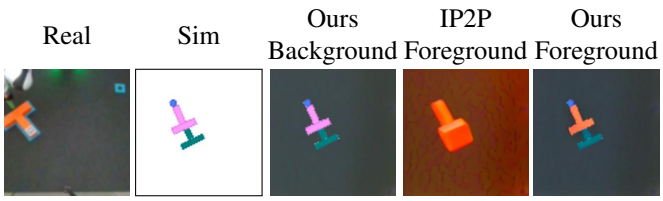

Figure 7. Comparison between the base InstructPix2Pix model and our specialized model on Sim-to-Real edits. We only show the performance of InstructPix2Pix on the foreground edit, following the background edit produced from our method, as a textual prompt is insufficient for it to reproduce the texture of the real-world table without a visual prompt. Textual prompts used for the background and foreground edits are: *"modify the background to appear more dark tabletop textured"* and *"change the T-shaped-block into a T-shaped-block with orange color"* respectively. The same visual prompt is used for both foreground and background edits, which is the real-world image. Our method captures both the texture of the table, the color of the foreground object, and adds some slight shading around, like there is in the real world, captured from a top-down view with a slight angle.

Method	MSE ↓
Random Init.	1.251
Sim Pretraining	0.192
Our Sim Pretraining	**0.158**

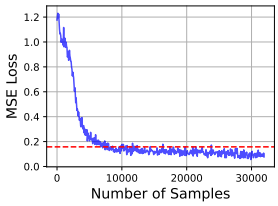

Figure 8. Comparison between performance of the policy after random initialization, pretraining on simulation data, and pretraining on our realistic simulation data (left), and evolution of randomly initialized policy's performance over one training epoch on real data (right). Red dotted line denotes the zero-shot performance of the policy after pretraining on our edited simulation data.

We report quantitative improvements in Fig. 8. We find that using this simulation data as pre-training for the robotic policy improves the zero-shot performance on the real-world data by $\sim 8\times$ compared to no pretraining, and by $\sim 20\%$ compared to pretraining with unrealistic simulation data. This performance reached with only 5 real-world samples that were used to train the generative model, would be attained with real data after training on 7000 samples, greatly alleviating the need for costly and time-consuming data collection.

5. Conclusion

In this paper, we introduce a novel approach to instruction-based image editing that enhances structural preservation and semantic alignment through few-steps finetuning, ef-

fectively mimicking human preferences without direct feedback. Our method demonstrates that these improvements can be achieved by leveraging AI-generated feedback, circumventing the need for extensive human annotations or large-scale datasets. Our models learn to capture and reproduce intricate details in visual prompts, with only 5 examples per concept, further reducing the reliance on elaborate textual prompts. Accordingly, these models significantly improve upon previous state-of-the-art methods in their balance between faithfulness to the input image and alignment with the instruction prompts. This combination results in samples with higher perceived realism. Our efficient fine-tuning approach with visual prompts enables complex sim-to-real edits using minimal reference images, demonstrating potential for high-quality training data generation in data-scarce domains.

While our approach shows significant improvements, we acknowledge certain limitations. The current method primarily excels at modifying textures and surfaces rather than altering global shapes of objects. Future work could explore more flexible constraints in masking and depth alignment operations, potentially allowing for more substantial structural modifications like adding and removing elements. Our model may also inherit biases from the pre-trained InstructPix2Pix model and the encoder used for semantic alignment supervision. However, this limitation can be mitigated by substituting these components with suitable alternatives in a modular 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. 3

[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, 3, 5, 6, 7

[12] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, 1986. 8

[13] 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, 3

[14] 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, 3

[15] 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. 8

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

[17] 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

[18] 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

[19] 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. 5, 6

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

[21] Rinon Gal, Or Patashnik, Haggai Maron, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators, 2021. 6

[22] 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

[23] 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, 3

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

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

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

[27] Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005. 14

[28] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models, 2022. 14

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

[30] 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, 3

[31] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022. 5

[32] 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. 7

[33] 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

[34] 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

[35] 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. 8

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

[37] Huan Ling, Karsten Kreis, Daiqing Li, Seung Wook Kim, Antonio Torralba, and Sanja Fidler. Editgan: High-precision semantic image editing, 2021. 3

[38] 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

[39] 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

[40] 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. 3

[41] 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, 3

[42] 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. 3

[43] 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. 3

[44] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024. 6

[45] 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

[46] 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. 6

[47] 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

[48] 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

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

[50] 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

[51] 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

[52] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. 1, 5

[53] 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. 3

[54] 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

[55] 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. 3

[56] 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

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

[58] 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. 3

[59] 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, 3, 7

[60] Yichun Shi, Xiao Yang, Yangyue Wan, and Xiaohui Shen. Semanticstylegan: Learning compositional generative priors for controllable image synthesis and editing, 2022. 3

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

[62] 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

[63] 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

[64] 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

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

[66] 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. 5

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

[68] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation, 2022. 1, 3

[69] 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

[70] 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

[71] 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. 3

[72] 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. 7

[73] 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

[74] Shaoan Xie, Zhifei Zhang, Zhe Lin, Tobias Hinz, and Kun Zhang. Smartbrush: Text and shape guided object inpainting with diffusion model, 2022. 3

[75] 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, 7

[76] 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

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

[78] 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. 3

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

[80] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing, 2024. 2, 3, 5, 6, 7

[81] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 3

[82] 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, 5

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

[84] Yinan Zhang, Eric Tzeng, Yilun Du, and Dmitry Kislyuk. Large-scale reinforcement learning for diffusion models, 2024. 5

[85] 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 InstructPix2Pix 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 $P(z|c_T,\ c_{I_{\text{sty}}},\ c_{I_{\text{in}}})$, which corresponds to the probability distribution of the image latents $z = \mathcal{E}(x)$ conditioned on an input original image $c_{I_{\text{in}}}$, a target style image $c_{I_{\text{sty}}}$, and a text instruction c_T. We arrive at our particular classifier-free guidance formulation by expressing the conditional probability as follows:

$$\begin{aligned} P(z|c_T,\ c_{I_{\text{sty}}},\ c_{I_{\text{in}}}) &= \frac{P(z, c_T,\ c_{I_{\text{sty}}},\ c_{I_{\text{in}}})}{P(c_T,\ c_{I_{\text{sty}}},\ c_{I_{\text{in}}})} \\ &= \frac{P(c_T|c_{I_{\text{sty}}}, c_{I_{\text{in}}}, z)P(c_{I_{\text{sty}}}|c_{I_{\text{in}}}, z)P(c_{I_{\text{in}}}|z)P(z)}{P(c_T,\ c_{I_{\text{sty}}},\ c_{I_{\text{in}}})} \end{aligned} \tag{10}$$

Diffusion models estimate the score [27] of the data distribution, i.e. the derivative of the log probability. Taking the logarithm of the expression above yields the following:

$$\begin{aligned} \log(P(z|c_T,\ c_{I_{\text{sty}}},\ c_{I_{\text{in}}})) = &\ \log(P(c_T|c_{I_{\text{sty}}}, c_{I_{\text{in}}}, z)) \\ &+ \log(P(c_{I_{\text{sty}}}|c_{I_{\text{in}}}, z)) \\ &+ \log(P(c_{I_{\text{in}}}|z)) \\ &+ \log(P(z)) \\ &- \log(P(c_T,\ c_{I_{\text{sty}}},\ c_{I_{\text{in}}})) \end{aligned} \tag{11}$$

Taking the derivative and rearranging, we obtain:

$$\begin{aligned} \nabla_z \log(P(z|c_T,\ c_{I_{\text{sty}}},\ c_{I_{\text{in}}})) = &\ \nabla_z \log(P(z)) \\ &+ \nabla z \log(P(c_{I_{\text{in}}}|z)) \\ &+ \nabla z \log(P(c_{I_{\text{sty}}}|c_{I_{\text{in}}}, z)) \\ &+ \nabla_z \log(P(c_T|c_{I_{\text{sty}}}, c_{I_{\text{in}}}, z)) \end{aligned} \tag{12}$$

6.2. Training Details

In training, we initialize our model from the InstructPix2Pix checkpoint. Depending on the specialization, we train over 15-40 steps, at 256×256 resolution with a total batch size of 512 samples. We posit that the biggest varying factor is how far out-of-distribution the requested edit is, as the KL-Divergence regularization limits the weights of the model from shifting too quickly too far from its initialization. For example, adding thick snow on the road needed 14 steps to converge, whereas turning the road into wood needed 43. For a fair comparison of our method, we do not conduct any hyperparameter optimization, and keep the same RL training hyperparameters as in D3PO.

We train our model to perform 7 types of edits of the road separately, these are adding snow (dense and sparse), rain, and sand on top; and changing it into gold and wood. We include a last one operating on the entire image, which is changing the time to nighttime. To train the latter, we handcraft the mask in the supervision to select the entire frame.

For baseline comparisons, we evaluate the InstructPix2Pix checkpoint from which we initialize our model, and the best publicly available versions of HIVE and MagicBrush based on StableDiffusion v1.5.

While our model is trained at 256×256 resolution, we find it generalized well to 512×512 resolution at inference time. We generate qualitative results in this paper with 100 denoising steps using an Euler ancestral sampler with denoising variance schedule proposed by Karras et al. [28]. Editing an image with our model takes roughly 9 seconds on an A100 GPU.

Figure 9. More comparisons between our method and other baselines when given the textual instruction prompt: *"change the road into gold"*

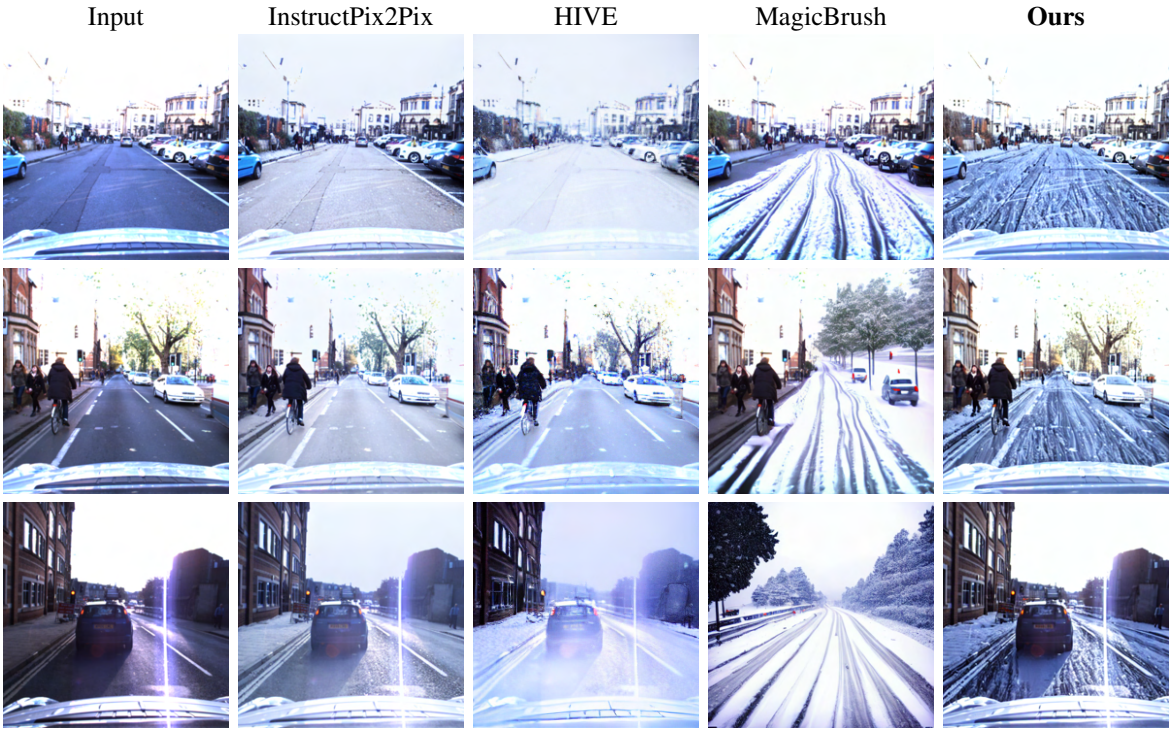

Figure 10. More comparisons between our method and other baselines when given the textual instruction prompt: *"add snow on the road"*

Figure 11. More comparisons between our method and other baselines when given the textual instruction prompt: *"change the time to nighttime"*

Figure 12. More comparisons between our method and other baselines when given the textual instruction prompt: *"add rain on the road"*

Figure 13. More comparisons between our method and other baselines when given the textual instruction prompt: *"change the road into wood"*

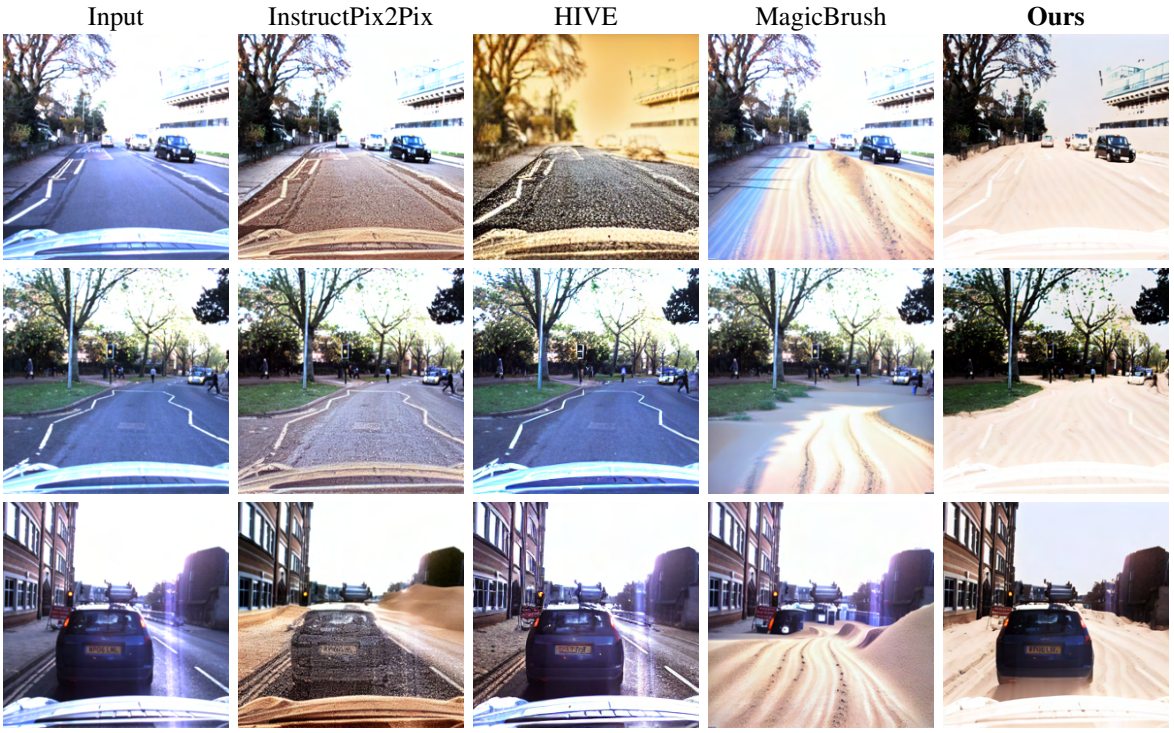

Figure 14. More comparisons between our method and other baselines when given the textual instruction prompt: *"add sand on the road"*

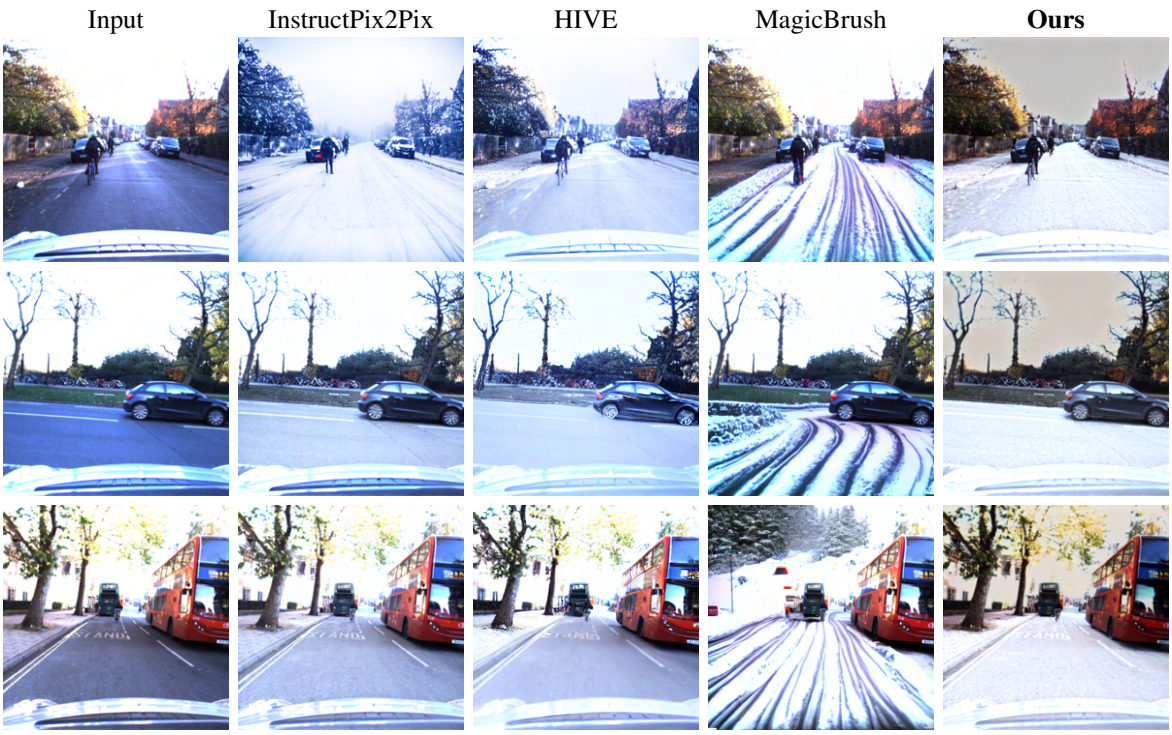

Figure 15. More comparisons between our method and other baselines when given the textual instruction prompt: *"add snow on the road"*