Instruction-Guided Precise Image Editing Using Multimodal LLMs

Marco Schouten[1,3] Mehmet Onurcan Kaya[1,3] Serge Belongie[2,3] Dim P. Papadopoulos[1,3]
[1] Technical University of Denmark [2] University of Copenhagen [3] Pioneer Centre for AI
https://poem.compute.dtu.dk

Abstract

Diffusion models have advanced text-to-image generation, producing high-quality images from textual prompts. However, editing specific objects remains challenging due to the need for precise modifications without affecting the overall scene. Text-driven methods often struggle with localized edits, while interactive approaches require manual input. We propose POEM (Precise Object-level Image Editing from Text Instruction via MLLMs), a framework that leverages multimodal large language models for fine-grained, instruction-driven editing. POEM generates object masks before and after transformation, guiding a diffusion-based process for accurate localization and modification without user input. To evaluate our method, we introduce VOCEdits, a benchmark based on PASCAL VOC 2012 with annotated prompts and ground-truth transformations. Experiments show that POEM improves precision and reliability over prior methods without relying on manual effort.

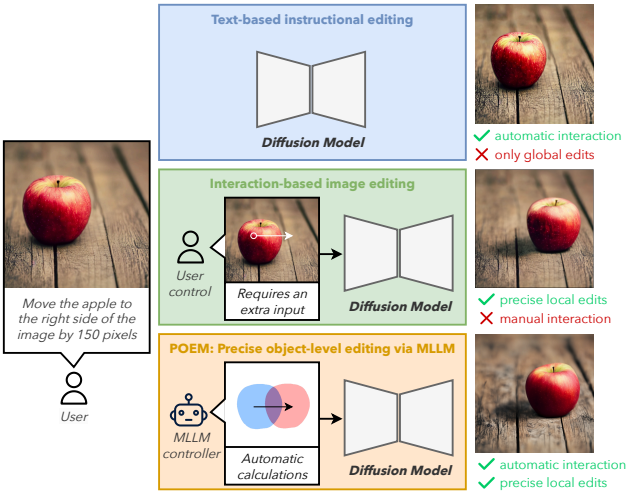

Figure 1. **POEM.** Existing text-based editing methods (top) struggle with precise object-level edits. Interaction-based approaches (middle) perform better but require manual user effort. Instead, we propose (bottom) leveraging MLLMs to interpret text-based prompts and automatically generate precise object masks and numerical transformations to support image editing pipelines.

1. Introduction

Diffusion models [27, 30] have significantly advanced high-resolution text-to-image generation, producing realistic images from textual prompts. Beyond generation, image editing [15, 16] has emerged as a key application, enabling users to modify images while preserving realism. A central challenge in image editing is precise object-level manipulation without disrupting global structure. While current methods support global edits [2], fine-grained transformations with high spatial accuracy remain burdensome [13].

Broadly, image editing methods fall into two categories: text-based instructional editing [2, 16, 17, 31] and image interaction-based editing [4, 7, 12, 15, 18, 19, 22, 32, 36]. Text-based methods like InstructPix2Pix [2], modifies input images based on a single edit prompt, making it efficient and user-friendly. Even though these methods have shown compelling results with global edits, they struggle with precise object-level shape transformations, often producing unintended global changes (Fig. 1, top). This is mainly because they purely rely on cross-attention text conditioning of a stable diffusion model [2, 16]. In contrast, interaction-based approaches require users to provide additional guidance through precise object masks [15, 19, 22, 36], object modification shapes [7] or click and drag [4, 12, 18]. While these methods can localize edits accurately and improve object-level editing, they demand manual effort, making them less scalable (Fig. 1, middle).

To address these limitations, we introduce **POEM** (**P**recise **O**bject-level Image **E**diting via from Text Instruction via **M**LLMs), a novel framework that decouples visual reasoning from the editor to achieve fine-grained object transformations (Fig. 1, bottom). Instead of requiring users to provide precise image interactions, POEM leverages Multimodal Large Language Models (MLLMs) to interpret instructional prompts, generate precise object masks before and after transformation, and provide image content descriptions. Inspired by recent advancements in large language models (LLMs) for complex reasoning [11, 35] and

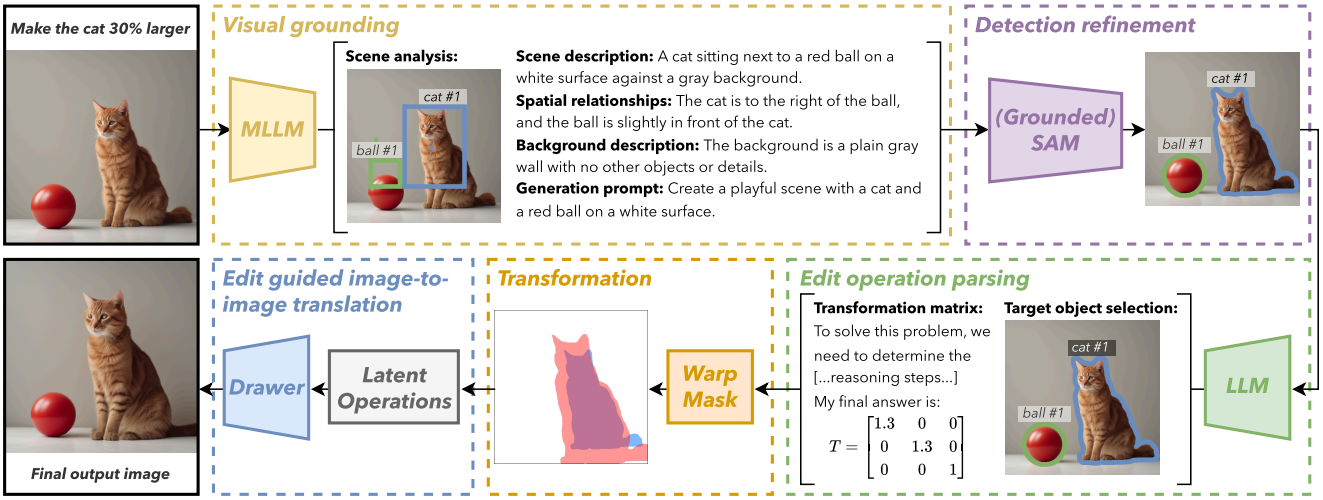

Figure 2. **POEM Pipeline.** Given an image and an edit prompt, we first use an MLLM to analyze the scene and detect objects. We then refine detections and enhance object masks using Grounded SAM. A text-based LLM predicts the transformation matrix for the target object. Finally, we perform image-to-image translation guided by the previous steps to generate the edited image.

MLLMs [9, 10, 23, 33, 34, 38, 39] for guiding diffusion processes, POEM ensures object localization and transformation without manual annotation.

Given an input image and a user edit instruction, POEM operates in two stages. In the reasoning stage, MLLMs generate structured editing instructions, including precise segmentation masks that define object boundaries before and after the transformation. These masks then guide the editing stage, where we apply controlled modifications in the latent space of a pre-trained diffusion model. By constraining the generation process with defined regions, POEM ensures fine-grained control over object transformations, surpassing previous text-based approaches in precision and reliability.

Existing datasets for image editing [37, 40] evaluate generic editing instructions, but they fail to capture the nuanced variations and details that are critical when assessing object shape edits. To address this gap and validate our method, we introduce a novel dataset, VOCEdits, by augmenting the training set of PASCAL VOC 2012 [8] with instructional edits and precise ground-truth object masks for before-and-after transformations. Our dataset enables a rigorous evaluation of our framework's ability to handle precise edits, which existing datasets do not fully account for.

Experiments demonstrate that POEM achieves significantly higher edit fidelity compared to existing text-based editing approaches while requiring no additional user annotations, unlike interaction-based methods.

Our contributions are two-fold: (a) we introduce a plug-and-play reasoning block that interprets user edit instructions with high numerical precision, generating accurate object masks and transformation matrices that enhance layout modifications and mask-guided diffusion editing; (b) we present VOCEdits, a novel dataset for evaluating precise object-level edits, establishing a comprehensive benchmark for detection, transformation, and synthesis tasks.

2. Method

Given an input image I and a textual instruction P, our objective is to produce an edited image $\hat{I}$ reflecting precise object-level transformations specified by P. To eliminate manual interaction, we leverage reasoning capabilities of Multimodal Large Language Models (MLLMs).

We propose **POEM** (Precise Object-level Image Editing via MLLMs), a framework that decouples reasoning from generation to enable fine-grained, automated edits.

POEM comprises five stages (Fig. 2): (a) Visual Grounding: an MLLM receives I and P and is prompted to detect all objects in the scene; (b) Detection Refinement: we refine detections into accurate object segmentation masks; (c) Edit Operation Parsing: we use an LLM that is instructed to select the target object and compute the transformation matrix; (d) Transformation: we apply the transformation to the segmented object mask; (e) Edit-Guided Image-to-Image Translation: given the initial input image and the masks of the target object before and after the transformation, we generate the final modified image while preserving spatial and visual coherence.

Visual Grounding. We use an MLLM to analyze I and P via zero-shot prompting, detecting all objects N and returning, for each object $i \in N$, a bounding box b_i, a segmentation point s_i, class c_i, and a unique object ID k_i. The MLLM also generates structured textual descriptions: scene layout (S), object relations (R), background appearance (P_{bg}), and generation intent (P_g). S and R support

Edit Operation Parsing estimating the transformation matrix, while P_{bg} and P_g support the Edit-Guided Image-to-Image Translation maintaining background consistency and apply object-specific edits.

Detection Refinement. Off-the-shelf MLLMs struggle to produce precise object-bounding boxes when for visual grounding [28]. To improve this, we use Grounded-SAM (a combination of Grounding DINO [14] and SAM [20]) as an open-set detector to obtain refined bounding boxes b'_i and segmentation masks m_i for each detected object i.

Edit Operation Parsing. Given a prompt P and a set of refined bounding boxes $B' = b'_i \mid i \in N$, our goal is to estimate the transformation matrix T and the ID k of the target object. However, when provided only with the prompt P, the MLLM struggles to infer T directly due to the lack of explicit scene information. For instance, if `P = "make the cat 100px wide"`, the required transformation depends on the cat's initial dimensions in the image. If the cat is initially 50px wide, the scaling factor should be 2; if it's 25px, the factor should be 4.

To address this, we use a text-based LLM optimized for mathematical reasoning to compute the transformation parameters. This separation allows for a more accurate estimation of scale, rotation, and translation transformations by explicitly incorporating object size information into the reasoning process. We use the input prompt P, the descriptive prompts S and R, and the coordinates of the detections B', and we directly instruct the LLM to predict the unique ID of the target object $i*$ and a 3×3 affine transformation matrix T. To ensure correct parsing, we employ a structured format where LLM matrices and object IDs are enclosed between the unique tokens `<MSTART>`, `<MEND>`, `<ISTART>`, and `<IEND>`. A regex-based parser extracts numerical values enclosed within the matrix tokens, ensuring the retrieval of transformation parameters.

Transformation. We select the segmentation mask m_{i*} corresponding to the selected id $i*$. Then, we perform image wrapping using T on the binary mask m_{i*} to generate the transformed mask $\hat{m}_{i*}$.

Edit Guided Image-to-Image Translation. We use the masks m_{i*} and $\hat{m}_{i*}$ of the target object, and the descriptive prompts P_{bg} and P_g from the first step to perform the image synthesis and generate the final input image $\hat{I}$. We apply these edits during the inference of pre-trained diffusion models without additional training or fine-tuning. Inspired by [35], we perform object-level shape manipulations in the latent space of diffusion models [30]. We use the region of the mask $\hat{m}_{i*}$ to define the area of interest, which is processed through backward diffusion to obtain its latent representation z_{repos}. The region of the initial mask m_{i*} is reinitialized with Gaussian noise $\mathcal{N}(0, I)$, and the new latent is blended into the image latent z as:

$$z_{\text{new}} = z \odot (1 - M_j) + z_{\text{repos}} \odot \hat{M}_j + \mathcal{N}(0, I) \odot M_j. \quad (1)$$

Table 1. **Evaluation on VOCEdits.** Methods are grouped according to different steps of our pipeline, as described in the paper.

Method	Average IoU (%)
Visual Grounding (BBox Prediction vs. GT)	
InternVL-8B [3]	17.4
InternVL-72B [3]	47.1
QwenVL-7B [26]	**55.5**
QwenVL-72B [26]	54.8
Detection Refinement (Mask Prediction vs. GT)	
QwenVL-7B + SAM [20]	27.3
QwenVL-7B + G-SAM [29]	**84.2**
Edit Operation Parsing & Transformation (Transformed Mask vs. GT)	
(QwenVL-7B + G-SAM) + DeepSeek [5]	25.3
(QwenVL-7B + G-SAM) + QwenM [25]	**49.2**
Oracle Mask + DeepSeek [5]	29.5
Oracle Mask + QwenM [25]	**55.6**
Edit Guided Image-to-Image Translation (Detected Mask vs. GT)	
(QwenVL-7B + G-SAM + QwenM) + SLD [35]	**38.4**
(QwenVL-7B + G-SAM + QwenM) + SLD + [24]	37.6
IP2P [2]	34.3
TurboEdit [6]	33.8
LEDITS++ [1]	35.0

A forward diffusion process refines the image, enhancing realism and coherence in edited and surrounding regions.

3. Experiments

Sec. 3.1 introduces VOCEdits, a new dataset designed to rigorously evaluate object-level image editing. Sec. 3.2–3.5 systematically analyze the design choices at each stage of our pipeline. We also provide a comparison between POEM and several state-of-the-art image editing methods [1, 2, 6].

3.1. VOCEdits Dataset

We present VOCEdits, a dataset for evaluating fine-grained object-level image editing involving affine transformations: flip, scale, rotation, translation, and shear. It is built upon PASCAL VOC 2012 [8] for its high-quality instance segmentation masks, enabling precise object-centric evaluation on real-world images. We augment PASCAL VOC images with instructional prompts, ground-truth transformations, and object masks before and after editing. We use images from the PASCAL VOC 2012 trainval segmentation set, containing 2913 images and 6929 object instances. We filter out images with multiple instances of the same class, truncated objects, extreme object sizes, or masks extending beyond image boundaries, resulting in 505 unique images.

To generate human-like edit instructions, we prompt GPT-4o [21] to paraphrase default prompts, producing diverse descriptions. Ground-truth segmentation masks from PASCAL VOC are then transformed using OpenCV for precise computation. Each image in the final set undergoes two

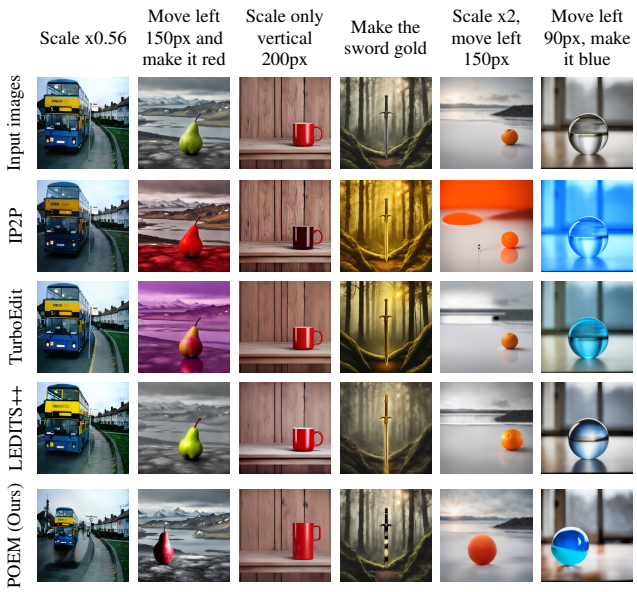

Figure 3. **Qualitative results.** We compare POEM with state-of-the-art image editing models across a diverse set of edit instructions, including geometric transformations (e.g., translation, scaling), appearance changes, and combinations of both.

randomly selected transformations, each with three paraphrased prompts, resulting in 3030 unique samples.

Our pipeline processes all samples excluding images with more than five foreground objects due to limitations in [35] when handling excessive occlusions. After filtering, 193 images and 921 samples remain for evaluation. Tab. 1 summarizes the results.

3.2. Visual Grounding

Evaluation protocol. To assess the quality of the detected bounding box, we compute Intersection over Union (IoU) with the ground truth. If the MLLM fails to detect a bounding box, we fallback to a prediction covering the entire image. For images with multiple objects, we evaluate only the bounding box corresponding to the target object.
Comparison. We compare Qwen2.5-VL [26] and InternVL-2.5 [3] in their 7B/8B and 72B variants. QwenVL-7B yields an average IoU of 55.5%, outperforming InternVL.

3.3. Detection Refinement

Evaluation protocol. We assess the segmentation quality by computing the IoU between the ground truth segmentation mask of the target object and the corresponding detected segmentation masks we obtain after the refinement stage. For images with multiple objects, we evaluate only the segmentation mask corresponding to the target object.
Comparison. We compare Grounded-SAM [29] to SAM2 [20]. Grounded-SAM is prompted with the predicted object class c_i while SAM2 is prompted with the predicted segmentation point s_i. G-SAM (Tab. 1) outperforms SAM2 with average IoU improvement of 56.9%.

3.4. Edit Operation Parsing and Transformation

Evaluation protocol. To assess transformation accuracy, we compute the ground-truth segmentation mask of the target object after applying the ground-truth transformation matrix. We then measure the IoU between this mask and the predicted transformed mask $\hat{m}_{i*}$. This allows us to measure implicitly the error between our predicted transformation matrix T and the ground-truth one.
Comparison. We evaluate Qwen2.5-Math-7B [25], relying on tool integrated reasoning (TOR), and DeepSeek-R1-Distill-Qwen-32B [5], relying only on internal knowledge. Transformations on segmentation maks are performed with OpenCV. We analyze two scenarios: (1) our pipeline's best models and (2) an oracle ground-truth mask, isolating LLM-based reasoning effects. The first measures cumulative error from imperfect segmentation, the second evaluates the transformations independently. QwenMath surpasses DeepSeek by 26.1% in average IoU on oracle masks.

3.5. Edit Guided Image-to-Image Translation

Evaluation protocol. We evaluate editing quality by measuring alignment between the edited image and the edit instruction, rather than relying on image quality metrics like FID. Specifically, we apply Grounded SAM to extract the segmentation mask of the transformed object in the edited image and compute its IoU with the mask obtained by applying the ground-truth transformation.
Comparison. We adopt Stable Diffusion v2.1 [30] as the base model and follow latent-space editing strategies from [35]. We also evaluate on a further refinement step on the edited image with SDXL [24] to enhance visual quality.
Comparison to state-of-the-art. Fig. 3 presents a qualitative comparison between POEM and state-of-the-art models, including IP2P [2], LEDITS++ [1], and TurboEdit [6]. The results demonstrate POEM's more effective editing ability. Tab. 1 provides quantitative results, where POEM achieves a score of 38.4%, outperforming IP2P (34.4%), TurboEdit (33.8%), and LEDITS++ (35.0%) by approximately 3%. These results underscore POEM's superior performance, delivering more precise edits and transformation parameters that more accurately reflect user intent.

4. Conclusion

We presented POEM, a framework that combines MLLMs and diffusion models for precise object-level image editing from natural language instructions. By aligning semantic reasoning with spatial control, POEM enables accurate edits. We also introduced VOCEdits, a benchmark for evaluating such tasks. Experiments show that POEM outperforms prior methods in precision and usability.

References

[1] Manuel Brack et al. LEDITS++: Limitless Image Editing using Text-to-Image Models. In *CVPR*, 2024. 3, 4

[2] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. InstructPix2Pix: Learning to Follow Image Editing Instructions. In *CVPR*, 2023. 1, 3, 4

[3] Zhe Chen et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *CVPR*, 2024. 3, 4

[4] Yutao Cui, Xiaotong Zhao, Guozhen Zhang, Shengming Cao, Kai Ma, and Limin Wang. StableDrag: Stable Dragging for Point-based Image Editing. In *ECCV*, 2024. 1

[5] DeepSeek-AI. DeepSeek-V3 Technical Report. In *arXiv*, 2024. 3, 4

[6] Gilad Deutch, Rinon Gal, Daniel Garibi, Or Patashnik, and Daniel Cohen-Or. Turboedit: Text-based image editing using few-step diffusion models. In *SIGGRAPH Asia*, 2024. 3, 4

[7] Dave Epstein, Allan Jabri, Ben Poole, Alexei A. Efros, and Aleksander Holynski. Diffusion Self-Guidance for Controllable Image Generation. In *NeurIPS*, 2023. 1

[8] Mark Everingham et al. The pascal visual object classes challenge: A retrospective. In *IJCV*, 2015. 2, 3

[9] Weixi Feng et al. LayoutGPT: Compositional Visual Planning and Generation with Large Language Models. In *NeurIPS*, 2023. 2

[10] Yutong Feng, Biao Gong, Di Chen, Yujun Shen, Yu Liu, and Jingren Zhou. Ranni: Taming Text-to-Image Diffusion for Accurate Instruction Following. In *CVPR*, 2024. 2

[11] Hanan Gani, Shariq Farooq Bhat, Muzammal Naseer, Salman Khan, and Peter Wonka. LLM Blueprint: Enabling Text-to-Image Generation with Complex and Detailed Prompts. In *ICLR*, 2024. 1

[12] Daniel Geng and Andrew Owens. Motion Guidance: Diffusion-Based Image Editing with Differentiable Motion Estimators. In *ICLR*, 2024. 1

[13] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-Based Real Image Editing with Diffusion Models. In *CVPR*, 2023. 1

[14] Shilong Liu et al. Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. In *ECCV*, 2024. 3

[15] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. RePaint: Inpainting using Denoising Diffusion Probabilistic Models. In *CVPR*, 2022. 1

[16] Chenlin Meng et al. SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations. In *ICLR*, 2022. 1

[17] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text Inversion for Editing Real Images using Guided Diffusion Models. In *CVPR*, 2023. 1

[18] Chong Mou, Xintao Wang, Jiechong Song, Ying Shan, and Jian Zhang. DiffEditor: Boosting Accuracy and Flexibility on Diffusion-based Image Editing. In *CVPR*, 2024. 1

[19] Alex Nichol et al. GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models. In *ICML*, 2022. 1

[20] Ravi Nikhila et al. SAM 2: Segment Anything in Images and Videos. In *ICLR*, 2024. 3, 4

[21] OpenAI. Gpt-4 technical report. In *arXiv*, 2023. 3

[22] Dong Huk Park et al. Shape-Guided Diffusion with Inside-Outside Attention. In *WACV*, 2024. 1

[23] Yuhan Pei et al. SOWing Information: Cultivating Contextual Coherence with MLLMs in Image Generation. In *arXiv*, 2024. 2

[24] Dustin Podell et al. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. In *ICLR*, 2024. 3, 4

[25] Qwen. Qwen2.5-Math Technical Report: Toward Mathematical Expert Model via Self-Improvement. In *arXiv*, 2024. 3, 4

[26] Qwen. Qwen2.5 Technical Report. In *arXiv*, 2025. 3, 4

[27] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical Text-Conditional Image Generation with CLIP Latents. In *arXiv*, 2022. 1

[28] Hanoona Rasheed et al. Glamm: Pixel grounding large multimodal model. In *CVPR*, 2024. 3

[29] Tianhe Ren et al. Grounded SAM: Assembling Open-World Models for Diverse Visual Tasks. In *ICCV*, 2023. 3, 4

[30] Robin Rombach et al. High-Resolution Image Synthesis with Latent Diffusion Models. In *CVPR*, 2022. 1, 3, 4

[31] Nataniel Ruiz et al. DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation. In *CVPR*, 2023. 1

[32] Andrey Voynov, Kfir Aberman, and Daniel Cohen-Or. Sketch-Guided Text-to-Image Diffusion Models. In *SIGGRAPH*, 2023. 1

[33] Zhenyu Wang, Aoxue Li, Zhenguo Li, and Xihui Liu. GenArtist: Multimodal LLM as an Agent for Unified Image Generation and Editing. In *NeurIPS*, 2024. 2

[34] Cong Wei, Zheyang Xiong, Weiming Ren, Xinrun Du, Ge Zhang, and Wenhu Chen. OmniEdit: Building Image Editing Generalist Models Through Specialist Supervision. In *ICLR*, 2025. 2

[35] Tsung-Han Wu, Long Lian, Joseph E. Gonzalez, Boyi Li, and Trevor Darrell. Self-correcting LLM-controlled Diffusion Models. In *CVPR*, 2024. 1, 3, 4

[36] Shaoan Xie, Zhifei Zhang, Zhe Lin, Tobias Hinz, and Kun Zhang. SmartBrush: Text and Shape Guided Object Inpainting with Diffusion Model. In *CVPR*, 2023. 1

[37] Qifan Yu et al. AnyEdit: Mastering Unified High-Quality Image Editing for Any Idea. In *arXiv*, 2024. 2

[38] Yongsheng Yu, Ziyun Zeng, Hang Hua, Jianlong Fu, and Jiebo Luo. PromptFix: You Prompt and We Fix the Photo. In *NeurIPS*, 2024. 2

[39] Yu Zeng, Zhe Lin, Jianming Zhang, Qing Liu, John Collomosse, Jason Kuen, and Vishal M. Patel. SceneComposer: Any-Level Semantic Image Synthesis. In *CVPR*, 2023. 2

[40] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. MagicBrush: A Manually Annotated Dataset for Instruction-Guided Image Editing. In *NeurIPS*, 2024. 2