Boost Text-Driven Image Editing with SAM and Residual Correction

Yuxiang Ying, Rui Xia, Longxin Wang, Xikai Ma
University of Pennsylvania
yingyx, xia7, longxin, maxikai@seas.upenn.edu

Abstract

This paper introduces a novel framework for text-driven image editing, addressing existing challenges in segmentation accuracy, content preservation, and fidelity during complex editing tasks. By integrating the Segment Anything Model (SAM) with ControlNet, we enhance segmentation accuracy and maintain consistency in unedited regions. Additionally, we use the "Editing Everything" framework, which combines SAM, CLIP, and Stable Diffusion to ensure precise segmentation, effective text alignment, and high-quality image generation. To further improve content fidelity, we integrate residual correction techniques into the IC-Light pipeline, which adaptively refines editing accuracy while preserving the structural and visual integrity of unedited areas. Experimental results on various benchmarks demonstrate significant improvements in LPIPS, SSIM, and CLIP metrics, highlighting the robustness and scalability of our approach across diverse editing tasks. This work provides a foundational contribution to advancing text-driven image editing, paving the way for real-time applications and multimodal editing solutions.

1. Introduction

Text-Driven image editing has long been a fundamental task in computer vision, enabling applications in fields such as content creation, graphic design, and visual storytelling. Traditional approaches often rely on manual tools or task-specific algorithms, which require substantial effort, expertise, and time to achieve high-quality results. Recently, deep learning-based generative models, particularly diffusion-based models, have demonstrated remarkable success in generating and manipulating realistic images. However, precise and efficient image editing, especially when conditioned on a target prompt, remains an open challenge.

The rise of diffusion models, such as Denoising Diffusion Probabilistic Models (DDPM) and Denoising Diffusion Implicit Models (DDIM) proposed by Song et al. [16], has significantly advanced the field of generative modeling. These models leverage a multistep denoising process to iteratively generate images from noise. DDIM further introduces a non-Markovian sampling process, which accelerates image synthesis while preserving high-quality outputs. Despite these improvements, existing methods often struggle with precise image editing tasks where fine-grained details of the original image must be preserved while aligning with new textual conditions.

To address these limitations, various methods have been proposed. Inversion-based methods, such as Direct Inversion [10] and Null Text Inversion [13], aim to map images to latent spaces for targeted edits. These approaches balance retention and editability, but are often less flexible for complex or global modifications. Instruction-driven methods, such as InstructPix2Pix [4] and InstructDiffusion [7], allow fine-grained edits guided by textual instructions, offering tailored transformations for specific tasks. Furthermore, latent diffusion-based models, such as Blended Latent Diffusion [2], introduce blending techniques allowing flexible and aesthetic image transformations. However, these methods frequently lack physical constraints, resulting in edits that may compromise structural consistency or realism.

Recent work has also explored control-based approaches for localized editing. For example, MasaCtrl [6] introduces self-attention mechanisms to provide precise spatial control, while Edit-Friendly P2P [9] focuses on facilitating localized and user-friendly manipulations. Meanwhile, methods like Pix2Pix Zero and Zero-Shot Image-to-Image Translation [14] aim to generalize editing tasks to unseen prompts or domains without additional fine-tuning. Despite their versatility, these methods often fail in ensuring coherence across global and local edits, particularly under complex lighting or semantic constraints.

The IC-Light framework [1] goes a step further by addressing the problem of light coordination in images. Based on the principle of light transmission coherence, IC-Light imposes constraints during training that modify only the illumination of the image while preserving intrinsic properties such as albedo and texture details. This physically based approach enhances the realism of editing and improves the adaptability to different lighting scenarios. However, IC-Light focuses primarily on illumination coordina-

tion and does not explicitly address the broader task of text-conditional image editing.

In this paper, we propose a novel framework for text-driven image editing that addresses the limitations of existing approaches through the following three key contributions. We introduce an approach that combines the Segment Anything Model (SAM) with a zero-order fine-tuning strategy to improve segmentation accuracy and the consistency of the unedited parts during editing. Furthermore, we develop a modular architecture called Editing Everything that integrates SAM, CLIP, and Stable Diffusion (SD) to ensure consistent segmentation results, correlation-based target selection, and high-quality generation. Additionally, we propose to integrate PnP Inversion with weighted residual refinement into the IC-Light pipeline, which enables adaptive control of editing fidelity and maintains the consistency of unedited regions.

Our approach not only improves the fidelity and accuracy of image editing, but also provides a more robust and scalable solution that can cope with a variety of complex editing tasks. With the ultilization of advanced segmentation and diffusion techniques, our method achieves satisfying performance on multiple benchmarks, demonstrating its great potential for research and practical applications.

2. Related Work

2.1. Evaluated Baseline

Denoising Diffusion Implicit Models (DDIM) by Song et al. [16] has introduced an efficient diffusion model that reduces the number of generation steps while maintaining high-quality outputs, laying the groundwork for more practical and real-time image generation applications.

Building on the efficiency of DDIM, Blended Diffusion by Avrahami et al. [3] has employed a diffusion-based framework specifically tailored for text-driven editing of natural images. This method integrates textual prompts with diffusion processes to achieve coherent and semantically accurate modifications to preserve the original image structure.

InstructPix2Pix has further enhanced the interaction between language and image editing. Brooks et al. [5] present a model that follows natural language instructions to perform diverse editing tasks. By learning the mapping from text to image edits, InstructPix2Pix enables more intuitive and flexible user-driven modifications and editing.

PnP Inversion by Ju et al. [10] has introduced a method to enhance diffusion-based editing with only three lines of code. This method enables researchers and developers to quickly apply and customize diffusion models for image editing through simplified implementation steps, which significantly simplifying the editing process of diffusion models and enhancing their flexibility and maneuverability.

The results of our evaluation of the four baselines are shown in the Table 1.

Method	Background Preservation		CLIP ↑
	LPIPS ↓	SSIM ↑	
DDIM + P2P	0.2084	0.7173	25.314
InstructPix2Pix	0.1581	0.7679	23.899
Direct + PnP	0.0433	0.8817	25.018
Blended LD	0.0358	0.8752	26.142

Table 1. Evaluation of the 4 baseline methods

2.2. Foundation Research

Our improvements are based on the following key studies that support our work both theoretically and methodologically:

Segment Anything by Kirillov et al. [11] introduces a versatile segmentation model capable of recognizing and segmenting objects in a variety of images. This work lays the foundation for accurate object manipulation in image editing tasks.

ControlNet: Adding Conditional Control to Text-to-Image Diffusion Models by Zhang et al. [20] present a neural network architecture designed to add spatial conditional control to large, pre-trained text-to-image diffusion models. ControlNet incrementally increases parameters through "zero-convolution" layers to ensure that no harmful noise is introduced during the fine-tuning process. The method can be combined with a variety of conditional controls (e.g., edge maps, depth maps, segmentation masks, etc.), which can significantly improve the controllability, generative quality, and combinability of the diffusion model, while reducing the training cost and making it suitable for real-world deployment.

Edit Everything: A Text-Guided Generative System for Image Editing [19] proposes a comprehensive system combining Segment Anything Model(SAM), CLIP [15] and Stable Diffusion (SD) that utilizes textual prompts to guide various image editing operations, demonstrating the effectiveness of text-driven methods in realizing diverse and complex modifications and editing.

Scaling In-the-Wild Training for Diffusion-Based Illumination Harmonization and Editing [1] explores techniques for image illumination harmonization and editing using diffusion models to ensure consistent light transmission and enhance the realism of edited images.

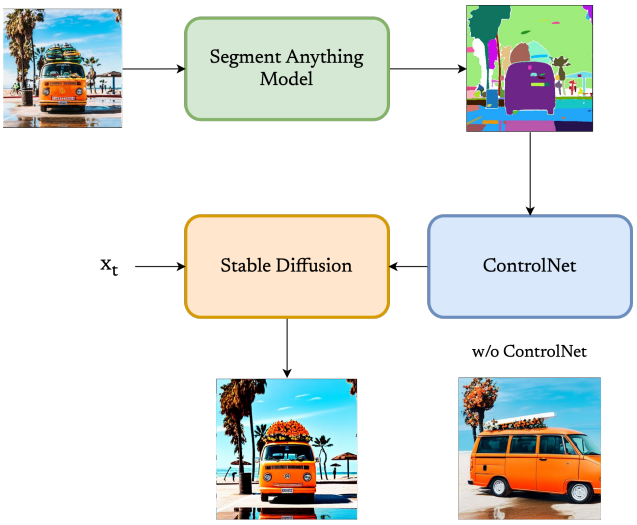


Figure 1. The overall architecture of the generation model with SAM and ControlNet

3. Method

3.1. Editing with Segment Anything Model

3.1.1. Segment Anything Model with ControlNet

Maintaining the consistency of unedited regions in text-driven editing tasks is crucial to ensure high-quality outcomes. To address this challenge, we propose a framework that integrates the Segment Anything Model (SAM) [11], ControlNet[20], and a text-driven editing framework. The process begins with SAM performing segmentation on the input image, generating multiple segment masks. Each mask is then assigned a random color, and the colored masks are combined into a single composite image. This composite image is subsequently fed into ControlNet, which guides the editing process while preserving the consistency of the unedited regions. Figure 1 shows the overall architecture of this framework. This approach ensures both precision in edits and fidelity to the original image structure.

3.1.2. Editing Everything Architecture

The text-guided generative framework, Editing Everything [19], consists of three primary components: Segment Anything Model (SAM), CLIP, and Stable Diffusion (SD). In Fig. 2, SAM is utilized to identify and segment all regions within an im- age, while CLIP evaluates these segments by aligning them with a provided source prompt [12]. The segment with the high- est alignment score is selected as the target for modification. Then Stable Diffusion leverages a target prompt to generate new object that replaces the chosen segment. This integrated approach offers a systematic and highly cus- tomizable solution for precise image editing.

3.2. Residual Correction on IC-Light

3.2.1. Method

In PnpInversion [10], the proposed technique involves adding the residual between the latent vector derived from the source prompt and the latent vector generated during the forward process. This residual correction is designed to enhance content preservation.

When a model performs poorly on unedited regions of the source prompt, we hypothesize that the latent direction may contribute to this issue. While the latent space is inherently non-linear, we explored whether adding a weighted correction to this ”incorrect velocity” could improve unedited content preservation performance, as illustrated in the left panel of Figure 3. Motivated by this idea, we proposed a source latent residual correction applied to the target latent, depicted in the right panel of Figure 3. Importantly, our method applies the correction only to the background mask in the latent space, with an MLP ensuring that the correction affects only the background in the pixel space.

To evaluate the impact of this weighted correction on the latent vectors in the target branch, we focused on scenarios involving significant alterations to the unedited portions of images. This approach allowed us to assess the method's potential for improving content preservation in challenging generative tasks.

3.3. Baseline Design and Setup

We designed a custom-designed model that combines the DDIM+P2P baseline with the advanced IC-Light[1] structure. This hybrid model, created specifically for this study, exhibited suboptimal LPIPS and SSIM performance, making it suitable for evaluating the impact of residual corrections. While the IC-Light model typically exhibits strong metric performance, including effective background conditioning, we opted to test our approach without the background conditioning for unbiased evaluation of content preservation.

The IC-Light structure in Fig. 9 employs a hooked U-Net as in Fig. 4 architecture built upon the SD1.5 backbone. It also adds a consistency loss shown in Eq. (4) on the vanilla loss in Eq. (1) to preserve the light transport consistency — the linear blending of an object's appearances under different illumination conditions is consistent with its appearance under mixed illumination. It will be further explained in this section. This setup fine-tunes an offset to improve editing capabilities.

Latent diffusion algorithms encode an input image I_L as a latent image representation $\epsilon(I_L)$, progressively adding noise to generate a noisy latent image $\epsilon(I_L)t$ at timestep t. These algorithms use the network δ to predict noise through

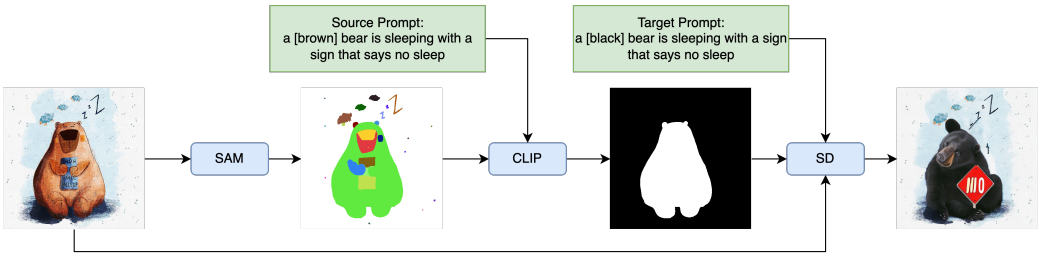


Figure 2. The Editing Everything framework operates through a systematic three-step process. First, the Segment Anything Model (SAM) divides the input image into multiple segments. These segments are then ranked according to a given source prompt, and the target segment is chosen based on the highest score calculated by our trained CLIP model. In the final step, Stable Diffusion (SD) generates new content to replace the chosen segment, guided by a target prompt. This process ensures seamless and efficient image editing, producing high-quality results.

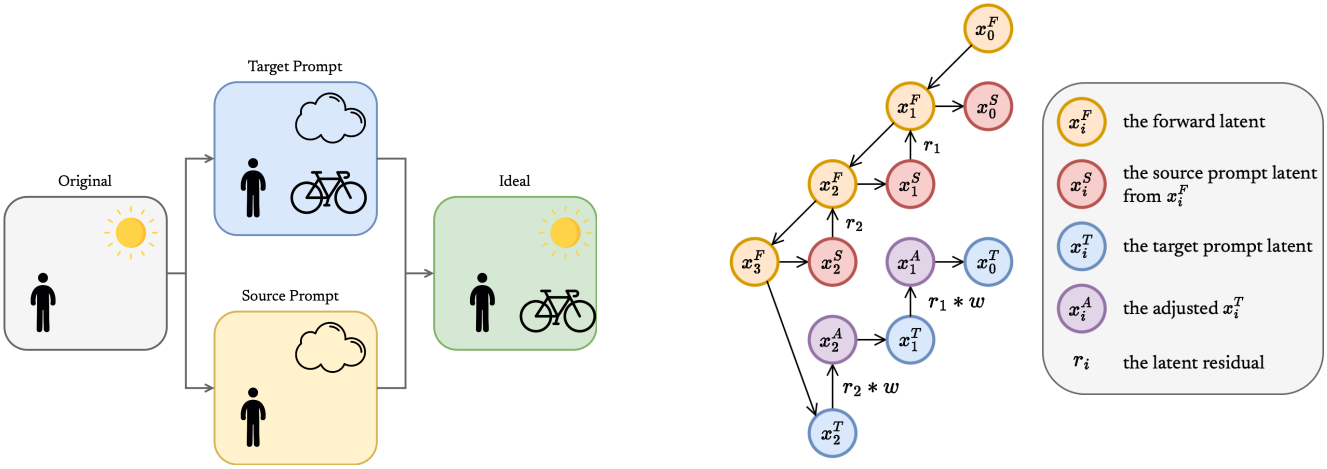


Figure 3. (Left) Motivation: Our method aims to correct errors introduced during DDIM denoising by addressing specific mistakes made for the source prompt, such as replacing a sun with a cloud. It seeks to correct these errors in the latent space while preserving the unedited parts of the image, resulting in better content similarity (refer to Tab. 4). (Right) Approach: At each time step t, the residual r_t between the forward latent x_t^F and the source latent x_t^S is calculated. A weighted correction $w \cdot r_t$ is applied to the target latent x_t^T, and the residual is added back to x_t^S to align it with x_t^F before the next step x_{t-1}^S. This process ensures better alignment and mitigates errors.

the cost function Eq. (1),

$$\mathcal{L}\text{vanilla} = |\epsilon - \delta(\epsilon(I_L)_t, t, L, \epsilon(I_d))|_2^2, \tag{1}$$

where ϵ represents the diffusion target (e.g., noise or v-target for prediction models), and $\epsilon(I_d)$ is derived from input degradation I_d. The objective trains the model to predict noise, facilitating image relighting tasks.

Light transport consistency, grounded in computational photography, ensures the intrinsic properties of light transport remain intact during modifications. As per Eq. (2), a matrix T relates latent appearance I_t^L and illumination condition L:

$$I_t^* = TL, \tag{2}$$

where $T \in \mathbb{R}^{(k\times3\times h\times w)\times(32\times32\times3)}$ maps illumination to appearance. By the principle of linearity in Eq. (3), merged illumination L_1+L_2 leads to combined appearance $\tilde{I}^*_{L_1+L_2}$:

$$\tilde{I}^*_{L_1+L_2} = T(L_1+L_2) = \tilde{I}^*_{L_1} + \tilde{I}^*_{L_2}. \tag{3}$$

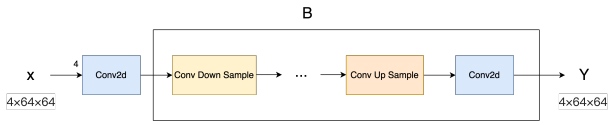
Using this principle, the IC-Light structure incorporates a consistency loss Eq. (4), defined for ϵ-prediction models to preserve light transport properties:

$$\mathcal{L}\text{cons} = |M \odot (\epsilon_{L_1+L_2} - \phi(\epsilon_{L_1}, \epsilon_{L_2}))|_2^2, \tag{4}$$

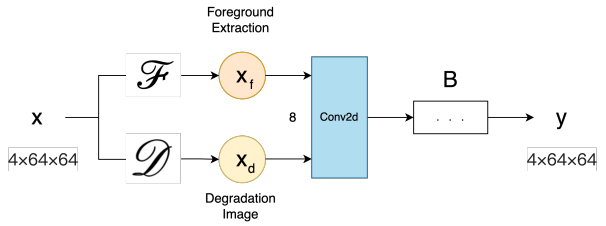
where $\phi(\cdot)$ is a 5-layer MLP with a hidden state of 128 channels, performing pixel-wise addition between ϵ_{L_1} and ϵ_{L_2}. M represents the merged illumination mask resized to match $L_1 + L_2$.

This consistency loss ensures linear merging of illumination, as demonstrated by Eq. (5):

$$\mathcal{L}\text{cons} = |M \odot (\epsilon_{L_1+L_2} - \delta_{L_1+L_2})|_2^2. \tag{5}$$



(a) StableBaseline 1.5 U-Net abstract structure.



(b) U-Net structure modified in the IC-Light model.

Figure 4. Comparison of U-Net architectures. (a) The abstract structure of StableBaseline 1.5 U-Net. (b) The modified U-Net structure used in the IC-Light model. In (b), a "degradation image" is an image that shares the same intrinsic albedo as the original image, but has completely altered illuminations.

The combination of Eq. (1) and Eq. (4) improves light transport consistency and enhances the model's ability to preserve visual realism in edited images, aligning with findings in [8]. This integration ensures that the light transport mechanism remains linear and first-order, making the IC-Light structure highly effective for tasks such as image relighting and high-fidelity image editing.

We integrated IC-Light into the DDIM+P2P baseline by inserting the image-processing modules (f, g) from IC-Light before passing the data into the Stable Diffusion Pipeline. Additionally, the standard U-Net within the pipeline was replaced with the modified IC-Light U-Net, and the learned offsets were applied to achieve refined output. Given that the high-resolution pipeline primarily enhances detail without significantly affecting content preservation, we excluded it to improve computational efficiency. To test our hypothesis, we experimented with residual correction applied to the IC-Light model variants that lack background conditioning. The goal was to assess whether the residual correction approach could enhance content preservation as measured by metrics such as LPIPS and SSIM while maintaining a robust CLIP score. Our findings suggest that targeted residual corrections can improve content preservation under these conditions.

4. Experiment

4.1. Evaluation Metrics

To illustrate the effectiveness and efficiency of our proposed methods, we use three metrics covering two aspects: background preservation (LPIPS [21] and SSIM [17] outside the annotated editing mask), edit prompt-image consistency (CLIPSIM[18] of the whole image and regions in the editing mask).

4.2. Experiment 1: SAM with ControlNet

4.2.1. Experiment Setup

For evaluation, we utilize the PIE-Bench dataset [10], which serves as a comprehensive benchmark for assessing image editing methods. The DDIM[16] algorithm is employed for inversion, while the Prompt-to-Prompt (P2P) method is used as the baseline editing framework. Our proposed framework builds upon this foundation by integrating the SAM and ControlNet. Specifically, we utilize SAM-1 to perform the segmentation task, and the ControlNet model is instantiated using a pre-trained image segmentation condition checkpoint. This configuration enables precise segmentation and controlled editing, ensuring robust performance in text-guided image manipulation tasks.

4.2.2. Experimental Result

We compare the editing framework with SAM and ControlNet with the framework without SAM and ControlNet. Table 2 shows the results of our framework.

Methods	LPIPS↓	SSIM↑	CLIP↑
w/o SAM and ControlNet	0.208	0.722	25.417
w/ SAM and ControlNet	**0.173**(16.8% ↓)	**0.717** (0.6% ↓)	**24.640** (3% ↓)

Table 2. **Performance of SAM and ControlNet.** This table presents a comparison of 2 different methods: the DDIM inversion and P2P editing framework with SAM&ControlNet and without them

Our evaluation indicates that the proposed framework effectively improves the consistency of the unedited regions in text-driven editing tasks. However, this improvement comes at the cost of reduced CLIP similarity between the generated image and the target prompt. This trade-off highlights a limitation in the current approach, suggesting room for further optimization.

4.3. Experiment 2: Editing Everything

4.3.1. Experiment Set up

For dataset, we focus on two subsets of PIE-Bench: Dataset 1 and Dataset 6. These subsets offer sufficient diversity and complexity to validate the effectiveness of our proposed approach while ensuring the feasibility of inference within our resource limits.

Our model includes CLIP-ViT-Large-Patch14, SAM, and Stable Diffusion 1.4, working together to achieve high-quality text-driven image editing. CLIP-ViT-Large-Patch14 aligns text prompts with visual content using a transformer-based text encoder and a Vision Transformer (ViT) image encoder. SAM (Segment Anything Model),

accurately segments the image into target regions using a vision transformer backbone. Finally, Stable Diffusion 1.4 generates realistic and context-aware outputs by integrating a Variational Autoencoder (VAE) for latent encoding, a U-Net for iterative denoising, and a text encoder to guide the generation process based on the input prompts.

We set the following key parameters to balance generation quality and computational efficiency: guidance scale = 7.5, which controls the adherence of the model to the input text prompts, ensuring that the generated content closely aligns with the target description; num inference step = 20, which determines the number of denoising steps, striking a balance between output quality and inference speed; and CLIP expand length = 12, which extends the length of the text embedding, enhancing the CLIP model's ability to capture and interpret complex semantic information.

4.3.2. Experimental Result

We choose some successful examples in Fig. 5, it demonstrates the effectiveness of the editing everything framework in generating visually coherent and contextually accurate results. Across various tasks, the system showcases precise segmentation of target regions, accurate alignment with text prompts, and high-quality content generation. For instance, objects such as animals and everyday items are seamlessly modified to match the target descriptions, while maintaining consistency in texture, lighting, and style with the surrounding context.

In evaluation, The DDIM algorithm is employed for inversion, while the Prompt-to-Prompt (P2P) method is used as the baseline editing framework. And we compare the editing everything architecture with method 1 and the baseline. Table 3 shows the results of our evaluation. The Editing Everything framework demonstrates significant improvements over the baseline and SAM and ControlNet methods. It achieves a reduction in LPIPS (from 0.208 to 0.106) and a noticeable increase in SSIM (from 0.722 to 0.839), indicating better perceptual quality and structural similarity.

Methods	LPIPS↓	SSIM↑	CLIP↑
Baseline	0.208	0.722	25.417
w/ SAM and ControlNet	0.173	0.717	24.640
w/ edit everything architecture	**0.106**	**0.839**	**25.244**

Table 3. **Performance of Editing Everything.** This table presents a comparison of 3 methods: the DDIM inversion and P2P editing framework with SAM&ControlNet, without them and our editing everything architecture.

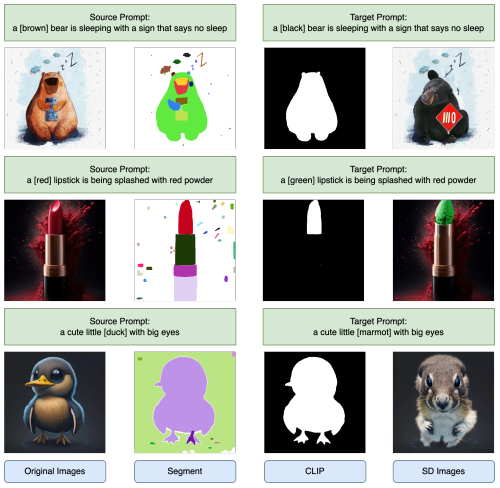


Figure 5. This figure illustrates some good examples of editing everything framework, where specific objects (a bear, lipstick, and duck) are accurately modified based on target prompts, demonstrating seamless integration and high-quality generation.

4.4. Experiment 3: Residual Correction

4.4.1. Experiment Set up

I run the evaluation experiments on my own laptop: NVIDIA GeForce RTX 3070 Laptop GPU Total Memory: 8.59 GB Compute Capability: 8.6.

4.4.2. Experimental Result

We compare our proposed method with the baseline model and the PnpInversion method applied to the baseline model. The baseline model, described in Sec. 3.3, is our integrated ddim+p2p framework. To enhance its performance, we evaluate two methods: (1) applying the PnpInversion method [10] and (2) our proposed residual correction method, detailed in Sec. 3.2.1.

The evaluation results are summarized in Tab. 4. Our proposed residual correction method demonstrates consistent improvements across all three metrics: LPIPS, SSIM, and CLIP.

Methods	LPIPS↓	SSIM↑	CLIP↑
Integrated IC	0.29	0.65	24.99
w/ PnpInversion	0.27(6% ↓)	0.63 (3% ↓)	27.54 (14% ↑)
w/ Residual Correction	**0.21** (27% ↓)	**0.71** (9% ↑)	**27.54** (14% ↑)

Table 4. **Performance Comparison Across Methods.** The table compares the integrated ddim+p2p model with IC-Light (baseline), the PnpInversion method applied to the baseline, and our proposed residual correction method. The residual correction method achieves notable improvements in all three metrics, particularly in LPIPS and SSIM.

Figure 6 presents visual outputs from different methods.

Our proposed weighted addition in the latent space avoids catastrophic errors in the generated image and preserves the background patterns more faithfully compared to the baseline model. This confirms the effectiveness of our approach in improving content preservation during target prompt generation.

Figure 6. Visual outputs of different methods for a sample prompt transformation. The input prompt changes from "fruit" to "candy." The proposed method demonstrates improved background preservation and overall quality compared to the baseline.

5. Discussion

5.1. Editing Everything

Since in Experiment 2, we found a few results that were partially or obviously wrong, we did some discussion for the different situations

5.1.1. Partial Success Examples

While our pipeline exhibits strong performance in text-driven image editing, certain examples highlight limitations that affect both the accuracy and visual coherence of the generated outputs. These limitations mainly arise from two key challenges: dependence on precise textual prompts and inconsistencies in integrating new content with the surrounding context.

Firstly, the pipeline is highly dependent on detailed and specific text prompts. When prompts are vague or imprecise, the generated results can appear unclear or unrealistic. For instance, in the bottom example of Fig. 7, although the source image already features a lake, the model generates an additional lake in front of the cabin, creating redundancy and disrupting logical consistency. This issue underscores the system's sensitivity to the precision of input prompts and its limited ability to infer contextual details beyond the provided description.

Secondly, directly replacing segmented areas using Stable Diffusion (SD) often leads to inconsistencies with the surrounding scene. Since SD focuses on generating content only within the segmented region, it may fail to account for the broader environment, resulting in mismatched texture, lighting, or style. For example, in the top and middle examples of Fig. 7, the added elements—such as the frog in the fishbowl or the rock beneath the cartoon girl—appear visually disconnected from the original background. Such misalignment disrupts the overall visual harmony, making the edits appear unnatural and out of place.

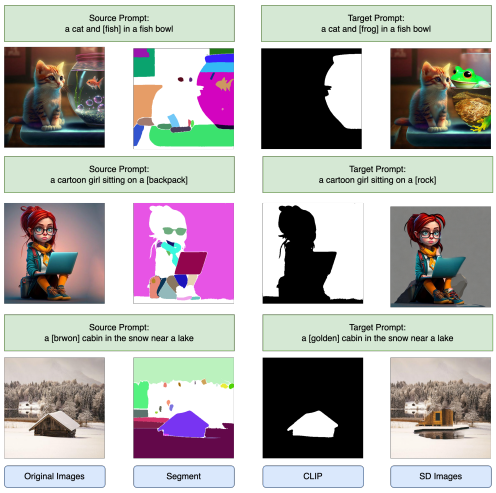


Figure 7. This figure shows some partially successful examples, where the correct results are generated according to the target prompt, but the generated content is separated from the background or the lighting information is lost.

5.1.2. Bad Examples

While our pipeline demonstrates strong capabilities, certain failure cases expose key limitations that affect the precision and consistency of the editing results. These shortcomings primarily stem from inaccurate segmentation by SAM and misidentified segment ranking by CLIP, especially in complex or ambiguous scenarios.

Firstly, SAM's imprecise segmentation can result in inconsistencies, particularly in scenes where distinguishing clear boundaries between objects and the background is critical. For instance, in the bottom image of Fig. 8, the cat in the primary region is successfully replaced with a tiger, but the reflection in the mirror still retains the original cat's shape and texture. This disrupts the overall visual coherence because the mask does not accurately differentiate between the reflected and primary objects. As a result, the generated content suffers from unnatural transitions, particularly at boundaries in reflective or intricate scenes.

Secondly, CLIP's segment ranking can incorrectly identify the intended editing target, leading to inaccurate modifications. In the top image of Fig. 8, CLIP mistakenly selects the leopard-print blanket as the target for editing instead of the kitten. Consequently, the blanket changes to yellow as specified in the prompt, while the kitten remains mostly unaltered. This outcome highlights CLIP's difficulty in accurately identifying the main subject when segmentation regions overlap or when the contrast between foreground and background is subtle.

These examples underline the need for more robust segmentation methods and improved ranking strategies to ensure accurate object targeting and seamless integration in complex visual environments.

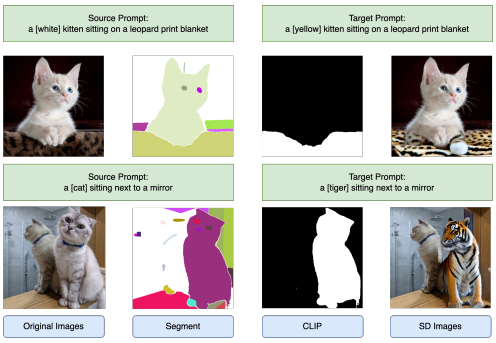


Figure 8. The figure shows two failure cases, one where the mirror reflection retains the original cat in the mirror after replacing the main object, and the other where the background blanket is mistakenly edited instead of the kitten.

5.2. Residual Correction

As shown in Tab. 4, the performance of our integrated IC-Light baseline method and the proposed improvements still leave significant room for enhancement. This is primarily due to how we integrated the components. The original IC-Light method, as illustrated in Fig. 4, operates by inputting only the extracted foreground and incorporating background conditioning. With this additional conditioning and advanced fine-tuning, IC-Light achieves state-of-the-art performance.

In contrast, we deliberately removed background conditioning because our proposed method is not designed to compete with state-of-the-art models directly. Instead, our goal was to evaluate the conceptual validity of our approach. Given that the input to our method excludes the background entirely, it is highly unlikely that our performance will surpass baseline methods, as evidenced in Tab. 1. However, our residual correction mechanism does improve unedited content preservation by addressing the background mask.

This result opens up new possibilities for generative model methods, suggesting alternative approaches to handling background elements in the latent space that could inspire further exploration and innovation.

6. Conclusion

In this paper, the experimental results of our three main methods clearly show that our work has improved editing fidelity, image quality, and control across multiple editing tasks.

Firstly, by comparing the frameworks with and without SAM and ControlNet, we find that the framework integrating SAM and ControlNet performs better on LPIPS and CLIP metrics. The LPIPS is reduced by about 16%, showing an improvement in the perceptual quality of the edited images; In addition, the CLIP score is improved by about 14%, indicating that the relevance of the generated images to the text prompt is significantly enhanced. This has also proven that the accurate segmentation capability of SAM and the control capability of ControlNet is playing an important role in the editing framework, especially in achieving high-quality editing while maintaining consistency in the unedited regions.

The performance of the Editing Everything framework is also shown by integrating SAM, CLIP, and Stable Diffusion, the Editing Everything framework demonstrates excellent performance in a variety of editing tasks. It has shown a further improvement in all three metrics comparing to the first method. This demonstrates that the Editing Everything framework not only generates high-quality images, but also accurately responds to the user's text prompts, ensuring that the edited content is highly consistent with the user's expectations.

Finally, for the combination of Residual Correction and PnP Inversion, by adding PnP Inversion and Residual Correction to the IC-Light pipeline, the system performance is further improved. The experimental results demonstrate that the addition of residual correction further decreases the LPIPS value by about 27% and improves the SSIM and CLIP values by about 9% and 14%, respectively. This result has strongly suggested that residual correction plays a key role in improving editing fidelity and adaptivity, especially in complex multi-target editing tasks.

In summary, our proposed methods has achieved comprehensive improvements in several metrics through modular design and integration of key technologies.Though further improvements can be made, these results has fully validated the potential and advantages of our framework in the field of text-driven image editing and provide a solid foundation for more complex editing tasks in the future. In the future, we will further explore the possibilities of real-time editing, finer-grained control, and multimodal generation.

7. Contribution

Rui Xia: Evaluation of Baseline 1, Method 1 coding and writing
Yuxiang Ying: Evaluation of Baseline 3, Method 3 coding and writing
Longxin Wang: Evaluation of Baseline 2, Method 2 coding and writing
Xikai Ma: Evaluation of Baseline 4, Method 1 and 2 Proposal and parts of evaluation, other parts writing

References

[1] Anonymous. Scaling in-the-wild training for diffusion-based illumination harmonization and editing by imposing consistent light transport. In *Submitted to The Thirteenth International Conference on Learning Representations*, 2024. under review. 1, 2, 3

[2] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18208–18218, 2022. 1

[3] Omri Avrahami, Dani Lischinski, and Ohad Fried. 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, 2022. 2

[4] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18392–18402, 2023. 1

[5] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 2

[6] 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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22560–22570, 2023. 1

[7] Zigang Geng, Binxin Yang, Tiankai Hang, Chen Li, Shuyang Gu, Ting Zhang, Jianmin Bao, Zheng Zhang, Houqiang Li, Han Hu, Dong Chen, and Baining Guo. Instructdiffusion: A generalist modeling interface for vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12709–12720, 2024. 1

[8] Paul Haeberli. Synthetic lighting for photography. *http://www. graficaobscura. com/synth/index. html*, 1992. 5

[9] Inbar Huberman-Spiegelglas, Vladimir Kulikov, and Tomer Michaeli. An edit friendly ddpm noise space: Inversion and manipulations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12469–12478, 2024. 1

[10] Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Pnp inversion: Boosting diffusion-based editing with 3 lines of code. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 2, 3, 5, 6

[11] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 2, 3

[12] Yuanze Lin, Yi-Wen Chen, Yi-Hsuan Tsai, Lu Jiang, and Ming-Hsuan Yang. Text-driven image editing via learnable regions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7059–7068, 2024. 3

[13] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6038–6047, 2023. 1

[14] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *ACM SIGGRAPH 2023 Conference Proceedings*, New York, NY, USA, 2023. Association for Computing Machinery. 1

[15] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2

[16] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. 1, 2, 5

[17] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. 5

[18] Chenfei Wu, Lun Huang, Qianxi Zhang, Binyang Li, Lei Ji, Fan Yang, Guillermo Sapiro, and Nan Duan. Godiva: Generating open-domain videos from natural descriptions, 2021. 5

[19] Defeng Xie, Ruichen Wang, Jian Ma, Chen Chen, Haonan Lu, Dong Yang, Fobo Shi, and Xiaodong Lin. Edit everything: A text-guided generative system for images editing. *arXiv preprint arXiv:2304.14006*, 2023. 2, 3

[20] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 2, 3

[21] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 5

Boost Text-Driven Image Editing with SAM and Residual Correction

Supplementary Material

8. IC-Light structure

The IC-Light repository implements two main pipelines: a text-to-image pipeline and an image-to-image pipeline. In the workflow, the text-to-image pipeline generates an initial image, which subsequently serves as a conditional input for the image-to-image pipeline. While this setup is effective, it does not introduce a novel methodology and adheres to standard practices within diffusion-based image generation frameworks.

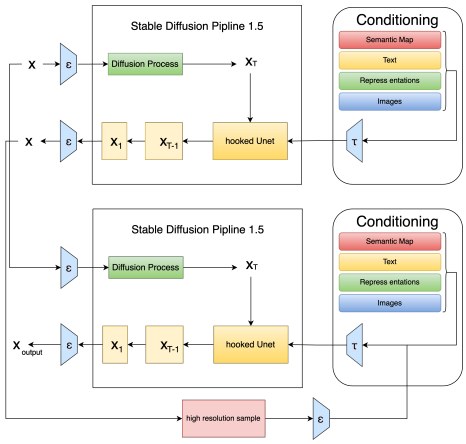


Figure 9. Overview of the IC-Light Architecture. The diagram illustrates the integration of text-to-image and image-to-image pipelines, where the output of the former conditions the latter to refine or modify the image.