RETHINKING SCRIBBLE-GUIDED IMAGE EDITING: A PROGRESSIVE SUPERVISION CURRICULUM FOR COHERENT MULTI-TASK EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

Scribble-guided image editing combines text with coarse strokes to specify both what to edit and where. However, current image editors often fail to follow scribble instructions, especially for multiple requests. We diagnose two gaps in the required training supervision. First, precise supervision and visually coherent supervision play complementary roles: the former supports reliable local editing, while the latter improves visual coherence with the source image. Second, exposure to multiple scribbles alone is insufficient for multi-task editing; complete multi-task supervision is needed for reliable binding between scribbles, instructions, and edits.

Based on these findings, we propose a *Progressive Supervision Curriculum*. Layered Synthetic Data provide precise supervision for single-task edits. Multi-Task Mosaicking extends these samples into complete multi-task tuples. Visually Coherent Synthetic Data provide supervision for visual coherence. Stage I learns single-task editing and multi-task binding from Layered Synthetic Data and mosaicked tuples. Stage II refines visual coherence with Visually Coherent Synthetic Data. We render scribbles directly into the input image and train only a LoRA adapter. On VIBE, our method achieves the best overall single-task result and reaches performance comparable to closed-source models on multi-task editing. We will release our dataset and model.

1 INTRODUCTION

Text-guided image editing has made strong progress in recent years (Shuai et al., 2024; Brooks et al., 2023), but text alone can leave the edit location underspecified. As illustrated in Figure 1, users may need a long spatial description to identify a particular target. Scribble-guided image editing instead lets users mark that target with a few coarse strokes, providing concise, localized guidance. Current models nevertheless struggle to execute scribble instructions reliably across single-task operations and multiple simultaneous operations; VIBE (Zhang et al., 2026a) exposes this gap most clearly for open-source models. The main challenge is to obtain training supervision that jointly aligns a source image, its edited counterpart, a localized scribble, and a matching instruction, while covering the capabilities required at inference time.

To diagnose these supervision gaps, we conduct two controlled comparisons. We first examine a simple, scalable form of precise supervision and a more costly form of visually coherent supervision to determine how they can be used efficiently. We then test whether multi-scribble visual exposure is sufficient for multi-task editing. Section 3 shows that precise supervision and visually coherent supervision provide complementary signals, whereas multi-scribble visual exposure alone is insufficient. These findings motivate a *Progressive Supervision Curriculum*: precise single-task supervision establishes single-task editing, complete multi-task supervision extends it to multi-task binding, and visually coherent supervision refines visual coherence with the source image. Layered Synthetic Data, Multi-Task Mosaicking, and Visually Coherent Synthetic Data instantiate these three forms of supervision, respectively.

Following the curriculum-learning principle of progressing from simpler, scalable supervision to more demanding supervision (Bengio et al., 2009), we organize training in two stages. **Stage I: Single-Task Editing and Multi-Task Binding Acquisition** learns coverage of the target edit operations and

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

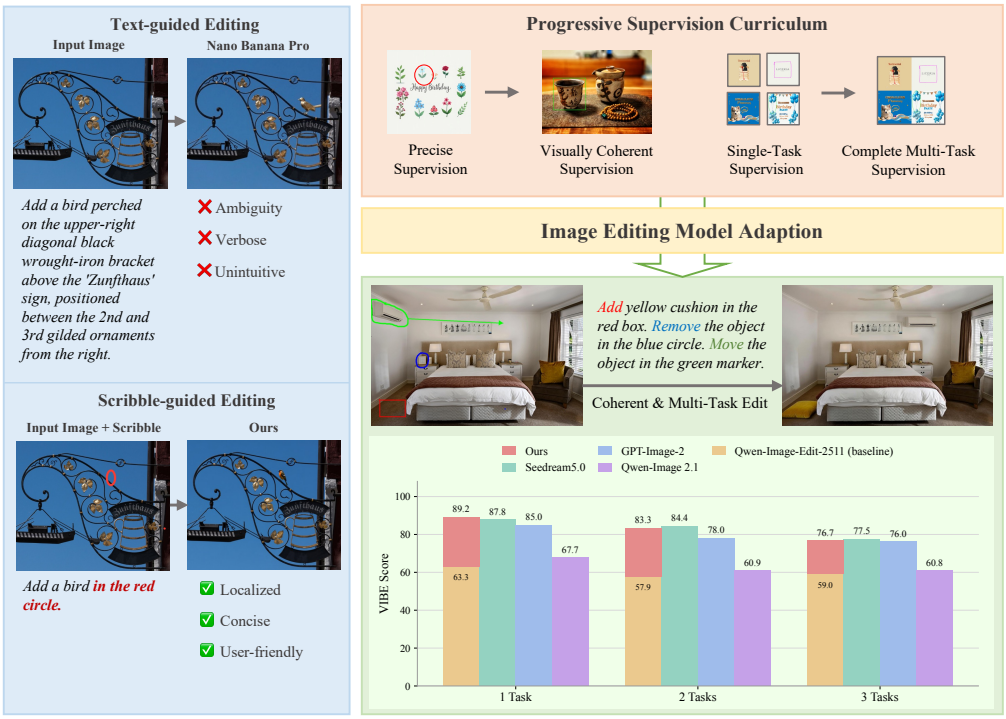


Figure 1: Overview. Text instructions can be ambiguous, verbose, and unintuitive for local edits, whereas scribbles provide localized, concise, and user-friendly guidance. Our Progressive Supervision Curriculum builds on precise supervision and extends it with visually coherent and complete multi-task supervision. It adapts an image editing model for coherent multi-task edits and achieves the best overall single-task result on VIBE, with multi-task performance comparable to closed-source models.

multi-task binding from Layered Synthetic Data and complete multi-task tuples. **Stage II: Visual Coherence Refinement** then fine-tunes on Visually Coherent Synthetic Data to improve visual coherence with the source image. Within Stage I, Multi-Task Mosaicking augments supervision for multi-task binding. Section 4 details the matched constructions and training procedure.

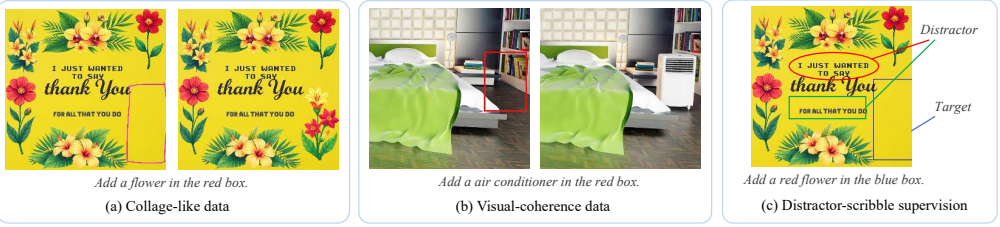
On the VIBE Deictic-Level benchmark, the curriculum raises Qwen-Image-Edit-2511 from a 1-Task average of 63.26 to 89.23, achieving the best overall single-task result. On multi-task editing, it reaches performance comparable to closed-source models.

Our main contributions are as follows:

- We diagnose the supervision required for scribble-guided editing: precise single-task supervision for reliable local editing, visually coherent supervision for visual coherence, and complete multi-task supervision for multi-task binding.
- Guided by these diagnoses, we introduce a Progressive Supervision Curriculum with Layered Synthetic Data, Multi-Task Mosaicking, and Visually Coherent Synthetic Data to instantiate the three supervision forms.
- By adapting our approach to an open-source image editing model, we achieves the best overall single-task result on VIBE and reaches performance comparable to closed-source models on multi-task editing.

2 Related Work

Text-To-Image generation. Recent image generation has shifted toward scalable DiT/MM-DiT backbones and flow-based training objectives (Peebles & Xie, 2023; Lipman et al., 2023; Esser et al., 2024). Modern systems increasingly unify generation and editing, supporting text-to-image synthesis, image-to-image transformation, reference conditioning, composition, and text rendering (Wu et al.,



(a) Collage-like data (b) Visual-coherence data (c) Distractor-scribble supervision

Figure 2: (a) Collage-like data pair a single-task instruction and scribble with an exact target, but may lack visual integration. (b) Visual-coherence data better preserve consistency with the source image. (c) Distractor-scribble supervision adds instruction-unmatched scribbles to a single-task example.

2025; Black Forest Labs, 2025a; ByteDance Seed Team, 2025; Google DeepMind, 2025). Controllable generation further introduces spatial conditions such as boxes, edges, depth, segmentation maps, poses, and sketches through adapters or grounding modules (Zhang et al., 2023; Mou et al., 2024; Li et al., 2023; Zhao et al., 2023), motivating lightweight spatial interaction for editing.

Image-To-Image editing. Image editing transforms a source image under a user's instruction while preserving irrelevant content. Text-guided methods are convenient (Hertz et al., 2022; Mokady et al., 2023; Kawar et al., 2023; Brooks et al., 2023; Shuai et al., 2024), but text alone often underspecifies the target region or object identity. Some methods improve localization through masks, sketches, or handle-target points (Avrahami et al., 2021; Couairon et al., 2022; Mao et al., 2023; Zhang et al., 2023; Mou et al., 2024; Shi et al., 2024; Chen et al., 2024), yet their input modes and supported operations are more rigid than scribbles. Scribble-guided methods use coarse visual marks as lightweight spatial cues (Xia et al., 2025; Zhang et al., 2026b; Qu et al., 2025), but most prior work remains centered on single-task edits. VIBE (Zhang et al., 2026a) further reveals substantial degradation in multi-task scribble-guided editing and large gaps across open-source and closed-source models.

Editing data construction. Large-scale editing datasets such as UltraEdit (Zhao et al., 2024) and AnyEdit (Yu et al., 2024) leverage VLM/LLM-assisted pipelines for scalable data construction. While sharing this goal, our pipeline is specifically designed for scribble-guided editing, exploiting its task-specific spatial characteristics to efficiently construct aligned edit–scribble–instruction supervision.

3 Diagnosing the Supervision Gaps in Scribble-Guided Editing

To understand why scribble-guided editing remains difficult, we use two controlled comparisons to diagnose gaps in its training supervision. The first examines supervision for single-task edits and visual coherence with the source image, while the second examines supervision for binding multiple scribbles, instructions, and edits. All comparisons use Qwen-Image-Edit-2511 with the same LoRA configuration and evaluation protocol; only the training supervision changes.

3.1 From Precise Supervision to Visual Coherence

We begin with two natural forms of supervision for single-task editing. We refer to their data characteristics as *collage-like data* and *visual-coherence data*, illustrated in Figure 2 (a) and (b), respectively. Collage-like data present edits as collage-like compositions, yielding exact edited targets and scribbles for single-task edits. Their edited regions have precise correspondence to the requested operation, but need not be visually integrated with the surrounding scene. This simple form of supervision can be constructed automatically at scale. In contrast, visual-coherence data better preserve the source image's lighting, perspective, texture, and object–background blending. Such supervision is more costly to obtain at scale, but provides the signal needed for visual coherence. Comparing these forms therefore helps determine how to learn the required capabilities efficiently.

We compare equal-size 4.5K subsets to examine how these two forms of supervision support the progression from single-task editing to visual coherence. Table 1 reports higher Instruction Adherence for collage-like data, alongside higher Visual Coherence for visual-coherence data. The results indicate complementary roles rather than a uniformly superior form of supervision: The

Table 1: VIBE scores for models trained on equal-size subsets of the two supervision sources.

Training data	Instruction Adherence	Visual Coherence
Collage-like data	**88.59**	89.28
Visual-coherence data	87.16	**90.25**

simpler precise supervision makes instruction adherence easier to learn, whereas visual-coherence supervision provides the visual-coherence signal needed to refine edits with the source image. This complementarity motivates one branch of our progressive curriculum: Scalable precise supervision first establishes broad coverage and reliable localized editing. Visually coherent supervision then refines visual integration with the source image.

3.2 From Single-Task Supervision to Multi-Task Binding

Single-task supervision supports individual edits, but performance degrades when multiple scribble requests are combined: the model may omit an instruction or bind an edit to the wrong scribble. One explanation is that it lacks exposure to multiple scribbles. We test this hypothesis by augmenting single-task examples with instruction-unmatched distractor scribbles, as illustrated in Figure 2 (c). This control adds multi-scribble visual exposure while retaining a single scribble–instruction–edit relation. We compare it with the original single-task supervision and complete multi-task supervision, which jointly specifies a multi-scribble image, multi-instruction prompt, and multi-edit target. All three supervision formats use equal-size training sets. Figure 3 shows that distractor scribbles yield only limited gains on 2- and 3-Task editing, whereas complete multi-task supervision substantially improves both. Its small 1-Task decrease may reflect the greater difficulty of learning from complete multi-task supervision. These results show that multi-scribble visual exposure alone is insufficient; reliable multi-task binding requires complete multi-task supervision.

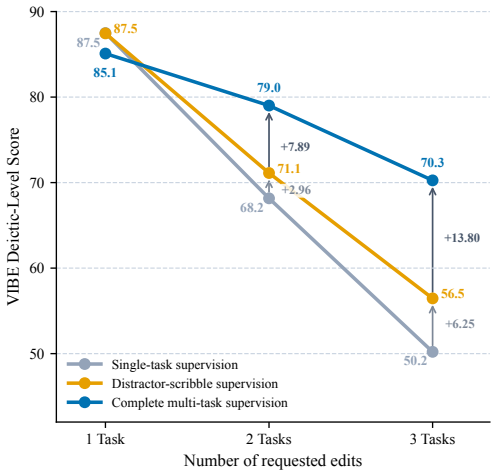


Figure 3: Performance across 1-, 2-, and 3-Task settings under different supervision formats.

4 Progressive Supervision Curriculum

4.1 Problem Formulation

Given an input image I, user scribbles S and a textual instruction T, the model produces an edited image $\tilde{I}$ that satisfies the edit intent specified by S and T while preserving image content not implicated by that intent. We render S directly into the input image so that the interface $(I \oplus S, T) \rightarrow \tilde{I}$ matches that of standard image editing models.

4.2 Progressive Supervision Construction

For the supervision forms identified in Section 3, we develop corresponding data construction approaches. The two supervision forms for single-task editing differ in both signal and construction cost. Layered Synthetic Data provide scalable precise supervision and coverage of the target edit operations. Visually Coherent Synthetic Data provide the more costly supervision needed to refine visual coherence with the source image. Building on the precise supervision, Multi-Task Mosaicking constructs complete multi-task tuples from single-task samples for multi-task binding. Figure 4 summarizes these constructions.

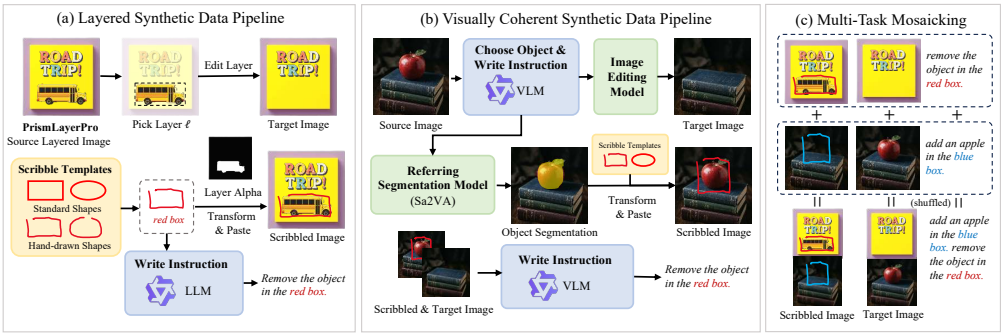


Figure 4: Overview of the Progressive Supervision Curriculum. Layered Synthetic Data establish precise single-task supervision; Multi-Task Mosaicking extends it to complete multi-task supervision; and Visually Coherent Synthetic Data refine visual coherence with the source image.

Layered Synthetic Data. The first diagnosis calls for precise supervision at a scale that covers diverse edit operations. To obtain it, we require an image source where single-object edits can be generated without per-example masks or learned inpainting. PrismLayersPro (Chen et al., 2025) provides such a source: its multi-layer RGBA compositions allow each single-object edit to be implemented by re-rendering the underlying layer stack. We use this source to construct Layered Synthetic Data. As illustrated in Figure 4 (a), the construction proceeds in three steps. **(1) Layer-level editing.** Given a source image I, we select a target layer ℓ and apply the task-specific edit to obtain $\tilde{I}$. **(2) Scribble synthesis.** We sample a template from a library that mixes geometric primitives with hand-drawn presets, transform it with the bounding box or alpha-matte contour of ℓ, and randomize its color, thickness, and placement to produce a scribble on I. **(3) Instruction generation.** An LLM is prompted with the task type, scribble type, and scribble color to write the matching instruction. Starting from 20K layered source images, this process yields 180K Layered Synthetic Data samples.

Visually Coherent Synthetic Data. To obtain visually coherent supervision, we leverage the ability of text-guided image editing models to generate edits that are visually coherent with their source images. We propose a pipeline that turns these edits into scribble-guided supervision, thereby constructing Visually Coherent Synthetic Data. As shown in Figure 4 (b), given an input image I and a target task, a VLM proposes a candidate object and coarse instruction T; an image editor (e.g., Nano Banana Pro (Google DeepMind, 2025)) produces the target $\tilde{I}$; a referring segmentation model (e.g., Sa2VA (Yuan et al., 2025)) localizes the affected region for scribble synthesis; and a VLM writes the final scribble-aware instruction. After human curation, we retain 3.8K automatically constructed samples. For challenging image-editing tasks such as translation, we supply an additional 700 samples with human-drawn scribbles. Together, these samples yield 4.5K Visually Coherent Synthetic Data samples for visual coherence refinement.

Multi-Task Mosaicking. The second diagnosis shows that multi-scribble visual exposure alone does not provide the complete multi-task supervision required for multi-task binding. This supervision must jointly specify a multi-scribble image, multi-instruction prompt, and multi-edit target. Directly synthesizing multiple independent edits within a single image is difficult because the edits can overlap, interfere visually, or disrupt correspondence among each scribble, instruction, and edited target. We therefore use *Multi-Task Mosaicking* to construct complete multi-task tuples from single-task samples. As illustrated in Figure 4 (c), given $k \in \{2, 4\}$ single-task samples, we mosaic their inputs into a 1×2, 2×1, or 2×2 grid, arrange the corresponding targets in the same layout, and concatenate the k instructions in random order. Since each source is a self-consistent scribble–instruction–edited-target triple, the composite explicitly pairs all three multi-task elements. The k sources use mutually distinct scribble colors to eliminate binding ambiguity.

4.3 Curriculum Training

Precise supervision and visually coherent supervision differ substantially in scalability. A one-stage mixture would require choosing a fixed ratio between readily scalable precise supervision and more

difficult-to-scale visually coherent supervision. We therefore follow the curriculum-learning principle of progressing from simpler, scalable supervision to more demanding supervision.

Stage I: Single-Task Editing and Multi-Task Binding Acquisition. We train on the 180K Layered Synthetic Data samples, mixing single-task and mosaicked multi-task samples at a 4:1 ratio. This stage covers the four target edit operations, establishes single-task editing, and exposes the model to complete multi-task tuples for multi-task binding. Because Layered Synthetic Data provide precise edit-region masks, we additionally use edit-region reweighting to emphasize the edited region, which is detailed in Section 5.1.

Stage II: Visual Coherence Refinement. We fine-tune the Stage-I model on 4.5K Visually Coherent Synthetic Data samples. This stage refines visual coherence with the source image through local blending, lighting, shadow, perspective, texture, and style consistency.

5 Experiments

5.1 Experimental Setup

Backbone and interface. We adopt Qwen-Image-Edit-2511 (Wu et al., 2025), an open-source image editing model. As described in Section 4.1, we paste the scribble S directly onto the input image, preserving the backbone's native image-text interface and supporting transfer to standard image editing models. We fine-tune with a standard LoRA adapter (Hu et al., 2021) of rank 32 and a learning rate of 10^{-4}.

Curriculum training. Following Section 4.3, **Stage I: Single-Task Editing and Multi-Task Binding Acquisition** trains on 180K Layered Synthetic Data samples, mixing single-task and mosaicked multi-task samples at a 4:1 ratio. **Stage II: Visual Coherence Refinement** fine-tunes the resulting model on 4.5K Visually Coherent Synthetic Data samples. Detailed training schedules and compute resources are reported in Appendix C.

Edit-region weighting. Because each layered synthetic edit is produced by a precise layer operation, its edit-region mask is directly available: thresholding the per-pixel difference $|I - \tilde{I}|$ and downsampling it to the latent resolution yields the mask without an additional annotation or segmentation step. We use this spatial prior during Stage I by adding a mask-weighted flow-matching term to the standard whole-image objective, with $\lambda = 0.1$ so that the two terms have comparable magnitude. This shifts optimization from the unchanged background toward the user-marked region. We do not use it for Visually Coherent Synthetic Data because image editing can induce diffuse changes in shadows, illumination, and local blending without a well-defined boundary.

5.2 Evaluation Protocol

We evaluate scribble-guided image editing using the Deictic Level of the VIBE benchmark (Zhang et al., 2026a). The evaluation covers four single-task operations: Addition (AD), Removal (RM), Replacement (RP), and Translation (TR), as well as multi-task settings involving 2 Tasks and 3 Tasks. Following the standard VIBE protocol, we employ its default LMM-as-a-judge evaluator (GPT-5.1) and report the average score over three independent runs. We compare our final model against representative closed-source models (Nano Banana Pro (Google DeepMind, 2025), Qwen-Image-2.0 Pro (Wu et al., 2025), Wan 2.7 Pro (Mao et al., 2026), Seedream 4.5 (ByteDance Seed Team, 2025), Seedream 5.0 (ByteDance Seed Team, 2026), and GPT-Image-2 (OpenAI, 2026)) and open-source models (Qwen-Image-2.1 and the unmodified Qwen-Image-Edit-2511 baseline (Wu et al., 2025), FLUX.2-Dev (Black Forest Labs, 2025b), HiDream-O1-Image (Cai et al., 2026), and HunyuanImage-3.0 (Tencent Hunyuan Foundation Model Team, 2025)).

5.3 Main Results

Table 2 reports the VIBE Deictic-Level scores. Our model improves on the unmodified Qwen-Image-Edit-2511 baseline across all settings, raising the 1-Task average from 63.26 to 89.23 and the 3-Task

score from 59.05 to 76.68. It achieves the best overall 1-Task result among all models. On multi-task editing, it scores 83.27 on 2 Tasks and 76.68 on 3 Tasks, reaching performance comparable to closed-source models. These scores are only 1.12 and 0.77 points below Seedream 5.0, respectively.

Table 2: VIBE Deictic-Level scores on single-task (AD, RM, RP, TR; 1 Task reports their average) and multi-task (2 Tasks, 3 Tasks) settings. Ours is a LoRA fine-tune of the open-source Qwen-Image-Edit-2511 backbone. Best per column in bold and second-best underlined.

Method	AD	RM	RP	TR	1 Task	2 Tasks	3 Tasks
Closed-source							
Nano Banana Pro	82.17	94.07	<u>88.26</u>	74.80	84.83	80.22	75.48
Qwen-Image-2.0 Pro	45.00	60.55	39.12	36.37	45.26	35.31	43.26
Wan 2.7 Pro	90.51	93.92	**88.34**	51.47	81.06	69.01	68.55
Seedream 4.5	83.31	**95.26**	84.00	48.03	77.65	70.20	61.28
Seedream 5.0	90.58	91.88	87.16	<u>81.56</u>	<u>87.80</u>	**84.39**	**77.45**
GPT-Image-2	86.65	87.23	86.68	79.30	84.97	77.95	76.04
Open-source							
Qwen-Image-2.1	**93.68**	70.36	83.82	22.95	67.70	60.85	60.75
Qwen-Image-Edit-2511 (baseline)	86.73	61.15	74.73	30.42	63.26	57.90	59.05
FLUX.2-Dev	58.90	18.42	44.26	12.16	33.43	19.84	18.36
HiDream-O1-Image	43.78	29.82	33.85	8.29	28.93	26.02	32.49
HunyuanImage-3.0	86.53	83.90	75.90	50.37	74.17	67.61	65.96
Ours	<u>90.68</u>	<u>94.79</u>	86.53	**84.91**	**89.23**	<u>83.27</u>	<u>76.68</u>

User study. To corroborate the automatic VIBE evaluation, we conduct a double-blind user study on 100 VIBE examples. We sample 20 examples from each of the four single-task operations and 20 multi-task examples. Five participants independently evaluate all 100 examples and select one or more models they judge best for each example. Figure 5 reports the total number of selections received by each model. Our method receives 281 selections, compared with 279 for Seedream 5.0, substantially exceeding GPT-Image-2 and Nano Banana Pro. The aggregate selection counts are broadly consistent with the automatic evaluation.

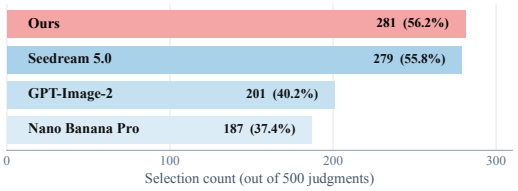


Figure 5: Double-blind user study on 100 VIBE examples. Bars show selection counts and rates over 500 judgments.

5.4 Qualitative Results

Figure 6 presents side-by-side examples of object single-task and multi-task editing. In the displayed single-task examples, baseline outputs can misinterpret instructions, modify unintended objects, produce inconsistent appearances, or leave artifacts such as lingering shadows. In the displayed multi-task examples, some baselines omit requested edits. Our outputs follow the requested edits and better preserve the source-image context in these examples.

5.5 Ablations

We ablate the Progressive Supervision Curriculum to assess the contribution of its Stage-I and Stage-II components. Table 3 examines the precise supervision in Layered Synthetic Data, edit-region weighting, and Multi-Task Mosaicking. We then examine whether Stage II further improves visual coherence. These ablations also provide additional evidence for the supervision gaps diagnosed in Section 3.

Stage-I components. The exact edit-region masks in Layered Synthetic Data allow edit-region reweighting to emphasize the intended edit. Removing this term leads to modest drops on most VIBE tasks. In contrast, removing Multi-Task Mosaicking leaves 1-Task performance largely unchanged

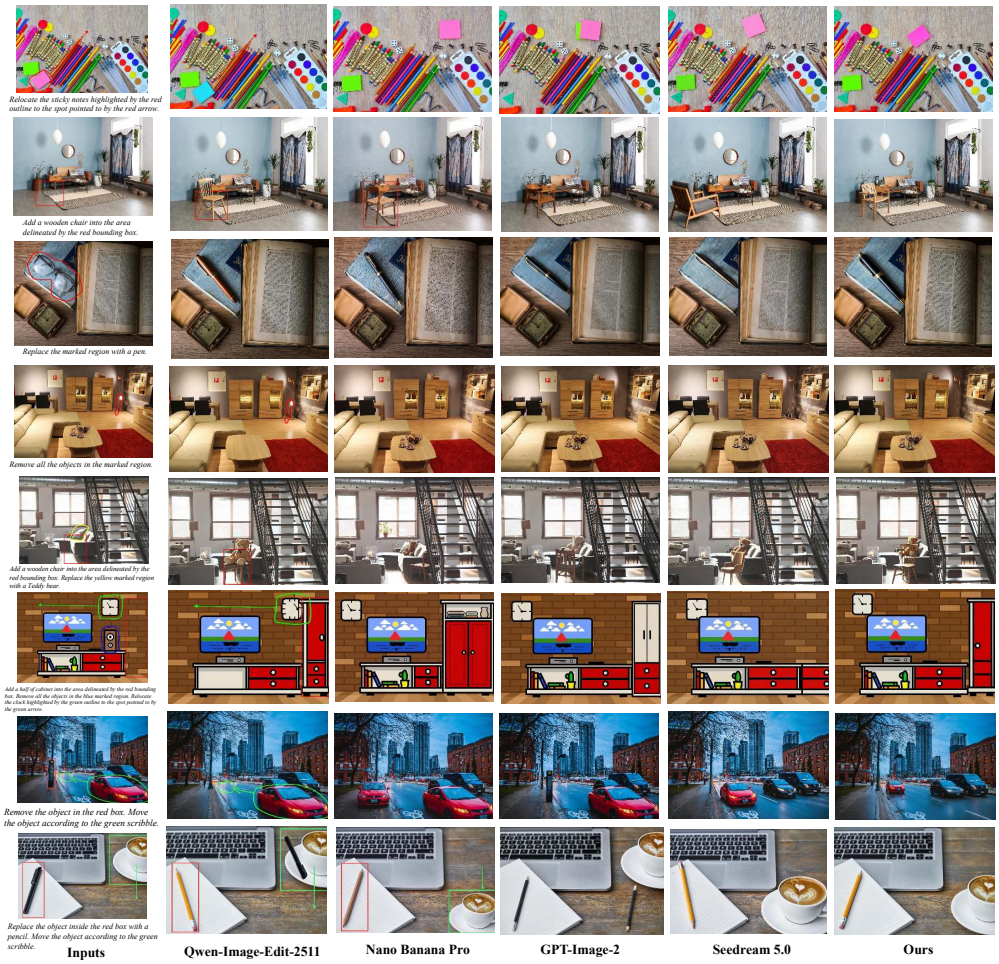


Figure 6: Qualitative comparison across single-task and multi-task editing.

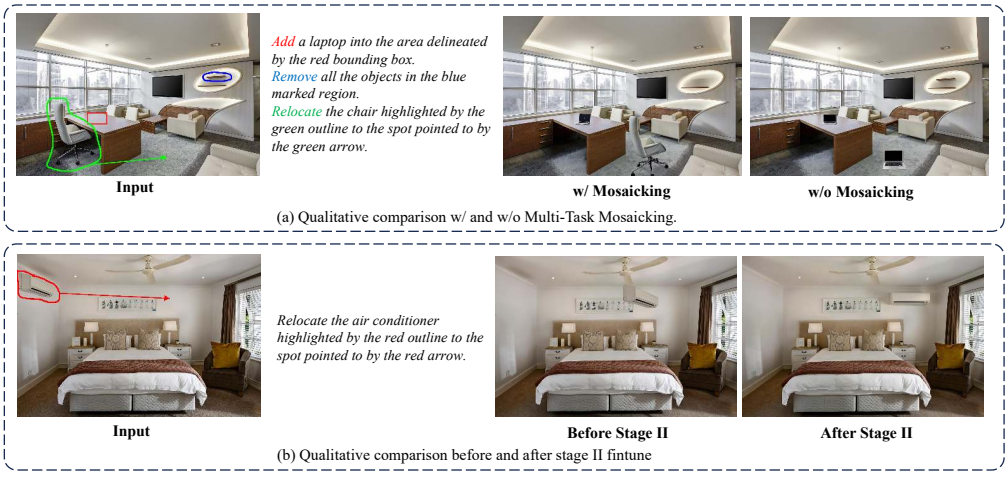


Figure 7: Qualitative effects of the Progressive Supervision Curriculum. (a) Multi-scribble editing with and without Multi-Task Mosaicking. (b) Visual coherence before and after Stage II: Visual Coherence Refinement.

Table 3: Component ablations on Layered Synthetic Data using VIBE Deictic-Level scores. In row labels, "Layered" abbreviates Layered Synthetic Data and "Mosaicking" abbreviates Multi-Task Mosaicking. Best per column in bold.

Method	AD	RM	RP	TR	1 Task	2 Tasks	3 Tasks
Layered w/o edit-region weighting	88.37	95.84	83.82	81.59	87.41	81.90	71.72
Layered w/o Mosaicking	**90.26**	**98.29**	**85.43**	**83.52**	**89.38**	73.21	49.67
Layered	88.39	98.18	84.60	81.24	88.10	**82.65**	**73.29**

but substantially degrades 2- and 3-Task editing. Figure 7 (a) illustrates the same pattern: without mosaicked data, the model can omit an instruction or bind an edit to the wrong scribble. Together with the distractor control in Section 3.2, these results support complete multi-task supervision for multi-task binding.

Table 4: Instruction Adherence ($\mathcal{IA}$) and Visual Coherence ($\mathcal{VC}$) sub-scores on VIBE before and after Stage II: Visual Coherence Refinement.

	Instruction Adherence ($\mathcal{IA}$)			Visual Coherence ($\mathcal{VC}$)		
Stage	1 Task	2 Tasks	3 Tasks	1 Task	2 Tasks	3 Tasks
Stage I	88.37	82.10	78.33	89.83	84.20	78.52
Stage II	**90.16**	**85.93**	**78.33**	**93.81**	**88.40**	**82.96**

Stage II: Visual Coherence Refinement. We next examine the second stage of the curriculum. Stage II fine-tunes the Stage-I model on Visually Coherent Synthetic Data to improve the visual coherence of edits with their source images. Table 4 compares the two stages on Instruction Adherence ($\mathcal{IA}$) and Visual Coherence ($\mathcal{VC}$). Stage II improves $\mathcal{VC}$ across all task counts, and maintains or improves the original $\mathcal{IA}$ scores.

Figure 8 summarizes this pattern on 1 Task: the Stage-II update moves the model primarily upward on $\mathcal{VC}$ and also rightward on $\mathcal{IA}$. Figure 7 (b) shows the corresponding qualitative improvement in perspective and object–background blending. Thus, Stage II primarily refines visual coherence while preserving the binding capability acquired in Stage I.

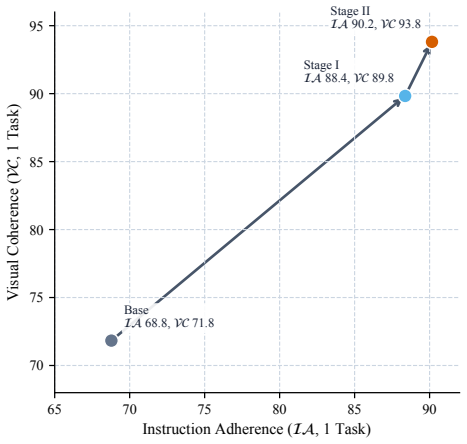


Figure 8: Adaptation trajectory on 1-Task Instruction Adherence ($\mathcal{IA}$) and Visual Coherence ($\mathcal{VC}$). Arrows trace the base model through Stage I and Stage II.

6 Conclusion

Scribbles give image editors a simple way to express spatial intent, but they also require supervision that connects each mark to an instruction and its intended edit. Our results show that this supervision can be effectively organized progressively. Large-scale precise supervision supports single-task editing, complete multi-task supervision supports binding across several requests, and visually coherent supervision improves the integration of edits with the source image. The resulting curriculum improves an open-source backbone to the best overall single-task result on VIBE and to multi-task performance comparable to closed-source models. We will release our dataset and model to support the future work.

AI USE STATEMENT

We used generative AI tools to construct synthetic training data. An LLM generated instructions for Layered Synthetic Data, while VLMs and an image editing model generated candidate instructions and edited targets for Visually Coherent Synthetic Data. AI coding agents also assisted in writing parts of the data construction code. We also used an LLM to improve the readability of the manuscript. All authors reviewed AI-assisted text and the data construction outputs; the automatically constructed visually coherent samples were additionally human curated as described in Section 4.2. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies controllable image editing, which can support creative and assistive workflows but can also be misused to manipulate visual content. We will document intended use and limitations, discourage deceptive applications, and apply appropriate safeguards to any user-facing deployment.

REPRODUCIBILITY STATEMENT

The data construction procedures, training configuration, evaluation protocol, and compute resources are described in Sections 4 and 5 and Appendix C. The appendix also provides full quantitative results and the VIBE-Text evaluation prompts.

REFERENCES

Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. *arXiv preprint arXiv:2111.14818*, 2021.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 41–48, 2009.

Black Forest Labs. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025a.

Black Forest Labs. FLUX.2: Frontier visual intelligence. `https://bfl.ai/blog/flux-2`, November 2025b. Accessed: 2026-09-26.

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)*, pp. 18392–18402, 2023.

ByteDance Seed Team. Seedream 4.0: Toward next-generation multimodal image generation. *arXiv preprint arXiv:2509.20427*, 2025.

ByteDance Seed Team. Deeper thinking, more accurate generation | introducing seedream 5.0 lite. `https://seed.bytedance.com/en/blog/deeper-thinking-more-accurate-generation-introducing-seedream-5-0-lite`, 2026. Accessed: 2026-09-26.

Qi Cai, Jingwen Chen, Chengmin Gao, Zijian Gong, Yehao Li, Yingwei Pan, Yi Peng, Zhaofan Qiu, Kai Yu, Yiheng Zhang, et al. HiDream-O1-Image: A natively unified image generative foundation model with pixel-level unified transformer. *arXiv preprint arXiv:2605.11061*, 2026.

Duosheng Chen, Binghui Chen, Yifeng Geng, and Liefeng Bo. Adaptivedrag: Semantic-driven dragging on diffusion-based image editing. *arXiv preprint arXiv:2410.12696*, 2024.

Junwen Chen, Heyang Jiang, Yanbin Wang, Keming Wu, Ji Li, Chao Zhang, Keiji Yanai, Dong Chen, and Yuhui Yuan. Prismlayers: Open data for high-quality multi-layer transparent image generative models. *arXiv preprint arXiv:2505.22523*, 2025.

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

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. *arXiv preprint arXiv:2403.03206*, 2024.

Google DeepMind. Nano banana pro (gemini 3 pro image). `https://deepmind.google/models/gemini-image/pro/`, 2025. Released November 2025.

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.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6007–6017, 2023.

Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22511–22521, 2023.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Chaojie Mao, Chen-Wei Xie, Chongyang Zhong, Haoyou Deng, Jiaxing Zhao, Jie Xiao, Jinbo Xing, Jingfeng Zhang, Jingren Zhou, Jingyi Zhang, et al. Wan-image: Pushing the boundaries of generative visual intelligence. *arXiv preprint arXiv:2604.19858*, 2026.

Qi Mao, Lan Chen, Yuchao Gu, Zhen Fang, and Mike Zheng Shou. Mag-edit: Localized image editing in complex scenarios via mask-based attention-adjusted guidance. *arXiv preprint arXiv:2312.11396*, 2023.

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)*, pp. 6038–6047, 2023.

Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

OpenAI. GPT-Image-2 model. `https://developers.openai.com/api/docs/models/gpt-image-2`, 2026. Accessed: 2026-09-09.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205, 2023.

Tianyuan Qu, Lei Ke, Xiaohang Zhan, Longxiang Tang, Yuqi Liu, Bohao Peng, Bei Yu, Dong Yu, and Jiayi Jia. Replan: Reasoning-guided region planning for complex instruction-based image editing. *arXiv preprint arXiv:2512.16864*, 2025.

Yujun Shi, Chuhui Xue, Jun Hao Liew, Jiachun Pan, Hanshu Yan, Wenqing Zhang, Vincent Y. F. Tan, and Song Bai. Dragdiffusion: Harnessing diffusion models for interactive point-based image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Xincheng Shuai, Henghui Ding, Xingjun Ma, Rongcheng Tu, Yu-Gang Jiang, and Dacheng Tao. A survey of multimodal-guided image editing with text-to-image diffusion models. *arXiv preprint arXiv:2406.14555*, 2024.

Tencent Hunyuan Foundation Model Team. HunyuanImage 3.0 technical report. *arXiv preprint arXiv:2509.23951*, 2025.

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.

Bin Xia, Bohao Peng, Jiyang Liu, Sitong Wu, Jingyao Li, Junjia Huang, Xu Zhao, Yitong Wang, Ruihang Chu, Bei Yu, et al. Dreamomni3: Scribble-based editing and generation. *arXiv preprint arXiv:2512.22525*, 2025.

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. *arXiv preprint arXiv:2411.15738*, 2024.

Haobo Yuan, Xiangtai Li, Tao Zhang, Yueyi Sun, Zilong Huang, Shilin Xu, Shunping Ji, Yunhai Tong, Lu Qi, Jiashi Feng, and Ming-Hsuan Yang. Sa2va: Marrying SAM2 with MLLM for dense grounded understanding of images and videos. *arXiv preprint arXiv:2501.04001*, 2025.

Huanyu Zhang, Xuehai Bai, Chengzu Li, Chen Liang, Haochen Tian, Haodong Li, Ruichuan An, Yifan Zhang, Anna Korhonen, Zhang Zhang, et al. How well do models follow visual instructions? vibe: A systematic benchmark for visual instruction-driven image editing. *arXiv preprint arXiv:2602.01851*, 2026a.

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)*, pp. 3836–3847, 2023.

Yudi Zhang, Yeming Geng, and Lei Zhang. Scribblesense: Generative scribble-based texture editing with intent prediction. *arXiv preprint arXiv:2601.22455*, 2026b.

Haozhe Zhao, Xiaojian Ma, Liang Chen, Shuzheng Si, Rujie Wu, Kaikai An, Peiyu Yu, Minjia Zhang, Qing Li, and Baobao Chang. Ultraedit: Instruction-based fine-grained image editing at scale. In *Advances in Neural Information Processing Systems*, 2024.

Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K. Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models. In *Advances in Neural Information Processing Systems*, 2023.

APPENDIX

A FULL EXPERIMENTAL RESULTS

Table 5 reports the full VIBE Deictic-Level results corresponding to Table 2 in the main paper, including the mean and standard deviation over three independent runs. Results for Nano Banana Pro are directly taken from the original VIBE paper, and the standard deviations of some of its entries are unavailable because they are not reported there either.

B GENERALIZATION ACROSS BACKBONES

We examine whether the Progressive Supervision Curriculum generalizes beyond Qwen-Image-Edit-2511 to another image editing backbone. We apply the same curriculum to FLUX.2-Dev. Scribbles are rendered directly into the input image, and only a LoRA adapter is trained. Stage I, *Single-Task Editing and Multi-Task Binding Acquisition*, trains on Layered Synthetic Data with Multi-Task Mosaicking and edit-region weighting. Stage II, *Visual Coherence Refinement*, then fine-tunes on Visually Coherent Synthetic Data.

Table 6 compares the original FLUX.2-Dev model with the adapted model. The curriculum raises the 1-Task average from 33.43 to 81.53 ($+48.10$), the 2-Task score from 19.84 to 61.66 ($+41.82$), and the 3-Task score from 18.36 to 52.05 ($+33.69$). These gains show that the curriculum can improve a second DiT image editing backbone under the same LoRA adaptation setting.

Table 5: Full VIBE Deictic-Level scores with mean and standard deviation over 3 independent runs.

Method	AD	RM	RP	TR	1 Task	2 Tasks	3 Tasks
Closed-source							
Nano Banana Pro	82.17 ± 0.66	94.07 ± 0.73	88.26 ± 1.97	74.80 ± 2.13	84.83	80.22	75.48
Qwen-Image-2.0 Pro	45.00 ± 1.94	60.55 ± 4.02	39.12 ± 1.10	36.37 ± 1.89	45.26 ± 2.48	35.31 ± 1.09	43.26 ± 6.52
Wan 2.7 Pro	90.51 ± 1.13	93.92 ± 1.57	88.34 ± 0.75	51.47 ± 0.57	81.06 ± 1.07	69.01 ± 2.76	68.55 ± 6.23
Seedream 4.5	83.31 ± 0.82	95.26 ± 0.51	84.00 ± 2.70	48.03 ± 1.59	77.65 ± 1.64	70.20 ± 3.07	61.28 ± 1.63
Seedream 5.0	90.58 ± 1.23	91.88 ± 1.15	87.16 ± 0.52	81.56 ± 1.68	87.80 ± 1.22	84.39 ± 2.21	77.45 ± 2.36
GPT-Image-2	86.65 ± 0.50	87.23 ± 1.33	86.68 ± 3.87	79.30 ± 2.12	84.97 ± 2.32	77.95 ± 0.49	76.04 ± 4.30
Open-source							
Qwen-Image-2.1	93.68 ± 0.36	70.36 ± 2.58	83.82 ± 0.42	22.95 ± 2.67	67.70 ± 1.88	60.85 ± 3.23	60.75 ± 6.04
Qwen-Image-Edit-2511	86.73 ± 0.25	61.15 ± 3.58	74.73 ± 0.37	30.42 ± 4.00	63.26 ± 2.69	57.90 ± 2.60	59.05 ± 4.62
FLUX.2-Dev	58.90 ± 1.15	18.42 ± 1.53	44.26 ± 0.20	12.16 ± 0.98	33.43 ± 1.08	19.84 ± 0.30	18.36 ± 4.55
HiDream-O1-Image	43.78 ± 1.18	29.82 ± 3.01	33.85 ± 1.16	8.29 ± 0.85	28.93 ± 1.77	26.02 ± 3.18	32.49 ± 3.31
HunyuanImage-3.0	86.53 ± 1.44	83.90 ± 2.52	75.90 ± 5.96	50.37 ± 2.84	74.17 ± 3.60	67.61 ± 3.18	65.96 ± 4.81
Ours							
Layered Data w/o edit-region weighting	88.37 ± 0.92	95.84 ± 0.44	83.82 ± 0.67	81.59 ± 0.33	87.41 ± 0.63	81.90 ± 2.80	71.72 ± 5.73
Layered Data w/o Multi-Task Mosaicking	90.26 ± 1.85	98.29 ± 1.12	85.43 ± 1.10	83.52 ± 1.25	89.38 ± 1.36	73.21 ± 2.56	49.67 ± 4.59
Layered Data	88.39 ± 0.48	98.18 ± 1.02	84.60 ± 0.10	81.24 ± 2.80	88.10 ± 1.51	82.65 ± 2.25	73.29 ± 3.48
Layered Data + Visually Coherent Data (final)	90.68 ± 1.00	94.79 ± 0.85	86.53 ± 1.41	84.91 ± 1.52	89.23 ± 1.23	83.27 ± 4.28	76.68 ± 2.42

Table 6: Generalization of the Progressive Supervision Curriculum to FLUX.2-Dev. Scores are reported as mean $\pm$ standard deviation over three independent runs.

Method	AD	RM	RP	TR	1 Task	2 Tasks	3 Tasks
FLUX.2-Dev	58.90 ± 1.15	18.42 ± 1.53	44.26 ± 0.20	12.16 ± 0.98	33.43 ± 1.08	19.84 ± 0.30	18.36 ± 4.55
Ours (FLUX.2-Dev)	$\mathbf{81.43 \pm 1.35}$	$\mathbf{93.19 \pm 0.44}$	$\mathbf{78.88 \pm 0.73}$	$\mathbf{72.60 \pm 4.31}$	$\mathbf{81.53 \pm 2.30}$	$\mathbf{61.66 \pm 1.76}$	$\mathbf{52.05 \pm 2.36}$

C Compute Resources Details

For the main training runs, we fine-tune the model on 16 NVIDIA H20 GPUs, each with 96GB of device memory. We use a total batch size of 64 across all GPUs. Training consists of 4,000 optimization steps in Stage I (Single-Task Editing and Multi-Task Binding Acquisition) on Layered Synthetic Data, followed by 500 steps in Stage II (Visual Coherence Refinement) on Visually Coherent Synthetic Data. In our implementation, training requires approximately 43 minutes per 100 optimization steps on this hardware configuration. This corresponds to approximately 32.3 wall-clock hours in total for one complete training run.

D More Qualitative Comparisons

Figure 9 provides more qualitative comparisons.

E Scribble-Guided Text Editing as an Application

Beyond object editing in the main paper, we explore scribble-guided *text* editing with a related structured synthetic-supervision pipeline. The text-editing data construction, the VIBE-Text benchmark, and current results are presented here as a self-contained appendix. It reuses the re-renderable local-correspondence principle of Layered Synthetic Data, but uses text-specific supervision, re-layout behavior, and evaluation criteria rather than the object-editing curriculum. We therefore keep it outside the main object-editing claim.

E.1 Text Editing Dataset

Text editing is not a special case of object editing: insertion, deletion, and replacement can change the length of a line and force the model to *re-layout* the surrounding text, which pure object-editing supervision provides no signal for. We build a dedicated text-editing dataset and keep it fully synthetic because rendered text gives exact pixel-level ground truth, while paired before/after text-editing examples with localized scribbles are difficult to collect at scale.

We cover four tasks: **Insertion**, **Deletion**, **Text Replacement**, and **Style Modification**. The first three operate on character content and always require re-layout; Style Modification triggers a re-layout only when the modified attribute changes character footprint (e.g., font size or spacing) and leaves

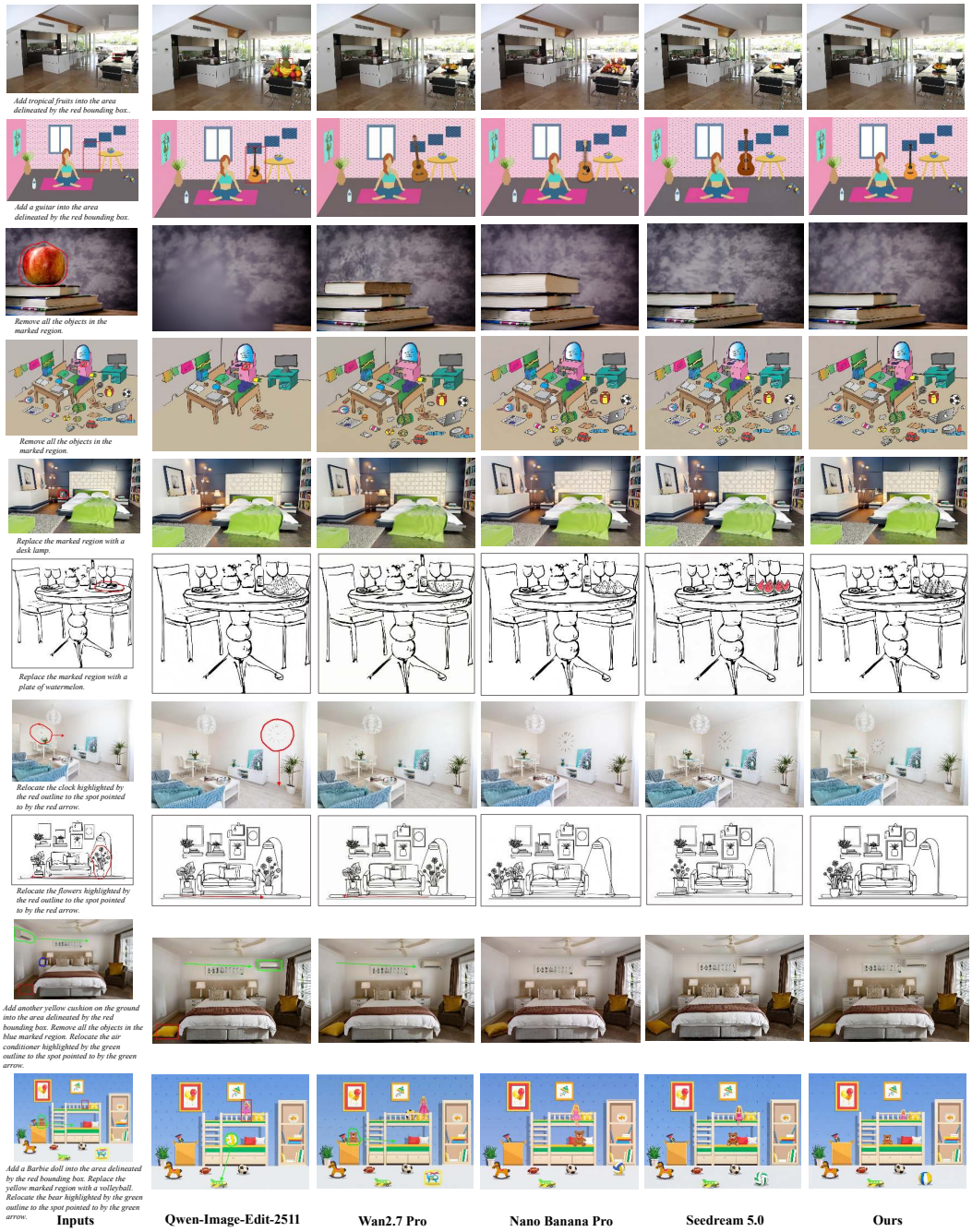


Figure 9: More qualitative results for scribble-guided editing.

the layout untouched otherwise (e.g., color or shadow). The instruction carries an explicit re-layout cue whenever a re-layout is needed, so the model triggers re-layout only when the prompt asks for it. To keep re-layout controllable, we restrict its scope to the last line of the edited block, leaving the rest of the page pixel-identical to the source.

The pipeline mirrors the construction of Layered Synthetic Data in Section 4.2, with rendered text in place of layered images. We first render a paragraph of mixed Chinese and English text on a solid-color canvas with randomized attributes. We then apply the edit at the source-string level and re-render with the same typography to obtain $\tilde{I}$, so that only the edited span changes pixels. The bounding box of the edited span drives scribble synthesis with randomized color, and an LLM is finally prompted with the task type, the edited text, and the scribble shape and color to write the matching instruction. This pipeline produces a synthetic text-editing dataset of $\sim$180K samples.

E.2 VIBE-TEXT BENCHMARK

VIBE's Deictic Level (Zhang et al., 2026a) evaluates scribble-guided object editing but does not cover scribble-guided text editing. We extend it minimally so that VIBE-Text scores remain directly comparable to VIBE Deictic-Level scores. The benchmark mirrors the four text-editing tasks of Section E.1 and adopts VIBE's three task-level criteria with text-specific adjustments: **Instruction Adherence** ($\mathcal{IA}$) gains a binary *Re-layout Compliance* sub-criterion that checks whether the model re-layouts exactly as the instruction requires; **Contextual Preservation** ($\mathcal{CP}$) requires character-level fidelity on unchanged spans while explicitly excluding instruction-required re-layouts from the violation count; and **Visual Coherence** ($\mathcal{VC}$) is augmented with text-style consistency (font, weight, size, color) and overall layout harmony. The geometric-mean aggregation and the gating of $\mathcal{VC}$ by $\mathcal{IA}$ are inherited from VIBE.

Evaluation prompts. The following listings give the three VIBE-Text evaluation prompts used for $\mathcal{IA}$, $\mathcal{CP}$, and $\mathcal{VC}$.

Listing 1: VIBE-Text prompt for Instruction Adherence.

```
You are given THREE images and ONE text prompt.

The first image:
- This is the original image.

The second image:
- Visual instructions are drawn on this image based on the original image by the user.

The third image:
- This is the image generated by a model after editing the second image.

TEXT PROMPT:
- {prompt}

Your task is to evaluate whether the Output Image (The third image) correctly follows the
    instruction.
The instruction consists of BOTH:
(1) the visual instruction drawn on the second image, and
(2) the textual description in the Text Prompt.

You must independently evaluate the following FOUR metrics and assign a binary score (1 or 0)
    to each:

1. Visual Instruction Localization Correctness
Did the main edit occur on the text target explicitly indicated by the visual instruction on
    the Input Image (The second image)?

2. Visual Operator Type Compliance
Was the type of edit consistent with the operation implied by the visual instruction?

3. Textual Action Semantic Compliance
Did the model execute the core textual action specified in the Text Prompt?

4. Text Re-layout Compliance
When the requested text edit changes the amount, size, or placement of text, did the model
    adjust the affected line, paragraph, or text block layout appropriately?

- Judge whether the new layout looks reasonable and natural for the affected line, paragraph,
    or text block.
```

```
- The result should not leave obvious blank gaps, squeezed text, overlapping characters, or
    broken layout.

Scoring rules:
- Score 1 if the requirement is clearly satisfied.
- Score 0 if the requirement is not satisfied or is ambiguous.
- If unsure, assign 0.
- Partial compliance must be scored as 0.

You may reason freely to reach your decision.
Then, for EACH metric, provide:
- "score": an integer value of 0 or 1.
- "reason": ONE short factual sentence describing an observable outcome.
This summary must strictly follow the output format specified below:

{
  "Visual_Instruction_Localization_Correctness": {
    "reason": "brief factual summary",
    "score": 1/0
  },
  "Visual_Operator_Type_Compliance": {
    "reason": "brief factual summary",
    "score": 1/0
  },
  "Textual_Action_Semantic_Compliance": {
    "reason": "brief factual summary",
    "score": 1/0
  },
  "Text_Re-layout_Compliance": {
    "reason": "brief factual summary",
    "score": 1/0
  }
}
```

Listing 2: VIBE-Text prompt for Contextual Preservation.

```
You are given TWO images and ONE text prompt.

IMAGE 1 (Input Image):
- Original image with user-drawn visual instructions.

IMAGE 2 (Output Image):
- Image generated by the model after editing IMAGE 1.
- The output may be cropped or reframed.

TEXT PROMPT:
- {prompt}

Your task is to evaluate Contextual Preservation.

Definition:
Contextual Preservation checks whether the model changed anything it was NOT supposed to
    change.
It does NOT judge whether the edit was correct, precise, or well aligned.
Errors in edit location, extent, or alignment belong to Instruction Adherence, not Contextual
    Preservation.

Evaluation rules (follow strictly):

1) Cropping rule
- If the output is cropped, only compare the overlapping visible region.
- Ignore content missing only due to cropping.

2) Difference listing (what counts as a difference)
- List ONLY meaningful differences in:
  - text content (characters, words),
  - text order (line order, word order),
  - typography (font family, size, weight, color, emphasis, case),
  - non-text visual entities (objects, background).
- Do NOT list differences caused only by:
  - minor blur or softness,
  - small texture or color shifts,
  - pixel-level noise,
  - slight position or alignment offsets.

3) Target rule
- Identify the intended edit target based ONLY on:
  (a) the visual instruction marks, and
  (b) the text prompt.
```

```
4) Re-layout scope rule
- Reasonable line wrapping, spacing changes, and local repositioning INSIDE the affected text
    block are IN_TARGET when they are a consequence of the requested edit, even if nearby
    unedited words shift to new line positions inside that same affected block.
- Failed or awkward reflow inside the affected block is NOT judged here.
- This rule NEVER licenses changing unrelated text content, unrelated typography, or unrelated
     text blocks; those remain OUT_OF_TARGET.

5) Classification rule
- IN_TARGET:
  - any change within the intended target,
  - OR any imperfect attempt to edit the target (including misplacement, offset, scale error,
    or incomplete coverage).
- OUT_OF_TARGET:
  - any change to unrelated objects or regions,
  - any addition or removal of unrelated semantic entities,
  - any structural damage to non-target objects.

6) Scoring
- Score = 1 if NO OUT_OF_TARGET differences exist.
- Score = 0 if ANY OUT_OF_TARGET difference exists.
- If unsure, score = 0.

Output format:
First provide a brief analysis with these sections:
- ## Differences
- ## Target
- ## Classification
- ## Decision

Then output the final JSON as the last part of your response:

{
  "Text_Contextual_Preservation": {
    "reason": "string",
    "score": 0
  }
}
```

Listing 3: VIBE-Text prompt for Visual Coherence.

```
You are given THREE images and ONE text prompt.

The first image:
- This is the original image.

The second image
- Visual instructions are drawn on this image based on the original image by the user.

The third image:
- This is the image generated by a model after editing the second image.

TEXT PROMPT:
- {prompt}

Your task is to evaluate the Visual Coherence of the Output Image.

Visual Coherence evaluates whether the edited result is visually unified and generatively
    sound.
This metric is STYLE-AGNOSTIC: you must NOT judge realism, beauty, or artistic preference.
Instead, you must assess whether the edited image remains consistent with the source image and
     whether the output avoids clear generative artifacts.

You must independently evaluate the following THREE metrics and assign a binary score (1 or 0)
     to each:

1. Text Style Consistency
Did the edited text region adopt the same visual text style as the surrounding or
    corresponding source text (e.g., font family, font size, weight, color, perspective,
    rotation, curvature, and surface deformation)?

Scoring:
- Score 1 if the edited / added text clearly belongs to the same visual text domain and
    matches the surrounding source text style.
- Score 0 if it introduces a noticeably different from the surrounding source text.
- Do NOT judge whether the style looks good or realistic--only whether it matches the source
    domain.

2. Text Layout Seamlessness
```

```
Is the edited text visually integrated with the surrounding text line, paragraph, surface, or
    document layout, with no obvious local discontinuity?

Focus on clear local layout discontinuities such as:
- unnatural seams or hard boundaries around the edited text,
- overlapping characters or visibly broken character spacing within the edit,
- broken baseline, mismatched line height, or irregular word spacing in the immediate edited
    area,
- misalignment with the original text grid, sign surface, page margin, or perspective plane in
     the immediate edited area.

Scoring:
- Score 1 if the edited text integrates seamlessly with its immediate surrounding layout.
- Score 0 if there are clear and noticeable local layout discontinuities.

3. Artifact-Free Text Generation
Does the Output Image avoid obvious text-specific or general generative artifacts?

Consider artifacts such as:
- unreadable, garbled, melted, duplicated, or malformed glyphs in the edited text,
- broken or inconsistent strokes,
- ghost text or leftover erased text under the edit,
- unintended blur, pixelation, warping, or rendering collapse,
- residual visual instruction marks such as circles, underlines, or masks that should not
    appear in the final image.

Scoring:
- Score 1 if no clear generative artifacts are present in the edited result.
- Score 0 if obvious artifacts are visible.
- Style-specific noise or abstraction is allowed if consistent with the source domain.

General rules:
- Evaluate ONLY visual coherence, not instruction correctness or contextual preservation.
- Minor pixel-level differences are acceptable.
- If you are unsure about a case, assign score 0.
- Partial compliance must be scored as 0.

You may reason freely to reach your decision.
Then, for EACH metric, you provide a summary:
- "score": an integer value of 0 or 1.
- "reason": ONE short factual sentence describing an observable outcome.
This summary must strictly follow the output format specified below:

{
  "Text_Style_Consistency": {
    "reason": "brief factual summary",
    "score": 1/0
  },
  "Text_Layout_Seamlessness": {
    "reason": "brief factual summary",
    "score": 1/0
  },
  "Artifact-Free_Text_Generation": {
    "reason": "brief factual summary",
    "score": 1/0
  }
}
```

E.3 Preliminary Results on VIBE-Text

We benchmark a model fine-tuned on the recipe of Section 4 extended with the text editing dataset of Section E.1. We use Gemini-3-Flash as the VLM judge because it is more accurate on our text-specific criteria. Table 7 reports VIBE-Text scores across the four text-editing tasks. Our model averages 53.46, compared with 44.33 for Nano Banana Pro and 14.64 for the unmodified Qwen-Image-Edit-2511 backbone. This is a $+9.13$ improvement over the closest closed-source competitor. The lead is consistent across all four tasks and is largest on Style Modification ($+13.32$ over Nano Banana Pro), suggesting that the visual constraints unique to text editing (font, color, layout) are where dedicated supervision matters most. This application suggests that our analysis of supervision for scribble-guided tasks and the resulting data construction approach extend beyond object editing.

Table 7: VIBE-Text scores with mean and standard deviation across four text-editing tasks. Best per column in bold.

Method	Deletion	Insertion	Style Mod.	Replacement	Avg.
Closed-source					
Nano Banana Pro	45.97 ± 3.42	47.47 ± 3.50	37.91 ± 2.39	45.95 ± 1.24	44.33 ± 2.79
Seedream 5.0	29.56 ± 5.39	43.80 ± 1.33	36.72 ± 4.64	42.69 ± 2.38	38.19 ± 3.81
Seedream 4.5	35.41 ± 2.10	26.58 ± 2.50	22.89 ± 2.48	43.87 ± 2.63	32.19 ± 2.43
Wan 2.7	19.27 ± 0.42	34.31 ± 3.77	34.52 ± 1.63	38.12 ± 2.95	31.55 ± 2.54
Wan 2.6	44.98 ± 3.32	29.17 ± 2.48	34.75 ± 2.78	46.05 ± 0.96	38.74 ± 2.54
Open-source					
Qwen-Image-Edit-2511	5.98 ± 0.51	21.88 ± 2.19	8.43 ± 2.56	22.26 ± 2.42	14.64 ± 2.09
Ours	$\mathbf{51.19 \pm 6.23}$	$\mathbf{53.46 \pm 0.70}$	$\mathbf{51.23 \pm 2.48}$	$\mathbf{57.95 \pm 3.61}$	$\mathbf{53.46 \pm 3.82}$



Figure 10: Qualitative examples for scribble-guided text editing.

E.4 Qualitative Results of Text Editing

Figure 10 shows representative VIBE-Text examples across text-editing tasks. The comparison is intended to highlight whether the model edits only the scribbled text span, preserves unrelated text, and performs local re-layout when the requested edit changes line length or typography.

F Limitations

This work focuses on scribble-guided editing at the *Deictic Level*, following the terminology introduced by VIBE. In this setting, the scribble gives an explicit spatial cue for the editing instruction, so the main challenge is to ground the marked region and perform the requested edit. This choice allows us to study spatial grounding and edit execution in a controlled way, but it also limits the scope of the paper. Higher-level instruction settings in VIBE require deeper visual understanding and reasoning: the model may need to infer object relations, recognize object states, resolve implicit references, or plan multi-step edits before it can decide where and how to edit. We leave these richer instruction forms to future work.