VeraRetouch: A Lightweight Fully Differentiable Framework for Multi-Task Reasoning Photo Retouching

YIHONG GUO, Zhejiang University, China
YOUWEI LYU, vivo BlueImage Lab, China
JIAJUN TANG, vivo BlueImage Lab, China
YIZHUO ZHOU, Zhejiang University, China
HONGLIANG WANG, University of Chinese Academy of Sciences, China
JINWEI CHEN, vivo BlueImage Lab, China
CHANGQING ZOU*, Zhejiang Lab, China; Zhejiang University, China
QINGNAN FAN, vivo BlueImage Lab, China

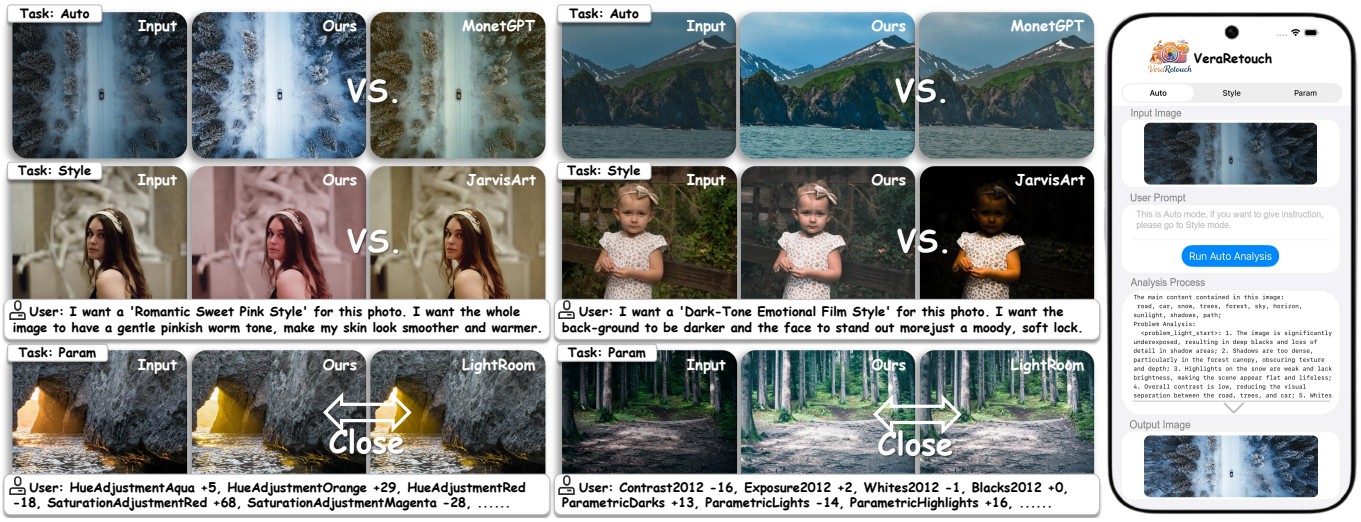

Fig. 1. We present **VeraRetouch**, a lightweight, fully differentiable framework for reasoning photo retouching in multiple scenarios: 1) ***Auto-Retouch*** (top left), with image input only; 2) ***Style-Retouch*** (middle left), with stylistic prompt, and 3) ***Param-Retouch*** (bottom left), parameter-driven; The mobile-oriented UI workflow (right) takes an input image with an optional user prompt and produces the retouched image with an interpretable analysis.

Reasoning photo retouching has gained significant traction, requiring models to analyze image defects, give reasoning processes, and execute precise retouching enhancements. However, existing approaches often rely on non-differentiable external software, creating optimization barriers and suffering from high parameter redundancy and limited generalization. To address these challenges, we propose VeraRetouch, a lightweight and fully differentiable framework for multi-task photo retouching. We employ a 0.5B Vision-Language Model (VLM) as the central intelligence to formulate retouching plans based on instructions and scene semantics. Furthermore, we develop a fully differentiable Retouch Renderer that replaces external tools, enabling direct end-to-end pixel-level training through decoupled control latents for lighting, global color, and specific color adjustments. To overcome data scarcity, we introduce AetherRetouch-1M+, the first million-scale dataset for professional retouching, constructed via a new inverse degradation workflow. Furthermore, we propose DAPO-AE, a reinforcement learning post-training strategy that enhances autonomous aesthetic cognition. Extensive experiments demonstrate that VeraRetouch achieves state-of-the-art performance across multiple benchmarks while maintaining a significantly smaller footprint, enabling mobile deployment.

CCS Concepts: • **Computing methodologies** → **Computer vision**.

*Corresponding author.

Authors' Contact Information: Yihong Guo, guoyihong@zju.edu.cn, Zhejiang University, Hangzhou, Zhejiang, China; Youwei Lyu, youweilv@gmail.com, vivo BlueImage Lab, Shanghai, China; Jiajun Tang, jiajun.t.cn@gmail.com, vivo BlueImage Lab, Hangzhou, Zhejiang, China; Yizhuo Zhou, zhouyizhuo@zju.edu.cn, Zhejiang University, Hangzhou, Zhejiang, China; Hongliang Wang, hl.wang.ucas@gmail.com, University of Chinese Academy of Sciences, Hangzhou, Zhejiang, China; Jinwei Chen, chenjinwei_1987@126.com, vivo BlueImage Lab, Hangzhou, Zhejiang, China; Changqing Zou, aaronzou1125@gmail.com, Zhejiang Lab, China; Zhejiang University, Hangzhou, Zhejiang, China; Qingnan Fan, fqnchina@gmail.com, vivo BlueImage Lab, Hangzhou, Zhejiang, China.

This work is licensed under a Creative Commons Attribution 4.0 International License.
SIGGRAPH Conference Papers '26, Los Angeles, CA, USA
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2554-8/2026/07
https://doi.org/10.1145/3799902.3811065

Additional Key Words and Phrases: Reasoning Photo Retouching, Lightweight VLM, Million-Scale Dataset, Mobile Deployment.

ACM Reference Format:

Yihong Guo, Youwei Lyu, Jiajun Tang, Yizhuo Zhou, Hongliang Wang, Jinwei Chen, Changqing Zou, and Qingnan Fan. 2026. VeraRetouch: A Lightweight Fully Differentiable Framework for Multi-Task Reasoning Photo Retouching. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers (SIGGRAPH Conference Papers '26), July 19–23, 2026, Los Angeles, CA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3799902.3811065

1 Introduction

As a cornerstone of digital photography post-processing, photo retouching refines visual aesthetics through precise tone and color adjustments while preserving the original content and fine-grained details. In practice, users' retouching needs vary significantly across diverse scenarios and stylistic preferences, yet meeting these demands typically requires mastery of commercial software (e.g., LightRoom, Photoshop) and specialized color/tonal expertise, creating significant barriers for non-professionals. This necessitates a shift toward reasoning-aware and interactive automatic retouching systems, capable of translating vague user intentions into a logical sequence of professional visual enhancements.

Numerous attempts have been made to automate photo retouching. Early supervised and unsupervised methods [Hu et al. 2018; Kosugi and Yamasaki 2020; Ouyang et al. 2023; Yang et al. 2022] aimed to learn expert retouching strategies from datasets like MIT-Adobe FiveK [Bychkovsky et al. 2011] and PPR10K [Liang et al. 2021]. These "black box" approaches lack an explicit reasoning process to understand the underlying scene semantics or aesthetic logic, while their generalization is limited due to the small-scale training datasets. With the advancement of diffusion models, diffusion-based approaches have achieved remarkable success in general image editing [Brooks et al. 2023; Labs et al. 2025; Wu et al. 2025; Zhang et al. 2023a]. However, they still struggle with inadequate instruction-following capabilities and insufficient preservation of fine-grained image textures in specialized retouching tasks. Recently, methods such as PhotoArtAgent [Chen et al. 2025] and JarvisArt [Lin et al. 2025] have integrated Multimodal Large Language Models (MLLMs) with professional retouching software or tools to enable instruction-driven and reasoning retouching. Nevertheless, these non-differentiable external tools create a fundamental optimization barrier that precludes direct pixel-level end-to-end training, ultimately compromising both retouching precision and generalization. Collectively, these limitations highlight four core challenges in the reasoning retouching field: (1) reliance on non-differentiable external retouching tools reduces the training precision of the model; (2) Suboptimal performance when simultaneously handling both automatic and instruction-based retouching within a single framework; (3) Extreme parameter redundancy and inefficient utilization of massive backbones for specialized color and tonal adjustments; and (4) Lack of large-scale training data limits generalization in complex, real-world scenarios.

To overcome these barriers, we propose VeraRetouch, a lightweight framework designed for multi-task and resolution-independent photo retouching. Centrally, a 0.5B Vision-Language Model (VLM) acts as the "brain" to analyze user instructions and scene semantics to formulate a retouching plan. To execute this plan without external software, we develop a fully differentiable Retouch Renderer. This module extracts three disentangled latents from the VLM's features to independently control lighting, global color, and specific color adjustment. By replacing non-differentiable tools with our pixel-faithful Retouch Renderer, VeraRetouch enables direct end-to-end gradient backpropagation during training. Extensive experiments demonstrate that VeraRetouch achieves superior results even with a significantly smaller model size than existing approaches, while maintaining the capability for efficient mobile deployment.

To address diverse real-world retouching needs, we define three core retouching tasks: (1) *Auto-Retouch*, which enhances images autonomously without user prompts; (2) *Style-Retouch*, which applies retouching styles based on language instructions; and (3) *Param-Retouch*, which executes precise pixel adjustments via operational parameters. To resolve the generalization bottleneck caused by small-scale data, we construct ***AetherRetouch-1M+***, the first million-scale dataset covering all three workflows. Specifically, we employ an inverse strategy to "degrade" high-quality images for *Auto-Retouch*, utilize over 5,000 style presets for *Style-Retouch*, and map parameters directly to pixels for *Param-Retouch*. By integrating VLM-generated reasoning chains, this dataset enables VeraRetouch to understand "why" behind each adjustment and generalize across complex scenes. In summary, our contributions can be summarized as follows:

(1) We propose VeraRetouch, the first fully differentiable framework to achieve multi-task reasoning photo retouching without any reliance on external retouching software or tools.

(2) We implement the VeraRetouch framework with a mere 0.5B VLM, outperforming existing SOTA methods in both quality and efficiency, while enabling mobile deployment.

(3) We develop an inverse "degradation" workflow to synthesize high-quality retouching pairs and construct AetherRetouch-1M+, the first million-scale dataset for multi-task professional retouching.

2 Related Work

2.1 Traditional Photo Retouching

Photo retouching represents an essential and widely adopted practice in the post-processing pipeline of digital photography. Early RL-based methods [Hu et al. 2018; Kosugi and Yamasaki 2020; Park et al. 2018; Yu et al. 2018] aimed to mimic human retouching by modeling it as a Markov process, however, they proved time-consuming due to multi-step iterations and consistently fell short in capturing artistic aesthetics. In contrast, another line of research formulates retouching as an end-to-end task, employing fully convolutional generators [Chai et al. 2020; Chen et al. 2018; Deng et al. 2018; Kim et al. 2020; Kneubuehler et al. 2020; Pan et al. 2021] to directly output enhanced images or predict parameters for physical models [Chai et al. 2020; Gharbi et al. 2017; Kim et al. 2021; Moran et al. 2020; Serrano-Lozano et al. 2024; Yang et al. 2022, 2024; Zeng et al. 2020] (e.g., 3D LUTs, tone curves). While efficient, these approaches inherently lack the capacity for deep user interaction and typically fail

to produce diverse stylistic outputs. More recently, diffusion models [Brooks et al. 2023; Labs et al. 2025; Wu et al. 2025; Zhang et al. 2023a] have been introduced for image editing, yet their application to photo retouching still faces challenges in preserving content integrity and fine-grained image details. Additionally, most of these works are built on the training of MIT-Adobe FiveK [Bychkovsky et al. 2011] and PPR10K [Liang et al. 2021] datasets. However, due to limitations in data scale and category coverage, the generalization ability of most methods is restricted in practical scenarios.

2.2 Reasoning Photo Retouching

Reasoning photo retouching is a newly proposed and highly focused research task that requires models to understand user instructions, reason about image defects, generate targeted retouching strategies or parameters, and ultimately output a retouched image. Monet-GPT [Dutt et al. 2025] first introduced VLMs to reasoning retouching tasks. To address the non-differentiable nature of certain retouching operations, it employed a puzzle-based training strategy combined with LoRA fine-tuning, enabling the VLM to indirectly comprehend retouching operations and acquire the ability to generate operation parameters step by step. Subsequently, PhotoArtAgent [Chen et al. 2025] constructed a training-free agent system by leveraging multiple VLMs and the LightRoom API, incorporating multi-round reasoning and self-feedback mechanisms. Further advancing this direction, JarvisArt [Lin et al. 2025] implemented a single-inference retouching agent through direct prediction of LightRoom parameters and fine-tuning via GRPO reinforcement learning. However, these methods are constrained by either multi-round reasoning or large model sizes, leading to persistent challenges in inference speed. Moreover, their reliance on external tools introduces issues of version dependency and potential copyright concerns. Additionally, since the retouching operations performed by these external tools are non-differentiable, they do not permit direct pixel-level gradient backpropagation, limiting end-to-end optimization.

3 Method

3.1 Retouch Encoder and Retouch Renderer

Existing Reasoning photo retouching methods [Dutt et al. 2025; Lin et al. 2025] rely on non-differentiable tools (e.g., LightRoom, Photoshop), creating optimization barriers for end-to-end pixel-level training. Drawing inspiration from the differentiable MLP retouching designs [Lin et al. 2023; Muruts Weldengus et al. 2025], we propose a fully differentiable dual-module framework (as shown in Fig. 2) comprising a Retouch Encoder E and a Retouch Renderer R that can replace professional retouching tools with precise, controllable retouching modeling.

The Retouch Encoder E, built on the ResNet structure [He et al. 2016], extracts disentangled control latents from pairs of input and target reference images $(I^{\text{in}}_{\text{ref}}, I^{\text{tar}}_{\text{ref}})$. Following the principle of independent adjustments in professional retouching workflows and prior studies [Chen et al. 2025; Dutt et al. 2025], we decompose the retouching space into three core latent dimensions:

$$(\mathbf{z}_{\text{l}}, \mathbf{z}_{\text{gc}}, \mathbf{z}_{\text{sc}}) = E(I^{\text{in}}_{\text{ref}}, I^{\text{tar}}_{\text{ref}}). \tag{1}$$

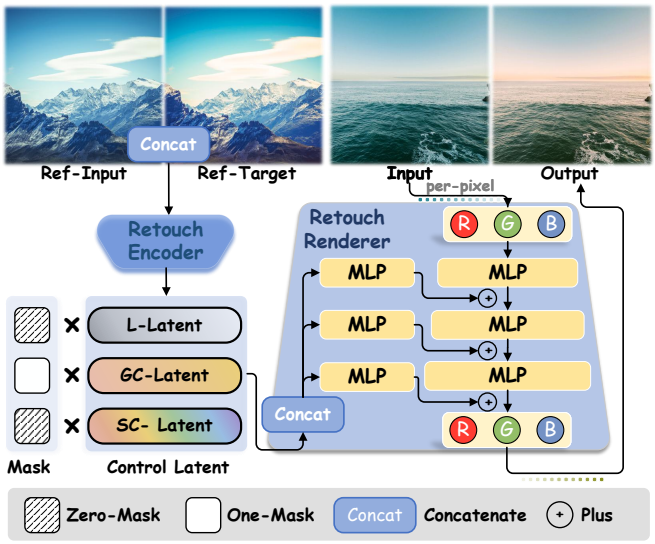

Fig. 2. Retouch Encoder and Retouch Renderer Structure. A reference pair (*Ref-Input*, *Ref-Target*) is concatenated and encoded into three *control latents*: **L-Latent** (light), **GC-Latent** (global color), and **SC-Latent** (specific color). A binary mask (zero/one) indicates which factors are activated. The control latents are then fed to the **Retouch Renderer**, implemented as stacked MLP-based RGB mappings, to produce the final retouched *Output*.

Here, $\mathbf{z}_{\text{l}}$, $\mathbf{z}_{\text{gc}}$, and $\mathbf{z}_{\text{sc}}$ target lighting (*e.g.*, exposure, shadows), global color (*e.g.*, tint, temperature), and specific color adjustments (*e.g.*, red luminance), respectively. To enforce disentanglement, we introduce binary masks $M_{\text{l}}, M_{\text{gc}}, M_{\text{sc}} \in \{0, 1\}$ during training, which selectively activate individual latents to form the composite control latent:

$$\mathbf{z} = \texttt{Concat}(M_{\text{l}} \cdot \mathbf{z}_{\text{l}},\ M_{\text{gc}} \cdot \mathbf{z}_{\text{gc}},\ M_{\text{sc}} \cdot \mathbf{z}_{\text{sc}}). \tag{2}$$

The Retouch Renderer R, implemented as a lightweight pure MLP for per-pixel color mapping, aims at translating the composite latent $\mathbf{z}$ into pixel-level retouching effects. It synthesizes the output image $I^{\text{out}} = R(I^{\text{in}}; \mathbf{z})$ from an input image I^{in} by additively injecting the latent $\mathbf{z}$ into its hidden layers. Unlike diffusion-based generators, Retouch Renderer enables color and tone adjustments while strictly preserving input structure and high-frequency details. During training, we randomly apply the same set of operations to two different image pairs, where one pair is fed into R and the other pair serves as input and target for E. Benefited from the modeling of E and R, retouch attributes can be effectively extracted from a pair of reference images and accurately reproduced on the input image. More experimental details can be found in the Appendix.

This encoder-renderer framework serves as the technical foundation for two critical aspects of our work. First, for the end-to-end training of the VeraRetouch framework (Sec. 3.3), the Retouch Renderer provides a differentiable bridge that allows the VLM to optimize parameters through direct image-based supervision. Second, the encoder-renderer framework enables the large-scale construction of the *AetherRetouch-1M+* dataset (Sec. 3.2). By utilizing the Retouch Encoder to extract control latents from existing expert-annotated pairs, we can synthesize realistic retouching pairs from any high-quality input with the Retouch Renderer, facilitating professional retouching data collection.

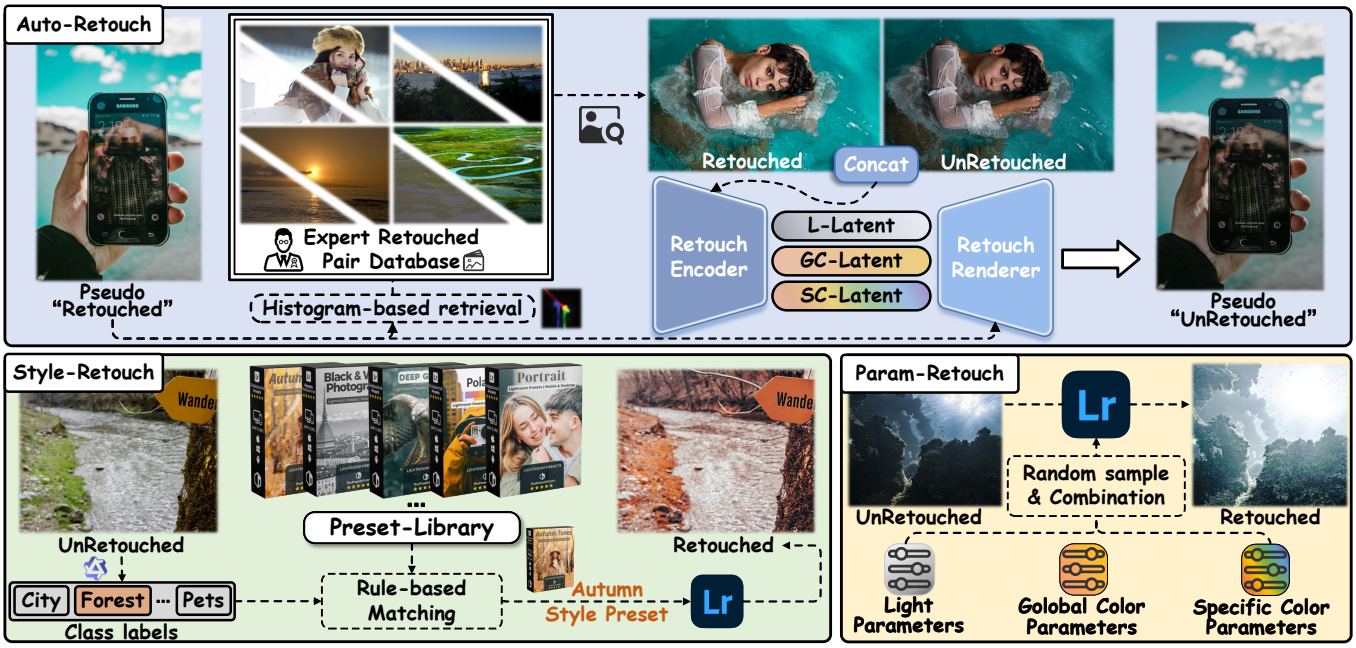

Fig. 3. Data synthesis pipelines for **AetherRetouch-1M+**. Three workflows generate a million-scale multi-task retouching dataset: (1) ***Auto-Retouch***: inverting expert retouching to synthesize pseudo unretouched images from high-quality images; (2) ***Style-Retouch***: applying LightRoom presets via rule-based matching; (3) ***Param-Retouch***: rendering images with randomly sampled LightRoom parameters.

3.2 *AetherRetouch-1M+* Dataset

To address the generalization limitations of existing methods caused by small-scale, narrowly covered datasets (*e.g.*, MIT-Adobe FiveK [Bychkovsky et al. 2011] and PPR10K [Schuhmann et al. 2021]), we construct the *AetherRetouch-1M+* dataset, with over 1 million retouching pairs tailored to three real-world user demands: Uninstructed Automatic Retouching (*Auto-Retouch*), Instructed Style-based Retouching (*Style-Retouch*), and Instructed Parameter-based Retouching (*Param-Retouch*). Fig. 3 presents our three data synthesis pipelines tailored for each retouching scenario.

***Auto-Retouch* Data Generation Pipeline.** This dataset is designed for scenarios where users only provide images without additional instructions. To circumvent the high cost of manual retouching, we adopt an inverse strategy: generating degraded "unretouched" versions from high-quality images. Specifically, we first curate an *Expert Pair Database* by filtering FiveK [Bychkovsky et al. 2011] and PPR10K [Liang et al. 2021] with aesthetic score to retain only high-quality improvements. Then we select high-aesthetic images from a large-scale photography dataset as pseudo "retouched" images. For each pseudo "retouched" image, we retrieve data pairs with the most similar histogram features from the *Expert Retouched Pair Database* as references. Feeding the pseudo "retouched" image and retrieved reference pairs into our Retouch Encoder and Retouch Renderer, the approach inverts expert retouching logic to generate a degraded "unretouched" image, which preserves the content structure of the original high-aesthetic input while embodying realistic flaws.

To further ensure input diversity and enhance generalization, we curate a supplementary dataset by extracting operator ranges and variances from FiveK and PPR10K. By applying randomly sampled operators to high-quality images, we generate disturbed "unretouched" images to expand our training distribution.

***Style-Retouch* Data Generation Pipeline.** For style-based instruction scenarios, we curate 5,030 online presets, categorized into 11 primary and 193 fine-grained subcategories. After sampling images from Unsplash dataset [Unsplash 2024], Qwen2.5-VL [Bai et al. 2025] classifies each image to match appropriate preset categories; one preset is randomly selected and applied via LightRoom API. Qwen3-VL [Wu et al. 2025] then generates **multiple variants of simulated user instructions** through semantic perturbations to expand the instruction diversity.

***Param-Retouch* Data Generation Pipeline.** For operation parameters instruction scenarios, we categorize retouching parameters into light, global color, and specific color adjustments (consistent with Sec. 3.1). Gaussian-random sampled operation parameter combinations are applied to randomly selected images via LightRoom.

Generation of Reasoning Processes. To provide reasoning processes for VLM training, we design a hierarchical structured reasoning process, featuring three key parts: (1) Key elements of image content; (2) A point-by-point issue analysis on the original image from three perspectives: light, global color, and specific color; (3) Detailed retouching plans point-by-point corresponding to above analysis. We feed the retouched image pairs and task requirements into Qwen3-VL [Wu et al. 2025] to simulate the reasoning process.

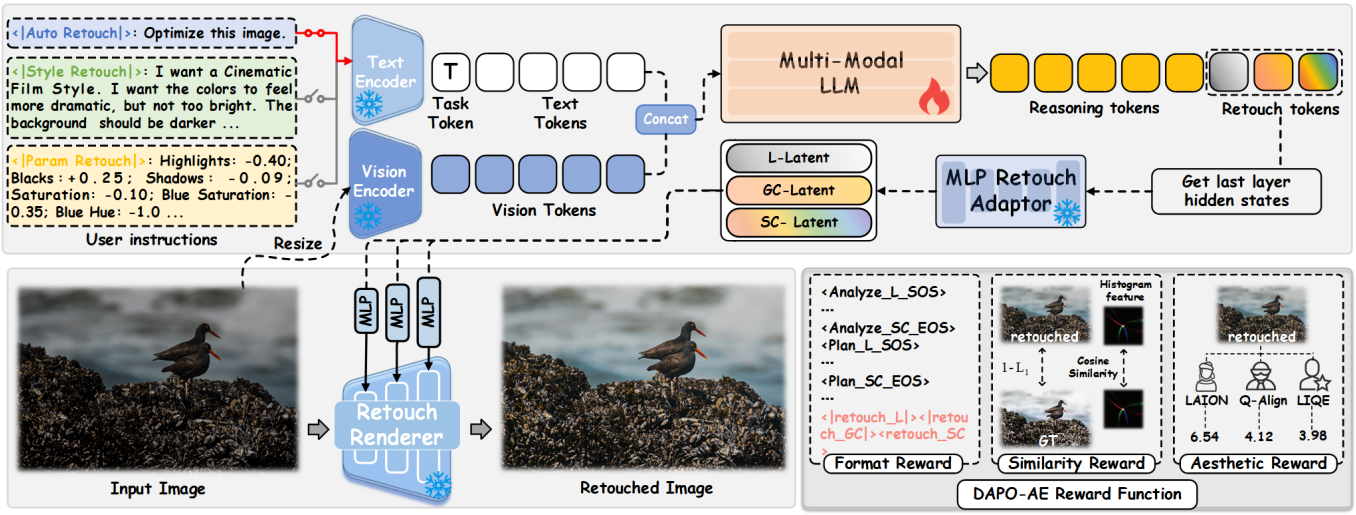

Fig. 4. Overview of the VeraRetouch framework. Our framework processes an image and optional prompts through a compact VLM to generate structured reasoning and disentangled retouching latents. These latents drive a fully differentiable renderer to produce the final enhancement. The bottom-right panel illustrates the components of our DAPO-AE reward functions, designed to optimize the model for both logical consistency and high-level aesthetic appeal.

Fig. 5. Directly training with pre-trained control latents leads to feature-space mismatch and degraded quality (w/o domain align). By introducing a lightweight adaptor and alignment pre-training, we successfully synchronize the latent space with LLM features, yielding results significantly closer to the ground truth (w/ domain align).

3.3 VeraRetouch Framework

As shown in Fig. 4, VeraRetouch consists of a FastViTHD Vision Encoder, Text Encoder, Multi-Modal LLM, MLP Retouch Adaptor, and Retouch Renderer. We build the framework on FastVLM-0.5B [Vasu et al. 2025] to reduce the model size and inference latency. The FastViTHD Vision Encoder encodes the input image into visual tokens, while the Text Encoder converts user instructions into prompt tokens. For user instructions, we design three special tokens: ⟨|Auto Retouch|⟩, ⟨|Style Retouch|⟩, and ⟨|Param Retouch|⟩, which enable task selection (Sec. 3.2) and reduce the model's task discrimination training burden. Then visual tokens are concatenated with prompt tokens, fed into the Multi-Modal LLM, and autoregressively generate reasoning completions; to extract retouching control latents from completions, three additional special *retouch tokens* for light, global color, and specific color adjustment (Sec. 3.1) are designed, their last hidden layer features are fed into the MLP Retouch Adaptor for alignment to generate distangled control latents, which are then used by the Retouch Renderer to convert the input image into the final retouched image.

Domain Align Pretraining. The retouch tokens generated by the Multi-Modal LLM exhibit a substantial distribution mismatch with the pre-trained control latents introduced in Sec. 3.1. As shown in Fig. 5, directly feeding control latents produced by Multi-Modal LLM into Retouch Renderer results in a severe degradation in the quality of the retouched images. To address this issue, we design a simple Retouch Adaptor (three-layer bottleneck MLP) for feature space transformation. Furthermore, starting from the pre-trained model, we freeze the Vision Encoder, and train the remaining components using the *Param-Retouch* dataset. The aligned control latents are obtained by autoregressively generating special *retouch tokens* from given parameters and rendering them into retouched images. For training losses, we adopt cross-entropy loss for token ids ($\mathcal{L}_{\text{CE}}^{\text{text}}$) and L1 loss for image reconstruction ($\mathcal{L}_{1}^{\text{img}}$), with the total loss computed as a weighted sum:

$$\mathcal{L}_{\text{total}} = \alpha \cdot \mathcal{L}_{\text{CE}}^{\text{text}} + \mathcal{L}_{1}^{\text{img}}. \tag{3}$$

Reasoning Supervised Fine-tuning (RSFT). In this stage, we freeze all other modules and only train the Multi-Modal LLM, initialized from its pre-trained weights, which prevents the probability shift during the domain alignment training from affecting multi-task training. The loss function formulation remains consistent with that used in the domain alignment stage. During training, we perform random sampling at an equal ratio across the *Auto-Retouch*, *Style-Retouch*, and *Param-Retouch* datasets. This phase enables the model to: (1) output results in the predefined structured format; (2) learn the causal relationship between reasoning processes and control latents during autoregressive training; (3) acquire precise retouch adjustment capabilities through direct pixel-level supervision.

DAPO-AE Post-Training. The RSFT stage has achieved robust instruction-following capabilities and high retouching quality. To further boost the model's aesthetic perception and elevate the visual appeal of output images, we introduce a Reinforcement Post-Training (RPT) stage, leveraging decoupled clip and dynamic sampling policy optimization [Yu et al. 2025] for aesthetic enhancement (DAPO-AE) to inject refined aesthetic nuances.

Unlike JarvisArt [Lin et al. 2025] employing numerous and complex rewards, our DAPO-AE consists of three simple rewards: The format reward R_f ensures adherence to the structured reasoning template

Table 1. Quantitative Comparison on **FiveK-Bench**. Red and blue indicate the best and second-best results, respectively.

Method	PSNR↑	SSIM↑	LPIPS↓	Hist-L↑	Hist-C↑	Hist-S↑	Hist-M↑	LAION↑	Q-Align↑	LIQE↑	DISTS↓	GMSD↓	TD↓
RSFNet	25.07	0.935	0.056	82.00%	72.08%	79.05%	77.71%	5.02±0.69	4.06±0.40	3.62±0.96	0.044	0.020	0.364
Nano Banana	20.30	0.616	0.137	82.38%	70.18%	63.80%	72.12%	5.18±0.75	4.15±0.39	3.54±0.95	0.075	0.142	1.654
Flux.1 Kontext	25.77	0.896	0.079	88.42%	95.04%	92.79%	92.09%	5.07±0.66	3.99±0.40	3.61±0.96	0.062	0.040	0.730
Qwen-Image-2509	17.81	0.572	0.193	61.58%	66.77%	76.33%	68.23%	4.87±0.62	3.91±0.50	3.13±0.98	0.102	0.164	1.659
MonetGPT	22.91	0.914	0.064	79.30%	65.78%	78.01%	74.36%	4.86±0.64	4.01±0.40	3.54±0.96	0.057	0.023	0.480
JarvisArt	21.52	0.865	0.149	72.74%	60.23%	76.69%	69.89%	5.14±0.61	4.05±0.45	3.02±0.95	0.108	0.039	0.771
Ours-SFT	26.04	0.936	0.053	90.44%	92.71%	95.33%	92.83%	5.13±0.68	4.18±0.39	3.92±0.93	0.040	0.061	0.694
Ours-DAPO-AE	26.85	0.939	0.049	96.35%	94.13%	92.13%	94.20%	5.10±0.68	4.15±0.39	3.88±0.94	0.039	0.045	0.607

and critical retouch tokens. The image similarity reward R_s aligns with the target's retouching trends. The aesthetic reward R_a, activated only for the *Auto-Retouch* task, enhances the visual aesthetic quality of the output. All individual rewards and their constituent scores are normalized to $[0, 1]$ for balanced optimization.

To adapt to different retouch tasks and avoid cross-task interference, we design task-specific reward configurations and training strategies: *Auto-Retouch* uses all three rewards (R_f, R_s, R_a), while *Style-Retouch* and *Param-Retouch* only adopt R_f and R_s. Specifically, we employ random alternating training where each step contains samples from a single task exclusively to mitigate cross-task reward interference in multi-task training. More details are provided in the appendix.

4 Experiments

4.1 Experimental Settings

Datasets. For *Auto-Retouch*, we employ approximately 500 samples from the MIT-Adobe FiveK [Bychkovsky et al. 2011] dataset (**FiveK-Bench**) for real-world performance evaluation. Additionally, we construct a synthetic dataset of 250 images (**Aether-Bench (Auto)**), subjected to randomized operational perturbations, to assess the model's generalization under complex and varied inputs. For *Style-Retouch*, we select presets that are outside the training distribution to generate 100 pairs of test samples (**Aether-Bench (Style)**). Finally, for *Param-Retouch*, we curate a test set of 350 samples (**Aether-Bench (Param)**) by applying seven distinct retouching protocols to 50 unseen images, with the specific parameters for each protocol randomly sampled from Gaussian distributions during application.

Metrics. To evaluate the fidelity of predictions relative to the ground truth, we employ PSNR, SSIM, and LPIPS. We further utilize histogram intersections to measure the distributional consistency of contrast, luminance, and color saturation. To assess texture preservation and retouching consistency, we adopt DISTS [Ding et al. 2020], GMSD [Xue et al. 2013], and TD [Dong et al. 2024] metrics. Image aesthetics and perceptual quality are quantified using LAION [Schuhmann and Beaumont 2022], Q-Align [Wu et al. 2023], and LIQE [Zhang et al. 2023b]. Following the evaluation protocol of MonetGPT [Dutt et al. 2025] for the FiveK-Bench, we report the maximum score across the five expert retouches for PSNR, SSIM, LPIPS, and DISTS, while histogram intersections are computed by considering the aggregate distribution across all experts.

Table 2. Quantitative evaluation on **Aether-Bench**. Each sub-task (Auto, Style, and Param) features its unique set of evaluation metrics. **Rea.** denotes reasoning capability.

Method	**Rea.**	**Experimental Results**						
		Aether-Bench (Auto)						
		Hist-M↑	**LAION↑**	**Q-Align↑**	**LIQE↑**	**DISTS↓**	**GMSD↓**	**TD↓**
RSFNet	×	89.17%	6.85	4.23	3.22	0.059	0.030	0.343
Nano Banana	✓	86.39%	6.83	4.26	3.26	0.088	0.143	1.547
Flux.1 Kontext	×	89.81%	6.76	4.25	3.31	0.063	0.039	0.665
Qwen-Image-2509	×	79.65%	6.51	4.02	2.87	0.120	0.161	1.523
MonetGPT	✓	85.03%	6.36	4.01	2.91	0.104	0.038	0.536
JarvisArt	✓	81.14%	6.34	4.06	2.60	0.123	0.043	0.709
Ours-SFT	✓	88.55%	6.83	4.27	3.30	0.061	0.035	0.435
Ours-DAPO-AE	✓	89.59%	6.82	4.25	3.28	0.055	0.026	0.360
		Aether-Bench (Style)						
		L_1 ↓	**PSNR↑**	**SSIM↑**	**LPIPS↓**	**DISTS↓**	**GMSD↓**	**TD↓**
Nano Banana	✓	0.125	16.66	0.596	0.242	0.138	0.151	1.579
Flux.1 Kontext	×	0.094	19.48	0.831	0.162	0.106	0.048	0.741
Qwen-Image-2509	×	0.158	14.34	0.494	0.289	0.196	0.174	1.725
JarvisArt	✓	0.147	15.72	0.677	0.288	0.170	0.100	1.235
Ours-SFT	✓	0.097	19.73	0.839	0.149	0.100	0.039	0.592
Ours-DAPO-AE	✓	0.092	20.12	0.847	0.145	0.099	0.036	0.526
		Aether-Bench (Param)						
		L_1 ↓	**PSNR↑**	**SSIM↑**	**LPIPS↓**	**DISTS↓**	**GMSD↓**	**TD↓**
Flux.1 Kontext	×	0.140	18.51	0.783	0.204	0.136	0.008	0.468
Qwen-Image-2509	×	0.283	13.55	0.380	0.484	0.257	0.198	1.843
Ours-SFT	✓	0.023	30.39	0.946	0.039	0.040	0.071	0.664
Ours-DAPO-AE	✓	0.024	30.18	0.947	0.041	0.042	0.067	0.644

Baselines. We compare our method against several state-of-the-art baselines, including RSFNet [Ouyang et al. 2023], Nano-Banana [Comanici et al. 2025], Flux.1 Kontext [Labs et al. 2025], Qwen-Image-2509 [Wu et al. 2025], MonetGPT [Dutt et al. 2025], and JarvisArt [Lin et al. 2025]. Specifically, we retrain RSFNet on the *Auto-Retouch* dataset. Furthermore, we implement LoRA fine-tuning for Flux.1 Kontext and Qwen-Image-2509 on our full AetherRetouch-1M+ dataset following the DiffSynth-Studio implementation.

4.2 Comparison

Quantitative Comparison. As shown in Tab. 1 and 2, VeraRetouch consistently achieves state-of-the-art performance across all benchmarks. For ***Auto-Retouch***, it reaches a peak PSNR of 26.85 dB on FiveK-Bench, outperforming Flux.1 Kontext by 1.08 dB and securing top aesthetic scores in Q-Align and LIQE. In ***Style-Retouch***, our model strikes a superior balance between visual enhancement and texture preservation, yielding the lowest Texture Distortion (TD: 0.526) and effectively suppressing generative artifacts. Furthermore, in ***Param-Retouch***, our method achieves a remarkable PSNR of 30.18 dB, significantly surpassing the fine-tuned diffusion baseline.

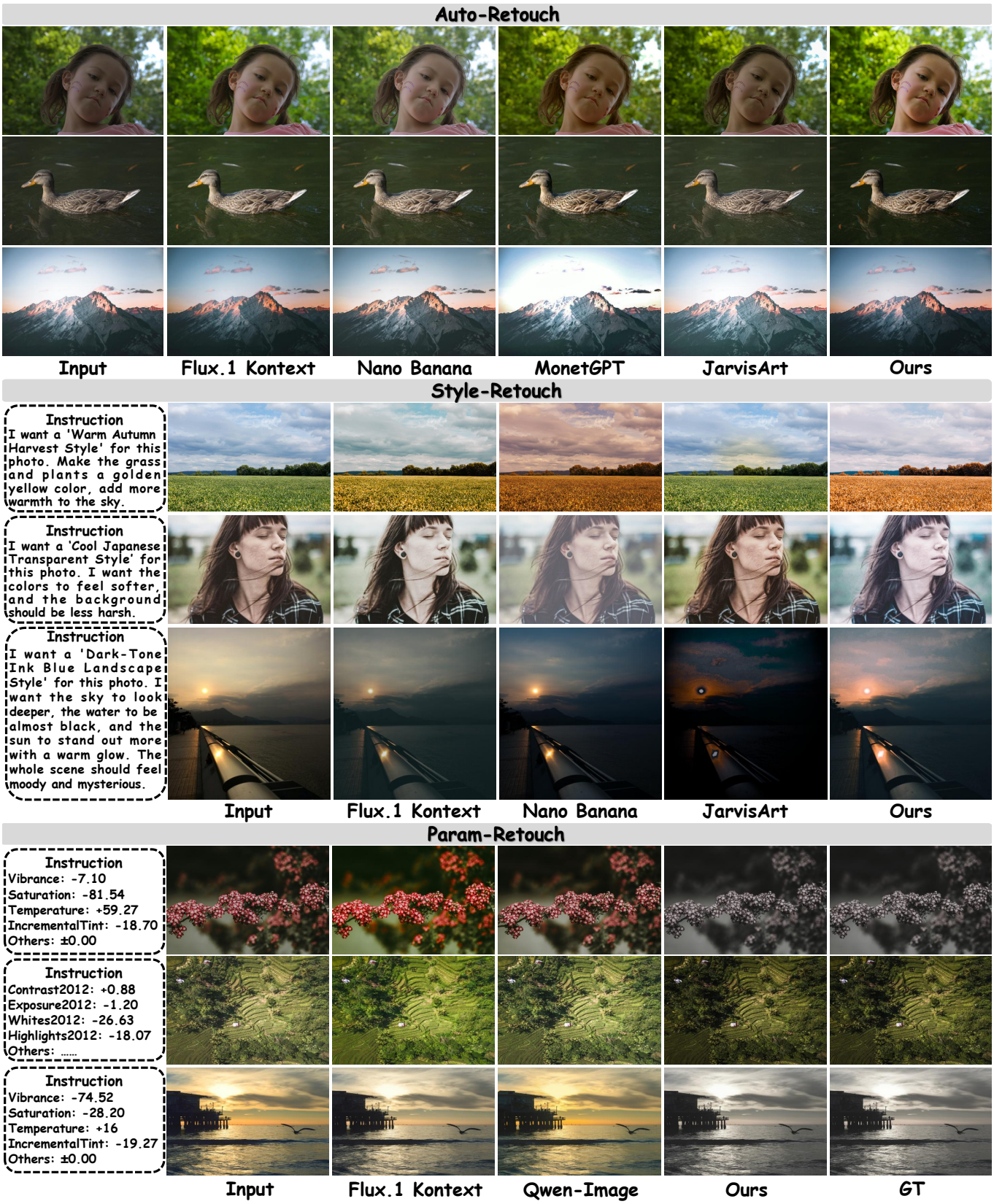

Fig. 6. Visual comparison with baseline methods on ***Auto-Retouch*** (image-only), ***Style-Retouch*** (text-guided), and ***Param-Retouch*** (parameter-driven).

Inference Time. We evaluate the efficiency of VeraRetouch by measuring the average inference time per image on a single NVIDIA H20 GPU with a batch size of 1 over a test set of 100 512p images. As shown in Tab. 3, our framework takes only 6.9s to process a single image, outperforming diffusion-based methods like Flux.1 Kontext (~16.8s) and large-scale agents such as JarvisArt (~14.3s) with a significant speedup of ~2.5×. Furthermore, we extend our evaluation to edge devices, including a Macbook Air (M4) and an iPhone 16 Pro. As reported in the last two rows of Tab. 3, our model achieves satisfactory inference speeds of 7.4s and 13.5s respectively, demonstrating the exceptional efficiency and deployment potential of our framework on consumer-grade hardware.

Table 3. Comparison on **model size** and **per-image latency**. Total Time denotes end-to-end latency, split into VLM Time (backbone) and Other Time (renderer/LightRoom/other tools, I/O).

Method	Task	Device	Params↓	Total Time↓	VLM Time	Other Time
Flux.1 Kontext	*Auto*	H20	16.87B	16.78s	–	–
Qwen-Image-2509	*Auto*	H20	28.85B	48.77s	–	–
MonetGPT	*Auto*	H20	8.29B	44.33s	28.69s	15.64s
JarvisArt	*Auto*	H20	8.29B	14.31s	14.11s	0.20s
Ours	*Auto*	H20	**0.63B**	**6.90s**	**6.86s**	**0.04s**
Ours	*Style*	H20	**0.63B**	**3.83s**	**3.78s**	**0.05s**
Ours	*Param*	H20	**0.63B**	**5.17s**	**5.14s**	**0.03s**
Ours	*Auto*	Macbook Air(M4)	**0.63B**	**7.46s**	**6.69s**	**0.77s**
Ours	*Auto*	iPhone16 pro	**0.63B**	**13.56s**	**11.58s**	**1.98s**

Qualitative Comparison. Fig. 6 demonstrates the visual superiority of VeraRetouch across three retouching tasks. To further validate these results, we conducted a user study with 38 participants. We collected blind rankings of model outputs and converted them into scores on a scale of 1 to 5 (higher is better). We randomly selected 10 images each from the *Auto-Retouch* and *Style-Retouch* test sets; the former was evaluated on *visual aesthetics* and *texture consistency*, while the latter focused on *instruction alignment*. As shown in Fig. 7, VeraRetouch consistently receives the highest scores in Aesthetics, Prompt Fidelity, and Texture Consistency. These results confirm that our method aligns more closely with human preferences and intent while better preserving original image content.

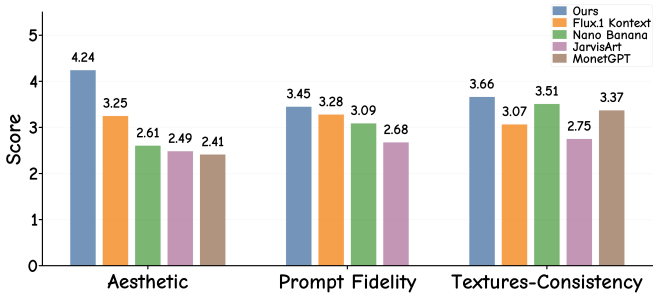

Fig. 7. User study results on Aesthetics (visual appeal), Prompt Fidelity (instruction alignment), and Texture Consistency (detail preservation).

4.3 Ablation Study

Latent-Prediction. We evaluate the effectiveness of our control latent prediction against direct parameter prediction on the MIT-Adobe FiveK Expert-C dataset. While the former utilizes a Retouch Renderer to interpret continuous latents, the latter predicts discrete LightRoom parameters integrated via the LightRoom API. As shown in Tab. 4, the latent-prediction approach consistently outperforms the parameter-prediction baseline across all quantitative metrics. This advantage is primarily attributed to the direct gradient back-propagation enabled by the differentiable renderer, which allows the VLM to bypass the discretization gap of traditional APIs and learn more precise, pixel-level aesthetic adjustments.

Table 4. Effectiveness of latent-prediction ablation study results, evaluated on the MIT Adobe-FiveK expert-C test dataset.

Method	L_1 ↓	PSNR↑	SSIM↑	LPIPS↓	DISTS↓
params-pred	0.125	18.07	0.800	0.155	0.086
latents-pred	**0.061**	**24.11**	**0.905**	**0.057**	**0.042**

Data Scaling. To assess the data scalability effect of VeraRetouch, we evaluate VeraRetouch on *Auto-Retouch* with 5%, 20%, and 100% training data. Quantitative metrics in Tab. 5 exhibit a consistent upward trend as the dataset expands. This improvement demonstrates that our model effectively leverages larger-scale data to refine its aesthetic reasoning and retouching precision. Additionally, our unified version (joint training on all three tasks without user instruction perturbations) achieves the best fidelity and consistency, verifying that improvements come from both larger data scale and multi-task supervision.

Table 5. Data-scaling ablation study, evaluated on **FiveK-Bench**.

Scale	PSNR↑	SSIM↑	LPIPS↓	Hist-M↑	LAION↑	Q-Align↑	LIQE↑
5%	25.39	0.931	0.058	87.81%	5.01±0.66	4.09±0.39	3.67±0.94
20%	26.06	0.935	0.052	94.49%	5.08±0.68	4.16±0.39	3.86±0.94
100%	26.57	0.935	0.052	94.46%	5.12±0.68	4.17±0.39	3.90±0.97
Unified	26.81	0.939	0.050	94.54%	5.10±0.68	4.16±0.39	3.89±0.95

DAPO-AE. Tables 1 and 2 compare the SFT baseline and the DAPO-AE training scheme. Despite marginal quantitative gains from the additional DAPO-AE stage, it plays a crucial role in refining model performance. We observe that DAPO-AE specifically benefits those challenging samples where the SFT model produces suboptimal reasoning and aesthetic results. Visual comparisons and preference user study are provided Appendix.

Disentanglement Ability. To verify the decoupling of operator categories(Light, Global Color and Specific Color), we performed an intervention study on 50 external images. We synthesized 350 Ground-Truth(GT) references by applying Gaussian-sampled parameters in various combinations. During inference, we provided all parameters while masking specific retouching latents to observe the model's isolation capability. As shown in Tab. 6, our method maintains an average PSNR>28 with corresponding GTs across all mask scenarios, confirming that the latent space for each operator category is effectively disentangled and independent. These quantitative findings are further corroborated by qualitative results. Such quantitative results are consistent with qualitative observations in Fig. 8 on MIT-Adobe FiveK. Masking L-Latent only changes illumination and preserves original color attributes. Masking GC-Latent adjusts global color tones without affecting lighting, and SC-Latent masking

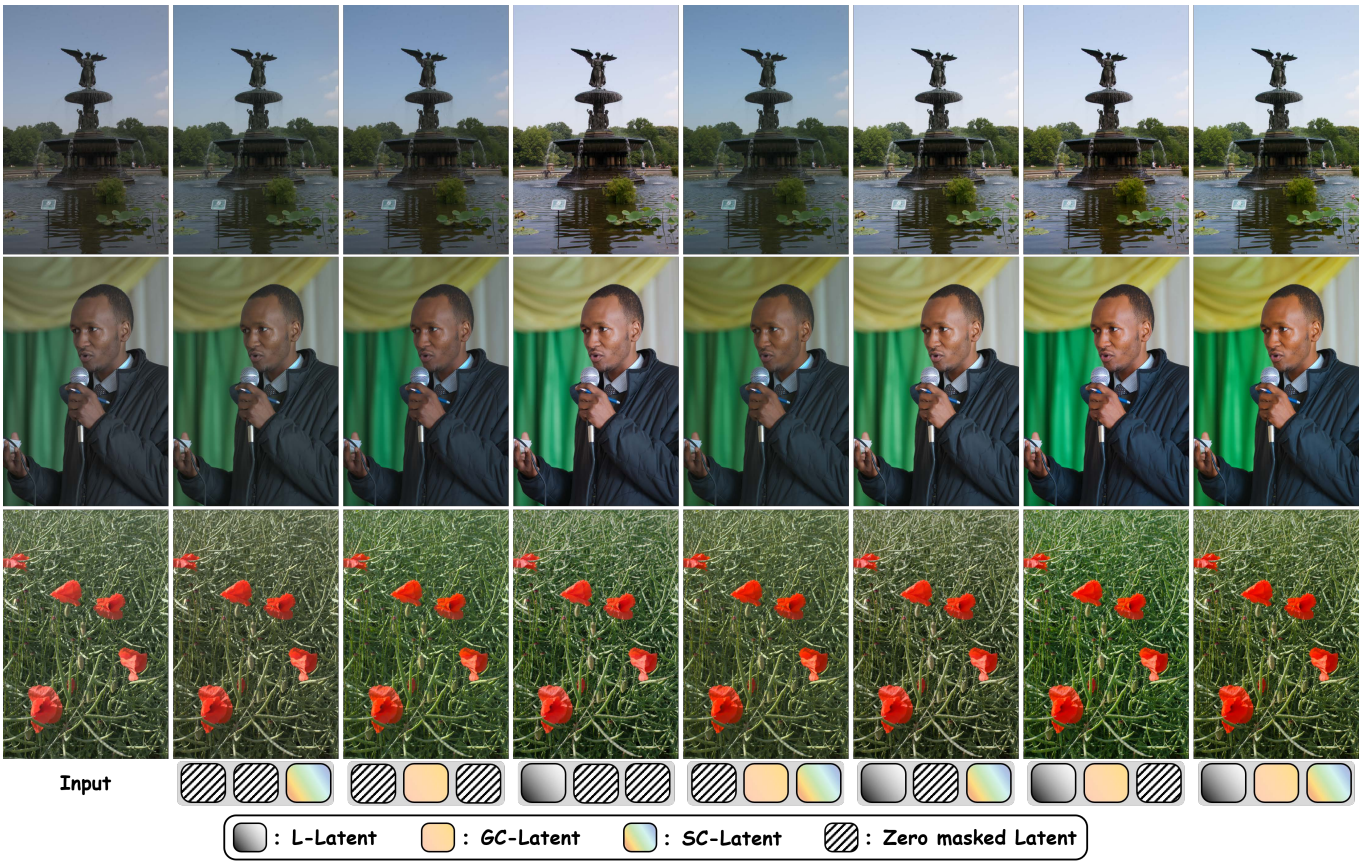

Fig. 8. To demonstrate the disentangling capability of our retouch renderer, we apply zero masking to individual control latents during the Auto-Retouch process on MIT-Adobe FiveK samples. The resulting images illustrate the independent impact of each latent—Lighting (L), Global Color (GC), and Specific Color (SC)—demonstrating the effective decoupling of our differentiable renderer.

selectively modulates local color components while maintaining overall color balance.

Table 6. Ablation study on the decoupling of operator categories. We report the performance under various mask interventions for *L*, *GC*, and *SC* latents.

mask-L	mask-GC	mask_SC	L1↓	PSNR↑	SSIM↑	LPIPS↓
×	✓	✓	0.025	30.38	0.939	0.033
✓	×	✓	0.024	29.61	0.947	0.037
✓	✓	×	0.053	28.12	0.925	0.030
×	×	✓	0.024	30.67	0.947	0.034
×	✓	×	0.030	28.20	0.917	0.060
✓	×	×	0.026	29.20	0.942	0.045
×	×	×	0.027	29.19	0.928	0.053

5 Conclusion

We present VeraRetouch, a new framework that integrates a 0.5B VLM with a fully differentiable Retouch Renderer for reasoning photo retouching. By formulating retouching as a structured autoregressive task, our method effectively bridges the gap between high-level retouching adjustment texts and low-level pixel adjustments. Supported by our million-scale AetherRetouch-1M+ dataset, extensive results demonstrate that even a lightweight model can achieve superior performance through meticulous data curation and model design, highlighting its potential for mobile deployment.

Limitations and Future Work. The current model still exhibits constrained capabilities in local retouching. In future work, we plan to enhance the flexibility of localized editing by incorporating pixel-wise mask mechanisms into the framework, enabling more precise and region-specific image manipulation.

6 Acknowledgments

We would like to thank Temesgen Muruts Weldengus, Fei Kou, Liqi Xue, Shiyang Li, and Yunlong Lin for their help with this work. In particular, we sincerely appreciate the photographer Yijing Chen for providing partial photographic materials used in this study. This research was supported by Zhejiang Provincial Natural Science Foundation of China under Grant No. LD24F020007.

References

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923* (2025).

Tim Brooks, Aleksander Holynski, and Alexei A Efros. 2023. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 18392–18402.

Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. 2011. Learning Photographic Global Tonal Adjustment with a Database of Input / Output Image Pairs. In *The Twenty-Fourth IEEE Conference on Computer Vision and Pattern Recognition*.

Yoav Chai, Raja Giryes, and Lior Wolf. 2020. Supervised and unsupervised learning of parameterized color enhancement. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 992–1000.

Haoyu Chen, Keda Tao, Yizao Wang, Xinlei Wang, Lei Zhu, and Jinjin Gu. 2025. PhotoArtAgent: Intelligent Photo Retouching with Language Model-Based Artist Agents. *arXiv preprint arXiv:2505.23130* (2025).

Yu-Sheng Chen, Yu-Ching Wang, Man-Hsin Kao, and Yung-Yu Chuang. 2018. Deep photo enhancer: Unpaired learning for image enhancement from photographs with gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 6306–6314.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261* (2025).

Yubin Deng, Chen Change Loy, and Xiaoou Tang. 2018. Aesthetic-driven image enhancement by adversarial learning. In *Proceedings of the 26th ACM international conference on Multimedia*. 870–878.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P Simoncelli. 2020. Image quality assessment: Unifying structure and texture similarity. *IEEE transactions on pattern analysis and machine intelligence* 44, 5 (2020), 2567–2581.

Yi Dong, Yuxi Wang, Zheng Fang, Wenqi Ouyang, Xianhui Lin, Zhiqi Shen, Peiran Ren, Xuansong Xie, and Qingming Huang. 2024. MovingColor: Seamless Fusion of Fine-grained Video Color Enhancement. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 7454–7463.

Niladri Shekhar Dutt, Duygu Ceylan, and Niloy J Mitra. 2025. MonetGPT: Solving Puzzles Enhances MLLMs' Image Retouching Skills. *ACM Transactions on Graphics (TOG)* 44, 4 (2025), 1–12.

Michaël Gharbi, Jiawen Chen, Jonathan T Barron, Samuel W Hasinoff, and Frédo Durand. 2017. Deep bilateral learning for real-time image enhancement. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–12.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Identity mappings in deep residual networks. In *European conference on computer vision*. Springer, 630–645.

Yuanming Hu, Hao He, Chenxi Xu, Baoyuan Wang, and Stephen Lin. 2018. Exposure: A white-box photo post-processing framework. *ACM Transactions on Graphics (TOG)* 37, 2 (2018), 1–17.

Hanul Kim, Su-Min Choi, Chang-Su Kim, and Yeong Jun Koh. 2021. Representative color transform for image enhancement. In *Proceedings of the IEEE/CVF international conference on computer vision*. 4459–4468.

Han-Ul Kim, Young Jun Koh, and Chang-Su Kim. 2020. PieNet: Personalized image enhancement network. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16*. Springer, 374–390.

Dario Kneubuehler, Shuhang Gu, Luc Van Gool, and Radu Timofte. 2020. Flexible example-based image enhancement with task adaptive global feature self-guided network. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*. Springer, 343–358.

Satoshi Kosugi and Toshihiko Yamasaki. 2020. Unpaired image enhancement featuring reinforcement-learning-controlled image editing software. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 11296–11303.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. 2025. FLUX. 1 Kontext: Flow Matching for In-Context Image Generation and Editing in Latent Space. *arXiv preprint arXiv:2506.15742* (2025).

Jie Liang, Hui Zeng, Miaomiao Cui, Xuansong Xie, and Lei Zhang. 2021. Ppr10k: A large-scale portrait photo retouching dataset with human-region mask and group-level consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 653–661.

Tianwei Lin, Honglin Lin, Fu Li, Dongliang He, Wenhao Wu, Meiling Wang, Xin Li, and Yong Liu. 2023. Adacm: adaptive colormlp for real-time universal photo-realistic style transfer. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 37. 1613–1621.

Yunlong Lin, Zixu Lin, Kunjie Lin, Jinbin Bai, Panwang Pan, Chenxin Li, Haoyu Chen, Zhongdao Wang, Xinghao Ding, Wenbo Li, et al. 2025. JarvisArt: Liberating Human Artistic Creativity via an Intelligent Photo Retouching Agent. *arXiv preprint arXiv:2506.17612* (2025).

Sean Moran, Pierre Marza, Steven McDonagh, Sarah Parisot, and Gregory Slabaugh. 2020. Deeplpf: Deep local parametric filters for image enhancement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 12826–12835.

Temesgen Muruts Weldengus, Binnan Liu, Fei Kou, Youwei Lyu, Jinwei Chen, Qingnan Fan, and Changqing Zou. 2025. InstantRetouch: Personalized Image Retouching without Test-time Fine-tuning Using an Asymmetric Auto-Encoder. *arXiv e-prints* (2025), arXiv–2602.

Wenqi Ouyang, Yi Dong, Xiaoyang Kang, Peiran Ren, Xin Xu, and Xuansong Xie. 2023. Rsfnet: A white-box image retouching approach using region-specific color filters. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 12160–12169.

Zhaoqing Pan, Feng Yuan, Jianjun Lei, Wanqing Li, Nam Ling, and Sam Kwong. 2021. MIEGAN: Mobile image enhancement via a multi-module cascade neural network. *IEEE Transactions on Multimedia* 24 (2021), 519–533.

Jongchan Park, Joon-Young Lee, Donggeun Yoo, and In So Kweon. 2018. Distort-and-recover: Color enhancement using deep reinforcement learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5928–5936.

Christoph Schuhmann and Romain Beaumont. 2022. LAION-AESTHETICS. Technical report and blog post. https://laion.ai/blog/laion-aesthetics/

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. 2021. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114* (2021).

David Serrano-Lozano, Luis Herranz, Michael S Brown, and Javier Vazquez-Corral. 2024. NamedCurves: Learned Image Enhancement via Color Naming. In *European Conference on Computer Vision*. Springer, 92–108.

Unsplash. 2024. Unsplash Dataset. https://unsplash.com/data. Accessed: 2025-06-20.

Pavan Kumar Anasosalu Vasu, Fartash Faghri, Chun-Liang Li, Cem Koc, Nate True, Albert Antony, Gokula Santhanam, James Gabriel, Peter Grasch, Oncel Tuzel, et al. 2025. Fastvlm: Efficient vision encoding for vision language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 19769–19780.

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

Haoning Wu, Zicheng Zhang, Weixia Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Yixuan Gao, Annan Wang, Erli Zhang, Wenxiu Sun, et al. 2023. Q-align: Teaching lmms for visual scoring via discrete text-defined levels. *arXiv preprint arXiv:2312.17090* (2023).

Wufeng Xue, Lei Zhang, Xuanqin Mou, and Alan C Bovik. 2013. Gradient magnitude similarity deviation: A highly efficient perceptual image quality index. *IEEE transactions on image processing* 23, 2 (2013), 684–695.

Canqian Yang, Meiguang Jin, Xu Jia, Yi Xu, and Ying Chen. 2022. AdaInt: Learning adaptive intervals for 3D lookup tables on real-time image enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 17522–17531.

Sidi Yang, Binxiao Huang, Mingdeng Cao, Yatai Ji, Hanzhong Guo, Ngai Wong, and Yujiu Yang. 2024. Taming Lookup Tables for Efficient Image Retouching. In *European Conference on Computer Vision*. Springer, 144–159.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. 2025. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476* (2025).

Runsheng Yu, Wenyu Liu, Yasen Zhang, Zhi Qu, Deli Zhao, and Bo Zhang. 2018. Deepexposure: Learning to expose photos with asynchronously reinforced adversarial learning. *Advances in neural information processing systems* 31 (2018).

Hui Zeng, Jianrui Cai, Lida Li, Zisheng Cao, and Lei Zhang. 2020. Learning image-adaptive 3d lookup tables for high performance photo enhancement in real-time. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 4 (2020), 2058–2073.

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. 2023a. Magicbrush: A manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems* 36 (2023), 31428–31449.

Weixia Zhang, Guangtao Zhai, Ying Wei, Xiaokang Yang, and Kede Ma. 2023b. Blind image quality assessment via vision-language correspondence: A multitask learning perspective. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 14071–14081.