Overcoming Source Object Grounding for Semantic Image Editing

Yeonjoon Jung[♡] **Seungtaek Choi**[♣] **Seung-won Hwang**[♠♡*]
[♡]Interdisciplinary Program in Artificial Intelligence, Seoul National University, South Korea
[♠]Department of Computer Science and Engineering, Seoul National University, South Korea
[♣]Division of Language & AI, Hankuk University of Foreign Studies, South Korea
{y970120, seungwonh}@snu.ac.kr
hist0134@naver.com

Abstract

Recent diffusion models have demonstrated remarkable capabilities in text-to-image generation. However, their stochastic denoising process often causes semantic image editing (SIE) models to misapply textual instructions. That is, models often leave the source object unchanged or erroneously alter the background. We refer to this challenge as source object grounding. To address this challenge, we introduce R-SIE, a region-wise SIE framework. During the inference, R-SIE models noise separately for distinct image regions, enabling precise control over the transformed areas. To reinforce the inference, we devise an automatic pipeline leveraging bounding boxes to generate unambiguous training data. Additionally, we propose two region-focused metrics, CLIP-Region Class (CLIP-RC) and CLIP-Global Context (CLIP-GC), to independently assess how well the source object is edited and the background is preserved, respectively. Experimental results demonstrate that region-wise diffusion improves existing baselines, and our data generation pipeline further enhances these improvements.[1]

1 Introduction

Recent large-scale diffusion models, such as Gemini 2.0 (DeepMind, 2025) and Grok 3 (xAI, 2025), have demonstrated remarkable capabilities in text-to-image generation. These models are trained to reconstruct clean images by denoising corrupted latent representations. To generate images aligned with user intent, diffusion models interpret user guidance provided in textual or visual form (Rombach et al., 2022).

In practical scenarios involving real photographs, there is a growing demand for precise, region-specific image editing, commonly referred to as Semantic Image Editing (SIE). Given an input image and a textual instruction, the goal of SIE is to transform a specific object region (*i.e.*, the source object) while preserving the surrounding scene (*i.e.*, the background; Oh et al., 2001). For example, in Figure 1, SIE modifies only the central cake into a donut. In the latent space, this transformation manifests as a localized difference between the input and output image representations (Souza et al., 2025), necessitating the diffusion model to capture region-specific latent variations.

We contend that a core challenge in SIE is accurately grounding the provided guidance to the corresponding image region, a problem we term source object grounding (Zhang et al., 2024). A major obstacle to effective grounding is the noisy latent's influence on determining which regions to modify (Wang et al., 2024). This latent guides the transformation process, encoding whether the diffusion model should prioritize consistency with the input image or adherence to the textual instruction.

Existing approaches typically address this by either (i) inpainting, which constrains edits to user-provided masks (Rombach et al., 2022), or (ii) noise-based techniques (Meng et al., 2022; Song et al., 2020), which encode the input image into a single noisy latent. As illustrated in Figure 2a, both strategies rely on a unified latent representation for the entire image, which fails to capture region-specific latent variation. Consequently, the diffusion model often applies unintended modifications across the image, such as altering background elements (*e.g.*, the left elephant in Figure 2a). These methods, however, often lack a precise mechanism to restrict changes exclusively to the source object. Inpainting, while avoiding edits outside the mask, frequently fails to meaningfully update the target region (Grechka et al., 2024).

*Corresponding author.

[1]Codes and data are available in https://github.com/ldilab/R-SIE.git.

Transactions of the Association for Computational Linguistics, vol. 13, pp. 1171–1185, 2025. https://doi.org/10.1162/tacl.a.34
Action Editor: Marco Baroni. Submission batch: 2/2025; Revision batch: 5/2025; Published 9/2025.

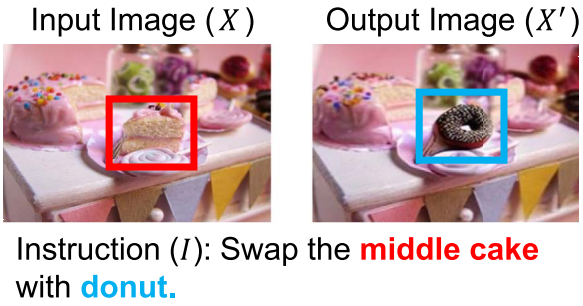


Figure 1: Example images edited by R-SIE. The source object (red) is successfully transformed into the target object (blue) without change in the background.

To overcome this, we propose a Region-wise Diffusion Process (RwDP) that explicitly reflects region-specific latent differences. Unlike prior methods, RwDP employs two distinct latents: (1) one representing the input image (background) and (2) another initialized with random noise for the source object, as shown in Figure 2b. By disentangling the latents per region, RwDP enables each part of the image to follow its respective guidance. During denoising, the background is reconstructed based on the original image latent, while the source object is generated using instruction-driven noise.

To support RwDP, we design an automatic data generation pipeline using bounding-box annotations as shown in Figure 3. This pipeline constructs triplets consisting of (1) an input image X, (2) a ground-truth (GT) output X', and (3) an instruction I. The resulting triplets are unambiguous: The instruction uniquely specifies the source object, and the output image reflects only the intended edit. Bounding boxes facilitate this process by (a) isolating the source object to preserve the background and (b) helping a language model generate instructions that describe the source object's relative location.

Finally, we address limitations in evaluating SIE, which considers entire images at once (Brooks et al., 2023; Gal et al., 2022; Ruiz et al., 2023). As such, these methods often overlook the specific regions where edits were intended. As a result, unintended background changes can misleadingly boost existing evaluation metrics. We address this by introducing two metrics: the CLIP-Region Class (CLIP-RC) score for the source object, and the CLIP-Global Context (CLIP-GC) score for the background. Each metric isolates the corresponding region. This penalizes unintended edits and yields a more precise measure of source object grounding.

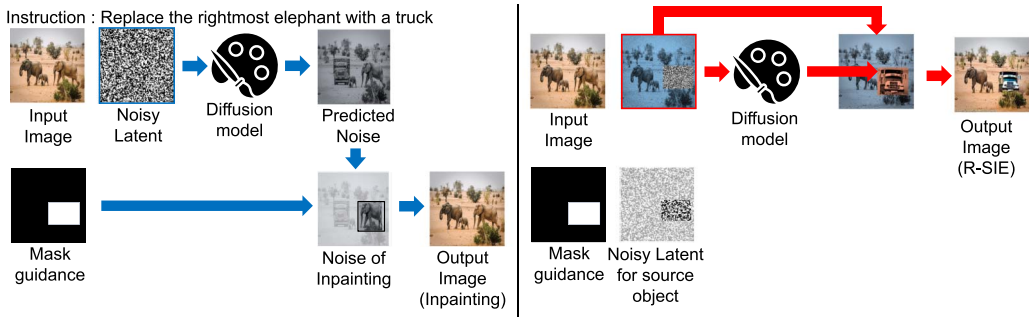
We propose R-SIE, a region-wise SIE framework that addresses training, inference, and evaluation. Our main contributions are threefold: 1) the Region-wise Diffusion Process (RwDP), which employs separate diffusion processes to condition each region on its respective guidance; 2) an automated data generation pipeline that crops and modifies the source object, specifically designed to train and reinforce the RwDP; and 3) two region-based evaluation metrics, CLIP-RC and CLIP-GC, which offer a more accurate assessment of source object grounding.

2 Related Work

2.1 Controlling Diffusion for SIE

A central challenge in SIE is modifying the source object while preserving the background. Previous studies have constrained diffusion models to preserve the input image context for SIE. A common strategy is inpainting, where mask guidance is given to restrict the modifications within the source object (Rombach et al., 2022). To preserve the background, early works copy-paste pixel values from the input image (Nichol et al., 2022) or constrain the diffusion process with the gradient of a CLIP score (Andonian et al., 2021; Avrahami et al., 2022) or repeated noise-denoise cycles (Lugmayr et al., 2022). These inpainting methods rely critically on the mask and can fail to apply edits if the mask or noise initialization is not well aligned (Grechka et al., 2024). Other methods focus on the noisy latents. For instance, SDEdit (Meng et al., 2022) adds Gaussian noise to the input image, while DDIM inversion (Song et al., 2020) approximately reverses the backward diffusion process of the input image. Then, DDIM inversion is paired with the inpainting (Couairon et al., 2022) or integrated with textual information (Mokady et al., 2023). A complementary line of work bypasses the source object grounding by introducing a post-hoc filtering scheme. When the SIE model is uncertain, it generates multiple candidate outputs and lets the user choose the preferred one (Shinagawa et al., 2020).

In line with other noise-based approaches, we aim to find a noisy latent that promotes a more effective SIE. Our key distinction is that 1) we strictly reconstruct the background by avoiding the inherent randomness of DDIM, and 2) we condition the source object to textual guidance, thereby promoting modification of the source objects.



(a) Inpainting model. (b) RwDP (Ours).

Figure 2: Comparison of inpainting models and the RwDP (Ours).

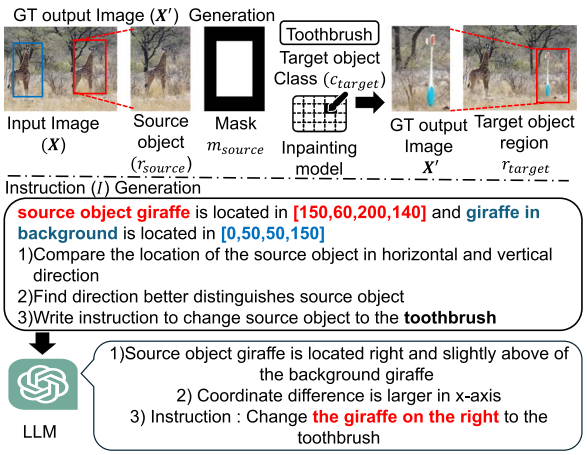


Figure 3: Training data generation process of R-SIE.

2.2 Training Data Generation for SIE

A recurring challenge in SIE is creating training data that pinpoints the source object while preserving the background (Hertz et al., 2023; Bar-Tal et al., 2022; Meng et al., 2022). Early works describing only the final image (e.g., ‘‘two elephants and a truck’’) introduce ambiguity about the source object (Hertz et al., 2023; Meng et al., 2022), while more recent methods adopt explicit instructions, e.g., ‘‘replace the elephant with a truck’’ (Brooks et al., 2023; Zhang et al., 2024; Liu et al., 2024). Building instruction-tuning data often involves automatic pipelines–combining large language models (LLMs) and text-to-image generation (Brooks et al., 2023). These pipelines risk mismatches between the instructions and the GT output images when attempting to scale up data generation (Zhang et al., 2024), whereas manual annotation is precise yet limited (Wang et al., 2023b). The semi-automatic approaches using inpainting (Zhang et al., 2024; Liu et al., 2024) have emerged, partially reducing manual effort yet still relying on carefully prepared guidance or masks.

In line with semi-automatic approaches, we leverage bounding-box annotations to build large-scale triplets of (X, I, X') where bounding boxes explicitly indicate the source objects. To generate GT output Image X', we separate the background from the modification and edit only the source object. In the instruction, we specify the relative position, *e.g.*, ‘‘the rightmost elephant’’, ensuring clarity in complex scenes.

3 Preliminary

3.1 Problem Definition

SIE aims to replace a specific object in an image with another object, guided by textual instruction, while preserving the rest of the scene. Formally, given an input image X containing n objects $O = \{o_i\}_{i=1}^{n}$, each with a bounding box b_i and category c_i, the objective is to produce an edited image X' where:

- A source object o_{source} (with bounding box b_{source} and category c_{source}) is replaced by a target object o_{target}.
- The location remains the same, *i.e.*, $b_{\text{source}} = b_{\text{target}}$.
- The category changes, *i.e.*, $c_{\text{target}} \neq c_{\text{source}}$.
- The other parts of the image, *e.g.*, all other objects in O, remain unchanged.

An instruction I specifies which object to modify and how, *e.g.*, ‘‘Replace the rightmost elephant with the truck.’’ The main challenge is to ensure that only the specified object is transformed

(the source object) and that the rest of the scene remains consistent.

3.2 Diffusion Model for SIE

We now formally define how the diffusion models perform SIE and how randomness is involved in the diffusion process. Briefly, during the training, the GT output image X' is corrupted with noise via the forward diffusion process. The resulting noisy latent is progressively denoised using the diffusion model $\tilde{\epsilon}(t, z_t, I, X)$ to generate the GT output image. The diffusion models are trained to estimate the noise introduced during the forward diffusion process. In inference, the noisy latent is sampled differently to compensate for the absence of the GT output image.

Forward Diffusion (Noise Addition) The goal of the forward diffusion process is to progressively corrupt the GT output image X'. Specifically, the forward diffusion process begins with noisy latent for the 0-th step, z_0, which is the latent representation of the GT output image obtained by using an encoder $\mathcal{E}$ of a variational auto-encoder (VAE):

$$z_0 = \mathcal{E}(X'). \tag{1}$$

Gaussian noise $\epsilon_t \sim \mathcal{N}(0, 1)$ is then added at each time-step t ($1 \leq t \leq T$, where T denotes the final diffusion step) by the forward scheduler γ_f:

$$z_{t+1} = \gamma_f(t, z_t, \epsilon_t), \tag{2}$$

so that z_T approximates the pure Gaussian random noise.

Backward Diffusion (Denoising) The goal of the backward diffusion process is to reconstruct the z_0 from z_T using the prediction of the diffusion model $\tilde{\epsilon}(t, z_t, I, X)$. For such goal, given the time step t and its corresponding noisy latent z_t, the backward diffusion scheduler γ_b uses the noise $\tilde{\epsilon}(t, z_t, I, X)$ predicted by the diffusion model to compute a less noisy latent z_{t-1}:

$$z_{t-1} = \gamma_b\big(t, z_t, \tilde{\epsilon}(t, z_t, I, X)\big). \tag{3}$$

Because the forward and backward schedulers use different time step schedules (Guo et al., 2025), re-using forward noise in the backward pass can distort the reconstruction, which harms the preservation of the background.

Training Objective During training, the diffusion models are optimized to predict the noise added in the t-th step of the forward diffusion process given the noisy latent z_t as follows:

$$\mathcal{L} = \mathbb{E}_{\substack{\mathcal{E}(X'), \mathcal{E}(X), T, \\ \epsilon \sim \mathcal{N}(0,1), t}} \Big[\|\epsilon - \tilde{\epsilon}(t, z_t, I, X)\|_2^2 \Big]. \tag{4}$$

As the diffusion models are given latent representation z_t during the training, they are trained to be sensitive to the initial noisy latent z_T (Wang et al., 2024).

Inference for Editing At test time, X' is unknown. Hence, the conventional diffusion process uses the Gaussian random noise as the initial noisy latent $z_T \sim \mathcal{N}(0, 1)$ and applies the backward diffusion repeatedly to generate a latent close to z_0. The VAE decoder $\mathcal{D}$ then produces the final edited image X'. While sensitive to the initial noisy latent z_T, randomly sampling it frequently results in mismatches between the regions and the guidance.

4 Proposed: Regional SIE Framework (R-SIE)

We propose R-SIE, which applies a region-wise diffusion process (RwDP) at inference—confining each region to the correct guidance—and a region-wise data generation pipeline that reinforces a diffusion model specifically for this region-wise approach. This section outlines our framework, detailing how it improves source object grounding.

4.1 Region-wise Diffusion Process (RwDP)

We now describe the RwDP, illustrated in Figure 2b, whose goal is to condition each region on its correct guidance. Unlike conventional SIE methods, which rely on a single diffusion process, RwDP conducts two distinct diffusion processes for each region. These two processes restrict the background to the input image and the source object to the instruction. Below, we detail how we implement forward and backward region-wise diffusion for each region.

Forward Diffusion Process Because diffusion models are sensitive to the initial noisy latent z_T, the RwDP for the forward diffusion process aims to sample an initial z_T such that the background is conditioned to the input image X and the source object to the instruction I. In contrast to the previous works, which sample a single

noisy latent z_T for the entire image, the RwDP constructs two latents: i) a reconstruction latent z_{reconst} that encodes the input image, and ii) a generation latent z_{gen} sampled independently from X, thereby letting the diffusion model condition on I, as depicted in Figure 2b.

To encode the input image X, we apply the conventional forward diffusion process to the reconstruction latent z_{reconst}. Specifically, we encode X into a latent $z_{\text{reconst},0} = \mathcal{E}(X)$, then progressively add noise $\epsilon_{\text{reconst}} \sim \mathcal{N}(0,1)$ (Rombach et al., 2022):

$$z_{\text{reconst},\,t+1} = \gamma_f(t,\, z_{\text{reconst},\,t},\, \epsilon_{\text{reconst}}), \quad (5)$$

from time $t = 0$ to $t = T$. Thus, $z_{\text{reconst},T}$ remains close to the input image, capturing its global appearance and thereby constraining the background to X during backward diffusion process.

The goal of the generation latent $z_{\text{gen},T}$ is to ensure the diffusion model conditions on I. To this end, we sample $z_{\text{gen},T} \sim \mathcal{N}(0,1)$ independently of X, making it free from any encoding of X. Because $z_{\text{gen},T}$ is sampled independently of X, the diffusion model can deviate from the input image and follow instruction I.

Finally, we align each region with the correct guidance by merging $z_{\text{reconst},T}$ and $z_{\text{gen},T}$ according to a binary mask m_{source}, where $m_{\text{source}} = 1$ for pixels belonging to the source object and $m_{\text{source}} = 0$ for background:

$$z_T \;=\; (1 - m_{\text{source}}) \cdot z_{\text{reconst},T} \;+\; m_{\text{source}} \cdot z_{\text{gen},T}. \quad (6)$$

This ensures that the background is guided by the input image while the source object is guided by the instruction, forming the initial noisy latent z_T for the entire image.

Backward Diffusion Process In the backward diffusion process, the RwDP denoises z_T by estimating noise $\tilde{\epsilon}$ for each region's correct guidance. As in the forward phase, we define two types of noise: i) the reconstruction noise $\tilde{\epsilon}_{\text{reconst}}$ that reconstructs X for the reconstruction latent, and ii) generation noise $\tilde{\epsilon}_{\text{gen}}$ that enforces the instruction I.

To condition the background on X, we reuse the noise $\epsilon_{\text{reconst}}$ from forward diffusion instead of using the model's prediction. Because $\epsilon_{\text{reconst}}$ is derived solely from X and independent of I, its use during the backward diffusion process aims to faithfully reconstruct the input image.

However, forward and backward processes involve different time schedules, causing distortions if we directly reuse $\epsilon_{\text{reconst}}$. To remedy this, we introduce a single scale parameter α that rescales $\epsilon_{\text{reconst}}$ during backward diffusion. At inference time, α is optimized once for each test image so the backward diffusion process can accurately reconstruct X. To optimize α for each test image, a preliminary backward diffusion process is performed to reconstruct the input image X. This process utilizes the identical scheduler and number of denoising steps as the main image generation process, applying $\alpha \cdot \epsilon_{\text{reconst}}$ in Eq. 2. α is then optimized to minimize the L_2 distance between this preliminary reconstruction's latent representation $\hat{z}_0(\alpha)$ and the input image's original latent $z_{\text{reconst},0} = \mathcal{E}(X)$:

$$\alpha^* \;=\; \arg\min_{\alpha} \,\| \, z_{\text{reconst},0} \;-\; \hat{z}_0(\alpha) \|^2, \quad (7)$$

Once we find the optimal α^*, we proceed with the backward diffusion process for SIE using the reconstruction noise $\tilde{\epsilon}_{reconst} = \alpha^* \cdot \epsilon_{\text{reconst}}$. Note that we do not use α during training and optimize α at inference. As the diffusion network is not involved, the additional latency remains modest.

For the source object, we let the diffusion model predict generation noise $\tilde{\epsilon}_{\text{gen}}(t, z_t, I, X)$, which captures the user instruction. We compute this via classifier-free guidance (Brooks et al., 2023), drawing noise predictions under null or partial guidance, *e.g.*, $\tilde{\epsilon}(t, z_t, I, \emptyset)$, $\tilde{\epsilon}(t, z_t, \emptyset, X)$, or $\tilde{\epsilon}(t, z_t, \emptyset, \emptyset)$, and combining them for high-quality edits. At each step t, the RwDP estimates the final noise $\tilde{\epsilon}_{RwDP}$ by merging the reconstruction noise and the generation noise weighted by the mask m_{source}:

$$\tilde{\epsilon}_{RwDP}(t, z_t, I, X) = m_{source} \cdot \tilde{\epsilon}_{gen} \quad (8)$$

$$+ (1 - m_{source}) \cdot \tilde{\epsilon}_{\text{reconst}}. \quad (9)$$

Thus, each region's noise follows its proper guidance. This arrangement discards any unintended background modifications by applying $\tilde{\epsilon}_{\text{reconst}}$ where $m_{\text{source}} = 0$. Conversely, $\tilde{\epsilon}_{\text{gen}}$ acts only on the source object. Finally, the backward scheduler γ_b subtracts $\tilde{\epsilon}(t, z_t, I, X)$ to produce z_{t-1}, iterating until the model yields an output image where the background reconstructs X and the source object is edited per I.

4.2 Region-wise Train Data Generation

We propose a region-wise train data generation procedure, illustrated in Figure 3, whose goal is to train a diffusion model to reliably estimate the generation noise $\tilde{\epsilon}_{gen}$ for the source object under complex visual contexts. Specifically, given an input image X, we aim to automatically generate i) a GT output image X' in which only the source object o_{source} is changed, and ii) an instruction I that unambiguously identifies the source object. To achieve this, we use bounding-box annotations to restrict modifications to the source object and to include its relative position in the instruction I, *e.g.*, ''the leftmost giraffe.''

Ground Truth Output Images (X') Given an input image X, the GT output image X' should resemble the background of X, while replacing only the source object o_{source} with a target object o_{target}. Following prior work (Zhang et al., 2024), we adopt an inpainting model to ensure the background remains unaltered. However, directly supplying the entire complex scene to the inpainting model often fails to ground the source object, leading to either unchanged source objects or requiring exhaustive annotations. This hinders the diffusion model's ability to condition the source object on the instruction.

To address this, region-wise train data generation simplifies and isolates the input image's context, making it easier for the inpainting model to focus on editing the source object. Specifically, we crop the source object region r_{source} with a small margin around its bounding box b_{source}, as shown in Figure 3. We then feed r_{source}, along with a mask m_{source} and a textual prompt specifying the target object category c_{target}, into an inpainting model Inpaint. Formally:

$$r_{\text{target}} = \text{Inpaint}(r_{\text{source}}, m_{\text{source}}, c_{\text{target}}). \quad (10)$$

As r_{source} provides a simplified context, the inpainting model is more inclined to replace o_{source} with o_{target} without extensive annotation effort. Additionally, the small margin around the bounding box b_{source} ensures that the generated target object region r_{target} remains coherent with the background.

Next, we verify that the inpainting model has produced the intended target object within the specified region. We run a CLIP zero-shot classifier on the generated target object region r_{target}, checking if the predicted class is the target object class c_{target}. If this check succeeds, we merge the generated target object region r_{target} back into the input image X, thereby replacing the source object region r_{source} and yielding the GT output image X'. This process ensures that only the source object is modified, while the background remains unchanged.

Instruction I with Relative Position Given X and the GT output X', the instruction I should uniquely identify the source object o_{source}. Previous methods typically refer to the source object as its object class, *e.g.*, ''Change the giraffe,'' which becomes ambiguous in complex scenes with multiple objects of the same category. This ambiguity compounds the stochasticity of diffusion models.

To mitigate this, we incorporate the source object's relative position into I, using bounding-box annotations to pinpoint the correct object. Concretely, we feed the bounding boxes b_{source} and others into an LLM. It compares b_{source} along both horizontal and vertical directions to determine where it most deviates from other objects, then returns a phrase that captures this relative position, *e.g.*, ''the rightmost cat,'' or ''the top-left lamp.'' By specifying the source object's location in the instruction, we remove ambiguity and ensure the diffusion model can more accurately ground the source object, even in complex visual contexts.

4.3 Evaluation: CLIP-RC and CLIP-GC Metrics

Proper evaluation of SIE must separate intended edits on the source object from unintended changes in the background. Existing SIE metrics, such as CLIP- or DINO-based image similarity scores, compare entire images without separating the source object from the background (Brooks et al., 2023; Gal et al., 2022; Ruiz et al., 2023). As a result, they can yield misleading scores if unintended alterations occur outside the source object. To address this, we propose CLIP-RC and CLIP-GC, two metrics that separately measure modifications to the source object and preservation of the background. By isolating each region, any unintended edits are penalized accordingly.

CLIP-RC: Source Object Evaluation The CLIP-RC score assesses whether the source object has been correctly transformed into the target

object. Specifically, we feed only the target object region r_{target} to the CLIP model, thus excluding the background from the evaluation. To do this, we crop out r_{target} from the output image X' based on its bounding box b_{target}, then use a CLIP zero-shot classifier to measure the cosine similarity between this cropped region and the textual representation of the target category. As only r_{target} is considered, any unintended changes in the background do not affect CLIP-RC. Hence, only correct replacements of the source object increase the CLIP-RC score. A higher CLIP-RC indicates more accurate object replacement.

CLIP-GC: Background Evaluation The CLIP-GC score measures how well the background is preserved, ignoring any modifications to the source object. Concretely, we remove the target object region from both the output image and the input image, then feed only these background regions to the CLIP model. We extract CLIP embeddings from both backgrounds and compute their cosine similarity to produce the CLIP-GC score. Because CLIP-GC focuses solely on the background, any unintended edits there will reduce its similarity and penalize the metric. Consequently, a higher CLIP-GC indicates the background remains faithful to the original image.

5 Experiments

We evaluate the R-SIE framework against multiple baselines for SIE. Below, we detail the baselines, our dataset construction and validation, and our implementation details.

5.1 Baselines

We compare R-SIE with three representative approaches. InstructEdit (Wang et al., 2023a) is a pipeline that combines a vision-language model, an LLM, and a segmentation model to enhance DiffEdit (Couairon et al., 2022), an inpainting model paired with DDIM. By leveraging a segmentation model, InstructEdit improves the source object grounding more precisely than DiffEdit. IP2P constructs a large-scale dataset by generating synthetic triplets with Prompt2Prompt (Hertz et al., 2023) and an LLM. Stable Diffusion v1.5 is then fine-tuned on these synthetic pairs, yielding an SIE model. MagicBrush (Zhang et al., 2024) finetunes IP2P with a smaller, semi-automatic SIE train dataset. Although it uses fewer training samples overall, its human-verified annotations aim to improve source object grounding and reduce unintended modifications.

5.2 Dataset Generation & Validation

Train Set Generation We build upon MS COCO bounding-box annotations (Lin et al., 2014) to create large-scale triplets for training in two scenarios: i) a simple setting with exactly one instance of the source object category, and ii) a complex setting with multiple instances of the source object category. We exclude bounding boxes smaller than 10% of the image's width or height or overlapping with other objects by more than 50%. After filtering, we obtain 25k valid candidates for the complex setting and 18k for the simple setting. Next, we use Stable Diffusion 2 Inpainting (Rombach et al., 2022) to generate GT output images for 'replace' operations. For 'delete' operations, we use LaMa (Suvorov et al., 2021) to generate GT output images where the source object. Following previous works (Bala et al., 2024; Lee et al., 2023), we formulate 'create' operations by reversing the input and GT output of the 'delete' operation. That is, the input image for the 'delete' operation becomes the GT output image for the 'create' operation and vice versa.

Textual instructions are generated via ChatGPT4 (Achiam et al., 2023). This process yields 67k training triplets for the complex scenarios and 47k for the simple scenarios, from which we reserve 2.5k and 2.3k, respectively, as the validation set.

Test Set Validation For the test set, we use MS COCO images not included in the training set. We apply the same inpainting-based pipeline to generate GT images and instructions, but then manually verify each sample to ensure: i) the source object is indeed transformed into the target object, ii) the background remains visually identical to the input image, and iii) the boundary between the new object and the original background is smooth and coherent. We also discard any samples whose generated instructions contain incorrect source object categories, operations, or relative positions. Only those passing all checks are further filtered to balance the source object category distribution, yielding 516 triplets in the complex setting and 400 in the simple setting.

5.3 Implementation Details

For a fair comparison with baselines, we follow IP2P in training R-SIE. Specifically, we fine-tune Stable Diffusion 1.5 on our training set for 10k gradient steps, using a batch size of 1,024 and AdamW at a 5e-5 learning rate. These hyperparameters align with IP2P's setup but are scaled for our larger dataset. During inference, we maintain the same hyperparameters for all baselines and R-SIE: we sample 100 denoising steps, set the text-guidance scale to 7.5, and keep the image-guidance scale at 1.2 for classifier-free guidance.

At test time, we additionally optimize α that rescales the reused reconstruction noise. We set $\alpha = 1$ initially, making $\tilde{\epsilon}_{\text{reconst}} = \epsilon_{\text{reconst}}$ at the start. The preliminary diffusion pass, performed to optimize α, uses 100 denoising steps, the same number employed in the main diffusion process to generate the output image. We then update α for 100 gradient steps. The optimization process engages only the scheduler, not the diffusion model. As a result, the added latency is modest and well within practical limits.

6 Results

We now present our experimental results, addressing the following research questions:

RQ1: Are the proposed evaluation metrics properly localized to only the intended change?

RQ2: Is the source object grounding in the complex scenes improved in the inference?

RQ3: Does the generated training data yield further improvements in the source object grounding?

RQ4: Does the source object grounding qualitatively improve SIE?

RQ5: Are the proposed methods generalizable to other datasets?

6.1 RQ1: Validity of CLIP-RC and CLIP-GC Score

To evaluate how effectively current metrics (CLIP and DINO-based image similarity) and CLIP-RC and CLIP-GC capture SIE quality, we present two output samples in Figure 4. Each sample has been labeled by human annotators as either a ''success'' or ''fail,'' and we show the corresponding metric

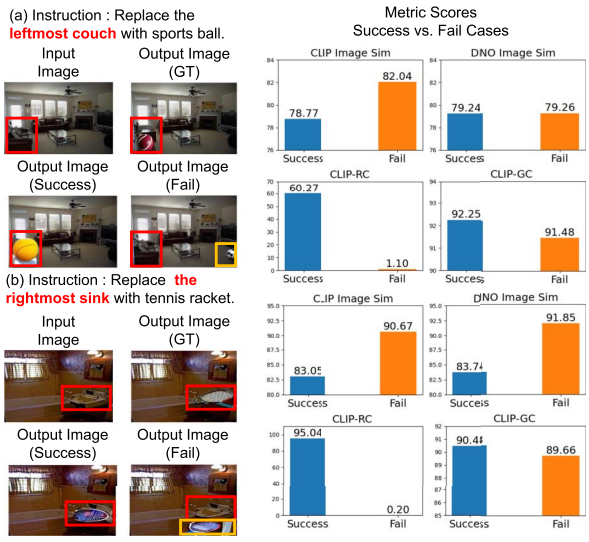


Figure 4: Example images evaluated by an expert human annotator as success or failure, alongside their corresponding metric scores (CLIP Image Sim, DINO Image Sim, CLIP-RC, CLIP-GC). We highlight the source object in red, while erroneous modifications are shown in orange.

scores alongside the images. In Figure 4(a), the success output accurately transforms the leftmost couch (the source object) into a sports ball while preserving the rest of the scene. By contrast, the fail output leaves the source object unchanged and instead edits a couch on the right, which is from the identical category but not the source object. Despite this clear error according to human judgment, CLIP Image Similarity assigns a higher score to the fail case (82.04) than to the success (78.77). DINO Image Similarity similarly favors the fail image (79.26) over the success image (79.24). These global similarity metrics overlook the localized mistake because they treat fine-grained edits as inconsequential. Meanwhile, CLIP-RC drops sharply from 60.27 (success) to 1.10 (fail) because the intended source object region was not correctly modified into the target object. Likewise, CLIP-GC penalizes the unintended background modification by giving the fail image a lower score (91.48) compared to 92.25 for the success image, echoing human evaluations. Figure 4(b) exhibits the same pattern: CLIP and DINO Image Similarity assign a higher score to the fail output than to the success, whereas CLIP-RC and CLIP-GC correctly recognize the better-edited image.

To confirm these qualitative observations quantitatively, we sampled 100 edited images from

Metric	Kendall's τ
CLIP Directional Similarity	0.23
CLIP-RC score	0.45

(a) Kendall's τ of CLIP directional similarity score and CLIP-RC score measuring modifying the source object.

Metric	Kendall's τ
CLIP Image Similarity	0.25
CLIP-GC Score	0.30

(b) Kendall's τ of CLIP Image similarity score and CLIP-GC score measuring preserving the background.

Table 1: Comparison of Kendall's τ for existing metrics and CLIP-RC and CLIP-GC score.

R-SIE, InstructEdit, and IP2P, then asked human annotators to label i) whether the source object was correctly modified, and ii) whether the background was preserved. Each image thus received a ''success'' or ''fail'' label under each criterion. We then computed Kendall's τ between these binary labels and each metric, as shown in Tables 1a and 1b. For the source object modification in Table 1a, CLIP-RC achieves a τ of 0.45 versus 0.23 for CLIP Directional Similarity, indicating closer alignment with human assessments of whether the source object was correctly replaced. For background preservation in Table 1b, CLIP-GC outperforms CLIP Image Similarity in correlating with human judgments. Overall, these results confirm that CLIP-RC and CLIP-GC provide more reliable, region-focused evaluations than existing metrics that assess the entire image without distinguishing the source object from the background.

6.2 RQ2: Effect of RwDP

We next investigate whether controlling the background and source object separately at inference alone improves source object grounding. We compare two baselines, IP2P and Magicbrush, in their conventional and RwDP. We also include InstructEdit, which leverages a user-provided mask, and provides it with GT mask guidance for fair comparison.

Table 2 illustrates that the CLIP-GC score consistently rises when IP2P or Magicbrush adopts the RwDP, suggesting better background preservation. For instance, the RwDP improves the CLIP-GC score in the replace operation of the complex setting for IP2P from 87.80 to 92.70, and Magicbrush from 90.90 to 92.58. Notably, this background preservation competes or even exceeds that of InstructEdit (+GT mask) across many subtasks.

However, the CLIP-RC score shows different trends. For IP2P, the RwDP raises the CLIP-GC score but hurts the CLIP-RC score. For example, IP2P's CLIP-RC score of the replace operation in the complex setting drops from 16.2 to 10.74. By contrast, Magicbrush shows improvements in the CLIP-RC score with the RwDP in most subtasks, *e.g.*, the replace operation in the complex setting from 6.9 to 8.73, suggesting the RwDP improves modifying the source objects. Overall, these findings confirm that the RwDP effectively preserves the background for both IP2P and Magicbrush, yet improvements in modifying the source object vary by model.

Furthermore, Table 2 shows that both DINO-I and CLIP-I scores also increase when IP2P or Magicbrush adopts the RwDP, suggesting more coherent image generation overall. For example, in the replace operation on the complex setting, IP2P's CLIP-I rises from 66.53 to 84.70 and DINO-I jumps from 43.05 to 80.04, whereas Magicbrush improves from 80.24 to 84.66 in CLIP-I and from 68.44 to 80.07 in DINO-I. These gains indicate that the RwDP generates an output image as coherent as the GT output image.

Finally, we quantify additional latency from introducing RwDP. The conventional diffusion process averages 3.56 s per image over 100 samples. Inference with RwDP takes 8.20s per image, which includes the search for the optimal α. Nevertheless, RwDP remains faster than DiffEdit (Couairon et al., 2022), the DDIM-inversion inpainting method used in InstructEdit, which averages 8.7 s.

6.3 RQ3: Effect of the Generated Train Data

We now examine whether the generated train data further boosts source object grounding beyond the RwDP alone. Table 2 compares R-SIE with IP2P and Magicbrush, both under their conventional and RwDP. When the conventional diffusion process is applied for inference (rows with ''-''), R-SIE already achieves higher CLIP-RC than either IP2P or Magicbrush in most tasks. For example, for the replace operation in the complex

Diffusion Model	Inference	Replace				Create				Delete			
		RC	GC	DINO-I	CLIP-I	RC	GC	DINO-I	CLIP-I	RC$_\downarrow$	GC	DINO-I	CLIP-I
InstructEdit	–	1.03	91.89	**83.27**	**87.45**	4.19	92.63	**84.22**	**89.97**	31.01	91.91	**88.52**	**90.94**
	+GT mask	1.96	91.97	83.23	**87.45**	7.96	92.63	83.8	89.57	24.39	91.99	87.72	90.61
IP2P	–	16.2	87.80	43.05	66.53	26.19	88.80	48.91	71.19	17.36	87.20	43.13	67.74
	+RwDP	10.74	**92.70**	80.04	84.70	11.04	**93.49**	80.04	86.97	30.05	**92.78**	84.55	87.51
Magicbrush	–	6.90	90.90	68.44	80.24	15.06	91.86	70.76	83.08	22.33	91.02	74.49	84.21
	+RwDP	8.73	92.58	80.07	84.66	16.39	93.44	80.01	87.05	24.00	92.07	84.52	87.49
R-SIE	–	19.12	88.73	54.12	72.46	**29.84**	93.03	78.32	86.43	17.40	91.31	73.88	83.73
	+RwDP	**26.46**	92.51	80.85	84.93	28.09	93.38	81.13	87.61	**15.94**	92.72	85.78	88.08

(a) Combined results in the complex settings.

Diffusion Model	Inference	Replace				Create				Delete			
		RC	GC	DINO-I	CLIP-I	RC	GC	DINO-I	CLIP-I	RC	GC	DINO-I	CLIP-I
InstructEdit	–	0.89	91.69	72.77	83.78	3.16	92.78	75.71	**87.39**	29.87	91.69	79.77	87.75
	+GT mask	2.35	91.67	74.04	**84.49**	5.16	92.42	74.60	86.83	19.51	91.71	79.71	**88.14**
IP2P	–	14.95	87.91	41.87	65.58	25.44	89.39	46.67	71.20	20.08	88.09	42.67	66.72
	+RwDP	10.37	**92.73**	72.19	81.72	15.24	93.59	72.60	84.82	26.94	**92.81**	79.69	85.76
Magicbrush	–	7.76	91.42	65.17	79.22	13.58	91.91	63.41	81.57	21.34	91.42	73.27	83.31
	+RwDP	7.88	92.71	72.90	82.21	14.66	93.42	72.47	84.72	19.20	92.76	80.33	85.93
R-SIE	–	36.76	90.80	65.84	80.33	32.42	93.18	74.14	85.43	6.69	91.64	79.06	85.56
	+RwDP	**40.15**	92.42	**77.47**	84.11	**33.53**	**94.42**	**77.54**	86.83	**5.91**	92.80	**84.60**	87.74

(b) Combined results in the simple settings.

Table 2: Combined results for the complex and simple tasks across different operations. RC: CLIP-RC score, GC: CLIP-GC score, DINO-I: DINO model-based image similarity score, CLIP-I: CLIP model-based image similarity score. Higher is better for all metrics; for those marked with $_\downarrow$, lower is better. **Bold**: best overall value, Underlined: second best value.

setting, R-SIE attains the CLIP-RC score of 19.12, substantially above IP2P's 16.20 or Magicbrush's 6.90, reflecting more accurate modification.

However, R-SIE's background preservation (CLIP-GC) is only competitive with Magicbrush, not dramatically higher. When the inference is conducted with the RwDP (rows ''+RwDP''), R-SIE gains a larger improvement in both the CLIP-RC and -GC score compared to the smaller improvements seen by IP2P or Magicbrush. For instance, the CLIP-RC score of R-SIE on the replace operation in the complex setting rises from 19.12 to 26.46, compared to that of Magicbrush from 6.90 to 8.73. Similarly, the CLIP-GC score also improves from 88.73 to 92.51, showing the similar background preservation of IP2P and Magicbrush with the RwDP. Thus, the generated training data further reinforces the RwDP.

Moreover, the DINO-based similarity (DINO-I) in Table 2 indicates that R-SIE often achieves comparable or even higher coherence scores than the baselines. Notably, in the simple setting for the create operation, R-SIE attains a DINO-I of 77.54, exceeding InstructEdit's 75.71. This gap suggests that, beyond accurately modifying the source object, our approach also integrates it coherently with the GT output image. Such coherence emerges even before applying RwDP (rows ''-''), sometimes surpassing IP2P or MagicBrush, and becomes more pronounced when combining R-SIE's training data with the RwDP. Thus, R-SIE not only strengthens source object grounding but also ensures that the edited regions blend smoothly into the surrounding scene.

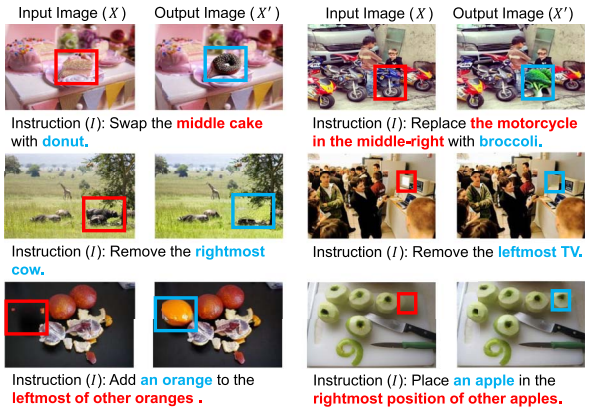


Figure 5: Example images edited by R-SIE. Each row shows a different operation–Replace (top), Delete (middle), and Add (bottom). The source object (red) is successfully transformed into the target object (blue) without change in the background.

6.4 RQ4: Qualitative Study

We first present qualitative results for R-SIE on six images with complex visual contexts in Figure 5.

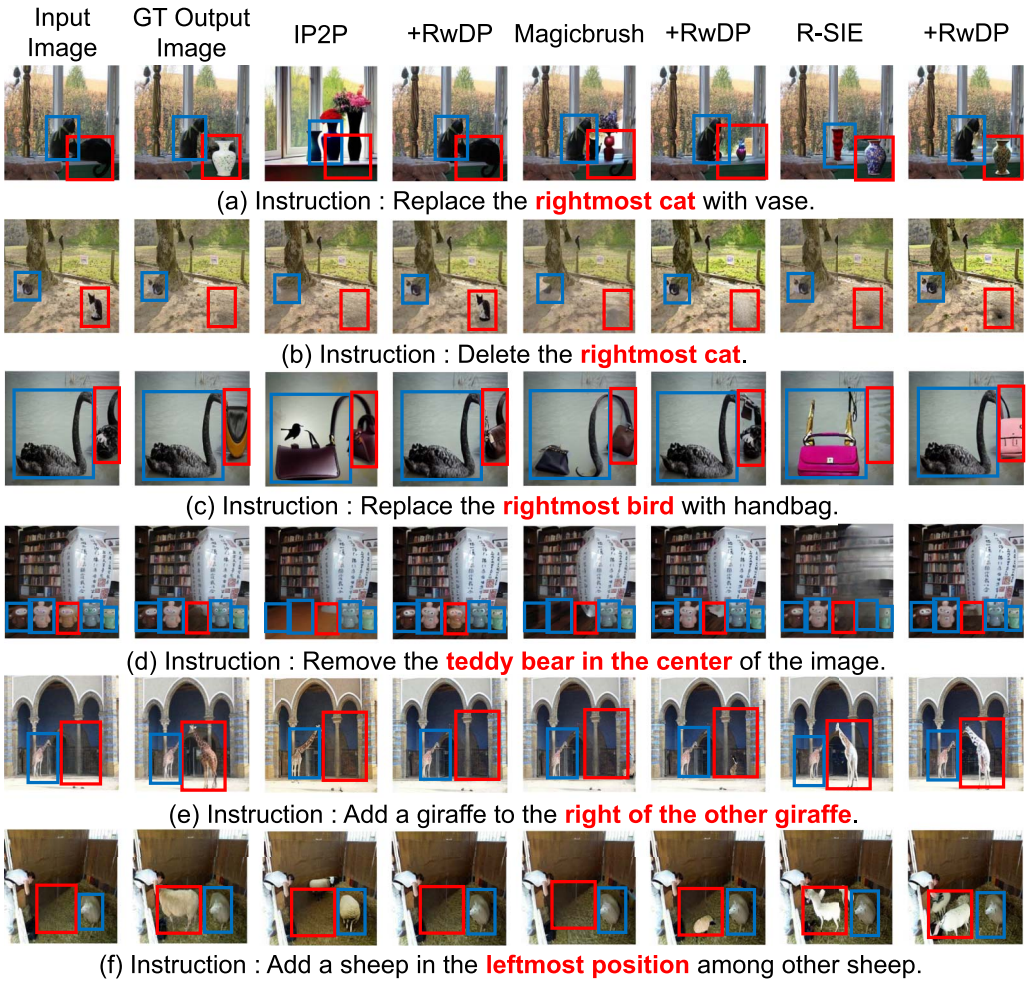


Figure 6: Qualitative comparison of R-SIE with the baselines on our test set. Source objects are in the red boxes and the other objects of identical category are in the blue boxes. +RwDP means that the region-wise diffusion process is used for inference.

Rows cover the three edit operations—Replace, Delete, and Add—and the examples vary the relative positions (middle, middle-right, leftmost, rightmost). Despite the complex visual contexts, R-SIE edits only the boxed source object. It successfully leaves the background unchanged. This behavior holds across diverse object pairs, *e.g.*, cake to donut and motorcycle to broccoli. These examples show that R-SIE grounds the source object even in complex scenes.

We qualitatively compare R-SIE with baselines IP2P and Magicbrush in Figure 6. Each sample demonstrates a different aspect of SIE: preserving the background, accurately modifying the source object, and handling complex instructions. In sample (a), both IP2P and Magicbrush disrupt the background: IP2P brightens the entire image, while Magicbrush distorts the cat into an oversized vase. Similarly, in sample (b), both methods remove every cat in the input image instead of editing only the intended one. In samples (c) and (d), IP2P and Magicbrush alter every bird or multiple teddy bears, respectively. These results reveal difficulty in source object grounding under a complex visual context. When enhanced by the RwDP, IP2P or Magicbrush can more reliably edit the source object while preserving the background.

Finally, samples (e) and (f) illustrate another failure mode. Even with RwDP, IP2P or Magicbrush sometimes fail to apply noticeable edits despite successfully preserving the background. For instance, IP2P makes no change at all, while Magicbrush adds the target object much smaller than the source object region, making it barely recognizable. By contrast, R-SIE accurately applies the modification with a size similar to the source object and keeps the background untouched.

Model	CLIP-I	DINO	L1$_{\downarrow}$	L2$_{\downarrow}$
HIVE	0.8519	0.7500	0.1092	0.0341
IP2P	0.8524	0.7428	0.1122	0.0371
R-SIE	0.8705	0.7691	0.1051	0.0420
+RwDP	**0.9192**	**0.8807**	**0.0678**	**0.0197**

Table 3: Results on Magicbrush test sets. +RwDP means that the region-wise diffusion process is used for inference. Higher values are better for all metrics except those marked with $_{\downarrow}$, where lower values are better.

We attribute this superior performance to the synergy between our RwDP and the generated training data, which further refines source object grounding and localizes edits exclusively to the intended region.

6.5 RQ5: Generalization of R-SIE

To evaluate how well our approach generalizes across datasets, we test R-SIE on the MagicBrush test set in a zero-shot manner. We compare it against two other models–HIVE and IP2P–likewise evaluated zero-shot. As Table 3 shows, we report CLIP-I and DINO scores, as well as L1 and L2 distances. Note that HIVE and IP2P can be finetuned on MagicBrush data to achieve an even higher score, *e.g.*, CLIP-I score with 0.9332, but we exclude those finetuned models from our main discussion here since we aim to show the generalization of R-SIE.

From Table 3, we see that R-SIE surpasses or matches HIVE and IP2P on key metrics. For instance, R-SIE zero-shot yields a CLIP-I of 0.8705, above IP2P's 0.8524 or HIVE's 0.8519. Similar trends are observed in other metrics. These findings indicate that R-SIE shows robust performance even in a zero-shot scenario, competing well against existing methods. When we conduct inference with the RwDP, performance further improves. CLIP-I rises to 0.9192 from 0.8705, and L2 drops to 0.0197 from 0.0420, reflecting enhanced background preservation and source object transformation.

These results demonstrate that R-SIE generalizes effectively to the MagicBrush test set without any additional finetuning, outperforming the other zero-shot baselines on multiple metrics. Moreover, inference with the RwDP yields further gains in both CLIP-I and distance measures, confirming the effectiveness of the RwDP outside the dataset on which it was originally tested.

7 Conclusion

We have presented R-SIE to address the issue of precise source object grounding in semantic image editing. By tailoring textual instructions with bounding-box–based positions, our method clarifies the otherwise ambiguous editing target in images with multiple or visually similar objects. During inference, the region-wise diffusion process distinctly handles noise in the foreground and background, enabling accurate object replacement without disrupting the rest of the scene. Empirical results confirm that R-SIE not only outperforms inpainting-based and instruction-tuned models but also maintains high fidelity for both the modified object and the surrounding context. Lastly, our proposed CLIP-RC and CLIP-GC metrics demonstrate improved alignment with human judgments, highlighting the importance of carefully isolating intended edits from the overall scene. Future work may explore broader classes of modifications and more nuanced instructions, further extending the adaptability of region-based SIE methodologies.

Acknowledgments

This research was partially supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2025-2020-0-01789) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation). This work was partly supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (no. RS-2021-II212068, Artificial Intelligence Innovation Hub and no. RS-2021-II211343), Artificial Intelligence Graduate School Program (Seoul National University) and Korea Evaluation Institute of Industrial Technology (KEIT) grant funded by the Korea government (MOTIE). This work was supported by Electronics and Telecommunications Research Institute (ETRI) grant funded by ICT R&D program of MSIT/IITP (2022-0-00995, Automated reliable source code generation from natural language descriptions).

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Alex Andonian, Sabrina Osmany, Audrey Cui, YeonHwan Park, Ali Jahanian, Antonio Torralba, and David Bau. 2021. Paint by word. *arXiv preprint arXiv:2103.10951*.

Omri Avrahami, Dani Lischinski, and Ohad Fried. 2022. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18208–18218. https://doi.org/10.1109/CVPR52688.2022.01767

Aniruddha Bala, Rohan Jaiswal, Loay Rashid, and Siddharth Roheda. 2024. Galaxyedit: Large-scale image editing dataset with enhanced diffusion adapter. *arXiv preprint arXiv:2411.13794*.

Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. 2022. Text2live: Text-driven layered image and video editing. In *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XV*, volume 13675 of *Lecture Notes in Computer Science*, pages 707–723. Springer. https://doi.org/10.1007/978-3-031-19784-0_41

Tim Brooks, Aleksander Holynski, and Alexei A. Efros. 2023. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402. https://doi.org/10.1109/CVPR52729.2023.01764

Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. 2022. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*.

Google DeepMind. 2025. Gemini 2.0 model updates: 2.0 flash, flash-lite, pro experimental. https://blog.google/technology/google-deepmind/gemini-model-updates-february-2025/. Accessed: 2025-03-24.

Rinon Gal, Or Patashnik, Haggai Maron, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. 2022. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13. https://doi.org/10.1145/3528223.3530164

Asya Grechka, Guillaume Couairon, and Matthieu Cord. 2024. Gradpaint: Gradient-guided in painting with diffusion models. *Computer Vision and Image Understanding*, 240:103928. https://doi.org/10.1016/j.cviu.2024.103928

Zhehao Guo, Jiedong Lang, Shuyu Huang, Yunfei Gao, and Xintong Ding. 2025. A comprehensive review on noise control of diffusion model. *arXiv preprint arXiv:2502.04669*.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. 2023. Prompt-to-prompt image editing with cross-attention control. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1–5, 2023*. OpenReview.net.

Yi Huang, Jiancheng Huang, Yifan Liu, Mingfu Yan, Jiaxi Lv, Jianzhuang Liu, Wei Xiong, He Zhang, Liangliang Cao, and Shifeng Chen. 2025. Diffusion model-based image editing: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(6):4409–4437. https://doi.org/10.1109/TPAMI.2025.3541625, PubMed: 40031849

Youngwon Lee, Ayoung Lee, Yeonjoon Jung, and Seung-won Hwang. 2023. On consistency training for language-based image editing interface. In *Proceedings of the Second Workshop on Natural Language Interfaces*, pages 22–30. https://doi.org/10.18653/v1/2023.nlint-1.2

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part v 13*, pages 740–755. Springer. https://doi.org/10.1007/978-3-319-10602-1_48

Chang Liu, Xiangtai Li, and Henghui Ding. 2024. Referring image editing: Object-level image editing via referring expressions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13128–13138. https://doi.org/10.1109/CVPR52733.2024.01247

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. 2022. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471. https://doi.org/10.1109/CVPR52688.2022.01117

Yiwei Ma, Jiayi Ji, Ke Ye, Weihuang Lin, Zhibin Wang, Yonghan Zheng, Qiang Zhou, Xiaoshuai Sun, and Rongrong Ji. 2024. I2EBench: A comprehensive benchmark for instruction-based image editing. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. 2022. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. 2023. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6038–6047. https://doi.org/10.1109/CVPR52729.2023.00585

Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob Mcgrew, Ilya Sutskever, and Mark Chen. 2022. Glide: Towards photo-realistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning*, pages 16784–16804. PMLR.

Byong Mok Oh, Max Chen, Julie Dorsey, and Frédo Durand. 2001. Image-based modeling and photo editing. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, pages 433–442. https://doi.org/10.1145/383259.383310

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695. https://doi.org/10.1109/CVPR52688.2022.01042

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. 2023. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510. https://doi.org/10.1109/CVPR52729.2023.02155

Seitaro Shinagawa, Koichiro Yoshino, Seyed Hossein Alavi, Kallirroi Georgila, David Traum, Sakriani Sakti, and Satoshi Nakamura. 2020. An interactive image editing system using an uncertainty-based confirmation strategy. *IEEE Access*, 8:98471–98480. https://doi.org/10.1109/ACCESS.2020.2997012

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

Matheus Souza, Yidan Zheng, Kaizhang Kang, Yogeshwar Nath Mishra, Qiang Fu, and Wolfgang Heidrich. 2025. Latent space imaging.

Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. 2021. Resolution-robust large mask inpainting with fourier convolutions. *arXiv preprint arXiv:2109.07161*. https://doi.org/10.1109/WACV51458.2022.00323

Qian Wang, Biao Zhang, Michael Birsak, and Peter Wonka. 2023a. Instructedit: Improving automatic masks for diffusion-based image editing with user instructions. *arXiv preprint arXiv:2305.18047*.

Ruoyu Wang, Huayang Huang, Ye Zhu, Olga Russakovsky, and Yu Wu. 2024. The silent prompt: Initial noise as implicit guidance for

goal-driven image generation. *arXiv preprint arXiv:2412.05101*.

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. 2023b. Imagen editor and editbench: Advancing and evaluating text-guided image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18359–18369. https://doi.org/10.1109/CVPR52729.2023.01761

xAI. 2025. Grok 3 beta – the age of reasoning agents. https://x.ai/news/grok-3. Accessed: 2025-03-24.

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. 2024. Magicbrush: A manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems*, 36.

8 Appendix

8.1 Limitations

While SIE encompasses diverse capabilities like attribute editing or complex semantic manipulations, this work focuses specifically on the fundamental challenge of source object grounding. To rigorously evaluate source object modification and background preservation, we concentrate our experiments on three core operations: object replacement, deletion, and addition.

These three operations are highly prevalent and serve as fundamental benchmarks in recent SIE evaluations (Huang et al., 2025; Ma et al., 2024). Focusing on these widely used operations allows a clear assessment of R-SIE framework's primary contribution regarding source object grounding. Addressing other valuable SIE tasks, which often involve different challenges beyond the specific object manipulation targeted by R-SIE's region-wise diffusion process, remains outside the current scope and is left for future work.