

# MAG-Edit: Localized Image Editing in Complex Scenarios via Mask-Based Attention-Adjusted Guidance

## Supplementary Materials

Anonymous Authors

### 1 SUMMARY

In this supplementary material, we present more implementation details, additional experiments, and additional results as follows.

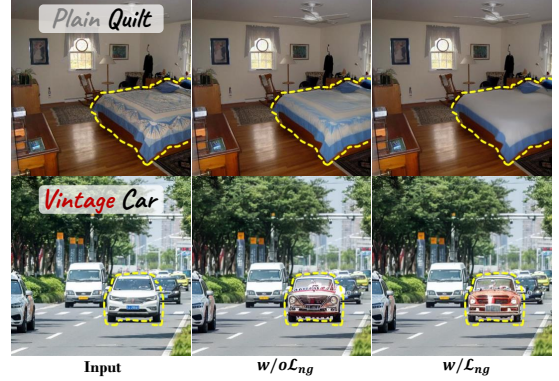
- We present more implementation details of MAG-Edit in Sec. 2. Furthermore, Sec. 5 illustrates more implementation details of the benchmark dataset, baselines, quantitative metrics, and user study.
- In Sec. 3, we present a comprehensive explanation and implementation details of the negative prompt constraint, demonstrating its effectiveness in assisting the editing process.
- In Sec. 4, we showcase additional results by applying our MAG-Edit to another attention-based editing method Plug-and-Play (PnP) [16] and the advanced inversion method PnP-Inversion [8], further highlighting the versatility and performance of our MAG-Edit approach.
- In Sec. 6, we extend our comparisons to encompass training and fine-tuning methods.
- We demonstrate additional qualitative results to complement the paper in Sec. 7.

### 2 IMPLEMENTATION DETAILS

We employ the DDIM method [15] over  $T = 50$  steps for the denoising sampling process, maintaining a constant classifier-free guidance scale of 7.5. CA injection is performed during  $[T, \mu]$ . For varying editing requirements, we set  $\mu = 10$  for color and texture edits, and  $\mu = 40$  for shape variation edits. For the gradient guidance process, we follow [3] by setting the gradient update scale  $\delta$  using a linear scheduling rate as  $\sqrt{(1 - \alpha_t)/\alpha_t}$ , particularly to optimize the token ratio constraint  $\mathcal{L}_{TR}$ . This approach modulates the gradient’s magnitude based on the denoising progress. On the contrary, for the constraint of the spatial ratio  $\mathcal{L}_{SR}$ , we keep  $\delta = 1$ . To further preserve the structure of the original image, we also consider incorporating self-attention as P2P [6] and replace them at diffusion steps  $t \in [T, 25]$ . Towards the end of the denoising process  $t \in [15, 0]$ , we implement a latent blending operation from P2P [6] to maintain information outside the edited region mask  $\mathcal{M}$ .

### 3 NEGATIVE PROMPT CONSTRAINT

**Details of Negative Prompt Constraint.** In real image editing, the latent noise feature  $z_T$  derived by the inversion methods still retains information related to the original image  $\mathcal{I}$ . Achieving the desired editing results can be challenging in some cases when there is a significant difference between the texture in the original image and modified prompt  $\mathcal{P}^*$ , such as transferring texture from “patterned” to “plain”, as shown in Fig. 1. Our proposed method can also be used to attenuate the textural information associated with the original image  $\mathcal{I}$  by employing negative prompts. In particular, we define a set of negative tokens  $\mathcal{S}_{ng}^*$  to present the texture of  $\mathcal{I}$  in contrast to the new



**Figure 1: Ablation study on the negative prompt constraint. Negative prompt constraints can amplify the effectiveness of editing by diminishing the influence of information from the original image.**

tokens  $\mathcal{S}^*$ . For example, if  $\mathcal{P}^*$  is “There is a bed with a plain quilt in the bedroom.” and the quilt in  $\mathcal{I}$  is patterned, then the negative token would be “patterned”.

Consequently, we can establish the negative prompt constraint  $\mathcal{L}_{ng}$  using the negative token’s corresponding CA value, which can be formulated as either  $\mathcal{L}_{TR}$  or  $\mathcal{L}_{SR}$ . By combining the two types of constraints calculated separately for the target token (e.g., “plain”) and the negative token (e.g., “patterned”), the total constraint can be defined as follows:

$$\mathcal{L}_{total} = \lambda_p \mathcal{L} - \lambda_{ng} \mathcal{L}_{ng}, \quad (1)$$

where  $\lambda_p$  and  $\lambda_{ng}$  aim to balance between positive and negative prompt constraints, we empirically set  $\lambda_p = 2.5$  and  $\lambda_{ng} = 5.5$ .

**Impact of Negative Prompt Constraint Guidance.** Fig. 1 demonstrates that negative prompt guidance is effective in diminishing the original image’s information, which is beneficial when dealing with original images that have information significantly contrast with the target prompt. For instance, as shown in the first row of Fig. 1, when altering the texture from patterned to plain, not applying negative constraints could lead to the edited image preserving some patterns. The negative prompt constraint, in such scenarios, efficiently reduces this residual patterned information.

### 4 APPLYING MAG-EDIT TO OTHER BASELINES

**Plug-and-Play Applied with MAG-Edit.** Plug-and-Play [16] (PnP) is an attention-based method that incorporates the use of feature and self-attention (SA) from the reconstruction branch into the editing process, to preserve the structure and layout of the source images. Although PnP [16] performs well in simple scenarios, it encounters challenges such as leakage and minimal editing effects when dealing



**Figure 2: Qualitative results of PnP [16] and its combination with MAG-Edit, namely PnP+Ours.** By employing our MAG-Edit approach, the alignment between the local edit region and the target prompt is significantly enhanced, leading to notable improvements in the edited region.

with images that have complex compositions and objects. This limitation stems from its reliance solely on text prompts for localization, which results in misalignment between the features and the prompts, as depicted in Fig. 2. However, by applying our MAG-Edit, denoted as PnP+Ours, the editing results exhibit significant improvement. As shown in Fig. 2, the desired editing effects become readily apparent in the target region.

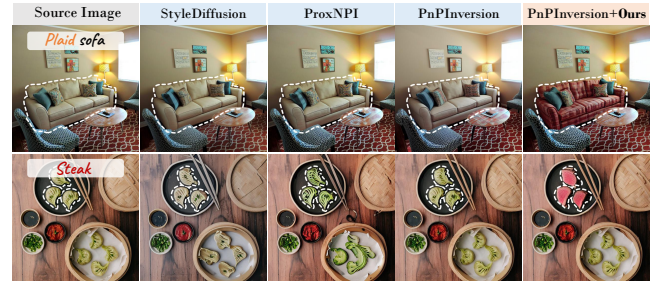
**PnPInversion Applied with MAG-Edit.** Recent advancements in inversion methods [10, 12, 13] focus on enhancing DDIM inversion [15] to achieve a better balance between editability and fidelity. However, these methods still rely on the use of Prompt-to-Prompt (P2P) [6] to facilitate image editing, which inherits its limitations, such as misalignment, resulting in minimal editing effects in the target region, as depicted in Fig. 3. Our method, MAG-Edit, is orthogonal to inversion-based methods but can be integrated with them to enhance performance. In particular, PnPInversion [8] aims to eliminate the trajectory offsets between the DDIM [15] inversion process and the reconstruction process, yielding optimal reconstruction and attention maps for the editing process. Nevertheless, it struggles to perform localized editing in complex scenarios, as illustrated in Fig. 3. When integrated with our MAG-Edit approach, now referred to as PnPInversion+Ours, there is a significant enhancement in the editing results, as showcased in Fig. 3.

## 5 DETAILS OF COMPARISONS WITH BASELINES

### 5.1 Benchmark Dataset

Existing well-recognized datasets for text-based image editing methods, such as *i.e.*, TEd-Bench [9] and PIE-Bench [8], primarily focus on simple scenes with prominent objects. For comparative analysis, we manually construct prompts and generate masks using the Segment Anything method<sup>1</sup> (SAM) for TEd-Bench [9]. For PIE-Bench [8], which already includes mask annotations, our evaluation focuses on its three specific categories: content, color, and material.

<sup>1</sup><https://github.com/facebookresearch/segment-anything>



**Figure 3: Qualitative comparisons of inversion methods and our MAG-Edit applied with PnPInversion [8].** The recent inversion methods combined with P2P [6] struggle to produce effective editing results in localized regions within complex scenarios. Our MAG-Edit approach is compatible with various inversion methods such as PnPInversion [8]. It is evident that when our method is employed, PnPInversion+Ours demonstrates notable advancements.

To enable a more thorough evaluation of our method particularly in complex scenarios, we have developed a benchmark dataset, named MAG-Bench, consisting of 200 images sourced from MSCOCO [11], ADE20K [19], Cityscape [4], and the Internet. This dataset features complex scenes with multiple objects in various real-world indoor and outdoor settings, encompassing a wide range of object categories like humans, furniture, animals, vehicles, and food. MAG-Bench is specifically designed to assess three types of local editing: (1) color editing, (2) texture editing which includes changes in material, background, and style, and (3) object replacement. For the generation of source and target prompts, we initially utilized GPT-4 [14], followed by manual refinement to ensure the accuracy and relevance of these prompts. The corresponding editing masks for each image are derived using SAM. Acknowledging the critical role of the mask's size in localized editing, we initially classify each image into three categories based on mask size: relatively small, medium, and relatively large. We then ensure a balanced distribution of varying sizes of editing regions across the datasets. Thus, each image in MAG-Bench is accompanied by three annotations: a source prompt, a target edit prompt, and an edit region mask, as illustrated in Fig. 4.

### 5.2 Implementation Details of Baselines

We use the official codes released by the authors for Blended LD<sup>2</sup>, PnP<sup>3</sup>, MasaCtrl<sup>4</sup> and P2P<sup>5</sup>. For DiffEdit [5], we adopt the implementation from InstructEdit<sup>6</sup>, which enhances automatic mask generation for scenarios involving multiple objects. This implementation, while improving upon mask generation, does not modify the core editing algorithm of DiffEdit [5]. To facilitate fair comparisons, all methods use *identical masks* provided in our benchmark dataset. Notably, for DiffEdit [5] and P2P [6], we utilize ground-truth masks instead of those generated through unsupervised learning or derived from average CA maps. In the case of P2P [6], we also integrate

<sup>2</sup><https://github.com/omriav/blended-latent-diffusion>

<sup>3</sup><https://github.com/MichalGeyer/plugin-and-play>

<sup>4</sup><https://github.com/TencentARC/MasaCtrl>

<sup>5</sup><https://github.com/google/prompt-to-prompt>

<sup>6</sup><https://github.com/QianWangX/InstructEdit>


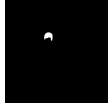





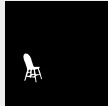


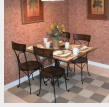

Scenarios	Edit Type	Source Image	Source Prompt	Target Prompt	Mask	Mask Type
Outdoor	Color		A couple and a kid with <b>black</b> hair are sitting on the bench	a couple and a kid with <b>blond</b> hair are sitting on the bench		Small
	Object		A green <b>truck</b> and some cars park under a tall building.	A green <b>bus</b> and some cars park under a tall building.		Medium
	Texture		Guinea fowl stand on <b>dry grass</b> under sky.	Guinea fowl stand on <b>desert</b> under sky.		Large
Indoor	Color		The wooden and round table is surrounded by four wooden chairs and a light <b>brown</b> chair is next to the windows.	The wooden and round table is surrounded by four wooden chairs and a light <b>red</b> chair is next to the windows.		Small
	Object		There are a <b>box</b> and lemons and several lemons on white sheet.	There are a <b>box</b> and lemons and several lemons on white sheet.		Medium
	Texture		There is a table with cups and four chairs on the <b>plaid</b> carpet.	There is a table with cups and four chairs on the <b>bohemian</b> carpet.		Large

Figure 4: Examples images and annotations in the MAG-Bench dataset.

Null-text inversion [13] as our approach for encoding real images. With the exception of Blended LD [1], which solely focuses on the target edit description for the foreground region and omits tokens for other unedited areas, all other methods employ target prompts identical to those used in our method.

### 5.3 Evaluation Details

We utilize the CLIP score with the CLIP ViT-L/14 model, as implemented in<sup>7</sup>, and the DINO-ViT self-similarity distance, available at<sup>8</sup>, as our evaluation metrics. To precisely evaluate localized editing, we crop the editing regions in both the source and edited images using bounding boxes as [7]. This approach enables us to specifically assess text prompt alignment within these localized regions by calculating the CLIP score on the target edited tokens with the respective cropped edited image. For instance, in a scenario where the editing objective is to alter a car's color to red, the CLIP score is computed using the phrase "red car." This calculation excludes common tokens shared between the source and target prompts and focuses solely on the cropped image depicting the edited car and the target phrase. To evaluate structure preservation within the localized editing regions, we utilize the DINO-ViT self-similarity by calculating the distance

between the cropped source image and the corresponding cropped edited image.

### 5.4 Details of User Study

We conduct a user study on the Amazon MTurk platform<sup>9</sup>. The user study comprises over 140 tasks, each evaluated by five human evaluators, as depicted in Fig. 5. In each task, participants are presented with a source image alongside two edited images: one generated by our proposed method and the other by a randomly selected baseline method, with their presentation order shuffled. To enhance the visibility of localized editing regions, we outline the prospective edit regions with white dashed lines in each pair of comparison images and their corresponding source images, as illustrated in Fig. 5. Additionally, a simplified version of the target edit prompt is displayed beneath the comparison images. We then pose three questions for the raters to answer:

- Text Alignment: In the dashed region, which image aligns better with the "edit prompt"?
- Structure Preservation: In the dashed region, which image preserves structures more similarly to the source image?
- Overall: In the dashed region, which image performs better overall?

<sup>7</sup><https://github.com/showlab/loveu-tgve-2023>

<sup>8</sup><https://github.com/omerbt/Splice>

<sup>9</sup><https://www.mturk.com>

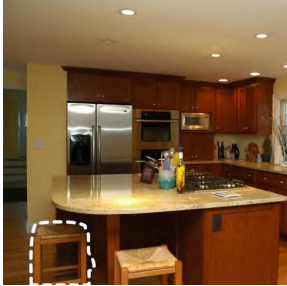
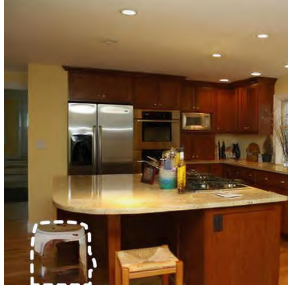
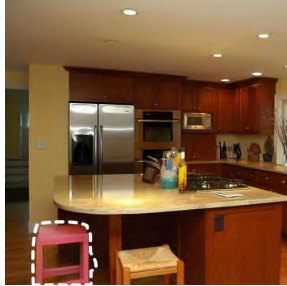


### Instructions

This task includes evaluating two AI based edits of real images in which we provided source image and target edit prompt. Moreover, we hope that **only dashed region** of corresponding image will be edited according to the target prompt. Please view the source image and target prompt and provided your feedback on the following criteria :

- Text Alignment:** In the **dashed region**, which image aligns better with the “**edit prompt**”?
- Structure Preservation:** In the **dashed region**, which image better preserves **structures** more similarly to the source image?
- Overall:** In the **dashed region**, which image performs better overall?

Our ultimate goal is to have the edited image and target edit prompt **aligned** as much as possible.

Source image                      Option 1                      Option 2

Target prompt: pink chair

- In the **dashed region**, which image aligns better with the “**pink chair**”?

☐ Option 1   ☐ Option 2

- In the **dashed region**, which image better preserves **structures** more similarly to the source image?

☐ Option 1   ☐ Option 2

- In the **dashed region**, which image performs better overall?

☐ Option 1   ☐ Option 2

Figure 5: Example of one task for 5 human raters on Amazon MTurk to complete.

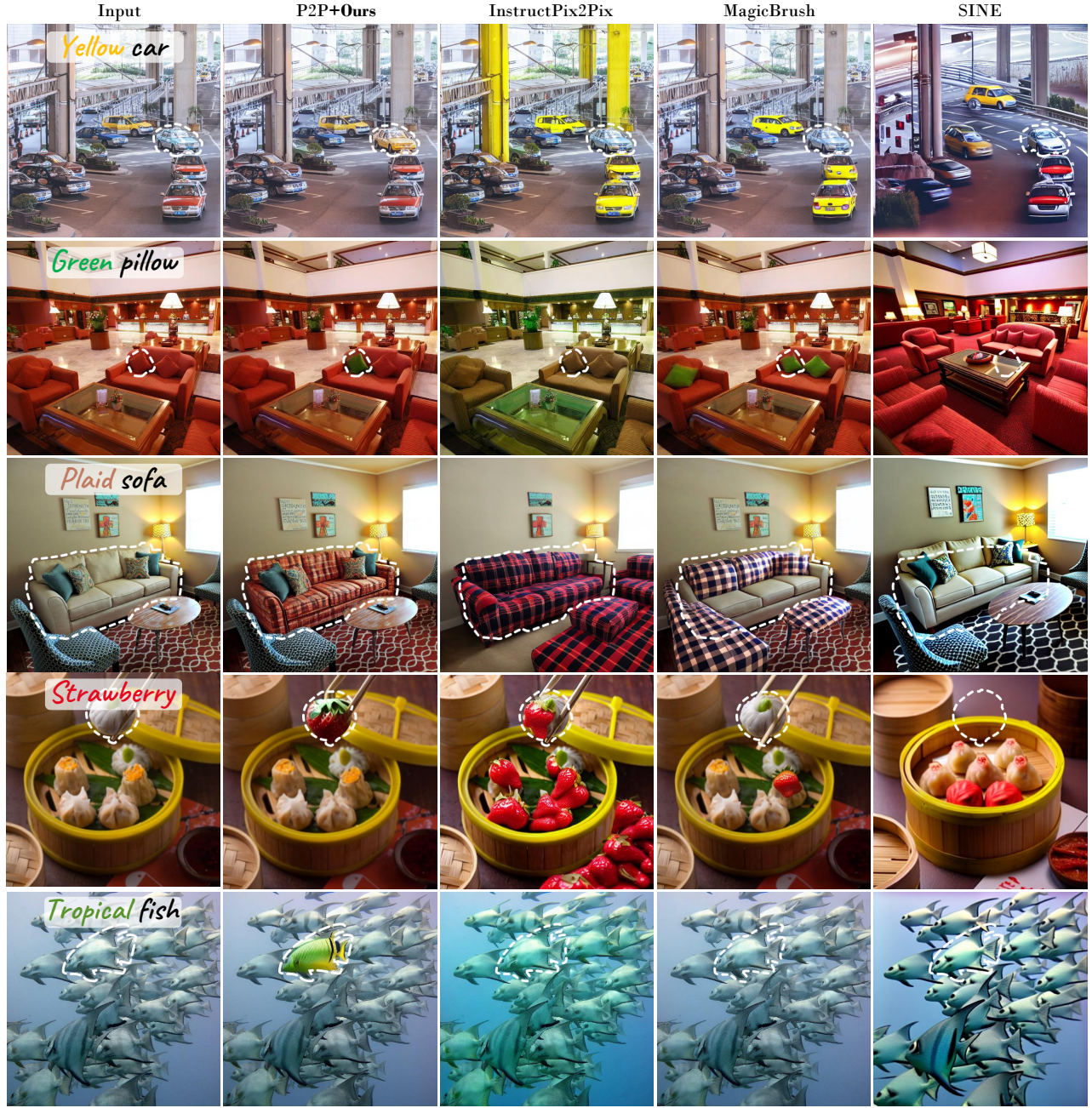
To ensure the credibility and reliability of our user study, we only involve Amazon MTurk workers with ‘Master’ status and a Human Intelligence Task (HIT) Approval Rate exceeding 90% across all Requesters’ HITs. In total, the 140 tasks garnered responses from 700 distinct human evaluators.

## 6 COMPARISONS WITH TRAINING AND FINE-TUNING METHODS

We conduct a comparison with existing training methods by evaluating InstructPix2Pix [2] and MagicBrush [17], utilizing their officially released codes and models. InstructPix2Pix [2] is trained on an extensive dataset, which includes instructions generated by GPT-3 and image examples modified by P2P [6]. This training facilitates instruction-based image editing during the inference phase. MagicBrush [17] harnesses a large-scale dataset of manually annotated real image editing triplets and optimizes the InstructPix2Pix model to improve editing capabilities. For our comparisons, we utilize editing instructions such as “make” and “change” to manipulate images. Fig. 6 illustrates that InstructPix2Pix, due to its lack of mask integration, frequently leads to substantial leakage into incorrect regions

during localized editing in complex scenes. In contrast, MagicBrush demonstrates better localized editing in some cases, thanks to mask-integrated examples in its dataset. However, MagicBrush encounters difficulties in precisely localizing individual objects within scenes containing multiple similar objects. This challenge is evident in the first and second rows of Fig. 6, where it struggles with tasks like coloring one car yellow and one pillow green. Moreover, as shown in the third row of Fig. 6, MagicBrush [17] tends to modify the underlying structure in areas undergoing texture changes. In contrast, our training-free method efficiently attains desired editing effects in the target local regions while preserving the original structure. A significant advantage of our approach is the elimination of the need for extensive training on large datasets, saving significant time and resources.

Subsequently, we compare our method with the existing fine-tuning method, SINE [18], using the code provided by its authors. SINE [18] proposes fine-tuning a pre-trained text-to-image (T2I) model with a single real image, incorporating model-based classifier guidance and patch-based guidance to prevent overfitting. However, as illustrated in Fig. 6, SINE fails to generate any noticeable editing



**Figure 6: Qualitative comparisons with training and fine-tuning methods for localized editing in complex scenarios. Training approaches such as InstructPix2Pix [2] and MagicBrush [17] demonstrate issues like leakage or unintended modifications in structure. The fine-tuning method SINE [18] is ineffective in both reconstructing and generating desired editing effects.**

effects in the intended regions. Furthermore, it faces difficulties in accurately reconstructing the original image in complex scenarios.

## 7 ADDITIONAL RESULTS

Our method offers a broad spectrum of localized editing capabilities, encompassing object attribute manipulation (*e.g.*, color and texture), object replacement, insertion, and removal, as exemplified in Fig. 7.

Additional examples of localized editing in complex scenarios are illustrated in Fig. 9 and Fig. 10. Furthermore, we demonstrate the controllability of our localized editing approach in terms of the magnitude of edits in Fig. 8. This allows for precise adjustment of editing granularity, catering to a variety of user requirements.



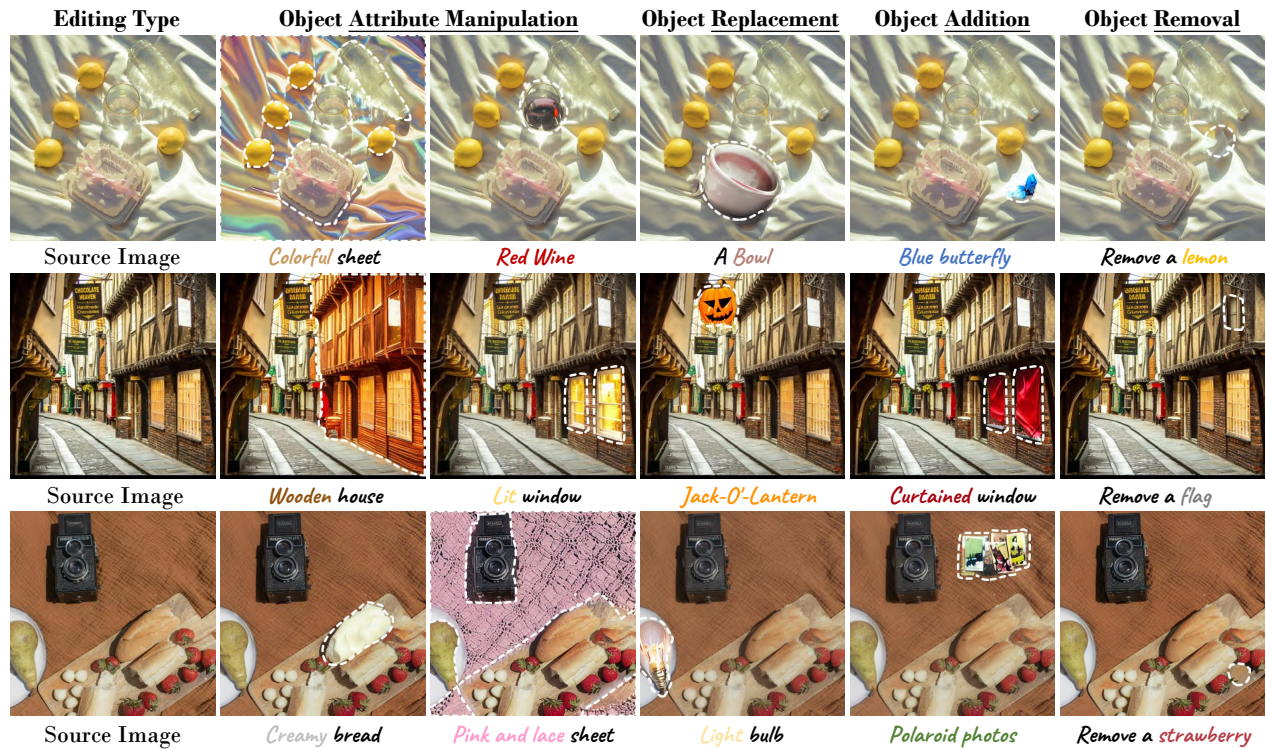


Figure 7: Various localized editing types. We provide a simplified version of the corresponding target prompt under each edited image.

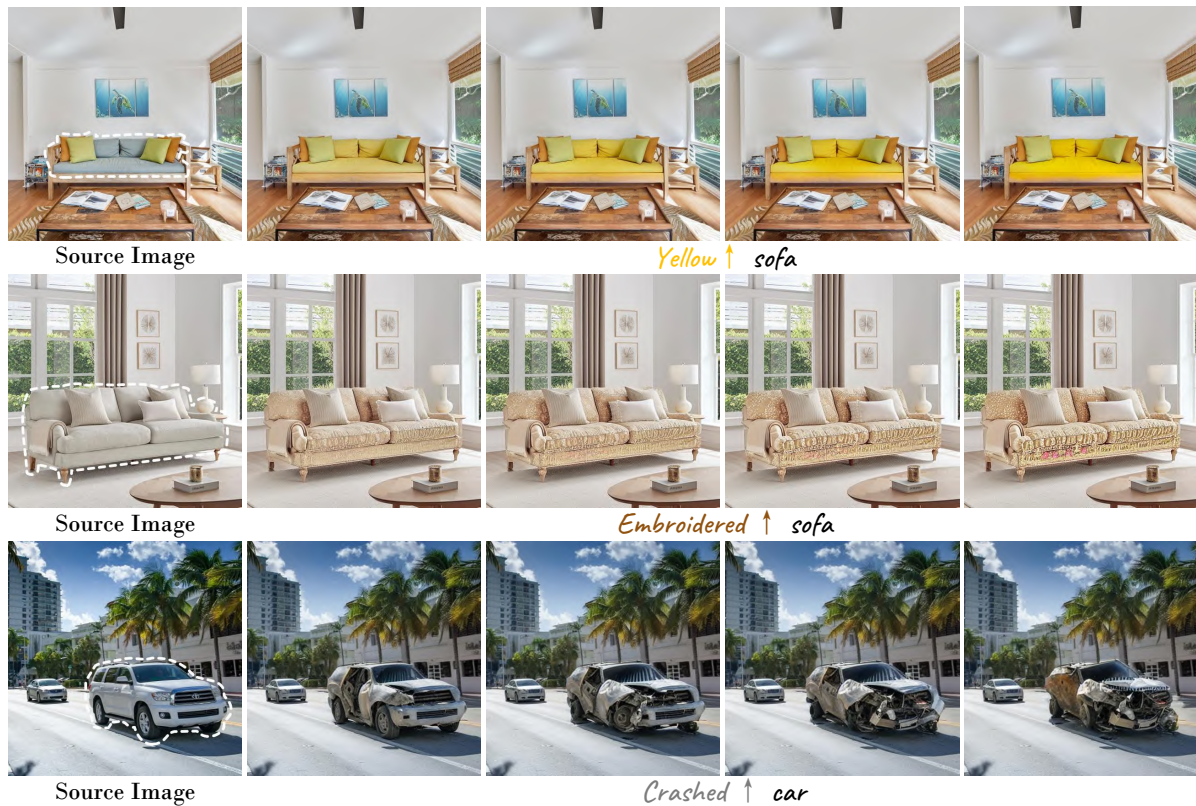


Figure 8: Granularity controllable localized editing. We present a simplified version of the corresponding target prompt under the edited images. ↑ denotes increasing the editing magnitude.



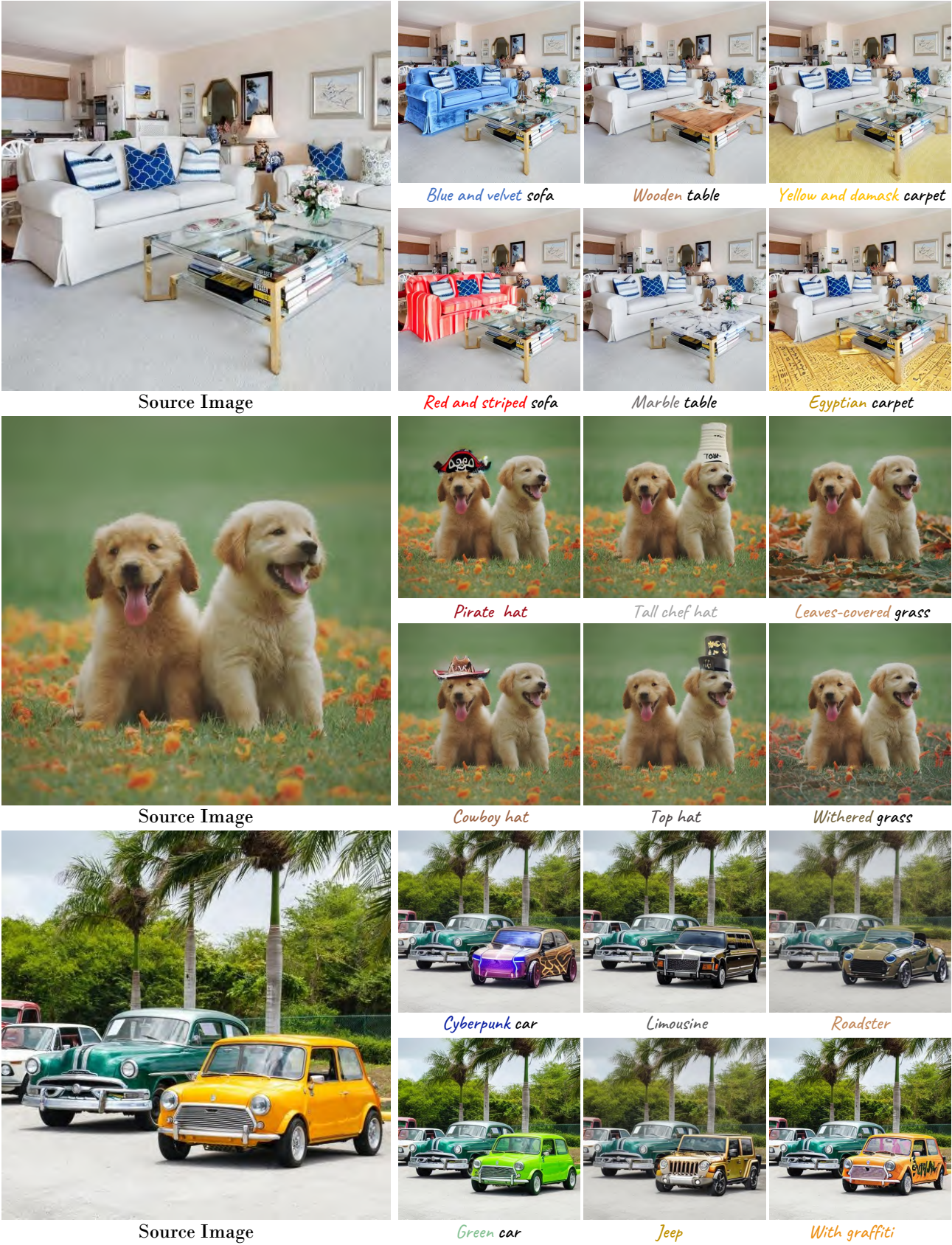


Figure 9: Various localized editing types. In each edited image, we present a simplified version of the corresponding target prompt.



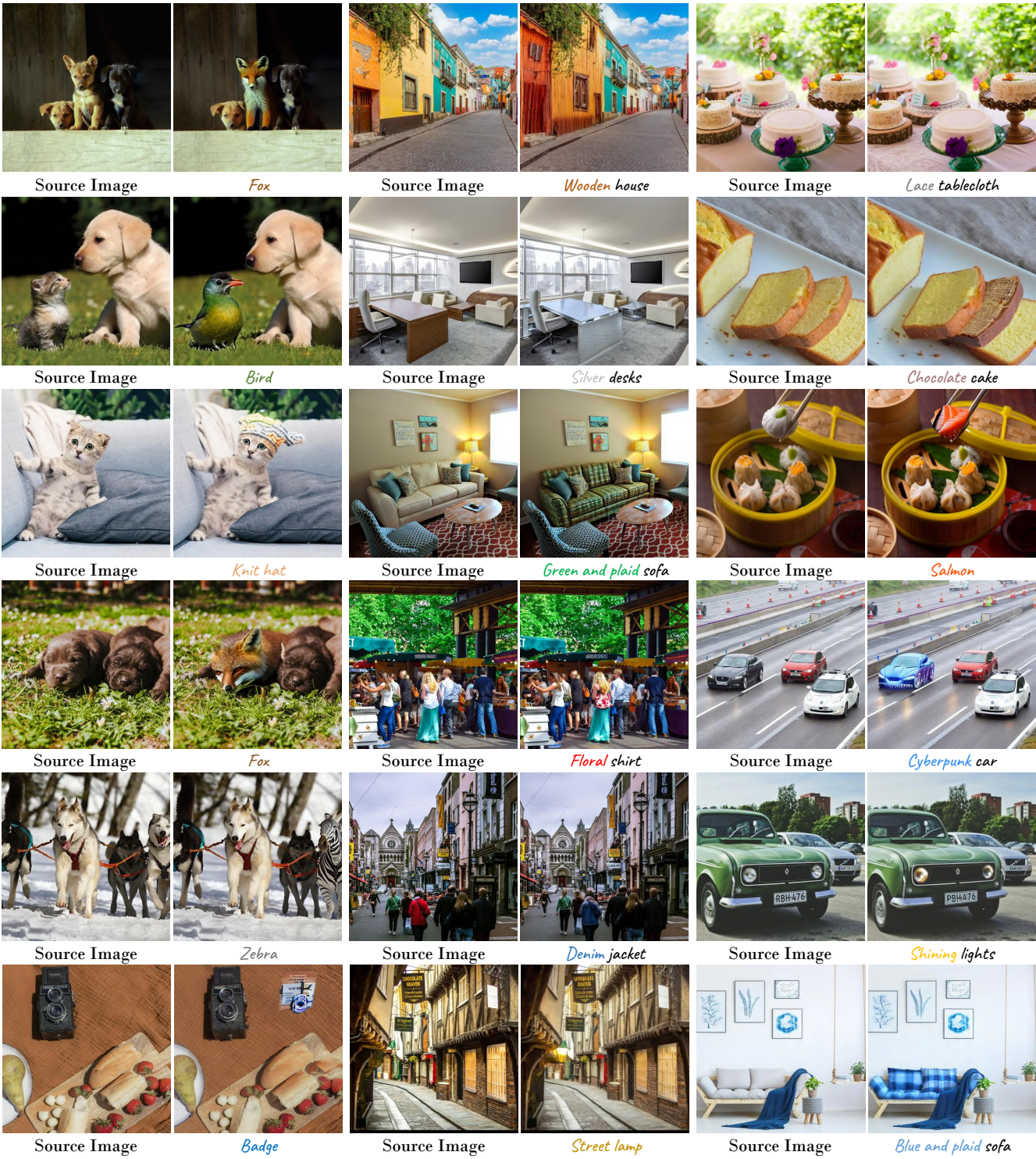


Figure 10: Additional results on localized editing in complex scenarios. We provide a simplified version of the target prompt beneath each edited image.



## REFERENCES

- [1] Omri Avrahami, Ohad Fried, and Dani Lischinski. 2023. Blended Latent Diffusion. *ACM Trans. Graph.* 42, 4 (2023), 149:1–149:11. <https://doi.org/10.1145/3592450>
- [2] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. 2023. InstructPix2Pix: Learning to Follow Image Editing Instructions. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17–24, 2023*. IEEE, 18392–18402. <https://doi.org/10.1109/CVPR52729.2023.01764>
- [3] Minghao Chen, Iro Laina, and Andrea Vedaldi. 2023. Training-Free Layout Control with Cross-Attention Guidance. *CoRR* abs/2304.03373 (2023). <https://doi.org/10.48550/ARXIV.2304.03373> arXiv:2304.03373
- [4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. 2016. The cityscapes dataset for semantic urban scene understanding. In *CVPR*. 3213–3223.
- [5] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. 2023. Diffedit: Diffusion-based semantic image editing with mask guidance. In *ICLR*. 1–22.
- [6] 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 *ICLR*. 1–36.
- [7] Wenjing Huang, Shikui Tu, and Lei Xu. 2023. PFB-Diff: Progressive Feature Blending Diffusion for Text-driven Image Editing. *arXiv preprint arXiv:2306.16894* (2023).
- [8] Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. 2024. PnP Inversion: Boosting Diffusion-based Editing with 3 Lines of Code. *International Conference on Learning Representations (ICLR)* (2024).
- [9] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. 2023. Imagic: Text-based real image editing with diffusion models. In *CVPR*. 6007–6017.
- [10] Senmao Li, Joost van de Weijer, Taihang Hu, Fahad Shahbaz Khan, Qibin Hou, Yaxing Wang, and Jian Yang. 2023. StyleDiffusion: Prompt-Embedding Inversion for Text-Based Editing. *arXiv preprint arXiv:2303.15649* (2023).
- [11] 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 *ECCV*. 740–755.
- [12] Daiki Miyake, Akihiro Iohara, Yu Saito, and Toshiyuki Tanaka. 2023. Negative-prompt Inversion: Fast Image Inversion for Editing with Text-guided Diffusion Models. *arXiv preprint arXiv:2305.16807* (2023).
- [13] 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 *CVPR*. 6038–6047.
- [14] OpenAI. 2023. GPT-4 Technical Report. *arXiv preprint arXiv:2303.08774* (2023).
- [15] Jiaming Song, Chenlin Meng, and Stefano Ermon. 2021. Denoising diffusion implicit models. In *ICLR*. 1–20.
- [16] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. 2023. Plug-and-play diffusion features for text-driven image-to-image translation. In *CVPR*. 1921–1930.
- [17] Kai Zhang, Lingbo Mo, Wenhui Chen, Huan Sun, and Yu Su. 2023. MagicBrush: A Manually Annotated Dataset for Instruction-Guided Image Editing. In *NeurIPS*.
- [18] Zhixing Zhang, Ligong Han, Arnab Ghosh, Dimitris N Metaxas, and Jian Ren. 2023. Sine: Single image editing with text-to-image diffusion models. In *CVPR*. 6027–6037.
- [19] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. 2017. Scene parsing through ade20k dataset. In *CVPR*. 633–641.

987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016  
1017  
1018  
1019  
1020  
1021  
1022  
1023  
1024  
1025  
1026  
1027  
1028  
1029  
1030  
1031  
1032  
1033  
1034  
1035  
1036  
1037  
1038  
1039  
1040  
1041  
1042  
1043  
1044