SPIE: Semantic and Structural Post-Training of Image Editing Diffusion Models with AI feedback

Elior Benarous[1,2,*] Yilun Du[1,3] Heng Yang[1]

[1]Harvard University [2]ETH Zürich [3]Google DeepMind

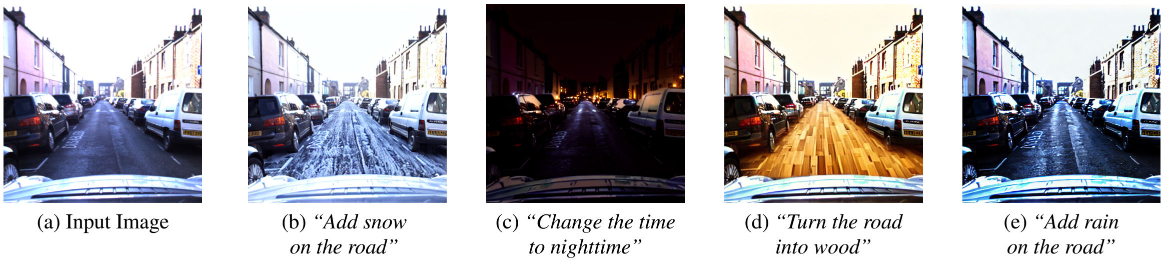

(a) Input Image
(b) *"Add snow on the road"*
(c) *"Change the time to nighttime"*
(d) *"Turn the road into wood"*
(e) *"Add rain on the road"*

Figure 1. Representative results showcasing our method's ability to perform precise and realistic edits. The input image is displayed alongside four diverse edits, highlighting SPIE's capacity to align with user intentions while preserving structural coherence.

Abstract

This paper presents SPIE: a novel approach for semantic and structural post-training of instruction-based image editing diffusion models, addressing key challenges in alignment with user prompts and consistency with input images. We introduce an online 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 and realism in two ways. First, SPIE captures fine nuances in the desired edit by leveraging a visual prompt, enabling detailed control over visual edits without lengthy textual prompts. Second, it achieves precise and structurally coherent modifications in complex scenes while maintaining high fidelity in instruction-irrelevant areas. 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 SPIE can perform intricate edits in complex scenes, after just 10 training steps. Finally, we showcase the versatility of our method by applying it to robotics, where targeted image edits enhance the visual realism of simulated environments, which improves their utility as proxy for real-world settings.

1. Introduction

Text-to-image (T2I) generative models have achieved remarkable success in creating visually compelling images from text prompts [27, 52, 55], driven by advancements in aligning captions with images [1, 50]. Leveraging these impressive generative capabilities, T2I models have facilitated the development of instructional image editing, offering a highly practical approach for semantic modifications [12, 20, 28, 29, 63, 84, 86, 89]. Unlike conventional image editing techniques [13, 24, 33, 38, 44, 71] that necessitate detailed descriptive captions for both the input and modified images, instruction-based image editing relies only on natural-language directives to specify changes while leaving unrelated attributes intact.

While instructional editing has gained popularity for creative applications, its potential remains underexplored in domains requiring precision and consistency. Simplistic synthetic data has proven effective for pretraining [7, 8, 32]. Yet, current data augmentation methods via semantic editing [3, 6, 17, 21, 60, 65, 66, 70, 83, 87, 91] fail to leverage instructional guidance for generating samples closely aligned with user intentions. To address this gap, we identify two key criteria for effective image editing models.

Semantic Alignment. Models should enable fine-grained control over desired modifications by leveraging both textual and visual prompts. Visual prompts capture stylistic nuances that are difficult to articulate in text alone, alleviating user effort while ensuring edits align closely with expectations.

Structural Alignment. Modifications should be confined to specified regions while preserving high fidelity

[*] Corresponding author: ebenarous@ethz.ch. Code is available at: https://github.com/ebenarous/SPIE

elsewhere. Current state-of-the-art methods like InstructPix2Pix [12] often struggle with precise edits due to limitations inherited from their training data [24]. These limitations can result in edits that inadvertently affect background elements or fail to maintain global coherence—an essential factor for achieving realism and preserving the overall structure of the original image.

To meet these criteria, we propose aligning diffusion models with human preferences through online reinforcement learning from AI feedback (RLAIF). Unlike traditional RLHF methods, which rely on human annotations, RLAIF enforces semantic and structural alignment from AI-generated preference feedback, effectively emulating human judgments. This approach also circumvents the limitations of the standard denoising objective, which requires an oracle to generate precise input-output pairs for training. Such oracles are often biased, lack scalability, or produce low-quality samples that necessitate extensive pruning [12, 14, 63, 84]. By moving away from the denoising objective, our method focuses on preserving high-level structural coherence and capturing nuanced semantic features, ensuring edits align closely with user expectations without compromising realism or precision.

In this work, we introduce **SPIE**: a novel framework for semantic and structural post-training of image editing diffusion models to produce edits highly aligned with visual prompts while preserving original structures in non-pertinent areas. Our method builds on InstructPix2Pix [12] and surpasses state-of-the-art baselines in both structural preservation, instruction adherence and predicted human preference. Beyond creative applications, we showcase its utility in robotics by enhancing simulated environments with realistic edits that improve alignment with real-world settings. Our contributions are summarized as follows:

1. We propose a novel self-play RLAIF-based framework addressing semantic and structural alignment challenges in image editing.
2. We adapt T2I diffusion models to capture nuanced visual styles from exemplars while adhering to simple textual instructions.
3. We conduct comprehensive quantitative and qualitative evaluations, demonstrating enhanced precision in intricate edits, stronger alignment with instruction prompts, and practical utility like improving the realism of simulation environments in robotics.

2. Related Works

Text-guided Image Editing. Prior approaches to text-guided image editing can be categorized into three distinct groups: architectural modifications, per-sample optimization, and large scale finetuning.

In the first category, methods like Prompt-to-Prompt (P2P) [24] manipulate attention maps in the diffusion model's U-Net [56] to control the layout and content of the editing. Plug-and-Play [71] injects self-attention maps and spatial features to improve structural coherence. While these approaches are effective for specific tasks, their reliance on architectural tweaks often limits their ability to handle complex scenes with intricate details. Other works leverage segmentation masks [2, 13, 39, 46, 64, 74, 78] or semantic masks [43, 45, 82, 85] during the forward pass to guide edits, ensuring modifications are both targeted and structurally consistent. However, these methods impose additional burdens on users by requiring them to provide these masks. MGIE [20] and SmartEdit [28] address this limitation by integrating multimodal large language models (LLMs) to enhance instruction comprehension and reasoning capabilities. Despite their advancements, these methods introduce significant architectural overhead by incorporating large additional components, greatly increasing computational demands.

In the second category, optimization-based methods like Null-Text Inversion [44] adjust the null-text embedding during the inversion for each input image. Imagic [33] finetunes model weights and embeddings to align with both the input image and the edit prompt. RB-Modulation [57] uses a stochastic optimal controller to align content and style with visual prompts. While effective, these methods are time-intensive as they require optimization for each individual sample during inference, resulting in slower generation speeds.

The third category includes methods that adopt standard denoising training on large synthetic datasets. InstructPix2Pix [12] trains on a dataset generated using P2P and instruction-based prompts. Emu Edit [63] expands this dataset with semantic and structural filters and employs multi-task training for improved generalization. SuTI [14], MagicBrush [84], HQ-Edit [29], and UltraEdit [89] rely on curated datasets synthesized using models like Imagen [59], DALL-E 2 and 3 [52], or LLMs. These datasets are often manually pruned or filtered to ensure quality. Alchemist [62], on the other hand, uses a rendering tool specifically designed for material attribute modifications. Nevertheless, all of these methods depend on an oracle to generate training data, which introduces biases, requires curation efforts, and may still carry limitations from the oracle itself.

Our method distinguishes itself from these approaches in several ways. Unlike architectural modification methods, we do not rely on large additional components in the forward pass architecture to improve performance, ensuring simplicity and efficiency. Unlike per-sample optimization techniques, our method does not require computationally expensive optimization steps during inference. Unlike large-scale finetuning approaches that depend on curated synthetic datasets generated by external oracles, we leverage the diffusion model's own samples for training without additional data generation pipelines or curation efforts. Building upon InstructPix2Pix, our approach addresses misalignments in structural preservation and prompt adherence through targeted finetuning steps while maintaining generalization to unseen input images. By focusing on simplicity and efficiency in both training and inference stages, SPIE achieves state-of-the-art performance without relying on external datasets or complex architectural modifications.

Visual Prompting. Most works on visual prompting for image generation have focused on style transfer, where the style of an image is modified across the entire frame [25, 67, 73]. Recent studies have also explored subject-driven editing by finetuning pre-trained T2I models using a set of reference images [23, 58]. However, these methods often require unique identifiers to encode the concept from the prompt into the edit, limiting their flexibility and usability. In contrast, our work focuses on local edits conditioned on visual prompts, paired with simple text instructions. By leveraging visual prompts, we reduce the user's burden to articulate complex edits in detail while maintaining precise control over the desired modifications. This combination of visual and textual guidance ensures alignment with user intentions without requiring lengthy or intricate text prompts. Moreover, prior approaches often rely on the diffusion denoising objective for style alignment, which can lead to reproductions of reference styles that fail to meet human expectations. Instead, we enforce alignment in the latent space of an encoder trained to match human judgments, capturing both high-level semantic features and subtle stylistic nuances while preserving structural fidelity. Our method enables localized edits with minimal user effort and moves beyond the limitations of traditional denoising objectives, ensuring results that are visually appealing and closely aligned with user preferences.

Reinforcement Learning for Diffusion. Aligning model outputs with human preferences has been widely successful in language modeling. For objectives that are difficult to define explicitly, reinforcement learning with human feedback (RLHF) [4, 15, 48, 68] has emerged as a popular strategy. RLHF involves training a reward function to mimic human preferences and using reinforcement learning algorithms like proximal policy optimization [61] to finetune models based on these rewards.

In the context of diffusion models, several works have explored using human feedback for T2I generation. Lee et al. [36] collect human annotations and perform maximum likelihood training where the reward is applied as a naive weight. Further, Wu et al. [76] design a reward model that captures fine-grained human preferences more effectively. DDPO [9] and DPOK [18] demonstrate that diffusion models can be trained with RL using a reward model emulating human preferences, such as ImageReward [79]. For instructional image editing specifically, HIVE [86] extends large-dataset supervised training by collecting human feedback on edits and performing offline RLHF training.

These methods rely heavily on reward models trained on large-scale human annotations, which introduce significant limitations. First, the annotation process is cumbersome and costly, and the resulting supervision often lacks consistency. Human evaluators' ability to detect structural preservation inconsistencies diminishes over time due to fatigue and attention variability. Second, semantic alignment remains vague as it is only compared to short instruction prompts, leaving room for subjective interpretation and disagreement among annotators.

Our method addresses these challenges by leveraging RLAIF [5, 35], eliminating the need for human-in-the-loop supervision. Instead of relying on human annotations, we use AI models to provide preference supervision tailored to address semantic and structural alignment issues. Additionally, unlike offline RL methods such as HIVE, we adopt an online training framework inspired by D3PO [80], which uses samples generated throughout training to ensure the learning process remains adaptable and unrestricted by the fixed distribution of pre-collected datasets. With such, SPIE only needs a few steps of post-training to produce edits that are semantically precise, structurally coherent, and aligned with user expectations, without relying on large-scale human annotations.

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 for Diffusion Models

Most RLHF methods train a reward model to then train a downstream model. However, Direct Preference Optimization (DPO) [51] showed that preference ranking can be used to train language models and circumvent reward models, which Wallace et al. [72] extended to diffusion models. In our work, we leverage the framework introduced by D3PO [80], 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 preferred and unpreferred samples respectively. Following the Bradley-Terry model [10], the human preference distribution p^* can be expressed by using a reward function r^* as:

$$p^*(y_w \succ y_l \mid x) = \frac{\exp(r^*(x, y_w))}{\exp(r^*(x, y_w)) + \exp(r^*(x, y_l))} \tag{1}$$

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

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

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:

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

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

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

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.

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

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 from the derived 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 [9], for the T-timesteps denoising process.

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

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 [80], 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

Having established the mathematical framework for converting preference rankings into a trainable loss, we now detail how we determine the relative ranking between two generated samples, I_{gen}. SPIE optimizes a composite objective that enforces both 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 overall objective into two separate scores.

Structural Score. To measure how well the structure of the input image is preserved, we employ a monocular depth estimation model [81]. Given a pair of input and edited images, we compute their respective depth map, and define the structural score as the L_1 distance between the two maps.

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

where f_ϕ is the depth model and $h \times w$ is the image resolution. This metric effectively captures any missing, additional, or deformed elements in the edit relative to the original input.

Semantic Score. The semantic alignment is evaluated within the region of interest, where the edit is expected. To identify this region, we use a text-conditioned segmentation model, grounded-SAM2 [41, 53, 54], which locates the element to be edited based on the instruction. Similarly, the visual prompt's style may not span the entire frame, so its relevant region is also segmented. For global edits that are intended to cover the whole frame, the mask can be defined to encompass the entire image. The semantic alignment score is computed by measuring the distance between embeddings of instruction-relevant patches in the generated image and those in the style prompt image. Additionally, we incorporate a pixel-space reconstruction objective for instruction-irrelevant regions that should remain unchanged. This acts as a regularizer to enforce sharper boundaries and prevent the style from spreading into areas outside the intended edit region. Accordingly, our semantic score is:

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

where $D(\cdot)$ is a distance metric, here the cosine distance, m_{in} and m_{sty} are binary segmentation masks for the input and style images respectively, $\odot$ defines the element-wise multiplication, and f_{θ} is an encoder. The hyperparameter λ balances the influence of the pixel reconstruction term relative to semantic alignment; empirically, we set $\lambda = 0.5$ for optimal performance. Among various encoders tested, DreamSim [19] most effectively captures task-relevant features (see Sec. 4.3 for ablations).

Similar to [88], we obtain *advantages* [69] by normalizing the scores on a per-batch basis using the mean and variance of each training batch. We then combine the distinct structural and semantic advantages into a unique score, with their relative contribution weighed by a hyperparameter α.

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

Our experiments indicate that $\alpha = 1$ is optimal; however α, like λ, can be tuned based on the model's zero-shot performance to accelerate learning convergence. Finally, we rank generated sequences according to their total score.

3.3. Architecture for Multiple Conditionings

Our architecture builds upon InstructPix2Pix [12], which itself adapts Stable Diffusion [55] with a key architectural modification: additional input channels in the first convolutional layer of the U-Net to incorporate the encoded input image. We extend this approach by adding further input channels to intake both the input and style images simultaneously. The weights for these newly added channels are initialized to zero to ensure stable training. We enhance performance by incorporating a cross-attention layer before feeding the visual prompt into the network. This mechanism helps better localize regions in both images that are relevant to the editing directive. Specifically, the query is formed from a linear projection of the concatenated VAE encodings of the input and style images, $\mathcal{E}(I_{\text{in}})$ and $\mathcal{E}(I_{\text{sty}})$, while the key and value are derived from projections of the CLIP-encoded instruction prompt. Importantly, the cross-attention output maintains the same spatial dimensions as the VAE-encoded images, preserving compatibility with the pre-trained U-Net architecture.

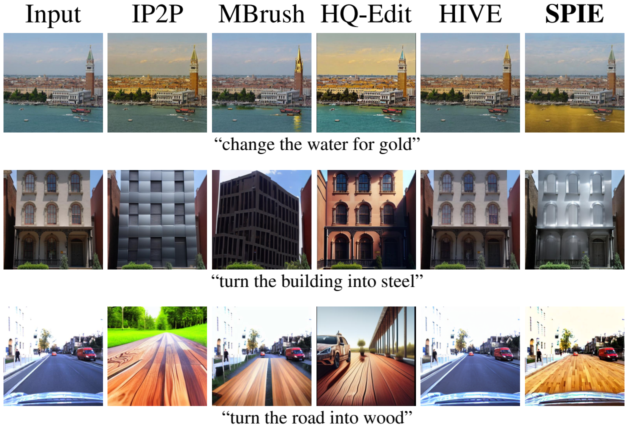

Input
IP2P
MBrush
HQ-Edit
HIVE
SPIE
"change the water for gold"
"turn the building into steel"
"turn the road into wood"

Figure 2. Qualitative comparison. SPIE outperforms its counterparts by significantly editing the image while sharply preserving the structure of regions unrelated to the instruction. We exclude the conditioning style image from the visualization since it is not applicable to the other methods. Additional samples are shown in Appendix 6.

For sampling, we leverage classifier-free guidance (CFG) [26], which shifts probability mass toward regions where an implicit classifier assigns high likelihood to the conditioning, thereby improving sample quality and faithfulness. In our case, we compose the CFG estimate with respect to both the input image, textual prompt, and visual prompt [40], allowing for more precise control over the editing process (see Appendix 6.1 for the complete derivation).

4. Experiments

This section presents a comprehensive analysis of our experimental results, including baseline comparisons, ablation studies, and applications in robotics. We demonstrate our method's effectiveness in performing precise and realistic edits across diverse scenarios. For consistency, we use the default guidance scale parameters from InstructPix2Pix and set our visual conditioning score to $s_{I_{\text{sty}}} = 3$ (discussed in Sec. 4.3).

Our evaluations focus on localized edits in complex scenes using images from the Oxford RobotCar [42] and Places [90] datasets, covering various edit types such as weather and material changes (see full list of edits in Sec. 6.2). We evaluate 29,500 images at 512 × 512 resolution, showing that our model outperforms baselines in realism and prompt alignment.

4.1. Baseline Comparisons

We evaluate SPIE against state-of-the-art baselines, including InstructPix2Pix (IP2P) [12], HIVE [86], MagicBrush (MBrush) [84], and HQ-Edit [29], all built on the stable diffusion v1.5 backbone.

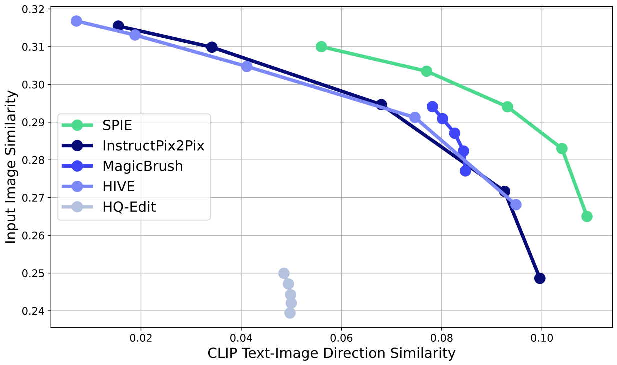


Figure 3. Quantitative comparison of instructional editing models. We plot the trade-off between input image consistency (Y-axis) and edit consistency (X-axis). Higher values indicate better performance for both metrics. For all methods, we fix the same parameters as in Brooks et al. [12] and vary $s_{I_{\text{in}}} \in [1.0, 2.0]$.

	Structural ↓		Semantic ↑				Human Preference ↑		
Method	Depth	$L_{2_{\text{out}}}$	CLIP_{in}	DINO_{in}	DSim_{in}	CLIP_{txt}	ImageReward	PickScore	HPSv2
IP2P	30.06	0.034	0.430	0.058	0.152	0.195	-0.648	19.18	21.60
MBrush	56.50	0.031	0.422	0.059	0.148	0.209	-0.420	19.16	21.67
HQ-Edit	90.81	0.135	0.399	0.057	0.141	0.206	-0.293	19.08	**24.79**
HIVE	18.09	**0.022**	0.438	0.064	0.145	0.177	-0.848	19.19	21.14
SPIE	**16.55**	0.025	**0.440**	**0.076**	**0.219**	**0.213**	**-0.284**	**19.21**	21.82

Table 1. Comparison of structural preservation through depth mask alignment and reconstruction metrics (left), semantic alignment with text and visual prompts (center), and human-preference-aligned metrics (right). Some metrics are computed for regions inside and outside the edit mask, denoted by indices "in" and "out" respectively. Text alignment is evaluated using descriptive prompts capturing all image information.

We begin by comparing the qualitative results in Figure 2. InstructPix2Pix often struggles to precisely locate the edit region, while MagicBrush and HQ-Edit generate unnatural edits with unrealistic prompts. HIVE tends to prioritize input fidelity over executing edits, sometimes resulting in insufficient modifications. In contrast, our method strikes a better balance between editing strength and input fidelity, producing edits that are both prompt-aligned and faithful to the original image. Additionally, SPIE mitigates text prompt-induced biases by suppressing irrelevant hallucinations, like mistaking "wood" for a forest.

We quantitatively assess the tradeoff between input fidelity and text alignment in Figure 3. Input fidelity is measured via cosine similarity of image patch embeddings outside the edit region, using grounded SAM2 for high-quality masking. We capture both high and low-level features by averaging scores across DINOv2 [47], CLIP [49], and DreamSim [19] encoders. Text alignment is evaluated using directional CLIP similarity [22], which measures how well the change in descriptive captions agrees with the change in input and generated images. Both metrics are antagonistic, increasing the desired edit strength will reduce the output's faithfulness to the input image. Our method achieves better directional similarity for the same level of image consistency compared to baselines. These quantitative results hence confirm that counterparts like HIVE emphasize preserving the input image over executing strong edits. In contrast, SPIE offers superior balance between edit strength and input fidelity. Additionally, Table 1 shows that while HIVE excels at reconstructing unchanged regions, our method closely matches this performance and outperforms others in depth mask alignment, indicating effective preservation of structure, crucial for visual coherence.

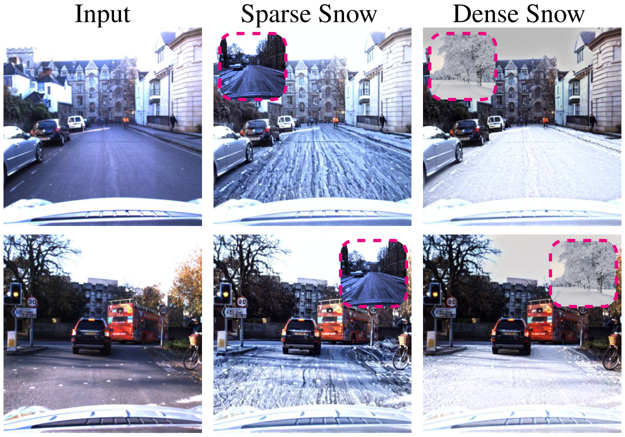


Figure 4. Visualization of the impact of the visual prompt (displayed in dashed lines contour) on generated samples when provided the text instruction "add snow on the road". SPIE effectively captures semantic nuances beyond that described in the text prompt. We present further applications of such nuanced control for materials in Fig. 10, demonstrating how our method can effectively handle diverse visual styles and attributes, such as varying colors, while maintaining structural coherence.

We demonstrate SPIE's ability to interpret subtle details beyond text prompts by training it to add dense and sparse snow layers without explicit text instructions. Figure 4 shows that our model effectively reproduces the visual style hinted at in the text prompt while respecting the input image's spatial composition. This underscores the benefit of leveraging a visual prompt to infuse fine-grained nuances without requiring a extensive text descriptions.

We quantitatively validate these observations in Table 1. Our evaluation focuses on two key aspects: visual semantic 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, across DINOv2, CLIP, and DreamSim embeddings. Text-image alignment (CLIP_{txt}) is measured through CLIP similarity between the edited image and output caption. SPIE surpasses all baselines, both in the visual and text-image alignments, confirming our method's efficacy to increase instruction fidelity.

Method	L1 ↓	L2 ↓	CLIP-I ↑	DINO ↑	CLIP-T ↑
IP2P	0.1122	0.0371	0.8524	0.7428	0.2764
MBrush	0.0740	0.0267	0.9166	0.8649	**0.2813**
HIVE	0.1092	0.0341	0.8519	0.7500	0.2752
SPIE	**0.0673**	**0.0187**	**0.9224**	**0.8743**	0.2768

(a) MagicBrush Test Set

Method	CLIP_{dir} ↑	CLIP_{im} ↑	CLIP_{out} ↑	L1 ↓	DINO ↑
IP2P	0.078	0.834	0.219	0.121	0.762
MBrush	0.090	0.838	0.222	0.100	0.776
HIVE	0.061	0.882	0.213	0.083	0.822
Emu Edit	**0.109**	0.859	**0.231**	0.094	0.819
SPIE	0.063	**0.897**	0.221	**0.078**	**0.858**

(b) Emu Edit Test Set

Table 2. Evaluation on benchmarks spanning diverse editing tasks.

We also compare the scores obtained from T2I synthesis preference prediction models, namely ImageReward [79], PickScore [34], and HPSv2 [75], which emulate human preferences. We find in Table 1 that our method outperforms its counterparts across the different edits. This confirms SPIE's superior ability to preserve the essential structural features of the input image, which are crucial for perceived realism, while also aligning effectively with the specified editing instructions.

Finally, we extend our baseline comparisons to standardized benchmarks that encompass a broader range of editing tasks. We evaluate our method in Tables 2a and 2b, without additional training. SPIE outperforms all baselines in structure preservation and surpasses IP2P in semantics, demonstrating strong transferability beyond region-based editing.

4.2. Sim-to-Real Editing

SPIE demonstrates utility beyond creative applications by enhancing the visual realism of simulated environments, addressing a critical limitation in their use for robotics research. In robotics, evaluating generalist manipulation policies poses significant challenges due to the scalability and reproducibility constraints of real-world testing. SIMPLER [37], a framework for simulation-based evaluation, aims to provide a reliable proxy for real-world assessments. A major challenge highlighted by Li et al. [37] is the visual disparity between simulated environments and their real-world counterparts, which can undermine the accuracy of policy evaluation. They mitigate this gap with a 2-step approach called visual matching (VisMatch), which overlays simulated elements onto real-world backgrounds, and bakes their textures and colors from real-world images. However, this method has notable limitations: it relies heavily on human effort, as texture matching is not automated and requires extensive curation of visual assets alongside access to 3D modeling tools. Hence, it does not scale efficiently, as assembling new scenes demands additional human input for each instance.

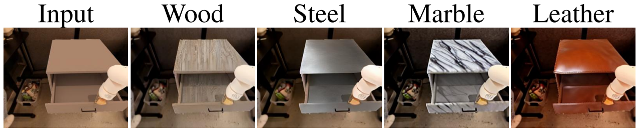


Figure 5. Examples of a simulated scene edited by our method, showcasing enhanced realism compared to the original image across various styles. See Appendix 6.4 for more variants.

	Visual Domain	Open	Close	Average
MMRV↓	SIMPLER-VarAgg	0.000	0.130	0.083
	SIMPLER-VisMatch	0.000	0.130	0.083
	SPIE	0.000	0.000	0.000
Pearson r↑	SIMPLER-VarAgg	0.915	0.756	0.964
	SIMPLER-VisMatch	0.987	0.891	0.972
	SPIE	0.917	0.978	0.966

Table 3. Comparison of visual domains for RT-1 policy evaluation on Google Robot tasks. Using our method results in much stronger correlation with real evaluation than using the SIMPLER methods. See Table 4 for a detailed breakdown of results per policy.

To further reduce the sim-to-real gap, we propose applying SPIE to simulated scenes. Specifically, we finetune an editing diffusion model using only 5 reference images to produce realistic edits in a given style within the environment. Such model can then be used to modify the robots' observations of their simulated environment during evaluation, introducing realistic textures and materials that better align with real-world settings. We conduct experiments in the opening/closing drawer task of the Google Robot environment, as the table texture domain shift is known for being one of the most difficult for policies to generalize effectively [77]. To evaluate our method, we specialize expert models to generate variants of the cabinet in diverse materials: wood, gold, leather, stone, steel, and marble.

Visual exemplars in Fig. 5 demonstrate that our method can convincingly alter simulated scenes to resemble real-world counterparts across various styles. Quantitatively, we assess the realism of our generated images by evaluating robot policies trained in the real-world on our edited images. Following Li et al. [37], we conduct evaluations using multiple RT-1 [11, 16] checkpoints (see Appendix Sec 6.4 for further experiment details). Table 3 summarizes the results on two metrics: Mean Maximum Rank Violation (MMRV) and Pearson correlation coefficient (Pearson r), which respectively measure the ranking and linear consistency between simulated and real-world performance. We evaluate SPIE, VisMatch, and the variant aggregation method (VarAgg) [37], which combines many visually randomized versions of a simulated scene, including variations in drawer texture. Some policies are highly sensitive to visual discrepancies between simulation and reality, which

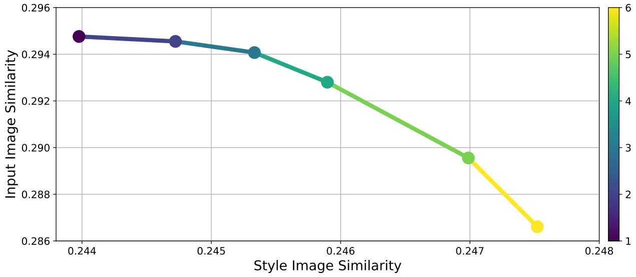


Figure 6. We plot the trade-off between consistency with the input image (Y-axis) and consistency with the visual prompt (X-axis). For both metrics, higher is better. We fix the same parameters as in Brooks et al. [12] and vary $s_{I_{\text{sty}}} \in [1.0, 6.0]$

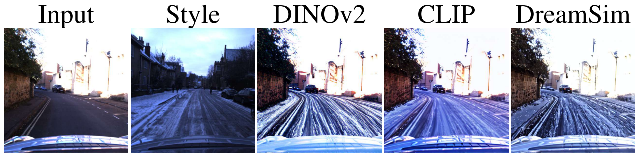


Figure 7. Qualitative comparison of the visual prompt's reproduction induced by different encoders. Best viewed zoomed-in.

exacerbates the challenges introduced by domain shifts and causes VarAgg to perform worse than VisMatch. Notably, SPIE outperforms both VarAgg and VisMatch, achieving higher MMRV scores—identified by Li et al. [37] as the more robust metric—and higher Pearson r scores for the closing task. These stronger results indicate that our edits provide a more realistic proxy for real-world evaluations. Improving finer details, such as drawer handles, presents a promising way to further enhance alignment between simulated and real-world environments, as suggested by the lower performance on the opening task against VisMatch. Overall, SPIE offers a scalable solution for bridging the visual realism gap in robotics policy testing by reducing reliance on manual efforts and enabling automated generation of realistic edits across diverse styles.

4.3. Ablation Study

Classifier-free-guidance scale. In Figure 6, we examine how varying the classifier-free-guidance scale $s_{I_{\text{sty}}}$ from Eq. 13 affects the edit's alignment with the visual prompt. Increasing its value enhances alignment with the visual prompt but reduces similarity with the input image. We find that values between 2 to 5 yield the best results, and use $s_{I_{\text{sty}}} = 3$ for quantitative evaluations in Sec. 4.1. 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 an optimal balance between faithfulness to the input and alignment with the visual prompt.

Encoder Choice. Different encoders capture distinct information, influencing the learning process and output quality. Other works commonly use DINOv2 and CLIP as evaluation metrics for generated sample quality [12, 63, 84]. However, DreamSim recently showed to outperform those encoders in alignment with human preferences. To assess the effect of encoder choice, we train three versions of SPIE on the same task—editing sparse snow—systematically replacing the encoder in Eq. 9. We focus on qualitative results, as semantic visual alignment metrics based on these same encoders lack impartiality. We find in Fig. 7 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, with an unwanted purple tint across the frame. Contrastingly, DreamSim better enforces the color and structure of the visual prompt. It does not lead to learning spurious cues like an unrelated tint or saturated colors, and best reproduces the structure of the snow stripes. This results in more realistic samples with stronger alignment to the prompt.

5. Conclusion

In this paper, we introduce SPIE, a novel approach to instruction-based image editing that enhances semantic alignment and structural preservation through few-steps post-training, effectively mimicking human preferences without direct feedback. Our method demonstrates that these improvements can be achieved by leveraging AI-generated supervision, circumventing the need for extensive human annotations or large-scale datasets. SPIE learns to capture and reproduce intricate details in visual prompts, with only 5 examples per concept, further reducing the reliance on elaborate textual prompts. Our approach significantly improves upon previous state-of-the-art methods in balancing faithfulness to the input image and alignment with instruction prompts, resulting in samples with higher perceived realism. The efficient finetuning approach with visual prompts enables complex sim-to-real edits using minimal reference images, demonstrating potential for high-quality simulated evaluation environments in robotics.

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. Future work could explore flexible constraints in masking and depth alignment operations, allowing for more substantial structural modifications like adding and removing elements. The model may inherit biases from the pre-trained InstructPix2Pix model and the AI models providing the alignment supervision signal. However, this limitation can be mitigated by substituting these components with suitable alternatives in a modular fashion.

We hope to inspire further research in online reinforcement learning for T2I models and additional studies on crafting AI-generated rewards for objectives that better align with human intent.

Acknowledgments

We would like to thank Sotiris Anagnostidis, Victor Miara and Kevin Kasa for insightful conversations, and Litu Rout for his generous feedback on this work.

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, 4

[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. 4

[11] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil J Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Kevin Sayed, Jaspiar Singh, Sumedh Sontakke, Austin Stone, Clayton Tan, Huong Tran, Vincent Vanhoucke, Steve Vega, Quan Vuong, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-1: Robotics transformer for real-world control at scale, 2023. 7

[12] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions, 2023. 1, 2, 5, 6, 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, 2

[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

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

[16] Embodiment Collaboration, Abby O'Neill, Abdul Rehman, Abhinav Gupta, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, Albert Tung, Alex Bewley, Alex Herzog, Alex Irpan, Alexander Khazatsky, Anant Rai, Anchit Gupta, Andrew Wang, Andrey Kolobov, Anikait Singh, Animesh Garg, Aniruddha Kembhavi, Annie Xie, Anthony Brohan, Antonin Raffin, Archit Sharma, Arefeh Yavary, Arhan Jain, Ashwin Balakrishna, Ayzaan Wahid, Ben Burgess-Limerick, Beomjoon Kim, Bernhard Schölkopf, Blake Wulfe, Brian Ichter, Cewu Lu, Charles Xu, Charlotte Le, Chelsea Finn, Chen Wang, Chenfeng Xu, Cheng Chi, Chenguang Huang, Christine Chan, Christopher Agia, Chuer Pan, Chuyuan Fu, Coline Devin, Danfei Xu, Daniel Morton, Danny Driess, Daphne Chen, Deepak Pathak, Dhruv Shah, Dieter Büchler, Dinesh Jayaraman, Dmitry Kalashnikov, Dorsa Sadigh, Edward Johns, Ethan Foster, Fangchen Liu, Federico Ceola, Fei Xia, Feiyu Zhao, Felipe Vieira Frujeri, Freek Stulp, Gaoyue Zhou, Gaurav S. Sukhatme, Gautam Salhotra, Ge Yan, Gilbert Feng,

Giulio Schiavi, Glen Berseth, Gregory Kahn, Guangwen Yang, Guanzhi Wang, Hao Su, Hao-Shu Fang, Haochen Shi, Henghui Bao, Heni Ben Amor, Henrik I Christensen, Hiroki Furuta, Homanga Bharadhwaj, Homer Walke, Hongjie Fang, Huy Ha, Igor Mordatch, Ilija Radosavovic, Isabel Leal, Jacky Liang, Jad Abou-Chakra, Jaehyung Kim, Jaimyn Drake, Jan Peters, Jan Schneider, Jasmine Hsu, Jay Vakil, Jeannette Bohg, Jeffrey Bingham, Jeffrey Wu, Jensen Gao, Jiaheng Hu, Jiajun Wu, Jialin Wu, Jiankai Sun, Jianlan Luo, Jiayuan Gu, Jie Tan, Jihoon Oh, Jimmy Wu, Jingpei Lu, Jingyun Yang, Jitendra Malik, João Silvério, Joey Hejna, Jonathan Booher, Jonathan Tompson, Jonathan Yang, Jordi Salvador, Joseph J. Lim, Junhyek Han, Kaiyuan Wang, Kanishka Rao, Karl Pertsch, Karol Hausman, Keegan Go, Keerthana Gopalakrishnan, Ken Goldberg, Kendra Byrne, Kenneth Oslund, Kento Kawaharazuka, Kevin Black, Kevin Lin, Kevin Zhang, Kiana Ehsani, Kiran Lekkala, Kirsty Ellis, Krishan Rana, Krishnan Srinivasan, Kuan Fang, Kunal Pratap Singh, Kuo-Hao Zeng, Kyle Hatch, Kyle Hsu, Laurent Itti, Lawrence Yunliang Chen, Lerrel Pinto, Li Fei-Fei, Liam Tan, Linxi ”Jim” Fan, Lionel Ott, Lisa Lee, Luca Weihs, Magnum Chen, Marion Lepert, Marius Memmel, Masayoshi Tomizuka, Masha Itkina, Mateo Guaman Castro, Max Spero, Maximilian Du, Michael Ahn, Michael C. Yip, Mingtong Zhang, Mingyu Ding, Minho Heo, Mohan Kumar Srirama, Mohit Sharma, Moo Jin Kim, Naoaki Kanazawa, Nicklas Hansen, Nicolas Heess, Nikhil J Joshi, Niko Suenderhauf, Ning Liu, Norman Di Palo, Nur Muhammad Mahi Shafiullah, Oier Mees, Oliver Kroemer, Osbert Bastani, Pannag R Sanketi, Patrick ”Tree” Miller, Patrick Yin, Paul Wohlhart, Peng Xu, Peter David Fagan, Peter Mitrano, Pierre Sermanet, Pieter Abbeel, Priya Sundaresan, Qiuyu Chen, Quan Vuong, Rafael Rafailov, Ran Tian, Ria Doshi, Roberto Mart’in-Mart’in, Rohan Baijal, Rosario Scalise, Rose Hendrix, Roy Lin, Runjia Qian, Ruohan Zhang, Russell Mendonca, Rutav Shah, Ryan Hoque, Ryan Julian, Samuel Bustamante, Sean Kirmani, Sergey Levine, Shan Lin, Sherry Moore, Shikhar Bahl, Shivin Dass, Shubham Sonawani, Shubham Tulsiani, Shuran Song, Sichun Xu, Siddhant Haldar, Siddharth Karamcheti, Simeon Adebola, Simon Guist, Soroush Nasiriany, Stefan Schaal, Stefan Welker, Stephen Tian, Subramanian Ramamoorthy, Sudeep Dasari, Suneel Belkhale, Sungjae Park, Suraj Nair, Suvir Mirchandani, Takayuki Osa, Tanmay Gupta, Tatsuya Harada, Tatsuya Matsushima, Ted Xiao, Thomas Kollar, Tianhe Yu, Tianli Ding, Todor Davchev, Tony Z. Zhao, Travis Armstrong, Trevor Darrell, Trinity Chung, Vidhi Jain, Vikash Kumar, Vincent Vanhoucke, Wei Zhan, Wenxuan Zhou, Wolfram Burgard, Xi Chen, Xiangyu Chen, Xiaolong Wang, Xinghao Zhu, Xinyang Geng, Xiyuan Liu, Xu Liangwei, Xuanlin Li, Yansong Pang, Yao Lu, Yecheng Jason Ma, Yejin Kim, Yevgen Chebotar, Yifan Zhou, Yifeng Zhu, Yilin Wu, Ying Xu, Yixuan Wang, Yonatan Bisk, Yongqiang Dou, Yoonyoung Cho, Youngwoon Lee, Yuchen Cui, Yue Cao, Yueh-Hua Wu, Yujin Tang, Yuke Zhu, Yunchu Zhang, Yunfan Jiang, Yunshuang Li, Yunzhu Li, Yusuke Iwasawa, Yutaka Matsuo, Zehan Ma, Zhuo Xu, Zichen Jeff Cui, Zichen Zhang, Zipeng Fu, and Zipeng Lin. Open x-embodiment: Robotic learning datasets and rt-x models, 2024. 7

[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] Tsu-Jui Fu, Wenze Hu, Xianzhi Du, William Yang Wang, Yinfei Yang, and Zhe Gan. Guiding instruction-based image editing via multimodal large language models, 2024. 1, 2

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

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

[23] 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. 3

[24] 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

[25] Amir Hertz, Andrey Voynov, Shlomi Fruchter, and Daniel Cohen-Or. Style aligned image generation via shared attention, 2024. 3

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

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

[28] Yuzhou Huang, Liangbin Xie, Xintao Wang, Ziyang Yuan, Xiaodong Cun, Yixiao Ge, Jiantao Zhou, Chao Dong, Rui Huang, Ruimao Zhang, and Ying Shan. Smartedit: Exploring complex instruction-based image editing with multimodal large language models, 2023. 1, 2

[29] Mude Hui, Siwei Yang, Bingchen Zhao, Yichun Shi, Heng Wang, Peng Wang, Yuyin Zhou, and Cihang Xie. Hq-edit: A high-quality dataset for instruction-based image editing, 2024. 1, 2, 5

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

[31] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models, 2022. 15

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

[33] 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

[34] 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

[35] 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

[36] 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

[37] 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. 7, 8, 16

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

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

[40] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European Conference on Computer Vision*, pages 423–439. Springer, 2022. 5

[41] 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

[42] 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, 15

[43] 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

[44] 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

[45] 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

[46] 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

[47] 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

[48] 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

[49] 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

[50] 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

[51] 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

[52] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022. 1, 2

[53] 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

[54] 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

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

[56] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. 2

[57] 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

[58] 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. 3

[59] 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

[60] 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

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

[62] 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

[63] 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, 8, 15

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

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

[66] 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

[67] 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. 3

[68] 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

[69] 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

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

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

[72] 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

[73] 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. 3

[74] 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

[75] 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

[76] 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

[77] Annie Xie, Lisa Lee, Ted Xiao, and Chelsea Finn. Decomposing the generalization gap in imitation learning for visual robotic manipulation, 2023. 7

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

[79] 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

[80] 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, 14

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

[82] 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

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

[84] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing, 2024. 1, 2, 5, 8, 15

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

[86] 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. 1, 3, 5

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

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

[89] Haozhe Zhao, Xiaojian Ma, Liang Chen, Shuzheng Si, Rujie Wu, Kaikai An, Peiyu Yu, Minjia Zhang, Qing Li, and Baobao Chang. Ultraedit: Instruction-based fine-grained image editing at scale, 2024. 1, 2

[90] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene

recognition using places database. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2014. 5, 15

[91] 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 image $c_{I_{\text{in}}}$, a reference 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 [30] 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:

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

This corresponds to the following formulation of classifier-free guidance, with three classifier-free-guidance scales:

$$\begin{aligned} \tilde{e}_\theta(z_t, c_{I_{\text{in}}}, c_{I_{\text{sty}}}, c_T) = &\ e_\theta(z_t, \varnothing, \varnothing, \varnothing) \\ &+ s_{I_{\text{in}}} \cdot (e_\theta(z_t, c_{I_{\text{in}}}, \varnothing, \varnothing) - e_\theta(z_t, \varnothing, \varnothing, \varnothing)) \\ &+ s_{I_{\text{sty}}} \cdot \big(e_\theta(z_t, c_{I_{\text{in}}}, c_{I_{\text{sty}}}, \varnothing) - e_\theta(z_t, c_{I_{\text{in}}}, \varnothing, \varnothing)\big) \\ &+ s_T \cdot \big(e_\theta(z_t, c_{I_{\text{in}}}, c_{I_{\text{sty}}}, c_T) - e_\theta(z_t, c_{I_{\text{in}}}, c_{I_{\text{sty}}}, \varnothing)\big) \end{aligned} \tag{13}$$

6.2. Training Details

In training, we initialize our model from the InstructPix2Pix checkpoint. Depending on the specialization, we train as few as six steps at a resolution of 256 × 256, with a total batch size of 256 samples. For a fair comparison, we maintain the same RL training hyperparameters as in D3PO [80] without conducting hyperparameter optimization. Additionally, we do not optimize for the best text prompt for each edit type, as we design SPIE to leverage the visual prompt as the primary carrier of semantic information, minimizing reliance on extensive textual descriptions.

The selection of hyperparameters λ and α requires balancing multiple objectives for optimal performance. While our recommended values work well across most scenarios, these parameters can be slightly fine-tuned based on the model's zero-shot capabilities to further optimize learning convergence. This flexibility allows practitioners to adapt the framework to specific task requirements while maintaining robust performance.

6.3. Real Image Experiment Details

Our evaluations focus on the ability to perform localized edits in complex scenes containing multi-level information, including local objects, global layout, and background environments that must remain unchanged unless explicitly instructed. We use two datasets for our experiments.

With the Oxford RobotCar Dataset [42], we train SPIE to perform seven types of edits: adding dense snow on the road, adding sparse snow on the road, adding rain on the road, adding sand on the road, changing the road to gold, changing the road to wood, and changing the entire scene to nighttime (using a full-frame mask).

With the Places Dataset [90], we train our model on six additional edit types: changing water to gold, changing a bed to leather, changing a building facade to steel, changing a telephone booth to stone, changing a lighthouse to terracotta, changing a train to wood.

When selecting edit types for our experiments, we aimed to cover a diverse range of materials, textures, and environmental modifications to demonstrate the versatility and robustness of our approach. This diverse selection allows us to assess our method's performance across both realistic modifications (weather changes) and more stylistic transformations (material changes), while testing its ability to maintain structural coherence across different scene types, object scales, and edit complexities.

To prevent overfitting on spurious cues, we alternate between five conditioning style images related to the same text prompt during training. Our evaluations cover 29,500 images at a resolution of 512×512 across both datasets and edit types (2,500 images for each of the seven Oxford RobotCar edits and 2,000 images for each of the six Places edits).

For baseline comparisons, we evaluate the InstructPix2Pix checkpoint from which we initialize our model, and the best publicly available versions of HIVE, HQ-Edit, and MagicBrush based on StableDiffusion v1.5. Some evaluation results on standardized benchmarks were taken directly from the ones reported in the original MagicBrush [84] and Emu Edit [63] papers.

While our model is trained at 256×256 resolution, we find it generalizes well to 512×512 resolution at inference time. We generate qualitative results at 512×512 resolution with 100 denoising steps using an Euler ancestral sampler with denoising variance schedule proposed by Karras et al. [31]. Editing an image with our model takes approximately 9 seconds on an A100 GPU.

6.4. Sim-to-real Experiment Details

Google Robot Evaluation Setup	Policy	Open / Close Drawer		
		Open	Close	Average
Real Eval	RT-1 (Converged)	0.815	0.926	0.870
	RT-1 (15%)	0.704	0.889	0.796
	RT-1-X	0.519	0.741	0.630
	RT-1 (Begin)	0.000	0.000	0.000
SIMPLER Eval (Variant Aggregation)	RT-1 (Converged)	0.270	0.376	0.323
	RT-1 (15%)	0.212	0.323	0.267
	RT-1-X	0.069	0.519	0.294
	RT-1 (Begin)	0.005	0.132	0.069
	MMRV↓	0.000	0.130	0.083
	Pearson r↑	0.915	0.756	0.964
SIMPLER Eval (Visual Matching)	RT-1 (Converged)	0.601	0.861	0.730
	RT-1 (15%)	0.463	0.667	0.565
	RT-1-X	0.296	0.891	0.597
	RT-1 (Begin)	0.000	0.278	0.139
	MMRV↓	0.000	0.130	0.083
	Pearson r↑	0.987	0.891	0.972
SPIE	RT-1 (Converged)	0.471	0.810	0.640
	RT-1 (15%)	0.259	0.619	0.439
	RT-1-X	0.180	0.608	0.394
	RT-1 (Begin)	0.021	0.058	0.040
	MMRV↓	0.000	0.000	0.000
	Pearson r↑	0.917	0.978	0.966

Table 4. Real-world, standard SIMPLER environment, and our visually-edited environment evaluation results across different policies on the Google Robot "(open/close) (top/middle/bottom) drawer" task. We present success rates for the 'Variant Aggregation' and 'Visual Matching' approaches, as well as our novel visually-edited environments with seven material styles. We calculate the Mean Maximum Rank Violation ('MMRV', lower is better) and the Pearson correlation coefficient ('Pearson r', higher is better) to assess the alignment between simulation and real-world relative performances across different policies.

In this section, we provide detailed descriptions of our robotics experiments using the SIMPLER environments. We follow the same evaluation protocol as established in SIMPLER [37], focusing on a language-conditioned drawer manipulation task where the robot must "(open/close) (top/middle/bottom) drawer." The robot is positioned in front of a cabinet with three drawers and must manipulate the specified drawer according to the instruction. For our simulation experiments, we also place the robot at 9 different grid positions within a rectangular area on the floor, resulting in $9 \times 3 \times 2 = 54$ total trials.

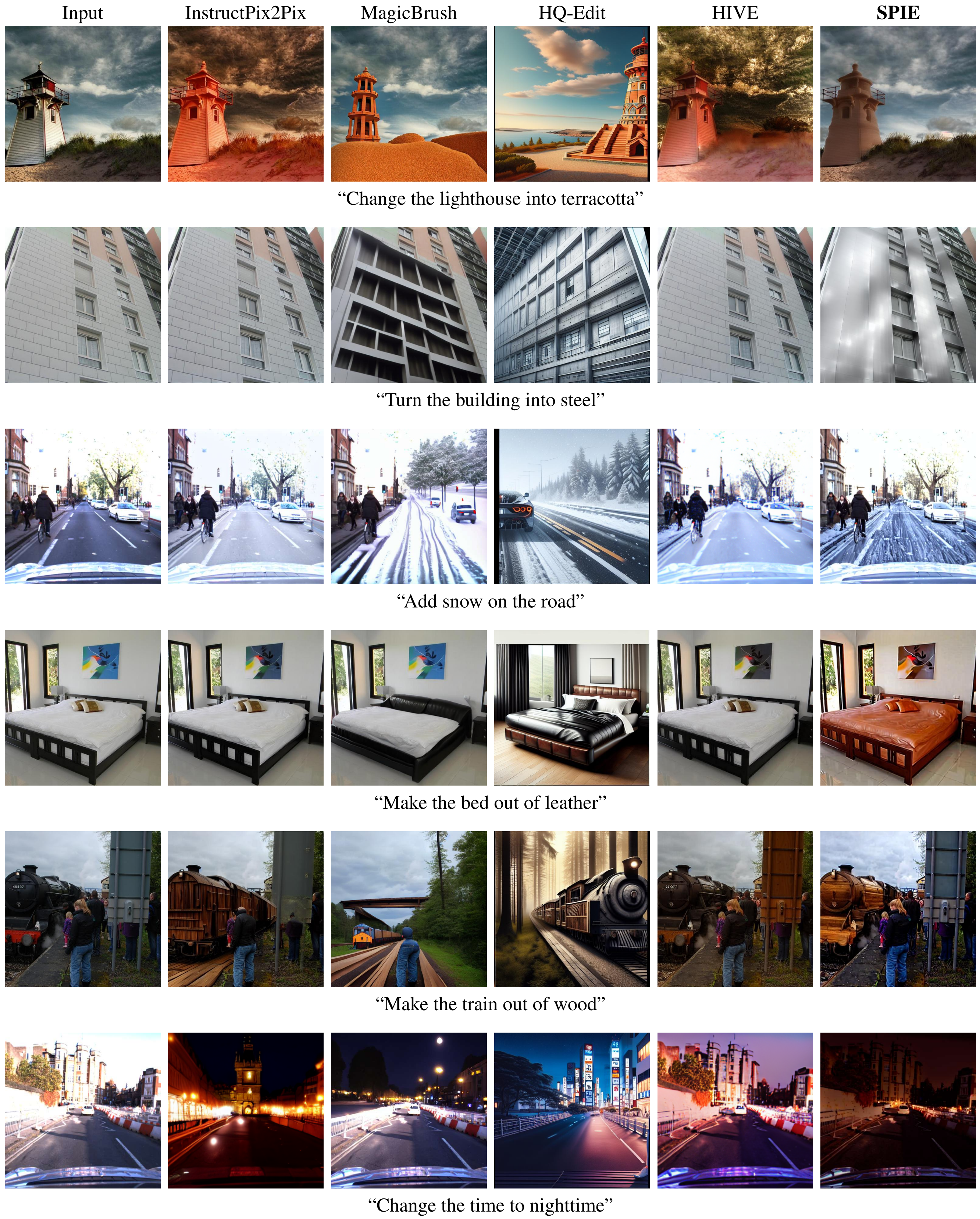
Following SIMPLER, we conduct experiments on RT-1 checkpoints at various training stages: RT-1-X, RT-1 trained to convergence (RT-1 Converged), RT-1 at 15% of training steps (RT-1 15%), and RT-1 at the beginning of training (RT-1 Begin).

We train our model to modify the cabinet's material using seven different styles: gold, leather, white marble, black marble, steel, stone, and wood. Importantly, we only modify the visual appearance of the cabinet without changing any physical properties such as friction coefficients, material density, center of mass, or static and dynamic friction. Since our method involves a non-deterministic diffusion process, we extend the SIMPLER protocol by averaging success rates across three different random seeds and across the seven different edit styles to produce lower-variance performance estimates.

For the VisMatch, VarAgg and Real evaluation results presented in Table 4, we directly reference the values reported in SIMPLER.

6.5. Additional Qualitative Results

This section of the appendix provides supplementary qualitative results, including a comparative evaluation against baselines (Figures 8 and 9), demonstrating SPIE's superior performance in structural preservation, semantic alignment, and realism. We also provided a extended visualization of the impact of different visual prompts on generated samples (Figure 10), showcasing how SPIE captures semantic nuances beyond text prompts. Additionally, we present examples of simulated scenes edited with enhanced realism (Figure 11), and highlight the flexibility of our approach in replicating visual prompts across diverse scenes (Figure 12). Finally, we display an extensive array of realistic edits generated by our method (Figures 13 and 14), illustrating precise structural preservation and semantic alignment across various scenes and styles.



“Change the lighthouse into terracotta”

“Turn the building into steel”

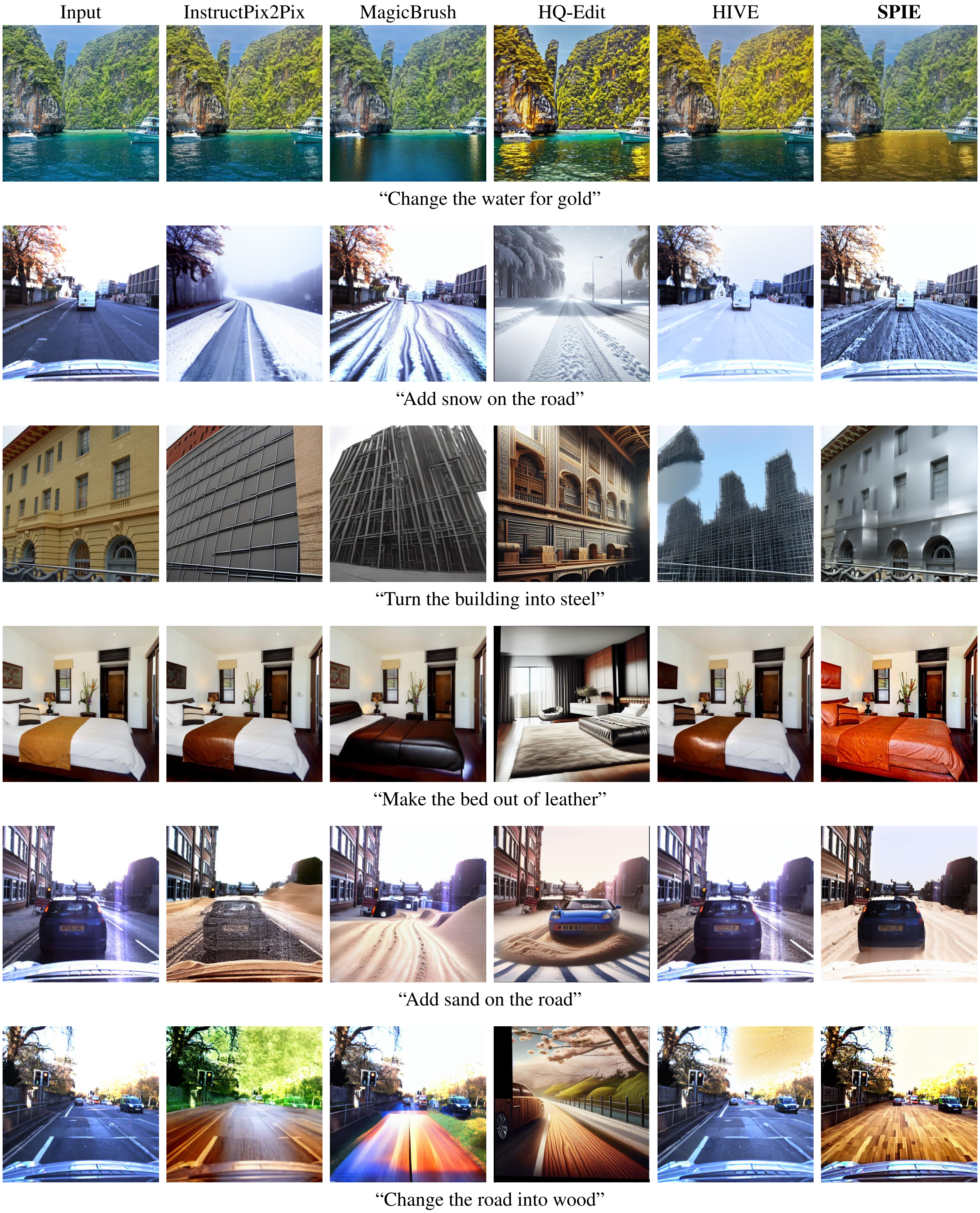
“Add snow on the road”

“Make the bed out of leather”

“Make the train out of wood”

“Change the time to nighttime”

Figure 8. Comparative evaluation of our method against baselines on a diverse set of prompts and images, highlighting superior performance in structural preservation, semantic alignment, and realism.



“Change the water for gold”

“Add snow on the road”

“Turn the building into steel”

“Make the bed out of leather”

“Add sand on the road”

“Change the road into wood”

Figure 9. Comparative evaluation of our method against baselines on a diverse set of prompts and images, highlighting superior performance in structural preservation, semantic alignment, and realism.

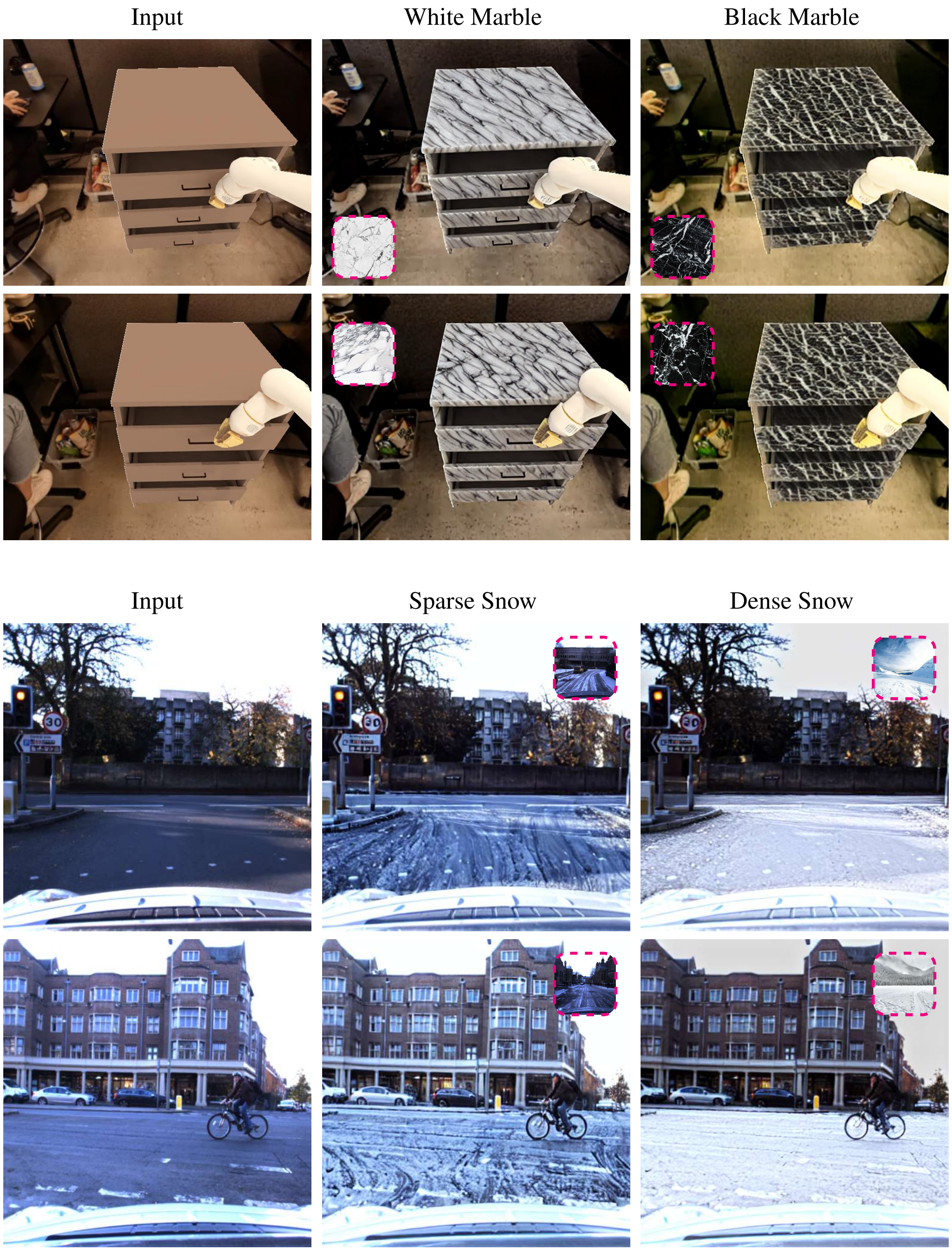


Figure 10. Visualization of the impact of different visual prompts on generated samples when provided a same text instruction. We show edits of both simulation and real-world images. Displayed within the dashed frame is one of the 5 style conditioning images relevant to this edit. Our method effectively captures semantic nuances beyond that described in the text prompt, understanding that the generated marble should be white or black, and that the generated snow should be sparse or dense.

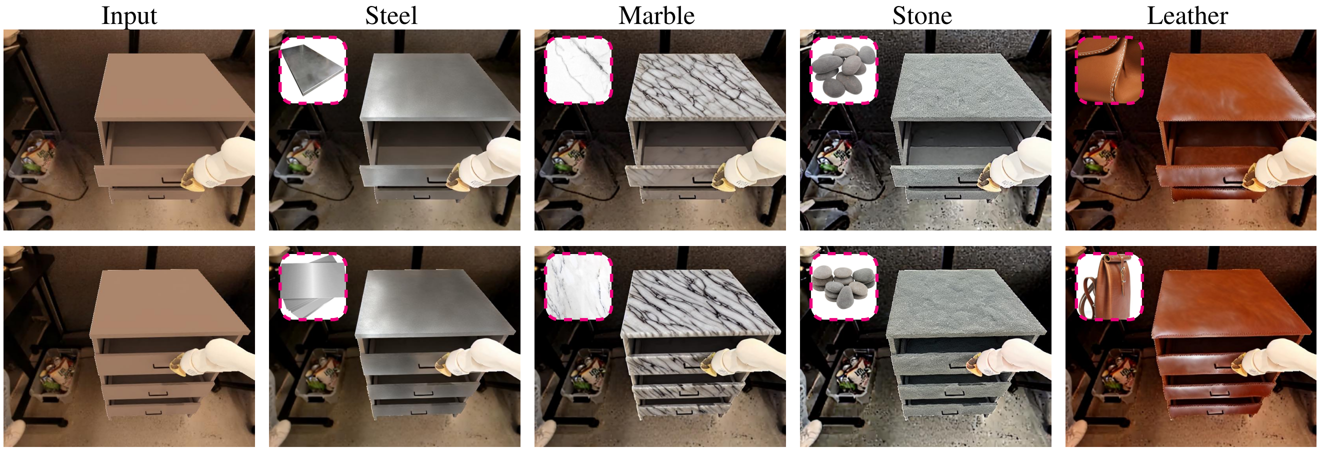


Figure 11. Examples of a simulated scene edited by our method, showcasing enhanced realism compared to the original image across various styles. When training the diffusion model across the diverse styles, we use the same text prompt format: *"make the cupboard out of [X]"*.

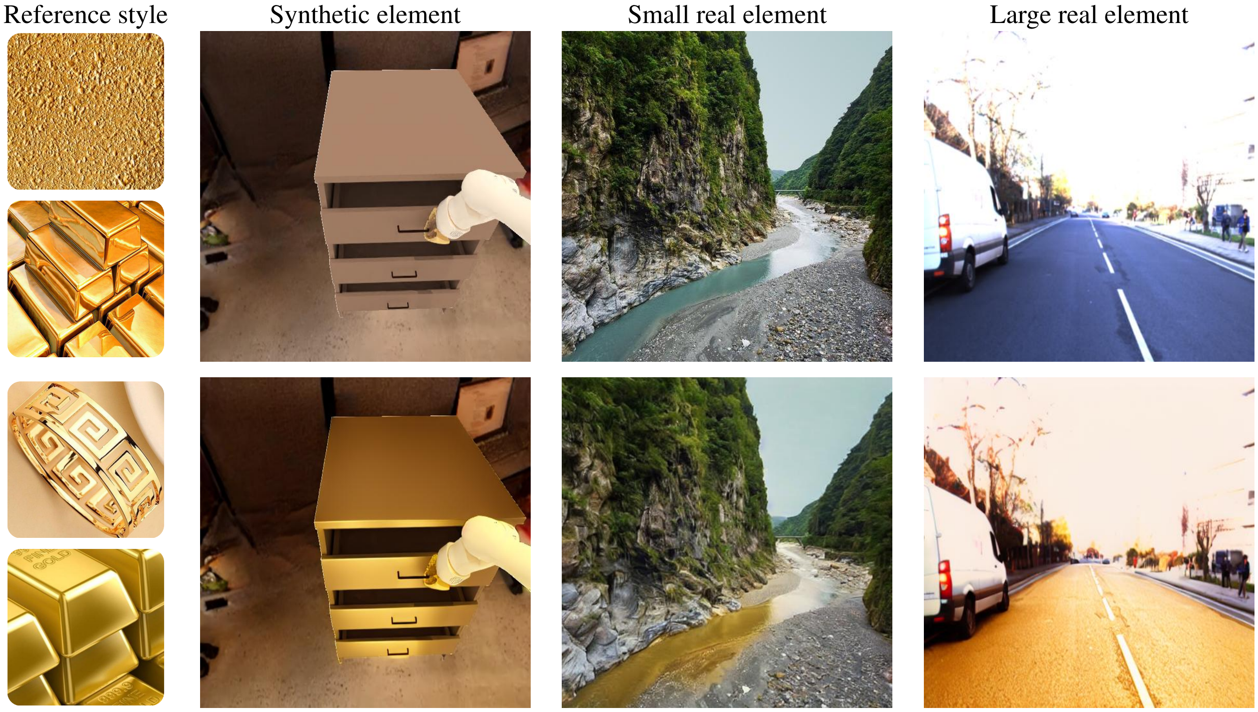
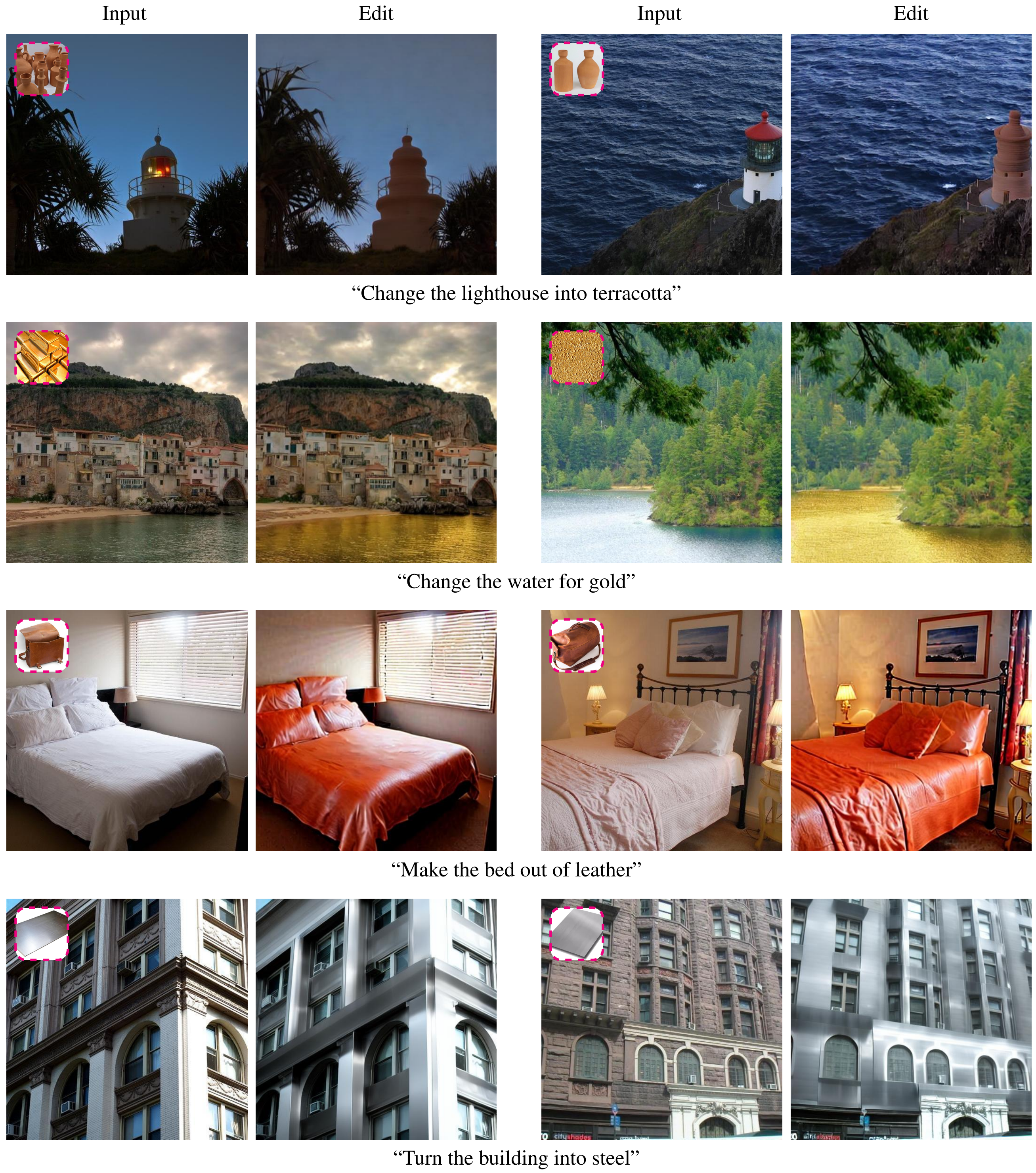


Figure 12. The visual prompts can be precisely replicated across scenes with diverse contexts and layouts, demonstrating the flexibility of our approach. We showcase variants of input images, including synthetic and real-world scenes, small and large elements, and various materials, to illustrate the model's ability to generalize effectively. Notably, our method achieves this versatility by leveraging only a few visual exemplars during training, ensuring robust performance across a wide range of inputs.



“Change the lighthouse into terracotta”

“Change the water for gold”

“Make the bed out of leather”

“Turn the building into steel”

Figure 13. Diverse examples of realistic edits generated by our method, demonstrating precise structural preservation and semantic alignment across various scenes and styles. We show the input images, as well as both the text and visual prompts used to generate these edits.

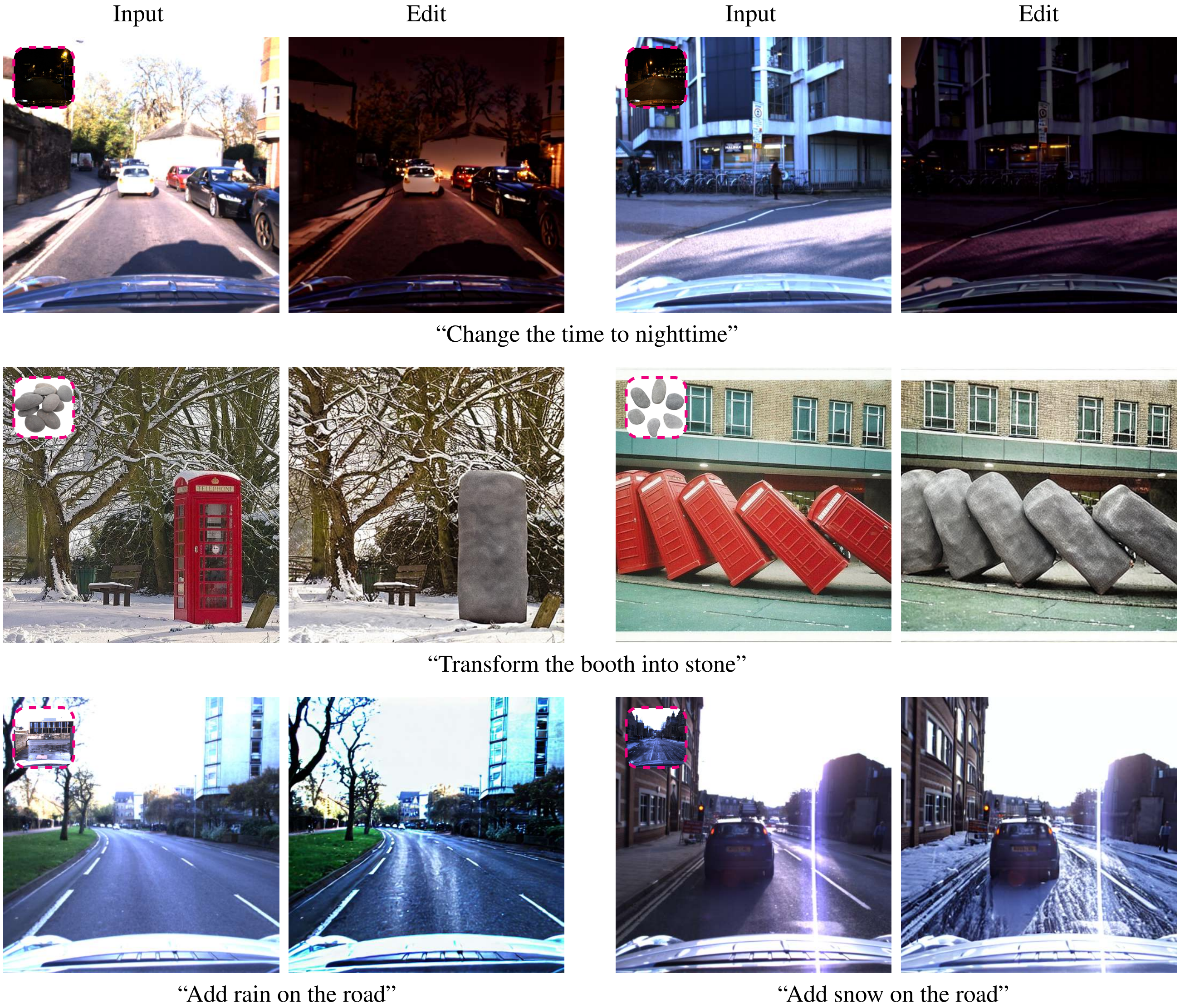

Input
Edit
Input
Edit
“Change the time to nighttime”
“Transform the booth into stone”
“Add rain on the road”
“Add snow on the road”

Figure 14. Diverse examples of realistic edits generated by our method, demonstrating precise structural preservation and semantic alignment across various scenes and styles. We show the input images, as well as both the text and visual prompts used to generate these edits