IRPO: Boosting Image Restoration via Post-training GRPO

Haoxuan Xu[1,2,†] Yi Liu[4,†] Tianfu Li[1] Ruolin Shen[5] Boyuan Jiang[6]
Jinlong Peng[7] Donghao Luo[8] Xiaobin Hu[3*] Shuicheng Yan[3] Haoang Li[1]

[1]The Hong Kong University of Science and Technology (Guangzhou)
[2]Shenzhen Loop Area Institute [3]National University of Singapore
[4]Tsinghua University [5]Technical University of Munich
[6]Zhejiang University [7]Shanghai Jiao Tong University [8]Fudan University

Abstract

Post-training has become effective for high-level generation, but its role in low-level vision remains underexplored. Existing image restoration methods often rely on fixed pixel-wise fitting to ground-truth images, which can lead to over-smoothing and weak generalization. We propose IRPO, a GRPO-based post-training framework for deterministic restoration models. IRPO is built around two axes: data formulation and reward modeling. For data formulation, we select the 30% underperforming samples from the pre-training stage, which improves both accuracy and training efficiency. For reward modeling, we combine fidelity-oriented and quality-aware feedback with three components: a General Reward for structural fidelity, an Expert Reward that uses a Vision-Language Model as a coarse visual-quality judge, and a Restoration Reward for task-specific low-level cues. Experiments on six in-domain and five out-of-domain (OOD) benchmarks show that IRPO improves the AdaIR baseline by 0.93 dB on in-domain tasks and 3.43 dB on OOD settings.

1 Introduction

Recent generative models cover diverse tasks, including virtual try-on with VTON-HandFit [44], portrait animation with Sonic [28], and controllable 3D asset generation with Hunyuan3D-Omni [25]. Post-training offers a way to refine pre-trained models using selected data and reward signals [33]. For example, DDPO [6] optimizes image generation against downstream rewards, while T2V-Turbo-v2 [38] combines data selection, reward feedback, and conditional guidance for video generation. Adversarial post-training also improves efficient video generation [45, 46]. The progress of post-training in high-level tasks [6, 38] motivates a related question for low-level vision. Image restoration (IR) aims to recover faithful, natural-looking images from degraded inputs. Quality-aware approaches include fine-grained image quality assessment for perceptual restoration [64] and IQA-driven RL for mixed-distortion restoration [91]. The key question is how to adapt post-training to IR, where paired supervision remains essential but the optimization signal should better handle hard cases and real-world degradations.

Existing restoration methods employ different architectural and training choices. **1) End-to-end supervised learning**: a network learns a mapping from degraded inputs to clean targets [27]. **2) Multi-stage designs**: MPRNet and Diff-Mamba use successive restoration stages [47, 80] to refine restored outputs. **3) Domain-specific modeling and adaptation**: methods such as APISR for anime super-resolution [68] and MoA-VR for video restoration [49] tailor models to their target domains. These choices provide complementary ways to improve restoration; our focus is how to post-train

*Corresponding author.
†Haoxuan Xu and Yi Liu contributed equally.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

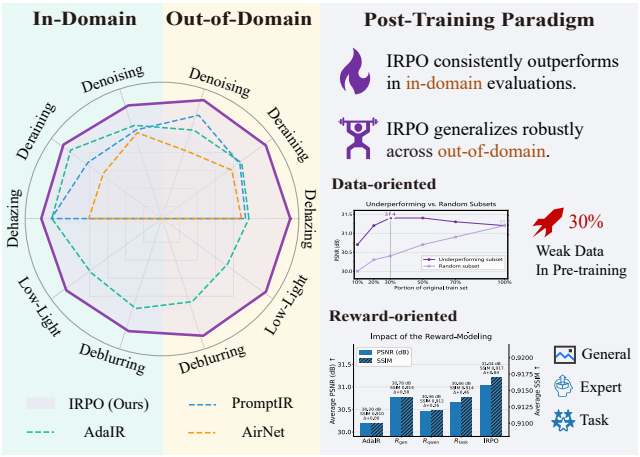


Figure 1: Overview of IRPO and its empirical behavior. (Left) Average PSNR comparison across in-domain and OOD settings. (Right) The two core components of IRPO: data-oriented supervision on the 30% underperforming subset, and reward-oriented optimization with complementary restoration rewards.

an existing restoration model using selected paired data and reward feedback. Recent work also explores training-time privileged information and iterative refinement for all-in-one restoration, as in SIPL [72]. More broadly, minimizing distortion alone does not ensure high perceptual quality, as formalized by the perception–distortion tradeoff [7]. This motivates studying complementary optimization signals for restoration.

These limitations motivate a post-training objective beyond fixed pixel losses. Many useful restoration criteria, such as perceptual similarity, visual plausibility, and degradation-specific quality, cannot be reduced to a single differentiable pixel loss. Reinforcement Learning (RL), including Group Relative Policy Optimization (GRPO) [63], offers a framework for reward-based policy optimization. We investigate its application to IR post-training. The design needs two components. **1) Data formulation**: unlike high-level tasks where multiple outputs can be ranked by preference, IR requires a stable rule for selecting samples that are worth post-training across degradations and domains. Paired data are also expensive to collect [34], while synthetic degradations may not cover real-world cases. **2) Reward modeling**: perceptual losses provide feature-based supervision beyond pixel-wise error [30]. Our goal is to complement supervision with rewards for structural fidelity, visual quality, and degradation-specific residuals.

To address these issues, we propose **I**mage **R**estoration with GR**PO**-based post-training (IRPO), a framework built on data-oriented supervision and reward-oriented optimization (Fig. 1). We first study data formulation through random sampling and performance-based selection. This leads to a simple data-oriented principle: post-training on the 30% underperforming samples from the pre-training stage gives a favorable balance between performance and efficiency. Next, low-level tasks differ substantially in degradation type, and their optimization objectives must preserve fidelity while also capturing quality cues that are not fully reflected by a single pixel loss. To this end, we build a reward system with three parts: (1) a General Reward for structural fidelity and broad quality; (2) an Expert Reward that uses a Vision-Language Model (VLM) as a coarse visual-quality judge; and (3) a Restoration Reward for degradation-dependent low-level cues. Finally, motivated by all-in-one IR [13, 56], we evaluate GRPO post-training under mixed-degradation training to test its adaptability to real-world settings.

In summary, our contributions are as follows:

- We introduce a low-level GRPO-based post-training framework that jointly studies data formulation and reward modeling for image restoration.
- We identify a performance-based data principle that directs post-training to hard samples without requiring additional data.
- We design a reward-oriented optimization scheme that combines fidelity, coarse visual-quality feedback, and degradation-specific restoration cues.

- We evaluate IRPO on six in-domain and five out-of-domain benchmarks, where it improves the AdaIR backbone and shows consistent robustness in real-world settings.

2 Related Work

2.1 Low-Level Training Paradigm

Related restoration approaches differ in their architectures and adaptation strategies: 1) Single-stage Training. Models are trained end-to-end to map degraded inputs to clean outputs. Early CNN-based restorers such as DnCNN [87] and SRCNN [17], and later transformer variants like SwinIR [42] and Restormer [81], demonstrate that strong architectures can learn the restoration mapping efficiently. 2) Multi-stage Designs. Multi-stage architectures such as MPRNet [80] and Diff-Mamba [47] refine restoration through successive stages. Related multi-scale architectures, such as MIRNet-v2 [82], aggregate features across resolutions. These designs can improve restoration quality, but introduce additional architectural or training complexity. 3) Domain-specific Modeling and Adaptation. Pre-trained models can be adapted to specific domains, as in FeRA [78] for diffusion adaptation. Domain-specific restoration also uses tailored data and priors: CBDNet [21] targets real camera noise, ESRGAN [70] perceptual super-resolution, and APISR [68] anime super-resolution. Other examples include 3D facial priors for face super-resolution and restoration [22, 23], and video restoration systems such as VRT [43] and MoA-VR [49]. These approaches motivate adaptation to the target degradation, although their training data and objectives may not transfer across domains.

Restoration objectives must account for the perception–distortion tradeoff [7]; perceptual losses offer one way to move beyond pixel-wise error [30]. We study a GRPO-based post-training framework for low-level tasks. Its two core components are data-oriented supervision and reward-oriented optimization, which are designed to improve robustness under diverse real-world degradations.

2.2 Reinforcement Learning

RL-based post-training improves LLM reasoning in OpenAI's o1 [55] and DeepSeek-R1 [15]. RLHF [5] commonly uses an actor–critic approach, whereas GRPO, introduced in DeepSeek-Math [63], estimates relative advantages within each sampled group without a separate value model. Direct Preference Optimization (DPO) [60] instead learns from preference pairs without an explicit RL optimization loop. GRPO-style methods extend to multimodal reasoning in R1-V [9], R1-OneVision [75], and R1-VL [86], multimodal table understanding [31], and action or navigation tasks in VLA-R1 [77] and VLN-R1 [57]. In related multimodal settings, SemMSA [41] and VT-FSL [39] use semantic guidance for sentiment analysis and few-shot recognition, while DVLA-RL [40] uses RL-based gating for vision–language alignment.

Despite these advances, GRPO-style post-training remains underexplored for low-level restoration. This setting raises questions about paired data [34], deterministic restoration backbones, and suitable rewards. In particular, restoration rewards must balance perceptual quality and fidelity, consistent with the perception–distortion tradeoff [7]. Reward optimization should therefore complement, rather than replace, supervised constraints. We study this setting through the two central questions of IRPO: data formulation and reward modeling.

3 Methodology

3.1 A Post-Training Paradigm for Low-Level Vision

The prevailing training scheme for image restoration is end-to-end supervision. This approach has two main limitations. First, it treats all data points equally and may underfit complex, long-tail degradations. Second, it relies on deterministic pixel-level objectives, which can produce overly smooth results and gives limited feedback about degradation-specific artifacts.

We therefore add a focused post-training phase after the initial restoration model is trained. The goal is to improve robustness while preserving restoration fidelity. As illustrated in Fig. 2, IRPO has two main pillars: Data-Oriented Supervision, which focuses on underperforming data (Sec. 3.2), and Reward-Oriented Optimization, which incorporates quality-aware and task-aware feedback (Sec. 3.3).

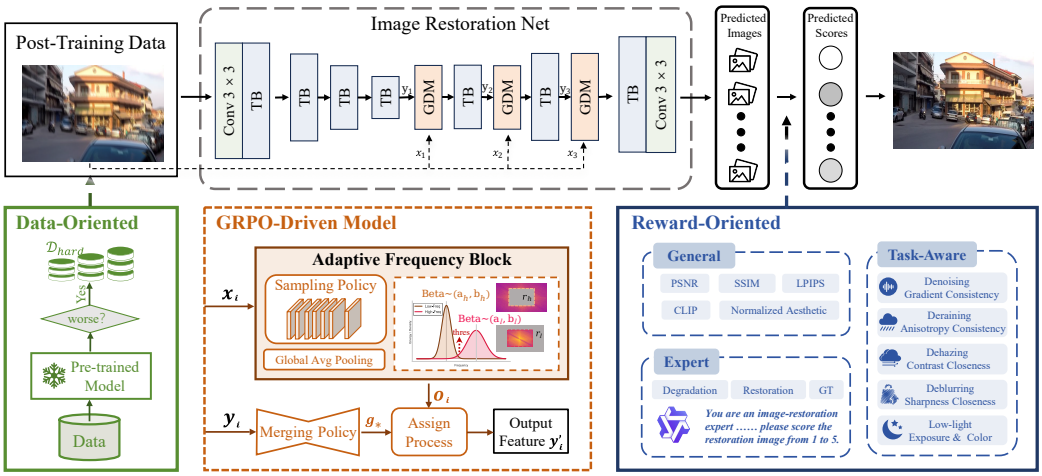


Figure 2: Overview of IRPO. Pillar 1 (Data-Oriented, left): a pre-trained model evaluates the full dataset and curates $\mathcal{D}_{\text{hard}}$ for post-training. Pillar 2 (Reward-Oriented, right): a multi-component reward model (General, Expert, Task-Aware) trains the policy π_θ. Lightweight GRPO-Driven Model (GDM) heads adapt the restoration backbone inside the AFLB modules.

3.2 Pillar 1: Data-Oriented Supervision

Exploration on Data Curation. We first investigate how data selection affects post-training. Fine-tuning on the full training dataset, or on randomly sampled subsets, yields only limited gains (Fig. 4). The results indicate that post-training is most useful when it focuses on the hard tail of the data distribution: samples where the pre-trained model performs poorly.

Based on this observation, we form a subset $\mathcal{D}_{\text{hard}} \subset \mathcal{D}$ from the worst-performing 30% of the training data. The Data-Oriented pillar applies the post-training supervision to this subset.

Differentiable Supervision on $\mathcal{D}_{\text{hard}}$. For this hard-data subset, we optimize the restoration backbone $\mathcal{F}$ using two differentiable, data-driven losses. First, we identify the best-scoring sample $y_{g^\star}$ (where $g^\star = \arg\max_i r_i$ via our reward model) from a group of G stochastic rollouts (see Sec. 3.3). We then impose a standard ℓ_1 loss:

$$\mathcal{L}_{\text{sup}} = \ell_1\big(y_{g^\star},\, \hat{y}\big), \tag{1}$$

where $\hat{y}$ is the ground-truth clean image. This provides a stable gradient signal to the backbone.

Second, to ensure stability, we compute a deterministic output y_{det} (using the mean of the Beta policy distributions) and enforce a consistency objective:

$$\mathcal{L}_{\text{cons}} = \ell_1\big(y_{g^\star},\, y_{\text{det}}\big), \tag{2}$$

This loss anchors the stochastic policy's best output to its stable, deterministic counterpart.

3.3 Pillar 2: Reward-Oriented Optimization

The supervised objectives in Pillar 1 primarily emphasize pixel-level fidelity (e.g., ℓ_1), which can be insufficient to capture perceptual artifacts and degradation-specific residuals. To incorporate richer feedback, we introduce reward terms that are important for restoration quality yet inconvenient (or impossible) to use as direct training losses, including R_{expert} (an expert-judgment score) and R_{aes} (an aesthetic score, as part of R_{gen}). Since these signals are non-differentiable with respect to the restoration output, we treat post-training as policy optimization and employ GRPO to update the policy π_θ under the composite reward.

Reward Modeling. We design a composite reward signal with three components:

$$R(y_{g^\star}, \hat{y}) = \lambda_{\text{gen}} R_{\text{gen}} + \lambda_{\text{expert}} R_{\text{expert}} + \lambda_{\text{task}} R_{\text{task}}, \tag{3}$$

where $y_{g^\star}$ is the restored output, $\hat{y}$ is the ground-truth, and $\lambda_{\text{gen}}, \lambda_{\text{expert}}, \lambda_{\text{task}}$ are fixed balancing hyperparameters.

- **Generic Quality** (R_{gen}): A reward that combines LPIPS, CLIP similarity, and Aesthetic score with PSNR and SSIM to preserve fidelity while adding quality cues.
- **Expert Quality** (R_{expert}): a coarse (1–5) score produced by an expert visual judge. We treat this term as a high-level quality complement rather than a standalone low-level artifact detector.
- **Task-Aware Reward** (R_{task}): A specialized heuristic reward (*e.g.,* gradient consistency for denoising) that guides the policy to restore the relevant attributes for each specific degradation task.

Appendix E gives the detailed reward definitions.

Policy Parametrization. Unlike generative methods that can sample diverse pixel-level outputs, restoration backbones such as AdaIR are deterministic. To apply GRPO, we adapt the backbone $\mathcal{F}$ into a stochastic policy π_θ by adding lightweight GRPO-Driven Model (GDM) heads to its AFLB modules. These heads do not generate images directly. They expose bounded continuous controls for frequency masks and feature fusion, so the policy acts as an input-adaptive router around the original restoration network. As shown in Fig. 2 (bottom center), the action $a = (r_h, r_l, g_f, g_o)$ contains two frequency-mask rates and two fusion weights. A policy head π_θ outputs Beta distribution parameters $(\alpha, \beta) \in \mathbb{R}^4_+$ for these controls:

$$a_k \sim \text{Beta}(\alpha_k, \beta_k), \quad \text{for } k \in \{h, l, f, o\}. \tag{4}$$

The joint log-probability $\log \pi_\theta(a \mid x)$ has a closed form. The first two actions, $(r_h, r_l)_i$, control the frequency-domain mask used to extract high- and low-frequency components from input image x_i and generate an intermediate feature O_i. The final two actions, $(g_f, g_o)_i$, dynamically fuse this feature with the original latent feature y_i:

$$y_i' = g_{f,i} \cdot y_i + g_{o,i} \cdot O_i, \quad \text{for } i \in \{1, 2, 3\}, \tag{5}$$

where y_i' is the new feature passed to the next decoder block. This stochastic modulation produces the restored candidate $y_g = \mathcal{F}(x)$. We sample G actions for each input image x. Appendix F provides the policy details.

Optimization Objective. We optimize the policy π_θ with GRPO under the non-differentiable reward R. Appendix D gives the GRPO background. GRPO constructs a *group-normalized* relative advantage A_i from a group of G rollouts, and uses it in a clipped PPO-style surrogate objective. Our RL objective, $\mathcal{L}_{\text{RL}}$, is:

$$\begin{aligned} \mathcal{L}_{\text{RL}}(\theta) = &- \frac{1}{G} \sum_{i=1}^{G} \min \Big(\rho_i(\theta) A_i, \text{clip}(\rho_i(\theta), 1-\epsilon, 1+\epsilon) A_i \Big) \\ &+ \beta \, \widehat{\mathcal{D}}_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}}) - \tau \, \mathcal{H}(\pi_\theta), \end{aligned} \tag{6}$$

where $\rho_i(\theta)$ is the likelihood ratio, A_i is the group-normalized advantage, β controls the KL penalty, τ is the entropy bonus, π_{ref} denotes the reference policy for KL regularization, and π_θ is the policy being optimized.

3.4 Final Hybrid Objective

Our final objective combines the pillars. We jointly optimize the policy π_θ (via $\mathcal{L}_{\text{RL}}$ from Pillar 2) and the backbone $\mathcal{F}$ (via $\mathcal{L}_{\text{sup}}$ and $\mathcal{L}_{\text{cons}}$ from Pillar 1). All computations are performed on our curated hard-data subset $\mathcal{D}_{\text{hard}}$:

$$\mathcal{L}_{\text{total}}(\theta) = \mathbb{E}_{(x,g) \sim \mathcal{D}_{\text{hard}}} \big[\, \mathcal{L}_{\text{RL}}(\theta) + \lambda_{\text{sup}} \mathcal{L}_{\text{sup}} + \lambda_{\text{cons}} \mathcal{L}_{\text{cons}} \, \big], \tag{7}$$

where λ_{sup} and λ_{cons} are weights with linear annealing schedules (see Appendix C).

4 Experiment

4.1 Experimental Setup

Datasets. For training and testing, we follow the standard protocols used in prior all-in-one restoration works [13, 36, 56].

Table 1: Comparisons for five-degradation all-in-one restoration. Each cell reports PSNR/SSIM, and denoising is evaluated at $\sigma = 25$. All-in-one baselines are reported under the same five-task protocol; IRPO improves the AdaIR backbone by 0.93 dB average PSNR.

Method	Dehaze SOTS	Derain Rain100L	Denoise CBSD68	Deblur GoPro	Low-Light LOL	Average PSNR/SSIM
NAFNet [11]	25.23/0.939	35.56/0.967	31.02/0.883	26.53/0.808	20.49/0.809	27.76/0.881
HINet [10]	24.74/0.937	35.67/0.969	31.00/0.881	26.12/0.788	19.47/0.800	27.40/0.875
MPRNet [80]	24.27/0.937	38.16/0.981	31.35/0.889	26.87/0.823	20.84/0.824	28.30/0.891
DGUNet [53]	24.78/0.940	36.62/0.971	31.10/0.883	27.25/0.837	21.87/0.823	28.32/0.891
MIRNet-v2 [82]	24.03/0.927	33.89/0.954	30.97/0.881	26.30/0.799	21.52/0.815	27.34/0.875
SwinIR [42]	21.50/0.891	30.78/0.923	30.59/0.868	24.52/0.773	17.81/0.723	25.04/0.835
Restormer [81]	24.09/0.927	34.81/0.962	31.49/0.884	27.22/0.829	20.41/0.806	27.60/0.881
DL [19]	20.54/0.826	21.96/0.762	23.09/0.745	19.86/0.672	19.83/0.712	21.05/0.743
Transweather [67]	21.32/0.885	29.43/0.905	29.00/0.841	25.12/0.757	21.21/0.792	25.22/0.836
TAPE [48]	22.16/0.861	29.67/0.904	30.18/0.855	24.47/0.763	18.97/0.621	25.09/0.801
AirNet [36]	21.04/0.884	32.98/0.951	30.91/0.882	24.35/0.781	18.18/0.735	25.49/0.846
IDR [85]	25.24/0.943	35.63/0.965	31.60/0.887	27.87/0.846	21.34/0.826	28.34/0.893
GridFormer [69]	26.79/0.951	36.61/0.971	31.45/0.885	29.22/0.884	22.59/0.831	29.33/0.904
InstructIR [12]	27.10/0.956	36.84/0.973	31.40/0.887	29.40/0.886	23.00/0.836	29.55/0.907
VLU-Net [83]	30.84/0.980	38.54/0.982	31.43/0.891	27.46/0.840	22.29/0.833	30.11/0.905
DFPIR [66]	31.64/0.979	37.62/0.978	31.29/0.889	28.82/0.873	23.82/0.843	30.64/0.913
AdaIR [13]	30.53/0.978	38.02/0.981	31.35/0.889	28.12/0.858	23.00/0.845	30.20/0.910
IRPO (Ours)	**31.85/0.984**	**38.72/0.987**	**32.13/0.896**	**28.89/0.863**	**24.08/0.858**	**31.13/0.918**

Table 2: OOD real-world evaluation with the same AdaIR backbone. Each cell reports PSNR/SSIM, and the average is computed over all five unseen degradation sources.

Method	RRSHID Dehaze	GT-Rain Derain	SIDD Denoise	ReLoBlur Deblur	LoLi-Street Low-Light	Average PSNR/SSIM
AdaIR	16.71/0.475	19.58/0.684	28.83/0.902	20.43/0.806	15.96/0.781	20.30/0.730
IRPO	**19.94/0.623**	**21.17/0.692**	**33.18/0.919**	**21.94/0.837**	**22.43/0.875**	**23.73/0.789**

In-Domain (ID) Datasets. Our training and evaluation cover five restoration tasks: dehazing on SOTS [35], deraining on Rain100L [74], deblurring on GoPro [54], low-light enhancement on LOL [71], and Gaussian denoising trained on BSD400 [2]+WED [50] and evaluated on BSD68 [51]/Urban100 [24] with $\sigma \in \{15, 25, 50\}$.

Out-of-Domain (OOD) Datasets. To evaluate generalization, we test the same all-in-one model on five OOD datasets: RRSHID [92] for real dehazing, GT-Rain [3] for real deraining, SIDD [1] for smartphone denoising, ReLoBlur [37] for local-motion deblurring, and LoLi-Street [26] for street-scene low-light enhancement.

Implementation Details. IRPO is initialized from AdaIR [13]. It keeps the 4-level encoder-decoder and three AFLB modules, and adds lightweight policy heads for the Beta actions (r_h, r_l, g_f, g_o). We fine-tune the worst-performing 30% training samples for 30 epochs with group size $G = 4$. We use Adam with learning rate 3×10^{-5} and a $6\times$ learning-rate multiplier for the policy heads. We keep AdaIR's 128×128 crops and augmentations, and add annealed $\mathcal{L}_1$ and consistency losses. Training uses two NVIDIA 4090 GPUs plus two GPUs serving Qwen2.5-VL-7B-Instruct [4]. At inference, the VLM is removed and the Beta policy uses deterministic expectations, so IRPO has no extra VLM overhead. Appendix C lists the full hyperparameters.

4.2 Main Experiments

We evaluate IRPO on standard all-in-one restoration settings and single-task benchmarks.

All-in-One Restoration. Table 1 compares IRPO with general IR methods and all-in-one restorers, including 2024–2025 baselines under the same five-task protocol. IRPO obtains the best average performance and improves AdaIR by 0.93 dB average PSNR, with gains on dehazing (+1.32 dB), low-light enhancement (+1.08 dB), and denoising (+0.78 dB). Table 3 further evaluates the three-degradation setting with haze, rain, and noise. IRPO remains the top performer and improves AdaIR by 0.74 dB average PSNR, suggesting that the post-training gain is consistent across both compact and broader all-in-one protocols. Fig. 3 provides synthetic visual comparisons; additional per-task and real-world visual examples are included in Appendix B.

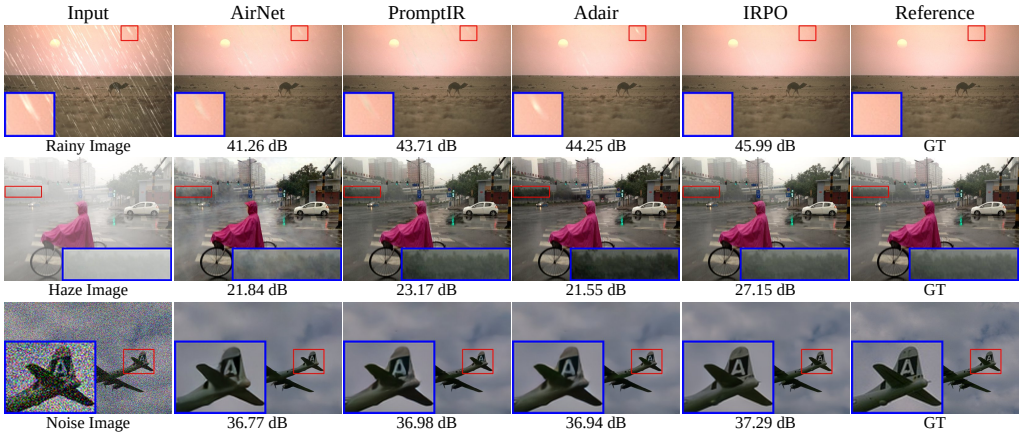


Figure 3: Qualitative comparisons on synthetic restoration benchmarks for deraining, dehazing, and denoising. IRPO removes residual degradations more effectively than AirNet, PromptIR, and AdaIR while preserving local structures and fine details.

Table 3: Comparisons under the three-degradation all-in-one setting. Each cell reports PSNR/SSIM; denoising is evaluated on CBSD68 at three noise levels. IRPO improves the AdaIR backbone by 0.74 dB average PSNR.

Method	Dehaze SOTS	Derain Rain100L	Denoising on CBSD68			Average
	PSNR/SSIM	PSNR/SSIM	$\sigma = 15$	$\sigma = 25$	$\sigma = 50$	PSNR/SSIM
BRDNet [65]	23.23/0.895	27.42/0.895	32.26/0.898	29.76/0.836	26.34/0.693	27.80/0.843
MPRNet [80]	25.28/0.955	33.57/0.954	33.54/0.927	30.89/0.880	27.56/0.779	30.17/0.899
AirNet [36]	27.94/0.962	34.90/0.968	33.92/0.933	31.26/0.888	28.00/0.797	31.20/0.910
PromptIR [56]	30.58/0.974	36.37/0.972	33.98/0.933	31.31/0.888	28.06/0.799	32.06/0.913
GridFormer [69]	30.37/0.970	37.15/0.972	33.93/0.931	31.37/0.887	28.11/0.801	32.19/0.912
InstructIR [12]	30.22/0.959	37.98/0.978	34.15/0.933	31.52/0.890	28.30/0.804	32.43/0.913
VLU-Net [83]	30.71/0.980	38.93/0.984	34.13/0.935	31.48/0.892	28.23/0.804	32.70/0.919
DFPIR [66]	31.87/0.980	38.65/0.982	34.14/0.935	31.47/0.893	28.25/0.806	32.88/0.919
AdaIR [13]	31.06/0.980	38.64/0.983	34.12/0.935	31.45/0.892	28.19/0.802	32.69/0.918
IRPO (Ours)	**31.88/0.984**	**39.13/0.988**	**34.54/0.938**	**32.04/0.905**	**29.56/0.823**	**33.43/0.928**

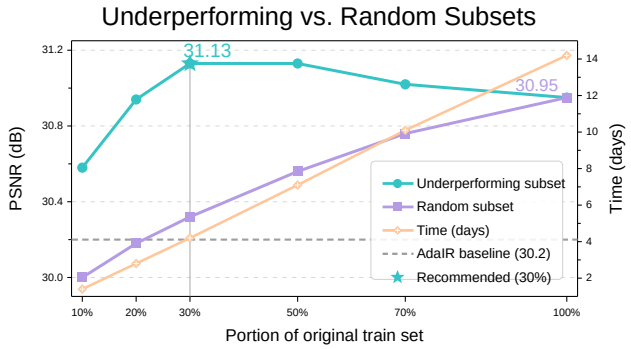


Figure 4: Post-training on underperforming *vs.* random subsets for the five-task all-in-one setting. The left y-axis shows average PSNR and the right y-axis shows time. The 30% underperforming subset reaches 31.13 dB in about 4.2 GPU days; 100% data reaches 30.95 dB with roughly three times the cost.

Single-Task Restoration. Appendix Table A.2 reports specialized single-task models. IRPO consistently improves over AdaIR and PromptIR across dehazing, deraining, and denoising, with the largest gain on Urban100 ($\sigma = 50$, +0.37 dB).

Table 4: Attribution of reward components under the same hard-data subset and GRPO schedule. Each cell reports PSNR/SSIM; the final column reports the average PSNR gain over AdaIR.

Method	Dehazing	Deraining	Denoising	Deblurring	Low-Light	Average PSNR/SSIM	ΔAvg. PSNR
AdaIR	30.53/0.978	38.02/0.981	31.35/0.889	28.12/0.858	23.00/0.845	30.20/0.910	0.00
+ R_{gen}	31.30/0.981	38.57/0.985	31.95/0.893	28.64/0.861	23.68/0.852	30.83/0.914	+0.63
+ R_{expert}	30.90/0.980	38.28/0.983	31.64/0.891	28.35/0.859	23.31/0.848	30.50/0.912	+0.30
+ R_{task}	31.12/0.981	38.42/0.984	31.83/0.893	28.57/0.861	23.57/0.850	30.70/0.914	+0.50
IRPO	**31.85/0.984**	**38.72/0.987**	**32.13/0.896**	**28.89/0.863**	**24.08/0.858**	**31.13/0.918**	**+0.93**

4.3 Evaluation on OOD Generalization

OOD generalization is central for all-in-one restoration, so we evaluate IRPO on unseen degradation sources in Table 2. IRPO raises AdaIR from 20.30/0.730 to 23.73/0.789 average PSNR/SSIM, with the largest gains on low-light (+6.47 dB) and denoising (+4.35 dB). These results support the main design of IRPO: hard-case post-training with quality-aware rewards improves OOD robustness while retaining supervised fidelity constraints. Real-world visual examples in Appendix B show the same trend; expert-source, group-size, efficiency, training-curve, data-strategy, and reward-weight analyses are in Appendix A for completeness.

4.4 Ablation Study

We focus on data selection and reward composition in the main paper, and report group-size, expert-source, efficiency, data-ratio, reward-weight, and stability analyses in Appendix A.

Effect of Data Portion. Fig. 4 shows that the 30% underperforming subset reaches 31.13 dB in about 4.2 GPU days, while 100% data reaches 30.95 dB with roughly three times the cost. Appendix A further analyzes the selected feature space, uniform *vs.* adaptive data ratios, and reward-weight sensitivity.

Effect of Reward Components. Table 4 isolates reward components. R_{gen} gives the largest individual gain (+0.63 dB) because it keeps full-reference fidelity signals. R_{task} (+0.50 dB) targets degradation residuals, and R_{expert} (+0.30 dB) acts as a coarse quality complement. Full IRPO reaches 31.13 dB, outperforming every single-reward variant.

Contribution of data and rewards. Fig. 4 changes the post-training data, while Table 4 fixes the hard subset and GRPO update to isolate reward channels. Thus, the gains come from hard-tail data selection plus reward-guided policy optimization, not from any single perceptual or VLM reward source alone or from score scaling.

Interaction with fidelity losses. The reward design does not replace supervised fidelity: GRPO selects among stochastic candidates on the hard subset, while $\mathcal{L}_{\text{sup}}$ and $\mathcal{L}_{\text{cons}}$ anchor updates to paired ground truth and deterministic behavior. Auxiliary quality rewards can therefore guide exploration without discarding distortion-oriented constraints during post-training.

4.5 Post-Training Analysis

Table 5(a) separates the sources of improvement in IRPO. Complementary to Table 4, it changes data, heads, reward access, and update rules rather than individual reward channels. Fine-tuning AdaIR on the hard subset improves the five-degradation average but not OOD robustness, showing that hard-tail data are useful but insufficient. Adding stochastic heads with supervised losses brings a larger OOD gain. Reward-ranked distillation uses the same sampled candidates and reward model as IRPO, but remains below full GRPO optimization, indicating that group-relative policy learning contributes beyond reward-based candidate selection.

We also evaluate quality-oriented metrics beyond PSNR/SSIM in Table 5(b). Full-reference LPIPS [90] and DISTS [16] are computed on paired OOD benchmarks. NIQE [52] and MUSIQ [32] are computed on the real-world visual set without references. IRPO improves these perceptual/IQA scores over AdaIR and DFPIR, which is consistent with the synthetic comparisons in Fig. 3 and the real-world visual examples in Appendix B for qualitative inspection. These metrics are used as

Table 5: Additional analysis of IRPO. (a) Attribution study under the same AdaIR initialization; both averages report PSNR. (b) Quality-oriented evaluation with full-reference OOD metrics and no-reference real-world IQA metrics.

(a) Attribution of post-training gains

Variant	Hard	Heads	Rew.	Rule	5D	OOD
AdaIR	–	–	–	–	30.20	20.30
Hard FT	✓	–	–	Sup.	30.43	20.22
Sup. GDM	✓	✓	–	Sup.	30.75	22.16
Reward distill.	✓	✓	✓	Distill.	30.91	22.89
IRPO-GRPO	✓	✓	✓	GRPO	**31.13**	**23.73**

(b) Perceptual/IQA evaluation

Method	LPIPS↓	DISTS↓	NIQE↓	MUSIQ↑
AdaIR	0.191	0.140	4.18	61.5
DFPIR	0.184	0.136	4.12	62.2
IRPO	**0.174**	**0.126**	**3.89**	**63.8**

supporting evidence for quality-aware restoration, while the main objective remains fidelity-oriented restoration with data-driven and reward-driven post-training for robust restoration.

Reading the controls. The attribution variants are deliberately matched in AdaIR initialization and then progressively add hard data, stochastic heads, reward access, and the update rule. This keeps the comparison focused on the source of improvement rather than on a hidden change in training data or model capacity. Reward-ranked distillation already benefits from the same sampled candidates and rewards as IRPO, but it collapses each group to a single pseudo-target; IRPO instead uses group-relative advantages to update the stochastic heads over frequency masks and feature-fusion gates. The IQA block should be read in the same spirit: LPIPS, DISTS, NIQE, and MUSIQ are diagnostics for whether the quality-aware rewards hurt visual naturalness, while PSNR/SSIM remain the primary restoration protocol.

Practical overhead. Appendix Table A.5 summarizes the computational cost. IRPO adds only lightweight policy heads to AdaIR (+0.05M parameters), and the VLM reward is used only during post-training. At inference, the Beta policy is evaluated by its deterministic expectation and the Qwen-VL scorer is removed. The deployed model therefore keeps the same 1080p latency as AdaIR while improving the OOD average by 3.43 dB. The extra computation is a one-time post-training cost.

Additional diagnostics. The appendix first provides the experimental evidence behind the main claims. Appendix A reports training curves, hard-sample visualization, single-task results, group-size and expert-source ablations, efficiency accounting, and reward-weight sensitivity. Appendix B provides real-world and additional per-task visual comparisons. The remaining sections make the recipe auditable: Appendix C lists hyperparameters, Appendix D derives the GRPO objective, Appendix E expands the reward definitions, and Appendix F gives the Beta-policy log probability.

Scope of the post-training wrapper. IRPO is a post-training recipe rather than a new restoration backbone. We instantiate it on AdaIR because its AFLB modules provide a natural place to attach stochastic policy heads, but the data-selection rule and reward model operate at the dataset and output-image levels. They only require a pre-trained restorer, a hard-sample subset, and a train-time mechanism for generating multiple candidate restorations. Thus, AdaIR determines the specific action parameterization used here, while the broader recipe can be adapted to other encoder-decoder or transformer restorers with architecture-specific stochastic heads. At deployment, the policy is evaluated deterministically and the model keeps the same input-output interface as the backbone. Appendix A.5 reports an additional evaluation on PromptIR.

5 Conclusion

This study introduces IRPO, a GRPO-based post-training framework for deterministic restoration models. IRPO has two core components. First, the data-oriented component focuses post-training on the 30% underperforming samples, which improves the accuracy-efficiency trade-off. Second, the reward-oriented component combines fidelity, coarse visual-quality feedback, and task-specific restoration cues. Experiments on in-domain and OOD benchmarks show that IRPO improves the AdaIR backbone and strengthens real-world generalization. These results indicate that data-driven and reward-driven post-training is a practical direction for robust image restoration.

Acknowledgments

This work was supported in part by the Natural Science Foundation of China under Grant 62403401, and in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2024A1515011992 and Grant 2026A1515012323. This work was supported by the National Natural Science Foundation of China under Grant No. 62320106007, and by NUS Grant A-0010106-00-00, A-8004365-00-00 and A-8004410-01-00.

References

[1] Abdelrahman Abdelhamed, Stephen Lin, and Michael S Brown. A high-quality denoising dataset for smartphone cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1692–1700, 2018.

[2] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *TPAMI*, 2010.

[3] Yunhao Ba, Howard Zhang, Ethan Yang, Akira Suzuki, Arnold Pfahnl, Chethan Chinder Chandrappa, Celso M De Melo, Suya You, Stefano Soatto, Alex Wong, et al. Not just streaks: Towards ground truth for single image deraining. In *European Conference on Computer Vision*, pages 723–740. Springer, 2022.

[4] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[5] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

[6] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023.

[7] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6228–6237, 2018.

[8] Bolun Cai, Xiangmin Xu, Kui Jia, Chunmei Qing, and Dacheng Tao. DehazeNet: An end-to-end system for single image haze removal. *TIP*, 2016.

[9] Liang Chen, Lei Li, Haozhe Zhao, Yifan Song, and Vinci. R1-V: Reinforcing super generalization ability in vision-language models with less than $3. `https://github.com/Deep-Agent/R1-V`, 2025.

[10] Liangyu Chen, Xin Lu, Jie Zhang, Xiaojie Chu, and Chengpeng Chen. Hinet: Half instance normalization network for image restoration. In *CVPR Workshops*, 2021.

[11] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *ECCV*, 2022.

[12] Marcos V. Conde, Gregor Geigle, and Radu Timofte. InstructIR: High-quality image restoration following human instructions. In *European Conference on Computer Vision*, 2024.

[13] Yuning Cui, Syed Waqas Zamir, Salman Khan, Alois Knoll, Mubarak Shah, and Fahad Shahbaz Khan. AdaIR: Adaptive all-in-one image restoration via frequency mining and modulation. In *The Thirteenth International Conference on Learning Representations*, 2025.

[14] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space. In *ICIP*, 2007.

[15] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[16] Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2567–2581, 2022. doi: 10.1109/TPAMI.2020.3045810.

[17] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):295–307, 2016.

[18] Yu Dong, Yihao Liu, He Zhang, Shifeng Chen, and Yu Qiao. FD-GAN: Generative adversarial networks with fusion-discriminator for single image dehazing. In *AAAI*, 2020.

[19] Qingnan Fan, Dongdong Chen, Lu Yuan, Gang Hua, Nenghai Yu, and Baoquan Chen. A general decoupled learning framework for parameterized image operators. *TPAMI*, 2019.

[20] Xueyang Fu, Borong Liang, Yue Huang, Xinghao Ding, and John Paisley. Lightweight pyramid networks for image deraining. *IEEE Transactions on Neural Networks and Learning Systems*, 31(6):1794–1807, 2020.

[21] Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. Toward convolutional blind denoising of real photographs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1712–1722, 2019.

[22] Xiaobin Hu, Wenqi Ren, John LaMaster, Xiaochun Cao, Xiaoming Li, Zechao Li, Bjoern Menze, and Wei Liu. Face super-resolution guided by 3D facial priors. In *European Conference on Computer Vision*, pages 763–780. Springer, 2020.

[23] Xiaobin Hu, Wenqi Ren, Jiaolong Yang, Xiaochun Cao, David Wipf, Bjoern Menze, Xin Tong, and Hongbin Zha. Face restoration via plug-and-play 3D facial priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):8910–8926, 2022.

[24] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *CVPR*, 2015.

[25] Team Hunyuan3D, Bowen Zhang, Chunchao Guo, Haolin Liu, Hongyu Yan, Huiwen Shi, Jingwei Huang, Junlin Yu, Kunhong Li, Linus, Penghao Wang, et al. Hunyuan3D-Omni: A unified framework for controllable generation of 3D assets. *arXiv preprint arXiv:2509.21245*, 2025.

[26] Md Tanvir Islam, Inzamamul Alam, Simon S Woo, Saeed Anwar, IK Hyun Lee, and Khan Muhammad. LoLI-Street: Benchmarking low-light image enhancement and beyond. In *Proceedings of the Asian Conference on Computer Vision*, pages 1250–1267, 2024.

[27] Viren Jain, Joseph F Murray, Fabian Roth, Srinivas Turaga, Valentin Zhigulin, Kevin L Briggman, Moritz N Helmstaedter, Winfried Denk, and H Sebastian Seung. Supervised learning of image restoration with convolutional networks. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007.

[28] Xiaozhong Ji, Xiaobin Hu, Zhihong Xu, Junwei Zhu, Chuming Lin, Qingdong He, Jiangning Zhang, Donghao Luo, Yi Chen, Qin Lin, et al. Sonic: Shifting focus to global audio perception in portrait animation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 193–203, 2025.

[29] Kui Jiang, Zhongyuan Wang, Peng Yi, Chen Chen, Baojin Huang, Yimin Luo, Jiayi Ma, and Junjun Jiang. Multi-scale progressive fusion network for single image deraining. In *CVPR*, 2020.

[30] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

[31] Xiaoqiang Kang, Shengen Wu, Zimu Wang, Yilin Liu, Xiaobo Jin, Kaizhu Huang, Wei Wang, Yutao Yue, Xiaowei Huang, and Qiufeng Wang. Can GRPO boost complex multimodal table understanding? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 12631–12644, 2025.

[32] Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. MUSIQ: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5148–5157, 2021.

[33] Hanyu Lai, Xiao Liu, Junjie Gao, Jiale Cheng, Zehan Qi, Yifan Xu, Shuntian Yao, Dan Zhang, Jinhua Du, Zhenyu Hou, et al. A survey of post-training scaling in large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2771–2791, 2025.

[34] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2Noise: Learning image restoration without clean data. *arXiv preprint arXiv:1803.04189*, 2018.

[35] Boyi Li, Wenqi Ren, Dengpan Fu, Dacheng Tao, Dan Feng, Wenjun Zeng, and Zhangyang Wang. Benchmarking single-image dehazing and beyond. *TIP*, 2018.

[36] Boyun Li, Xiao Liu, Peng Hu, Zhongqin Wu, Jiancheng Lv, and Xi Peng. All-in-one image restoration for unknown corruption. In *CVPR*, 2022.

[37] Haoying Li, Ziran Zhang, Tingting Jiang, Peng Luo, Huajun Feng, and Zhihai Xu. Real-world deep local motion deblurring. In *proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 1314–1322, 2023.

[38] Jiachen Li, Qian Long, Jian Zheng, Xiaofeng Gao, Robinson Piramuthu, Wenhu Chen, and William Yang Wang. T2V-Turbo-v2: Enhancing video generation model post-training through data, reward, and conditional guidance design. *arXiv preprint arXiv:2410.05677*, 2024.

[39] Wenhao Li, Qiangchang Wang, Xianjing Meng, Zhibin Wu, and Yilong Yin. VT-FSL: Bridging vision and text with LLMs for few-shot learning. In *NeurIPS*, volume 38, pages 113697–113727, 2025.

[40] Wenhao Li, Xianjing Meng, Qiangchang Wang, Zhongyi Han, Zhibin Wu, and Yilong Yin. DVLA-RL: Dual-level vision-language alignment with reinforcement learning gating for few-shot learning. In *ICLR*, 2026.

[41] Wenhao Li, Zhibin Wu, Chong Xiao, and Qiangchang Wang. SemMSA: Latent semantic-aided robust multimodal sentiment analysis with incomplete data. *arXiv preprint arXiv:2609.30238*, 2026.

[42] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. SwinIR: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 1833–1844, 2021.

[43] Jingyun Liang, Jiezhang Cao, Yuchen Fan, Kai Zhang, Rakesh Ranjan, Yawei Li, Radu Timofte, and Luc Van Gool. VRT: A video restoration transformer. *IEEE Transactions on Image Processing*, 33:2171–2182, 2024.

[44] Yujie Liang, Xiaobin Hu, Boyuan Jiang, Donghao Luo, Xu Peng, Kai Wu, Chengming Xu, Wenhui Han, Taisong Jin, Chengjie Wang, et al. VTON-HandFit: Virtual try-on for arbitrary hand pose guided by hand priors embedding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22616–22626, 2025.

[45] Shanchuan Lin, Xin Xia, Yuxi Ren, Ceyuan Yang, Xuefeng Xiao, and Lu Jiang. Diffusion adversarial post-training for one-step video generation. *arXiv preprint arXiv:2501.08316*, 2025.

[46] Shanchuan Lin, Ceyuan Yang, Hao He, Jianwen Jiang, Yuxi Ren, Xin Xia, Yang Zhao, Xuefeng Xiao, and Lu Jiang. Autoregressive adversarial post-training for real-time interactive video generation. *arXiv preprint arXiv:2506.09350*, 2025.

[47] Lei Liu, Luan Ma, Shuai Wang, Jun Wang, and Silas N Melo. Two-stage mamba-based diffusion model for image restoration. *Scientific Reports*, 15(1):22265, 2025.

[48] Lin Liu, Lingxi Xie, Xiaopeng Zhang, Shanxin Yuan, Xiangyu Chen, Wengang Zhou, Houqiang Li, and Qi Tian. TAPE: Task-agnostic prior embedding for image restoration. In *ECCV*, 2022.

[49] Lu Liu, Chunlei Cai, Shaocheng Shen, Jianfeng Liang, Weimin Ouyang, Tianxiao Ye, Jian Mao, Huiyu Duan, Jiangchao Yao, Xiaoyun Zhang, et al. MoA-VR: A mixture-of-agents system towards all-in-one video restoration. *arXiv preprint arXiv:2510.08508*, 2025.

[50] Kede Ma, Zhengfang Duanmu, Qingbo Wu, Zhou Wang, Hongwei Yong, Hongliang Li, and Lei Zhang. Waterloo exploration database: New challenges for image quality assessment models. *TIP*, 2016.

[51] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, 2001.

[52] Anish Mittal, Rajiv Soundararajan, and Alan C. Bovik. Making a completely blind image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2013. doi: 10.1109/LSP.2012.2227726.

[53] Chong Mou, Qian Wang, and Jian Zhang. Deep generalized unfolding networks for image restoration. In *CVPR*, 2022.

[54] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, 2017.

[55] OpenAI, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. OpenAI o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

[56] Vaishnav Potlapalli, Syed Waqas Zamir, Salman H Khan, and Fahad Shahbaz Khan. PromptIR: Prompting for all-in-one image restoration. *NeurIPS*, 2023.

[57] Zhangyang Qi, Zhixiong Zhang, Yizhou Yu, Jiaqi Wang, and Hengshuang Zhao. VLN-R1: Vision-language navigation via reinforcement fine-tuning. *arXiv preprint arXiv:2506.17221*, 2025.

[58] Yanyun Qu, Yizi Chen, Jingying Huang, and Yuan Xie. Enhanced pix2pix dehazing network. In *CVPR*, 2019.

[59] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[60] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

[61] Wenqi Ren, Si Liu, Hua Zhang, Jinshan Pan, Xiaochun Cao, and Ming-Hsuan Yang. Single image dehazing via multi-scale convolutional neural networks. In *ECCV*, 2016.

[62] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. LAION-5B: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022.

[63] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. doi: 10.48550/arXiv.2402.03300. URL `https://arxiv.org/abs/2402.03300`.

[64] Xiangfei Sheng, Xiaofeng Pan, Zhichao Yang, Pengfei Chen, and Leida Li. Fine-grained image quality assessment for perceptual image restoration. *arXiv preprint arXiv:2508.14475*, 2025.

[65] Chunwei Tian, Yong Xu, and Wangmeng Zuo. Image denoising using deep CNN with batch renormalization. *Neural Networks*, 2020.

[66] Xiangpeng Tian, Xiangyu Liao, Xiao Liu, Meng Li, and Chao Ren. Degradation-aware feature perturbation for all-in-one image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28165–28175, 2025.

[67] Jeya Maria Jose Valanarasu, Rajeev Yasarla, and Vishal M Patel. TransWeather: Transformer-based restoration of images degraded by adverse weather conditions. In *CVPR*, 2022.

[68] Boyang Wang, Fengyu Yang, Xihang Yu, Chao Zhang, and Hanbin Zhao. APISR: Anime production inspired real-world anime super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 25574–25584, 2024.

[69] Tao Wang, Kaihao Zhang, Ziqian Shao, Wenhan Luo, Bjorn Stenger, Tong Lu, Tae-Kyun Kim, Wei Liu, and Hongdong Li. GridFormer: Residual dense transformer with grid structure for image restoration in adverse weather conditions. *International Journal of Computer Vision*, 132 (10):4541–4563, 2024. doi: 10.1007/s11263-024-02056-0.

[70] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. ESRGAN: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018.

[71] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. *arXiv preprint arXiv:1808.04560*, 2018.

[72] Gang Wu, Junjun Jiang, Kui Jiang, and Xianming Liu. Boosting all-in-one image restoration via self-improved privilege learning. *arXiv preprint arXiv:2505.24207*, 2025.

[73] Haoning Wu, Zicheng Zhang, Weixia Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Yixuan Gao, Annan Wang, Erli Zhang, Wenxiu Sun, Qiong Yan, Xiongkuo Min, Guangtao Zhai, and Weisi Lin. Q-align: Teaching lmms for visual scoring via discrete text-defined levels. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

[74] Wenhan Yang, Robby T Tan, Jiashi Feng, Zongming Guo, Shuicheng Yan, and Jiaying Liu. Joint rain detection and removal from a single image with contextualized deep networks. *TPAMI*, 2019.

[75] Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, et al. R1-Onevision: Advancing generalized multimodal reasoning through cross-modal formalization. *arXiv preprint arXiv:2503.10615*, 2025.

[76] Rajeev Yasarla and Vishal M Patel. Uncertainty guided multi-scale residual learning-using a cycle spinning CNN for single image de-raining. In *CVPR*, 2019.

[77] Angen Ye, Zeyu Zhang, Boyuan Wang, Xiaofeng Wang, Dapeng Zhang, and Zheng Zhu. VLA-R1: Enhancing reasoning in vision-language-action models. *arXiv preprint arXiv:2510.01623*, 2025.

[78] Bo Yin, Xiaobin Hu, Xingyu Zhou, Peng-Tao Jiang, Yue Liao, Junwei Zhu, Jiangning Zhang, Ying Tai, Chengjie Wang, and Shuicheng Yan. FeRA: Frequency-energy constrained routing for effective diffusion adaptation fine-tuning. *arXiv preprint arXiv:2511.17979*, 2025.

[79] Zhiyuan You, Xin Cai, Jinjin Gu, Tianfan Xue, and Chao Dong. Teaching large language models to regress accurate image quality scores using score distribution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14483–14494, 2025.

[80] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14821–14831, 2021.

[81] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *CVPR*, 2022.

[82] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Learning enriched features for fast image restoration and enhancement. *TPAMI*, 2022.

[83] Haijin Zeng, Xiangming Wang, Yongyong Chen, Jingyong Su, and Jie Liu. Vision-language gradient descent-driven all-in-one deep unfolding networks. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 7524–7533, 2025.

[84] He Zhang and Vishal M Patel. Density-aware single image de-raining using a multi-stream dense network. In *CVPR*, 2018.

[85] Jinghao Zhang, Jie Huang, Mingde Yao, Zizheng Yang, Hu Yu, Man Zhou, and Feng Zhao. Ingredient-oriented multi-degradation learning for image restoration. In *CVPR*, 2023.

[86] Jingyi Zhang, Jiaxing Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng Tao. R1-VL: Learning to reason with multimodal large language models via step-wise group relative policy optimization. *arXiv preprint arXiv:2503.12937*, 2025.

[87] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising. *TIP*, 2017.

[88] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. Learning deep CNN denoiser prior for image restoration. In *CVPR*, 2017.

[89] Kai Zhang, Wangmeng Zuo, and Lei Zhang. FFDNet: Toward a fast and flexible solution for CNN-Based image denoising. *TIP*, 2018.

[90] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[91] Xiaoyu Zhang, Wei Gao, Ge Li, Qiuping Jiang, and Runmin Cong. Image quality assessment–driven reinforcement learning for mixed distorted image restoration. *ACM Transactions on Multimedia Computing, Communications and Applications*, 19(1s):1–23, 2023.

[92] Zeng-Hui Zhu, Wei Lu, Si-Bao Chen, Chris HQ Ding, Jin Tang, and Bin Luo. Real-world remote sensing image dehazing: Benchmark and baseline. *IEEE Transactions on Geoscience and Remote Sensing*, 2025.

A Additional Experimental Results and Analysis

This section reports additional results and analyses. We first show training stability and hard-sample structure, then provide single-task evaluation, group-size and expert-source ablations, efficiency analysis, and reward-weight sensitivity.

A.1 Training and Data-Selection Diagnostics

Figure A.1 shows stable post-training dynamics. The total loss decreases smoothly, while the mean reward increases and reaches a plateau, indicating that the reward-guided stage remains numerically stable rather than oscillatory.

In Pillar 1 (Data-Oriented Supervision), fine-tuning on the 30% underperforming samples gives the selected accuracy-efficiency trade-off. To examine this subset, we visualize the pre-trained feature space using T-SNE in Figure A.2. The selected 30% samples mainly lie in long-tail, high-entropy regions of the distribution. This indicates that the subset contains difficult cases, such as severe noise and complex occlusion, where the deterministic baseline performs less reliably.

The main experiments use a uniform 30% ratio across degradation tasks. We also test an adaptive task-specific ratio, such as assigning 40% to low-light enhancement and 20% to deraining. As shown in Table A.1, adaptive sampling slightly improves deblurring and low-light enhancement, but the uniform 30% strategy gives the best average performance (31.13 dB).

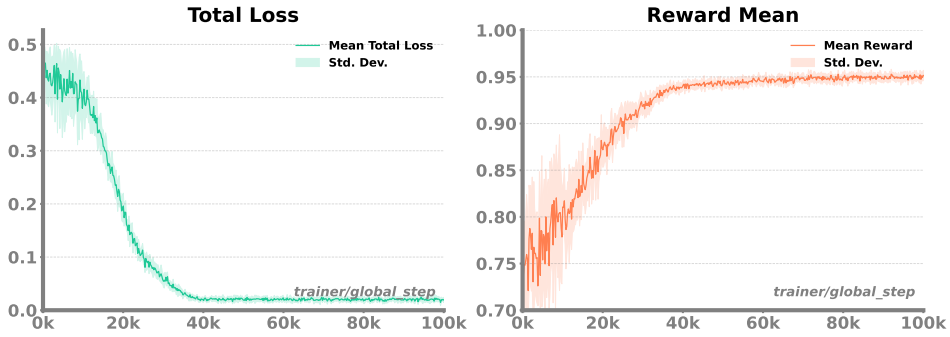


Figure A.1: Training curves for IRPO post-training (*mean* and *std. dev.* over 30 epochs). The total loss smoothly converges, while the mean reward steadily increases and reaches a stable plateau.

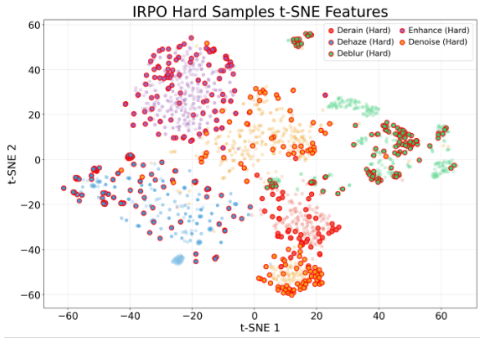


Figure A.2: T-SNE feature visualization of the hard samples ($\mathcal{D}_{hard}$) across five tasks compared to the full dataset. The selected samples naturally cluster in complex, long-tail regions.

Table A.1: Comparison between uniform and adaptive data selection strategies. Performance is measured by average PSNR/SSIM on the all-in-one benchmark.

Strategy	Ratio Config	Dehaze	Derain	Denoise	Deblur	Low-Light	Average
Uniform	All 30%	**31.85/0.984**	**38.72/0.987**	**32.13/0.896**	28.89/0.863	24.08/0.858	**31.13/0.918**
Adaptive	30/20/25/35/40%	31.58/0.983	38.55/0.984	31.98/0.893	**29.02/0.865**	**24.21/0.860**	31.07/0.917

Table A.2: Single-task comparisons for dehazing (a), deraining (b), and denoising (c). For this setting, each model is trained separately on its respective dataset. All metrics shown are PSNR/SSIM.

(a) Dehaze (SOTS)		(b) Derain (Rain100L)		(c) Denoise (Urban100+BSD68)								
Method	PSNR/SSIM	Method	PSNR/SSIM	Method	Urban100 $\sigma = 15$	Urban100 $\sigma = 25$	Urban100 $\sigma = 50$	BSD68 $\sigma = 15$	BSD68 $\sigma = 25$	BSD68 $\sigma = 50$	Average
DehazeNet [8]	22.46/0.851	DIDMDN [84]	23.79/0.773	CBM3D [14]	33.93/0.941	31.36/0.909	27.93/0.840	33.50/0.922	30.69/0.868	27.36/0.763	30.80/0.874
MSCNN [61]	22.06/0.908	UMR [76]	32.39/0.921	DnCNN [87]	32.98/0.931	30.81/0.902	27.59/0.833	33.89/0.930	31.23/0.883	27.92/0.789	30.74/0.878
EPDN [58]	22.57/0.863	MSPFN [29]	33.50/0.948	IRCNN [88]	27.59/0.833	31.20/0.909	27.70/0.840	33.87/0.929	31.18/0.882	27.88/0.790	29.90/0.864
FDGAN [18]	23.15/0.921	LPNet [20]	33.61/0.958	FFDNet [89]	33.83/0.942	31.40/0.912	28.05/0.848	33.87/0.929	31.21/0.882	27.96/0.789	31.05/0.884
AirNet [36]	23.18/0.900	AirNet [36]	34.90/0.977	BRDNet [65]	34.42/0.946	31.99/0.919	28.56/0.858	34.10/0.929	31.43/0.885	28.16/0.794	31.44/0.889
Restormer [81]	30.87/0.969	Restormer [81]	36.74/0.978	AirNet [36]	34.40/0.949	32.10/0.924	28.88/0.871	34.14/0.936	31.48/0.893	28.23/0.806	31.54/0.897
PromptIR [56]	31.31/0.973	PromptIR [56]	37.04/0.979	PromptIR [56]	34.77/0.952	32.49/0.929	29.39/0.881	34.34/0.938	31.71/0.897	28.49/0.813	31.87/0.902
AdaIR [13]	31.80/0.981	AdaIR [13]	38.90/0.985	AdaIR [13]	34.96/0.953	32.74/0.931	29.70/0.885	34.36/0.938	31.72/0.897	28.49/0.813	32.00/0.903
IRPO (Ours)	**32.31/0.983**	**IRPO (Ours)**	**39.38/0.988**	**IRPO (Ours)**	**35.08/0.954**	**32.95/0.933**	**30.07/0.888**	**34.45/0.940**	**31.91/0.899**	**28.71/0.815**	**32.19/0.905**

Table A.3: Ablation study on the GRPO group size (G). Performance is measured by average PSNR/SSIM on the all-in-one 5-task benchmark.

Group	Dehazing	Deraining	Denoising	Deblurring	Low-Light	Average
3	31.50/0.982	38.59/0.985	31.89/0.892	28.32/0.854	24.03/0.856	30.87/0.914
4	**31.85/0.984**	**38.72/0.987**	32.13/0.896	28.89/0.863	**24.08/0.858**	**31.13/0.918**
5	31.58/0.983	38.71/0.986	**32.15/0.897**	**28.91/0.864**	24.05/0.857	31.08/0.917

A.2 Single-Task Restoration Results

Table A.2 reports the single-task setting where each model is trained separately on its respective dataset. IRPO consistently improves over the AdaIR backbone across dehazing, deraining, and denoising, indicating that the post-training recipe is not limited to the mixed all-in-one setting.

A.3 Additional Ablations

Table A.3 studies the rollout group size. Increasing G from 3 to 4 improves the average result, while $G = 5$ gives no clear gain and costs more rollouts. We therefore use $G = 4$ in the main experiments. Table A.4 shows that Qwen-VL, Q-Align, and DeQA-Score lead to similar quantitative outcomes, suggesting that IRPO does not rely on a single expert model.

A.4 Sensitivity Analysis of Reward Weights

We analyze the reward weights $(\lambda_{gen}, \lambda_{expert}, \lambda_{task})$ on the Dehaze (SOTS) task. Table A.6 shows that the generic physical constraint λ_{gen} should remain dominant (≥ 0.5) to preserve structural fidelity. The expert-quality weight λ_{expert} works as a coarse visual regularizer and is kept relatively low (≈ 0.1) to avoid deviations from the ground truth. The setting $\lambda_{gen} = 0.6$, $\lambda_{expert} = 0.1$, and $\lambda_{task} = 0.3$ gives the best balance in this study.

A.5 Transfer to a Second Restoration Backbone

We additionally evaluate the post-training recipe on the released PromptIR model [56]. We replace its three native routers with Dirichlet policy weights, keeping the prompt bases and encoder–decoder fixed. This adaptation uses a different policy parameterization from the Beta controls used for AdaIR. The post-training controls share the hard-data subset, 30 training epochs, crop settings, ℓ_1/consistency supervision, and deterministic inference. The real-world evaluation covers RRSHID, GT-Rain, SIDD, ReLoBlur, and LoLi-Street.

Table A.7 reports the routing, GDM, and full IRPO variants. PromptIR-IRPO improves over the original PromptIR by 0.34 dB on the three-task average and 0.83 dB on the five real-world datasets. These results provide evidence of transfer beyond AdaIR with an architecture-specific policy interface.

B Additional Visual Results

In addition to the synthetic comparison in Fig. 3, we provide real-world visual examples and more per-task qualitative comparisons against AirNet [36], PromptIR [56], and AdaIR [13].

Image Dehazing (Figure B.2). Figure B.2 shows that competing methods can suffer from color distortion or residual haze. In the second row, AirNet and AdaIR leave dense haze in distant regions,

Table A.4: Comparisons of different expert reward sources on the all-in-one 5-task benchmark. Performance is measured by average PSNR/SSIM.

Expert Reward Model	Dehazing	Deraining	Denoising	Deblurring	Low-Light	Average
w/ Q-Align [73]	31.59/0.984	38.70/0.987	32.12/0.895	28.88/0.863	24.06/0.857	31.07/0.917
w/ DeQA-Score [79]	31.47/0.982	38.65/0.985	31.98/0.893	28.76/0.861	23.95/0.854	30.96/0.915
w/ Qwen2.5-VL [4]	**31.85/0.984**	**38.72/0.987**	**32.13/0.896**	**28.89/0.863**	**24.08/0.858**	**31.13/0.918**

Table A.5: Practical overhead of IRPO. The VLM reward is used only during post-training and is removed at inference.

Method	Params (M)	Post-Training Time	Inference Time (1080p)	OOD Avg. PSNR
AdaIR	28.78	∼14 GPU days pre-training	2.0 s	20.30
IRPO	28.83	+4.2 GPU days post-training	2.0 s	**23.73**
Difference	*+0.05 M (+0.17%)*	*one-time cost*	*+0 ms*	***+3.43***

Table A.6: Sensitivity analysis of the reward weights on the SOTS (Dehaze) dataset.

λ_{gen}	λ_{expert}	λ_{task}	PSNR (dB)	SSIM
0.3	0.4	0.3	31.45	0.980
0.5	0.2	0.3	32.12	0.981
0.6	**0.1**	**0.3**	**32.31**	**0.983**
0.7	0.1	0.2	32.24	0.983

Table A.7: Post-training on PromptIR. Both columns report average PSNR/SSIM; the three-task setting follows the protocol in Table 3.

Method	Three-task PSNR/SSIM	Five real-world PSNR/SSIM
PromptIR	32.06/0.913	21.71/0.687
PromptIR + routing	32.12/0.914	21.91/0.691
PromptIR + GDM	32.20/0.915	22.03/0.694
PromptIR-IRPO	**32.40/0.917**	**22.54/0.702**

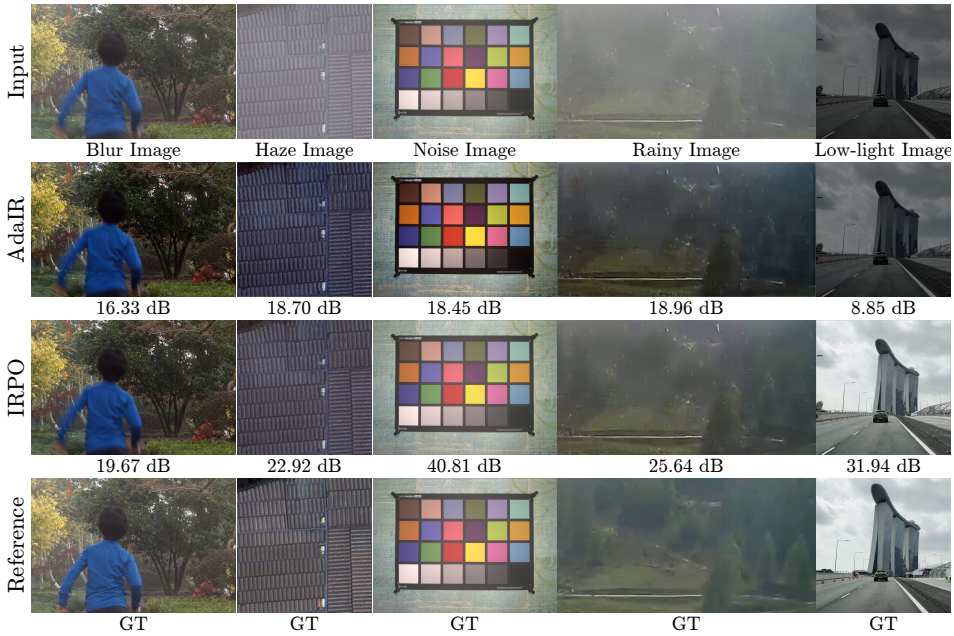


Figure B.1: Visual comparisons on real-world datasets. From left to right: deblurring, dehazing, denoising, deraining, and low-light enhancement. IRPO reduces residual degradations while preserving local structures.

leading to low contrast. IRPO recovers clearer visibility while maintaining more natural sky regions, which is consistent with the structural role of the *General Reward*.

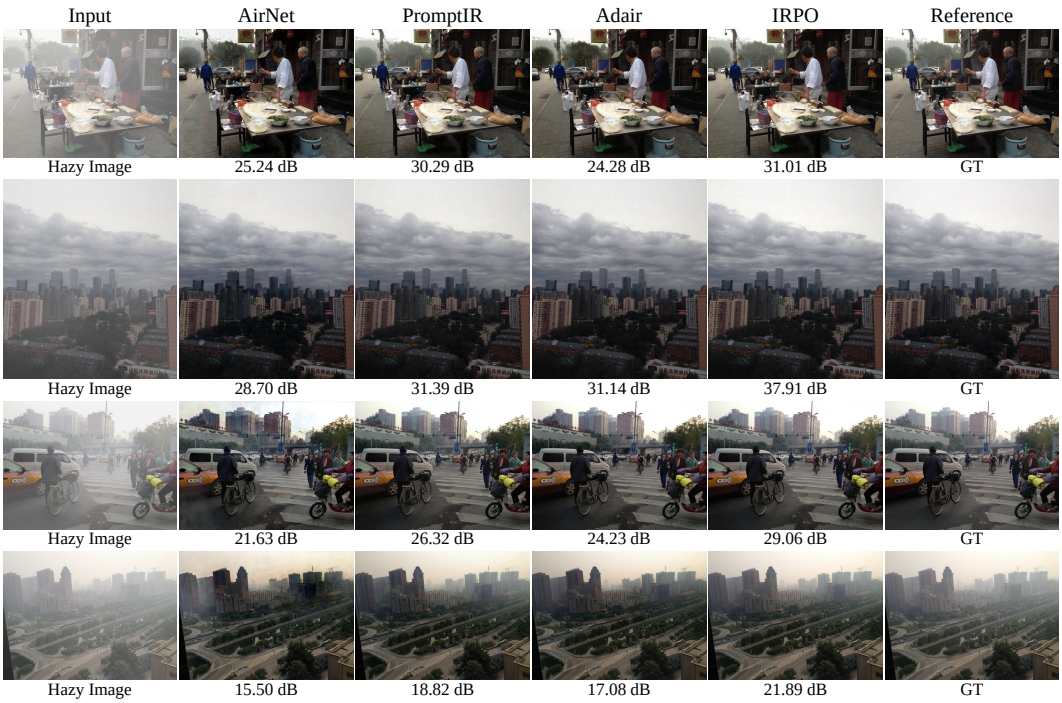


Figure B.2: Supplementary visual comparisons for image dehazing on synthetic datasets. IRPO yields clearer visibility and higher PSNR in these examples.

Image Denoising (Figure B.3). Figure B.3 reports results at the $\sigma = 50$ noise level. Severe noise often leads models to over-smooth details. In the "Face" example (fourth row), PromptIR and AdaIR blur facial features and hair textures. IRPO, guided by the *Task-Aware Reward* based on gradient consistency, better balances noise removal and texture retention.

Image Deraining (Figure B.4). Rain streaks often have clear directionality and varying density. In Figure B.4, baselines leave rain artifacts in the first row and blur background details in the third row. IRPO uses the anisotropy consistency reward to separate rain streaks from background textures, leading to cleaner removal with fewer artifacts.

These qualitative comparisons are consistent with the quantitative results and support the data-oriented and reward-oriented design of IRPO.

C Additional Implementation Details

Table C.1 lists the hyperparameters used in IRPO post-training. The table covers the model architecture, data handling, optimization, and GRPO training settings.

D Background on GRPO

Our reward-oriented optimization pillar is based on Group Relative Policy Optimization (GRPO). Here, we provide the detailed formulation for completeness.

Given a context x (the degraded image), a behavior policy $\pi_{\theta_{\text{old}}}(\cdot \mid x)$ draws a group of G responses $\{a_i\}_{i=1}^{G}$ (actions, i.e., control parameters). Each corresponding output y_i is scored by our reward model R (Sec. 3.3) to produce $\{r_i\}_{i=1}^{G}$.



Figure B.3: Supplementary visual comparisons for image denoising (noise level 50) on synthetic datasets. IRPO preserves more fine details in these examples.

GRPO constructs a *group-normalized* relative advantage A_i within the same input group, which serves as a stable, value-free advantage estimate:

$$\bar{r} = \frac{1}{G}\sum_{i=1}^{G} r_i, \qquad s = \sqrt{\frac{1}{G}\sum_{i=1}^{G}(r_i - \bar{r})^2 + \varepsilon}, \qquad A_i = \frac{r_i - \bar{r}}{s}. \tag{D.1}$$

Let the likelihood ratio be:

$$\rho_i(\theta) = \frac{\pi_\theta(a_i \mid x)}{\pi_{\theta_{\text{old}}}(a_i \mid x)}. \tag{D.2}$$

GRPO then maximizes a clipped surrogate objective $\mathcal{J}_{\text{GRPO}}(\theta)$, which is identical to PPO but replaces the standard advantage with A_i. It further enforces a KL penalty on a reference policy π_{ref}:

$$\begin{aligned}\mathcal{J}_{\text{GRPO}}(\theta) =& \mathbb{E}_{x\sim\mathcal{D},\{a_i\}\sim\pi_{\theta_{\text{old}}}}\Bigg[\frac{1}{G}\sum_{i=1}^{G}\min\Big(\rho_i(\theta)A_i,\ \text{clip}\big(\rho_i(\theta),1-\epsilon,1+\epsilon\big)A_i\Big) \\ &- \beta\,\widehat{\mathcal{D}}_{\text{KL}}\big(\pi_\theta(\cdot \mid x)\,\|\,\pi_{\text{ref}}(\cdot \mid x)\big)\Bigg].\end{aligned} \tag{D.3}$$

The KL divergence term is estimated on the batch using samples from $\pi_{\theta_{\text{old}}}$:

$$\widehat{\mathcal{D}}_{\text{KL}}\big(\pi_\theta \| \pi_{\text{ref}}\big) = \frac{1}{G}\sum_{i=1}^{G}\rho_i(\theta)\left[\log\pi_\theta(a_i \mid x) - \log\pi_{\text{ref}}(a_i \mid x)\right]. \tag{D.4}$$

The full objective used in the main paper, $\mathcal{L}_{\text{RL}}$ (Eq. 6), is the negative of Eq. (D.3) plus an entropy bonus. A_i is treated as a stop-gradient statistic in practice.

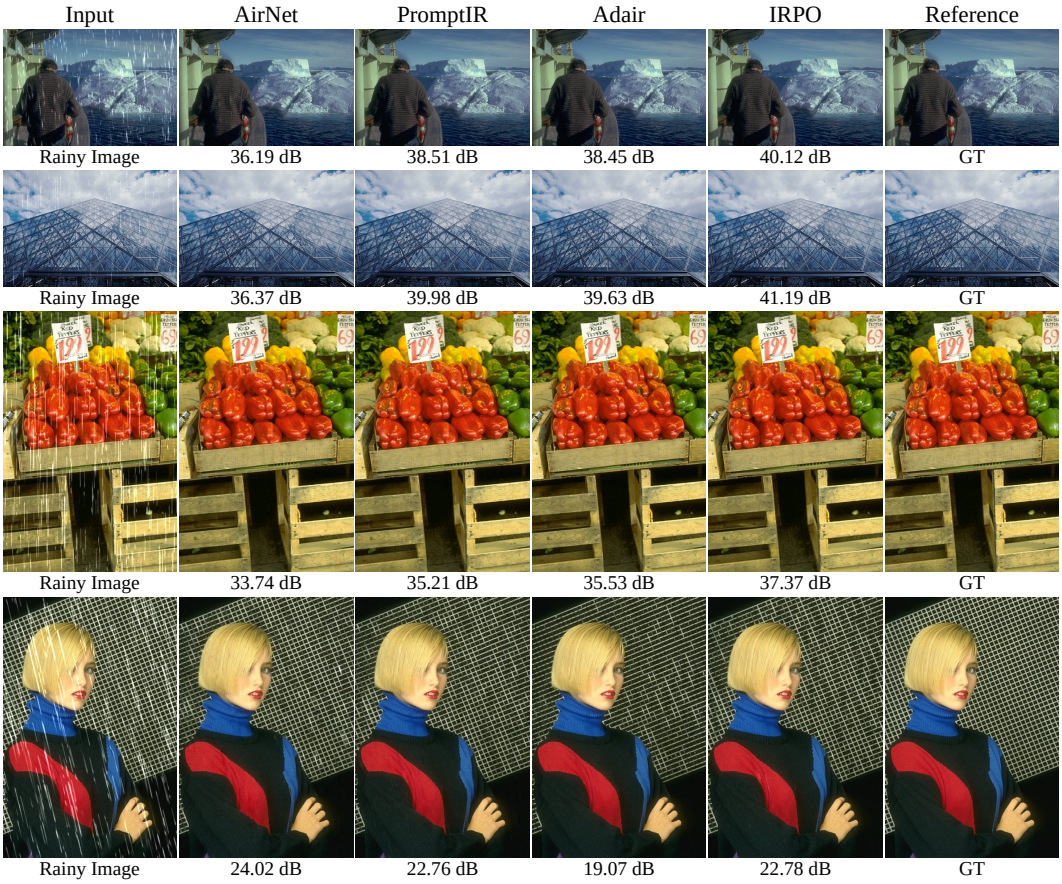


Figure B.4: Supplementary visual comparisons for image deraining on synthetic datasets. IRPO removes rain streaks while preserving background details.

E Reward Details

Notation and utilities. We work with linear RGB in $[0, 1]$. Let $\hat{y}$ denote the ground-truth image and y the restored output. Let $\text{clip}_{[0,1]}(t) = \min(\max(t, 0), 1)$ and $\varepsilon = 10^{-6}$. For luminance we use $Y = 0.299R + 0.587G + 0.114B$. Sobel gradients of an image Z are $G_x = \text{Sobel}_x(Z)$, $G_y = \text{Sobel}_y(Z)$, and magnitude $M(Z) = \sqrt{G_x^2 + G_y^2 + 10^{-12}}$. When needed, $\mu(\cdot)$, $\sigma^2(\cdot)$, and $\text{std}(\cdot)$ denote spatial mean, variance, and standard deviation.

E.1 Generic Quality Term R_{gen}

This reward is a weighted combination of five metrics, each normalized to $[0, 1]$.

$$\begin{aligned} R_{\text{gen}}(y, \hat{y}) = \quad & 0.25 \cdot R_{\text{clip}}(y, \hat{y}) + 0.25 \cdot R_{\text{LPIPS}}(y, \hat{y}) + 0.15 \cdot R_{\text{aes}}(y) \\ & + 0.20 \cdot R_{\text{PSNR}}(y, \hat{y}) + 0.15 \cdot R_{\text{SSIM}}(y, \hat{y}), \end{aligned}$$

where R_{clip} is the CLIP similarity [59], R_{LPIPS} is the normalized LPIPS perceptual score [90], R_{aes} is a normalized aesthetic score from a pre-trained predictor [62], R_{PSNR} is the PSNR score mapped to $[0, 1]$ via fixed thresholds $(\tau_{\min}, \tau_{\max})$, and R_{SSIM} is the standard SSIM score.

E.2 VLM-based Verifier R_{expert}

We use Qwen2.5-VL-7B-Instruct [4] as a visual-quality verifier. The model receives the degraded input (x), the restored output (y), and the ground truth $(\hat{y})$. It returns a score from 1 to 5, which we linearly rescale. The exact verifier prompt is provided at the end of this appendix section, after the reward definitions.

Table C.1: Detailed hyperparameters for IRPO post-training.

Hyperparameter	Value / Description
Base Channel Width (model_dim)	48
Encoder-Decoder Levels	4 levels
Transformer Blocks	[4, 6, 6, 8]
Multi-head Attention Heads	[1, 2, 4, 8]
Refinement Blocks	4
FFN Expansion Factor	2.66
Frequency Policy (AFLB)	3 modules inserted between decoder levels
Action Space	4D continuous $(r_h, r_l, g_f, g_o) \in (0, 1)^4$
Input Patch Size	128×128
Augmentation	Random horizontal/vertical flips
Hard-Mining Ratio	Top 30% hardest samples
Optimizer	Adam ($\beta_1 = 0.9, \beta_2 = 0.999$)
Base Learning Rate (LR)	3×10^{-5}
Total Epochs	30 (default for main results)
Batch Size	1 per GPU (constrained by $G = 4$)
Group Size (G)	4
Entropy Regularization	0.01
Supervised Loss Weight (λ_{sup})	Linear decay: $0.35 \to 0.1$
Consistency Loss Weight (λ_{cons})	Linear decay: $0.2 \to 0.05$

E.3 Task-aware Shaping R_{task}

For each degradation type, we define a specialized reward.

(1) Denoising — gradient consistency. Let $M_g = M(Y_g)$, $M_{\text{rest}} = M(y)$. Define the GT baseline and mean magnitude deviation:

$$B = \mu\big(M_g\big), \qquad D = \mu\big(|M_{\text{rest}} - M_g|\big). \tag{E.1}$$

The reward encourages the restored gradients to approach the clean target:

$$R_{\text{grad}} = \text{clip}_{[0,1]}\left(1 - \frac{D}{B + \varepsilon}\right). \tag{E.2}$$

This reward encourages the restored texture and edge gradients (M_{rest}) to match the ground truth (M_g), reducing over-smoothing during denoising.

(2) Deraining — anisotropy consistency. Compute average absolute Sobel responses on luminance:

$$E_x = \mu\big(|G_x(y)|\big), \qquad E_y = \mu\big(|G_y(y)|\big). \tag{E.3}$$

Define normalized anisotropy

$$A = \frac{|E_x - E_y|}{\max(E_x + E_y, \varepsilon)}. \tag{E.4}$$

The reward favors isotropy (i.e., $A \approx 0$):

$$R_{\text{aniso}} = 1 - \text{clip}_{[0,1]}(A). \tag{E.5}$$

Rain streaks are anisotropic and create a directional bias in image gradients ($E_x \neq E_y$). A clean, rain-free image is closer to isotropic ($E_x \approx E_y$). This reward encourages isotropic outputs after deraining.

(3) Dehazing — contrast closeness. Let $C(Z) = \text{std}(Y_Z)$ be a proxy for the contrast of image Z. We measure symmetry-aware contrast closeness to ground truth:

$$R_{\text{contrast}} = \text{clip}_{[0,1]}\left(\min\left(\frac{C_{\text{rest}}}{C_g + \varepsilon}, \ \frac{C_g}{C_{\text{rest}} + \varepsilon}\right)\right). \tag{E.6}$$

Haze is a low-frequency component that reduces image contrast (lower $\text{std}(Y)$). This reward encourages the restored contrast C_{rest} to match the ground truth C_g, penalizing both under-enhancement (remaining haze) and over-enhancement (oversaturation).

(4) Deblurring — sharpness closeness. Using the mean gradient magnitude $S(Z) = \mu(M(Z))$ as a proxy for sharpness:

$$R_{\text{sharp}} = \text{clip}_{[0,1]}\left(\min\left(\frac{S_{\text{rest}}}{S_g + \varepsilon},\ \frac{S_g}{S_{\text{rest}} + \varepsilon}\right)\right). \tag{E.7}$$

Blur removes high-frequency information and lowers gradient magnitudes. This reward encourages the mean sharpness S_{rest} to match the ground truth S_g, which supports edge and detail recovery.

(5) Low-light — exposure & color consistency. Exposure alignment on luminance means:

$$d = \left|\mu(y) - \mu(Y_g)\right|, \qquad R_{\text{exp}} = \text{clip}_{[0,1]}\left(1 - \frac{d}{0.5}\right). \tag{E.8}$$

Color consistency sums mean-channel deviations ($c \in \{R, G, B\}$):

$$d_c = \sum_c \left|\mu(c_{\text{rest}}) - \mu(c_g)\right|, \qquad R_{\text{color}} = \text{clip}_{[0,1]}\left(1 - \frac{d_c}{0.6}\right). \tag{E.9}$$

Low-light enhancement has two goals: fidelity and brightness. R_{exp} ensures the overall luminance $\mu(y)$ matches the target, while R_{color} prevents color shifts (e.g., a green or purple tint) by penalizing deviation in individual R, G, B channels.

Putting the task terms together. For each sample, we select the reward(s) corresponding to its degradation:

$$\begin{aligned} R_{\text{task}}^{(\text{denoise})} &= R_{\text{grad}}, \quad R_{\text{task}}^{(\text{derain})} = R_{\text{aniso}}, \\ R_{\text{task}}^{(\text{dehaze})} &= R_{\text{contrast}}, R_{\text{task}}^{(\text{deblur})} = R_{\text{sharp}}, \\ R_{\text{task}}^{(\text{low-light})} &= 0.2 \cdot R_{\text{exp}} + 0.1 \cdot R_{\text{color}}. \end{aligned} \tag{E.10}$$

E.4 Final Reward Combination

The final reward signal used in Eq. (3) of the main paper is the weighted sum of the three components:

$$R(y, \hat{y}) = \quad 0.6 \cdot R_{\text{gen}}(y, \hat{y}) + 0.1 \cdot R_{\text{expert}}(y) + 0.3 \cdot R_{\text{task}}^{(k)}(y, \hat{y})$$

where k is the known degradation type for the training sample.

E.5 VLM Verifier Prompt

The following prompt is used to obtain the expert score before linear rescaling.

Qwen-VL Verifier Prompt

You are an image-restoration expert. You will be given three images:

1. The degraded input that suffers from a certain type of degradation.
2. The restored output generated by a model.
3. The clean ground-truth reference.

Task:

1. Identify the most plausible degradation type of the input image. Consider categories such as denoising (0/1/2, different noise levels), deraining (3), dehazing (4), deblurring (5), or low-light enhancement (6). Briefly justify your reasoning.
2. Compare the restored output against the ground truth with respect to the identified degradation type. Pay attention to:
   - Noise or streak removal quality for denoising/deraining.
   - Contrast and haze removal for dehazing.
   - Sharpness recovery for deblurring.
   - Exposure and color constancy for low-light enhancement.
3. Highlight specific improvements and any remaining artifacts.
4. Provide a final quality score from 1 to 5, where:
   - 1: severe artifacts or almost no improvement,
   - 2: minor improvement but significant issues remain,
   - 3: moderate improvement with noticeable gaps,
   - 4: good restoration with only small flaws,
   - 5: near-perfect restoration indistinguishable from ground truth.

Respond in the following XML-style format:

```
<Assessment>
  <Degradation>
    [type and reasoning]
  </Degradation>
  <Analysis>
    [detailed comparison]
  </Analysis>
  <Score>X</Score>
</Assessment>
```

F Policy Parametrization Details

Our policy π_θ consists of 6 independent policy heads distributed across the 3 AFLB modules in the AdaIR decoder. Each AFLB module contains two heads: (1) a **Policy Rate** head, and (2) a **Policy Fuse** head, which correspond to the actions (r_h, r_l) and (g_f, g_o) respectively.

Policy Rate Head (r_h, r_l). The Policy Rate head determines the size of the frequency-domain mask.

- **Input:** It takes the initial degraded image x (which is fed to all AFLB modules) and passes it through an FFT.
- **Process:** A small CNN head takes the pooled features of x and outputs four positive scalars $(\alpha_h, \beta_h, \alpha_l, \beta_l)$.
- **Action:** These scalars parameterize two Beta distributions. During training, we sample $r_h \sim \mathrm{Beta}(\alpha_h, \beta_h)$ and $r_l \sim \mathrm{Beta}(\alpha_l, \beta_l)$. These actions $(r_h, r_l) \in (0, 1)^2$ define the height and width of the low-frequency mask. This mask is applied to the FFT of x to separate frequency components, which are then used to generate the intermediate feature O_i (the "Output Feature" in Figure 2).

Policy Fuse Head (g_f, g_o)**.** The "Policy Fuse" head determines how to blend the intermediate feature with the decoder's latent feature.

- **Input:** It takes the latent feature y_i (the "Latent Feature" in Figure 2) coming from the previous decoder block.
- **Process:** A separate small CNN head takes the pooled features of y_i and outputs four positive scalars $(\alpha_1, \beta_1, \alpha_2, \beta_2)$.
- **Action:** We sample $g_f \sim \mathrm{Beta}(\alpha_1, \beta_1)$ and $g_o \sim \mathrm{Beta}(\alpha_2, \beta_2)$. These actions $(g_f, g_o) \in (0, 1)^2$ are the dynamic weights used in the final fusion, as shown in Eq. (5) in the main paper.

Log-Probability Calculation. The policy π_θ is the joint distribution of these 4 independent actions. The log-probability $\log \pi_\theta(a \,|\, x, y)$ is required for the policy gradient calculation (e.g., in Eq. D.4). This function has the following closed form:

$$\begin{aligned} \log \pi_\theta(a \,|\, x, y) = & \sum_{k \in \{h,l,f,o\}} \Big[(\alpha_k - 1)\, \log a_k + (\beta_k - 1)\, \log\big(1 - a_k\big) \Big] \\ & - \sum_{k \in \{h,l,f,o\}} \log B(\alpha_k, \beta_k), \end{aligned} \tag{F.1}$$

where a_k is the sampled action for $k \in \{h, l, f, o\}$, (α_k, β_k) are the distribution parameters output by the corresponding policy head, and $B(\cdot, \cdot)$ is the Beta function.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state IRPO as a GRPO-based post-training framework for image restoration, with data-oriented supervision, reward-oriented optimization, and empirical gains on in-domain and OOD benchmarks. The claims are framed as fidelity-oriented and quality-aware restoration rather than direct human-preference alignment.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The main text discusses the scope of the post-training wrapper, the AdaIR-specific stochastic-head parameterization, the one-time VLM-assisted post-training cost, and the role of IQA metrics as supporting diagnostics rather than a replacement for human preference studies.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper is an empirical method paper and does not claim new theoretical guarantees. The GRPO objective and its instantiation for restoration are described for completeness.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper reports datasets, evaluation protocols, metrics, baselines, model initialization, optimization settings, group size, reward components, and hyperparameters. Additional implementation details and ablations are provided in the appendix.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The experiments use public benchmark datasets and cite their original sources. A public code URL is not provided in this manuscript.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The experimental section specifies the in-domain and OOD datasets, all-in-one and single-task settings, evaluation metrics, optimizer, learning rate, crop size, group size, training epochs, reward weights, and compute setup. The appendix lists detailed hyperparameters.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Following common image-restoration benchmark practice, the main tables report deterministic benchmark PSNR/SSIM and IQA scores rather than repeated-run confidence intervals. Training stability curves are included, but repeated-seed error bars for every benchmark are not reported.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The implementation details specify the GPU setup used for IRPO post-training and VLM reward serving, and the main paper reports post-training time, parameter overhead, and inference latency.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics?

   Answer: [Yes]

   Justification: The work studies image-restoration training on public benchmarks, does not collect private user data, and does not conduct human-subject experiments. The paper avoids unsupported claims of human-preference alignment.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The method can improve restoration robustness for images captured under adverse degradations, which may benefit downstream visual analysis. Potential negative uses include over-reliance on enhanced imagery or deployment in sensitive surveillance settings; the method should be used with task-specific validation and provenance awareness.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse?

    Answer: [N/A]

    Justification: The paper does not introduce a high-risk dataset or a model designed for harmful generation. It uses public restoration benchmarks and releases a restoration training recipe rather than private data or identity-recognition capability.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper cites the original datasets, baselines, metrics, and Qwen2.5-VL source used for expert reward scoring. Public assets are used for research benchmarking under their respective release terms.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: The paper documents the method, hyperparameters, reward components, and evaluation protocols, but a public code and configuration release is not linked in this manuscript.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not include crowdsourcing studies or direct human-subject experiments.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether IRB approvals were obtained?

    Answer: [N/A]

    Justification: No human-subject protocol is conducted in this study.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research?

    Answer: [Yes]

    Justification: The paper explicitly describes Qwen2.5-VL as a train-time expert reward model. It is used only to score restoration candidates during post-training and is removed at inference.