SketchAssist: A Practical Assistant for Semantic Edits and Precise Local Redrawing

Supplementary Material

Model	PSNR↑	PSNR-M↑	SSIM↑	SSIM-M↑
SketchEdit [18]	18.49	10.66	0.853	0.601
BrushNet [5]	16.17	7.85	0.798	0.502
MagicQuill [10]	17.78	9.19	0.868	0.583
Ours	**19.03**	**11.41**	**0.872**	**0.688**

Table 1. Additional quantitative results for Line-guided Region Redrawing. PSNR-M and SSIM-M denote PSNR and SSIM computed only within the masked region.

1. Detailed Evaluation Protocol

In this section, we provide further details on the evaluation settings, including the specific prompt configurations used for MLLM-based automated evaluation (VIEScore [6]), supplementary pixel-wise metrics, and the rationale behind the evaluation scope for different experimental tables.

1.1. Supplementary Pixel-wise Evaluation

Complementing the perceptual and semantic evaluations, we additionally provide pixel-wise evaluations (PSNR and SSIM [15]) in this section as supplementary metrics. While pixel-wise metrics are known to correlate poorly with semantic editing quality in sparse sketch domains due to background dominance and sensitivity to spatial misalignments [1, 20], we include them only to offer a reference for the model's precision in low-level reconstruction, rather than as primary indicators of overall performance.

- **Global Metrics (PSNR/SSIM):** These provide a coarse proxy for **structural fidelity in unmasked regions**, loosely reflecting whether the model preserves the original context and blends the edited region without introducing obvious artifacts.
- **Masked-Region Metrics (PSNR-M/SSIM-M):** These offer a limited indication of **control precision**. Since the guidance lines in our test set are derived from the ground truth, higher scores within the masked region may suggest that the model is able to follow the provided guidance and reconstruct geometry consistent with the target, although such pixel-wise metrics remain imperfect for evaluating semantic editing quality.

Our method achieves state-of-the-art results across both dimensions, demonstrating its ability to perform precise local reconstruction while maintaining superior consistency with the global composition.

1.2. VIEScore Prompt Configuration

For the automated evaluation of Instruction-Guided Editing, we utilized the VIEScore [6] protocol with Gemini-2.0-Flash and Qwen3-VL-30B. As dictated by the evaluation protocol, the multimodal model returns a list of two sub-scores for both **Semantic Consistency (SC)** and **Perceptual Quality (PQ)**. To compute the final metric for a single image, we take the minimum of its two sub-scores for both SC and PQ, thereby strictly penalizing any singular failure mode (e.g., severe artifacts or significant over-editing). The **Overall** score per image is then computed as the geometric mean of its final SC and PQ scores ($\sqrt{\text{SC} \times \text{PQ}}$), ensuring that high performance is required across both semantic adherence and visual quality. Finally, the metrics reported in our quantitative tables represent the arithmetic mean of these image-level scores across the entire test set. To ensure reproducibility, we provide the exact system prompts used for this evaluation in Table 2.

2. Comparison of Data Generation Methods

To justify the necessity of our proposed data generation pipeline—which integrates generation with strict post-filtering—we conducted a study to evaluate the quality of the constructed training pairs. We compared the editing pairs produced by our pipeline against those generated by standard paradigms. To ensure a rigorous comparison, we employed state-of-the-art generative backbones (SDXL [11] and FLUX) for these baseline methods. Our goal is to demonstrate that our pipeline provides significantly more accurate and spatially consistent training data, establishing a superior foundation for training the SketchAssist model compared to:

1. **Prompt-to-Prompt Editing** [2, 3]: This method manipulates image content by modifying the text prompt while keeping the model's cross-attention maps aligned with the original image (implemented via SDXL). By adding, removing, or replacing certain words in the prompt, local semantic changes can be introduced without altering unrelated image regions.
2. **Object detection based Local Editing** (adapted from [17]): This approach first detects objects or body parts in the image using *Grounding Dino* [8] and *SAM2* [13]. The detected region is then erased and refilled via an inpainting model. To represent the state-of-the-art, we evaluated two variants of this paradigm: one using *SDXL-Inpainting* and another using *FLUX.1-Fill*.

Table 2. **System Prompts for VIEScore Evaluation.** The detailed instructions and scoring criteria provided to the MLLM.

Metric	System Prompt / Instruction
Perceptual Quality (PQ)	RULES: The image is an AI-generated image. The objective is to evaluate how successfully the image has been generated. **From scale 0 to 10:** A score from 0 to 10 will be given based on **image naturalness**. (0 indicates that the scene in the image does not look natural at all or gives an unnatural feeling such as wrong sense of distance, or wrong shadow, or wrong lighting. 10 indicates that the image looks natural.) A second score from 0 to 10 will rate the **image artifacts**. (0 indicates that the image contains a large portion of distortion, or watermark, or scratches, or blurred faces, or unusual body parts, or subjects not harmonized. 10 indicates the image has no artifacts.) Put the score in a list such that output score = [naturalness, artifacts]
Semantic Consistency (SC)	**From scale 0 to 10:** A score from 0 to 10 will be given based on the **success of the editing**. (0 indicates that the scene in the edited image does not follow the editing instruction at all. 10 indicates that the scene in the edited image follows the editing instruction text perfectly.) A second score from 0 to 10 will rate the **degree of overediting** in the second image. (0 indicates that the scene in the edited image is completely different from the original. 10 indicates that the edited image can be recognized as a minimally edited yet effective version of original.) Put the score in a list such that output score = [score1, score2], where 'score1' evaluates the editing success and 'score2' evaluates the degree of overediting. Editing instruction: `<instruction>`

Prompt-to-Prompt: We constructed a total of 101 editing pairs using the Prompt-to-Prompt method [3], following the data generation paradigm widely utilized by InstructPix2Pix [2]. Editing instructions were randomly sampled with lengths between 1 and 5 operations, enabled by Prompt-to-Prompt's ability to sequentially modify multiple words or phrases within a single prompt—effectively simulating multi-step edits in one generation. Participants rated each pair as success or failure. As shown in Table 3, our method achieves a substantially higher average success rate (87.88%) compared to Prompt-to-Prompt (46.89%) under the same evaluation protocol. Fig. 1 illustrates typical failure cases of Prompt-to-Prompt. While Prompt-to-Prompt supports compositional edits by adding, replacing, or removing attributes in the prompt, it often fails to maintain consistent human poses and scene composition across edits. In the examples shown, although the target attributes are correctly modified, the resulting images exhibit noticeable changes in body posture or spatial arrangement, breaking the intended continuity between the original and edited versions.

Method	Average Success Rate (%)
P2P	46.89
Ours	87.88

Table 3. User study success rate (%) for Prompt-to-Prompt editing pairs compared with our method.

Object Detection + Inpainting: We constructed a total of 64 editing items using an object-detection-based local editing approach adapted from [17]. *Grounding Dino* [8] and *SAM2* [13] were used to detect target regions, which were then erased or replaced via inpainting. Although the detection-based pipeline can be extended to multi-step edits by repeatedly detecting and erasing different regions, such repetition increases complexity and may introduce structural inconsistencies. Therefore, we restrict our comparison to the single-step case.

The quantitative results in Table 4 show that our method achieves a significantly higher average success rate (91.50%) compared to SDXL-inpainting (16.28%) and FLUX.1-fill (35.31%).

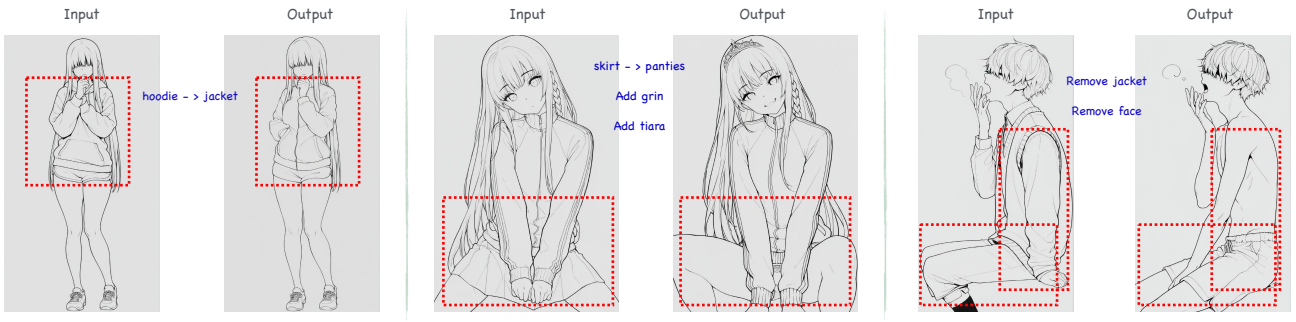

Figure 1. Failures of generated edit pairs using Prompt-to-Prompt [2, 3]. While Prompt-to-Prompt supports compositional edits by adding, replacing, or removing attributes in the prompt, it often fails to preserve consistent human poses and scene composition across edits. The examples illustrate typical cases where pose or layout consistency is lost despite correct attribute modifications. Regions enclosed by red dashed boxes indicate areas where the human pose has changed between the source and edited image.

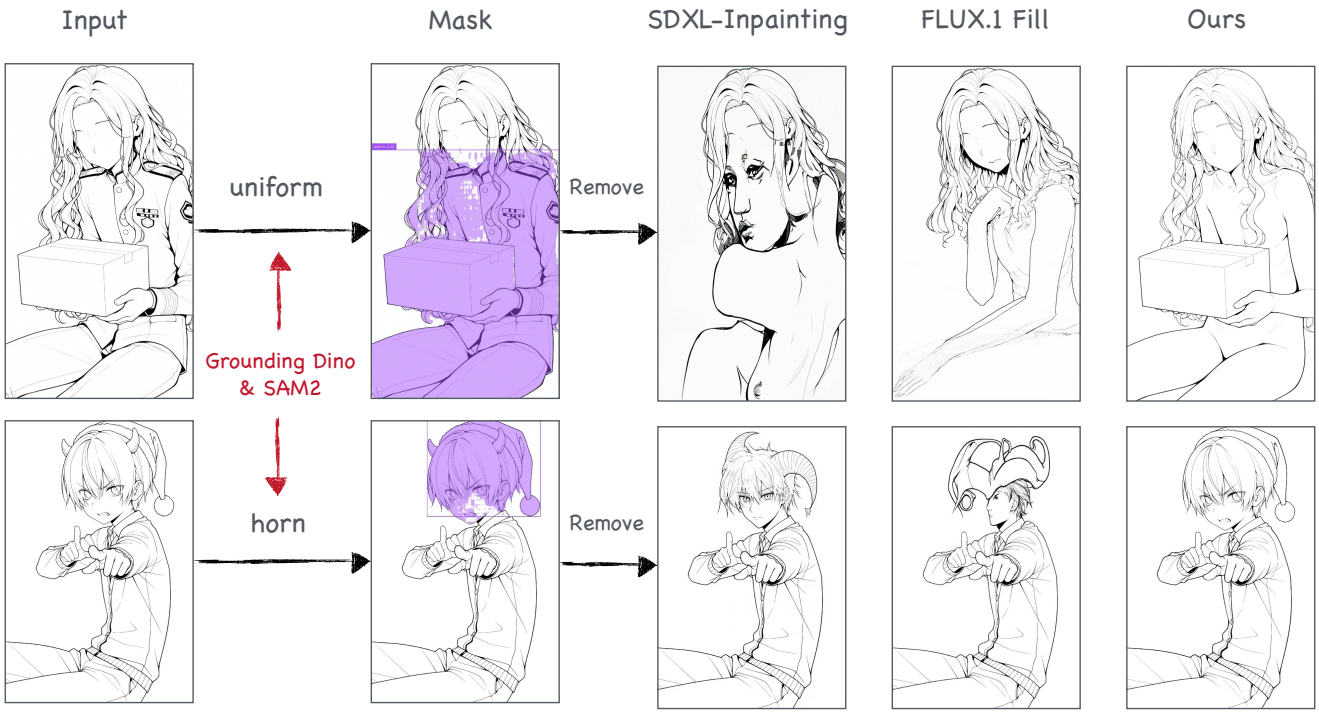

Figure 2. Qualitative comparisons between our method and an object-detection-based inpainting approach (SDXL-Inpainting and FLUX.1-Fill), where both baselines generate content based on segmentation masks, while our method synthesizes results via an addition sequence. In this setting, the detected target region is used to construct a "remove" edit pair by erasing the region and filling it via inpainting. Two common failure cases of the object-detection-based pipeline are illustrated: Top — when the segmented region is overly large (e.g., covering clothing), the inpainting process fails to recover the original human pose; Bottom — under coarse sketch guidance, Grounding Dino and SAM2 produces inaccurate segmentation masks, leading to incorrect inpainting regions.

Beyond the quantitative advantage, Fig. 2 presents qualitative comparisons between our method and an object-detection-based approach. In this setting, the detected target region is used to construct a "remove" edit pair by erasing the region and filling it via inpainting. Two typical failure cases of object-detection-based inpainting are shown: (1) when the segmented region is too large (e.g., covering clothing), the inpainting process fails to restore the original human pose; and (2) when guided by a coarse sketch, SAM2 produces inaccurate segmentation masks, resulting in incorrect inpainting regions.

Method	Average Success Rate (%)
SDXL-inpainting	16.28
FLUX.1-fill	35.31
Ours	91.50

Table 4. User study average success rate (%) for Object Detection + Inpainting editing pairs, compared with our method.

3. More Details of Data Generation

3.1. Instruction-guided Editing Data

Attribute Addition Sequence: We generate **10,000** distinct base sketches using anime-style *Stable Diffusion XL (SDXL)* models. Each base sketch is then extended into multiple distinct atomic addition sequences. Each sequence begins with the initial sketch state and is expanded through multiple successive attribute-adding chains. At each step of the sequence, the sketch generated in the preceding step is fed into a *lineart-conditioned ControlNet [19]* as structural guidance, thereby preserving the underlying line structure while allowing semantic content to evolve according to the specified attributes.

Filtering Process: We apply the *HumanArt* [4] model in conjunction with CLIP-based [12] similarity scoring to evaluate structural changes between source–target pairs. Pairs exhibiting significant composition, pose, or structural changes are discarded. Furthermore, we utilize the WD14 Tagger [14] to confirm the precise presence or absence of target attributes, and employ Qwen-VL to validate the overall alignment with the editing instruction. Only pairs that pass both the structural-consistency check and the attribute-correctness verification are retained, resulting in a curated subset of roughly **100,000** images.

Style Diversification: To expand stylistic coverage, we trained an attribute-removal model based on FLUX.1-Kontext [7] using our synthetic line art sequences and applied it to approximately 4,000 diverse real-world sketches. Crucially, the remover generalizes effectively to these unseen styles by leveraging the backbone's in-context generation design, where texture synthesis is driven by surrounding pixel cues rather than solely on fine-tuned weights. This stands in contrast to the *addition* task, which requires hallucinating new content and is prone to overfitting to the synthetic line art domain of the source data. Thus, these generated multi-style pairs act as essential **style regularizers**, preventing domain overfitting and enabling SketchAssist to support diverse artistic renditions.

Simulated Guidance Lines: We first extract line art structure from a diverse collection of source sketches using the anime lineart preprocessor provided in ControlNet [19], capturing the essential contours of the subject. Based on these structural maps, we employ *Stable Diffusion XL (SDXL)* combined with a *line art-conditioned ControlNet* to synthesize the final guidance images. To emulate the imperfect quality of real hand-drawn hints, we incorporate prompt modifiers such as "bad quality" and "sketch", intentionally producing output lines with lower visual fidelity and jittery strokes. Crucially, this synthetic data construction introduces a domain gap between the rough guidance lines and the precise geometry of the original sketch. This forces the model to learn semantic correspondence rather than rigid pixel alignment, enabling it to robustly correct user stroke errors and spatial misalignments during inference.

Mask Generation: As described in the main text, body-part masks are detected using *HumanArt* [4], and object masks are obtained using *Grounding Dino* and *SAM2*.

4. Quantitative Verification of Input Encoding and Mask Fidelity

Our framework employs a unified input strategy where the sketch (R), mask (G), and guidance (B) are encoded via the repurposed RGB channels of the pre-trained VAE. While this design significantly streamlines the architecture, the fidelity of the **mask encoding** is paramount. In the context of local editing, the binary mask acts as the definitive hard constraint that delimits the editing scope. Therefore, preserving the precise spatial geometry of the mask during VAE encoding/decoding is the absolute prerequisite for successful instruction adherence and structural preservation.

To verify that our input strategy meets this critical requirement, we conducted a rigorous quantitative analysis on the mask regions of our entire test set ($N = 200$). Specifically, we passed the composite inputs through the VAE, applied a standard binary threshold ($\tau = 0.5$) to the reconstructed G-channel, and computed the **Intersection-over-Union (IoU)** against the ground-truth masks.

Results. As shown in Table 5, the model achieves a near-perfect **Mean IoU of 0.9995**. Even in the worst-case scenario, the **Minimum IoU** remains at **0.9982**. These results confirm that despite the domain gap of the pre-trained VAE, the spatial definition of the editable region is preserved with effectively zero loss. This ensures that the DiT backbone receives accurate localization guidance, establishing a solid foundation for precise local redrawing.

Metric	Mean	Min	Std
IoU	**0.9995**	0.9982	0.0003

Table 5. **Quantitative analysis of Mask Reconstruction Quality.** Evaluated on the test set ($N = 200$). The near-perfect IoU scores demonstrate that the proposed encoding strategy incurs negligible loss in spatial localization accuracy, ensuring precise editing control.

5. Training Configuration

Cross-Sequence Sampling During training, we employ a cross-sequence sampling strategy, in which edit pairs are constructed by randomly composing between **1–5** atomic operations from a predefined set (e.g., *add*, *remove*, *replace*). Accordingly, the resulting edit pairs span an edit distance range of $D = 1$ to 5, as defined in the main paper. This design enables the model to learn from both simple single-operation edits and more complex multi-operation transformations, while remaining compatible with the unified framework.

Task-guided Mixture-of-Experts (T-MoE) Architecture The *Shared LoRA* module is configured with a fixed rank of 24, and four *expert LoRA* modules are instantiated with a rank of 12 each. The router selects the *top-2* experts for each instruction based on a learned gating mechanism conditioned on concatenated text and visual features. This design allows the model to distinguish between text-based editing mode and line-based local redrawing mode, while sharing low-level style and structure representations. For fairness in the ablation study, the non-MoE baseline is configured with a single LoRA module of rank 48, such that the total number of activated LoRA parameters matches that of the T-MoE configuration.

6. More Qualitative Comparison

We provide more qualitative comparisons of instruction-guided editing task with ICEdit [21], Qwen-Image-Edit [16], FLUX.1 Kontext [7], and Step1X-Edit [9] in Fig. 3 and Fig. 4. In these experiments, we deliberately adopt complex and multi-faceted editing instructions, requiring simultaneous modifications to multiple attributes, objects, or regions. Our method demonstrates a strong ability to accurately follow these instructions while preserving the original scene composition and human pose, without introducing unintended structural changes. This highlights both the advantage of our curated data construction and the efficacy of our Cross-Sequence Sampling strategy in handling multi-instruction edits within a single generation.

We further provide qualitative comparisons on a line-guided region redrawing task—where the target mask is specified and structural guidance is provided via line drawings—with SketchEdit [18], BrushNet [5], and MagicQuill [10] in Fig. 5. SketchEdit is designed for local sketch-to-image translation and therefore cannot operate directly on complete sketch images. BrushNet and MagicQuill can perform local redrawing on sketch images, but in our evaluation they fail to reliably follow the provided guidance lines, producing structures misaligned with the intended line geometry. Our method, by comparison, redraws the masked region in strict accordance with the guidance lines while maintaining high stylistic fidelity to the original image, ensuring structural accuracy and visual coherence.

References

[1] Hmrishav Bandyopadhyay, Ayan Kumar Bhunia, Pinaki Nath Chowdhury, Aneeshan Sain, Tao Xiang, Timothy Hospedales, and Yi-Zhe Song. Sketchinr: A first look into sketches as implicit neural representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12565–12574, 2024. 1

[2] 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. 1, 2, 3

[3] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 1, 2, 3

[4] Xuan Ju, Ailing Zeng, Jianan Wang, Qiang Xu, and Lei Zhang. Human-art: A versatile human-centric dataset bridging natural and artificial scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 618–629, 2023. 4

[5] Xuan Ju, Xian Liu, Xintao Wang, Yuxuan Bian, Ying Shan, and Qiang Xu. Brushnet: A plug-and-play image inpainting model with decomposed dual-branch diffusion. In *European Conference on Computer Vision*, pages 150–168. Springer, 2024. 1, 5

[6] Max Ku, Dongfu Jiang, Cong Wei, Xiang Yue, and Wenhu Chen. Viescore: Towards explainable metrics for conditional image synthesis evaluation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12268–12290, 2024. 1

[7] Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025. 4, 5

[8] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *Euro-*

Figure 3. Qualitative comparisons of instruction-based editing. The text beneath each example corresponds to the editing instruction applied.

Figure 4. Qualitative comparisons of instruction-based editing. The text beneath each example corresponds to the editing instruction applied.

pean conference on computer vision, pages 38–55. Springer, 2024. 1, 2

[9] Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, et al. Step1x-edit: A practical framework for general image editing. *arXiv preprint arXiv:2504.17761*, 2025. 5

[10] Zichen Liu, Yue Yu, Hao Ouyang, Qiuyu Wang, Ka Leong Cheng, Wen Wang, Zhiheng Liu, Qifeng Chen, and Yujun

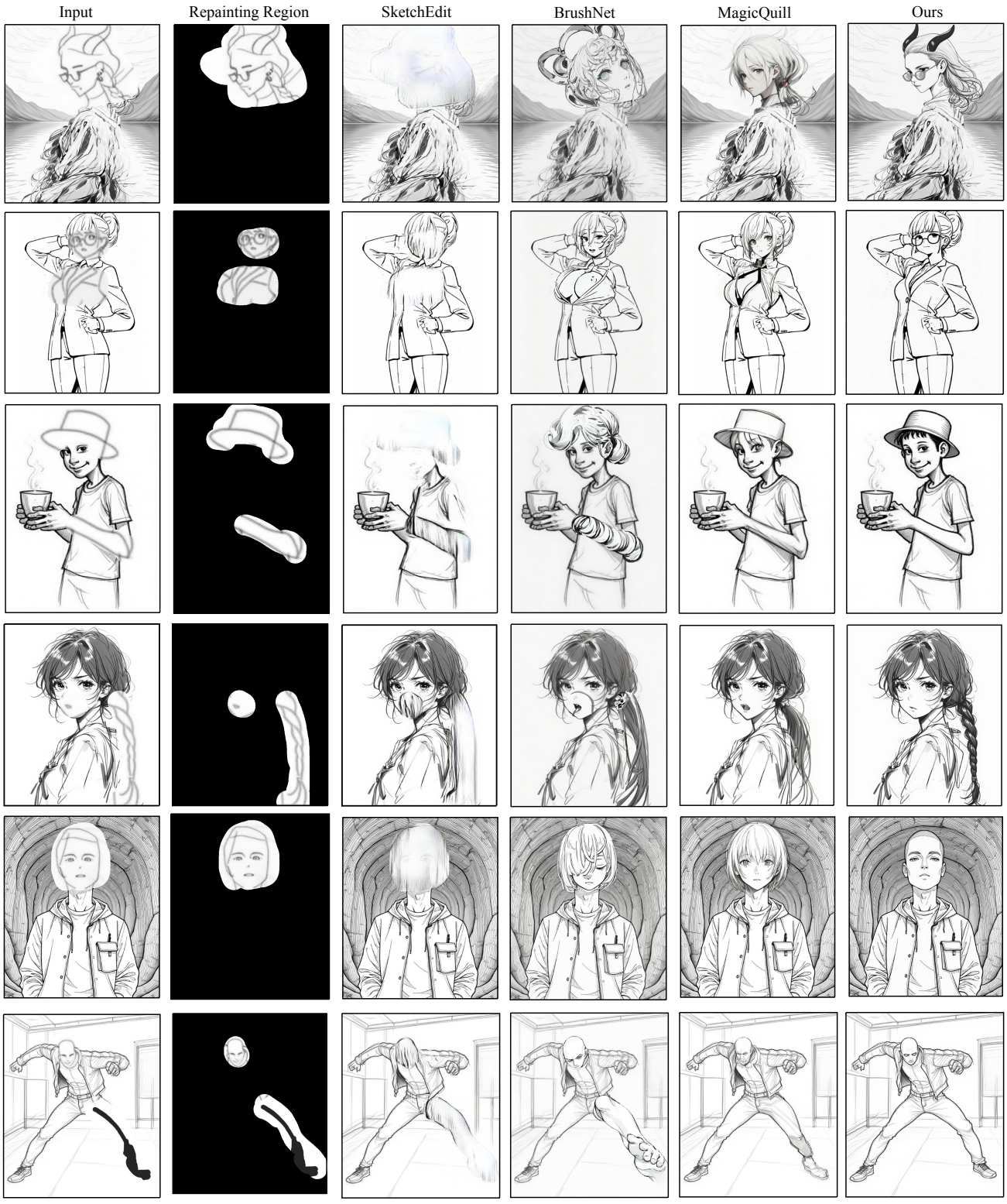

Figure 5. Qualitative comparisons of Line-guided Redrawing

Shen. Magicquill: An intelligent interactive image editing system. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13072–13082, 2025. 1, 5

[11] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 1

[12] 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. 4

[13] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 1, 2

[14] SmilingWolf. Wd 14 tagger v3 (eva02 large). https://huggingface.co/SmilingWolf/wd-eva02-large-tagger-v3, 2024. 4

[15] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 1

[16] Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025. 5

[17] Qifan Yu, Wei Chow, Zhongqi Yue, Kaihang Pan, Yang Wu, Xiaoyang Wan, Juncheng Li, Siliang Tang, Hanwang Zhang, and Yueting Zhuang. Anyedit: Mastering unified high-quality image editing for any idea. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26125–26135, 2025. 1, 2

[18] Yu Zeng, Zhe Lin, and Vishal M Patel. Sketchedit: Mask-free local image manipulation with partial sketches. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5951–5961, 2022. 1, 5

[19] 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 (ICCV)*, pages 3836–3847, 2023. 4

[20] 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. 1

[21] Zechuan Zhang, Ji Xie, Yu Lu, Zongxin Yang, and Yi Yang. Enabling instructional image editing with in-context generation in large scale diffusion transformer. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*. 5