Restore-R1: Efficient Image Restoration Agents via Reinforcement Learning with Multimodal LLM Perceptual Feedback

Jianglin Lu[1,2]*, Yuanwei Wu[1], Ziyi Zhao[1], Hongcheng Wang[1], Felix Jimenez[1], Abrar Majeedi[1,3], Yun Fu[2]
[1]Amazon, [2]Northeastern University, [3]University of Wisconsin-Madison
jianglinlu@outlook.com, {ywuvlms,zhaziyi,hongchw}@amazon.com, yunfu@ece.neu.edu

Abstract

Complex image restoration aims to recover high-quality images from inputs affected by multiple degradations such as blur, noise, rain, and compression artifacts. Recent restoration agents, powered by vision-language models and large language models, offer promising restoration capabilities but suffer from significant efficiency bottlenecks due to reflection, rollback, and iterative tool searching. Moreover, their performance heavily depends on degradation recognition models that require extensive annotations for training, limiting their applicability in label-free environments. To address these limitations, we propose a policy optimization-based restoration framework that learns an lightweight agent to determine tool-calling sequences. The agent operates in a sequential decision process, selecting the most appropriate restoration operation at each step to maximize final image quality. To enable training within label-free environments, we introduce a novel reward mechanism driven by multimodal large language models, which act as human-aligned evaluator and provide perceptual feedback for policy improvement. Once trained, our agent executes a deterministic restoration plans without redundant tool invocations, significantly accelerating inference while maintaining high restoration quality. Extensive experiments show that despite using no supervision, our method matches SOTA performance on full-reference metrics and surpasses existing approaches on no-reference metrics across diverse degradation scenarios.

1. Introduction

Image restoration aims to reconstruct high-quality images from degraded observations, where the underlying corruption process is often unknown and involves complex interactions [36, 51, 52]. In real-world scenarios, low-quality images rarely suffer from a single type of distortion; instead, they typically exhibit a mixture of blur, rain streaks, noise, haze, low-light conditions, and compression artifacts, among others [10, 12, 24, 28, 50]. This setting, referred to as *complex image restoration (CIR)* [4, 14, 23, 74], is considerably more challenging than classical single-degradation restoration due to compound, nonlinear, and spatially varying corruption patterns. CIR further introduces fundamental difficulties, including ambiguous degradation sources, the absence of paired clean supervision, and degradation-stacking effects that amplify visual deterioration.

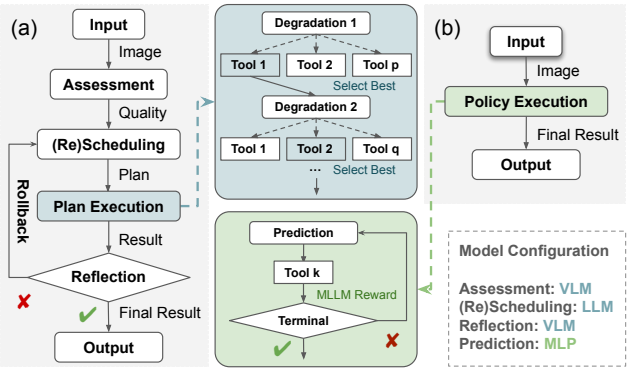


Figure 1. (a) Existing restoration agents [4, 23, 74] typically consist of assessment, scheduling, execution, reflection, and rollback, using VLMs for degradation recognition and LLMs for plan making; (b) Our Restore-R1 agent determines the tool-calling sequence via a single policy execution, avoids the need for iterative trial-and-error, and generalizes to label-free environments.

To address the CIR problem, a variety of methods have emerged in recent years. A prominent direction focuses on all-in-one restoration, which handles heterogeneous and often unknown degradations with unified restoration models [15, 17, 22, 34, 63]. Early solutions primarily adopted unified CNN architectures to learn general low-level image priors, such as MPRNet [68] and MIRNet [67]. More recent methods leverage Transformer-based architectures, including Restormer [69] and Uformer [57], to better capture long-range dependencies beneficial for joint restoration. To

*Corresponding author. Work done during an internship at Amazon.

Table 1. Comparison with existing image restoration agents, where ✓ indicates the feature is required and ✗ indicates that it is not.

Agentic Method	# of Params	Restoration Planner	Ground-Truth	Recognition Model	Reflection	Rollback
RestoreAgent [4]	8B	Llava-Llama3 [48]	Optimal Sequence	Llava-Llama3 [48]	✓	✓
AgenticIR [74]	7B	GPT-4 [1]	Image+Label+Level	DepictQA [64]	✓	✓
MAIR [14]	7B	GPT-4o [1]	Image+Label+Level	DepictQA [64]	✓	✗
Q-Agent [73]	7B	Greedy Strategy	Image+Label+Level	Qwen2-VL [55]	✓	✗
HybridAgent [23]	8.2B	Llama3.2-Instruct [9]	Image+Label	Co-instruct [60]	✓	✓
Ours	0.28B	Image Encoder + MLP	✗	✗	✗	✗

further improve adaptability to heterogeneous distortions, various conditioning and modulation strategies have been introduced, such as degradation embeddings (e.g., AirNet [22]), prompt-based conditioning (e.g., PromptIR [37], InstrucIR [6]), dynamic expert routing (e.g., AMIR [63]), CLIP-driven control (e.g., DA-CLIP [29]), and generative priors (e.g., AutoDIR [15]). While providing a general solution, all-in-one models often struggle with complex real-world degradation mixtures, exhibiting limited flexibility and sub-optimal performance due to the need to fit broad degradation distributions [4, 50, 52, 74].

With the rapid progress of intelligent agents across multiple domains, agent-based image restoration has recently emerged as a compelling alternative paradigm. Inspired by human-like decision-making, these methods leverage large language models (LLMs) [1, 9, 28, 48] and vision-language models (VLMs) [55, 64, 70, 71] to autonomously recognize image degradations, plan restoration pipelines, invoke appropriate restoration tools, and progressively refine outputs toward high-quality results. The state-of-the-art (SOTA) approaches include RestoreAgent [4], AgenticIR [74], MAIR [14], Q-Agent [73], and HybridAgent [23] (see Table 1 for a detailed comparison among them). While demonstrating promising performance, these agent-driven frameworks suffer from substantial efficiency limitations. Determining the optimal sequence of restoration tools, especially involving iterative reflection, trial-and-error, and tool re-scheduling (as shown in Figure 1), introduces heavy computational overhead and prolonged inference time. This process becomes particularly more time-consuming and suboptimal as the degradation patterns become more diverse and complex. Moreover, most existing agents rely on a recognition model to identify degradations for tool selection, which requires extensive annotations for training, such as degradation labels [14, 74] or optimal tool-calling trajectories [4]. Such supervision dependencies restrict their universality and deployment in real-world, label-free environments.

To overcome these challenges, we propose a policy optimization-based image restoration framework that learns a lightweight agent, named Restore-R1, to select tools and determine their execution order, aiming to maximize overall restoration quality. Our framework adopts an actor–critic architecture, where the actor network sequentially selects most appropriate restoration tools based on the observed image state, while the critic network estimates the expected return and provides feedback guidance to assess whether the chosen actions outperform the expected behavior. This formulation enables our agent to progressively refine degraded inputs through a sequence of learned restoration decisions. To handle the label-free environments, we introduce a perceptual reward mechanism powered by multimodal large language models (MLLMs) [7, 8, 19, 26, 27], which serve as a perception evaluator to provide human-aligned feedback on restored outputs. The actor is then optimized based on these reward signals to encourage operation sequences that yield higher cumulative gains. To ensure stable policy learning and avoid abrupt policy shifts, we adopt the clipping strategies during training as in [43, 44]. During inference, our agent executes its learned policy in a single forward pass to deterministically produce restoration sequences, eliminating iterative trial-and-error loops and redundant tool invocations typically required in prior agent-based solutions [4, 14, 74] (as shown in Figure 1). In summary, our main contributions are as follows:

- To the best of our knowledge, we introduce the first label-free agentic framework for complex image restoration, requiring no ground-truth images, degradation labels, or optimal sequence guidance.
- We design a lightweight agent that directly determines tool-calling sequences through a single policy execution, eliminating expensive trial-and-error loops used in prior agentic approaches [4, 14, 74].
- We propose a novel and general reward mechanism that leverages MLLMs to provide human-like perceptual feedback for action evaluation and policy optimization.
- Extensive experiments show that our agent matches supervised SOTA methods on full-reference metrics and outperforms them significantly on no-reference metrics.

2. Related Work

2.1. Complex Image Restoration

All-in-One Image Restoration. All-in-one methods [15, 17, 22, 34] aim to handle diverse and often unknown degradations within a single unified framework. For example,

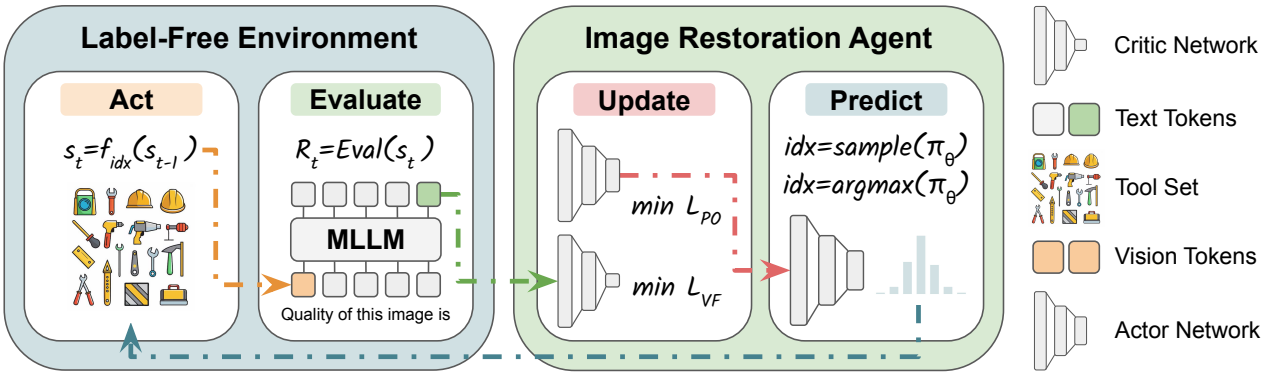


Figure 2. Framework overview. The restoration agent predicts the next action based on the current input status (sampling actions during training while selecting the highest-probability action during inference, see Sec. 3.2). The environment executes the chosen action, evaluates the restored output with an MLLM, and returns a feedback signal to update the agent's policy (see Sec. 3.3). Through this iterative interaction, the agent progressively refines its decision-making policy without ground-truth supervision.

AirNet [22] learns a unified backbone capable of adapting to various corruption types. PromptIR [37] introduces employs prompt-based conditioning to steer restoration and dynamically handle different degradations. DA-CLIP [29] leverages vision-language priors and natural-language instructions to flexibly control restoration behavior, and InstrucIR [6] similarly aligns restoration outputs with human instructions. AutoDIR [15] leverages latent diffusion models to automatically infer and correct degradations without explicit user instructions. Although these approaches provide a general solution, all-in-one models often struggle with complex real-world degradation mixtures, exhibiting limited flexibility and sub-optimal performance due to the need to fit broad degradation distributions [23, 51, 52, 74].

Agentic Image Restoration. Agent-based approaches treat restoration as a process of iterative decision making, where an intelligent agent analyzes the input image, identifies degradation types, and composes a sequence of specialized restoration tools. For example, RestoreAgent [4] fine-tunes a VLM to directly generate execution plans, while AgenticIR [74] integrates VLM-based quality analysis with LLM-based planning following a human-inspired pipeline. Q-Agent [73] improves efficiency via a greedy planner driven, while MAIR [14] categorizes degradations into scene, imaging, and compression types, and reverses them in an inverse order. HybridAgent [23] designs routing strategy to balance accuracy and efficiency. Despite notable progress, existing agent-based systems often suffer from high inference overhead due to extensive search, and their effectiveness heavily relies on auxiliary recognition models trained with ground-truth sequences or degradation labels, which limits their practicality and generalization in label-free real-world scenarios. Table 1 presents a comprehensive comparison between these approaches.

2.2. Reinforcement Learning

Reinforcement learning (RL) [31–33, 45, 46, 53] frames decision making as an interactive optimization problem, where an agent refines its actions through feedback from the environment to achieve maximal cumulative benefit. Most RL algorithms fall into two categories: value-based and policy-based. Value-based techniques estimate action-dependent returns through a value function and induce a decision strategy by favoring actions with higher predicted returns. Representative examples include Q-learning [58] and its deep variant DQN [31]. In contrast, policy-based approaches explicitly optimize the policy parameters, typically via gradient-based optimization, to maximize expected rewards. Representative examples include REINFORCE [59], TRPO [41], PPO [43], RLHF [33] and the recently proposed GRPO [44]. In this work, we utilize RL to determine optimal tool-execution strategies for complex image restoration. Unlike prior RL-based approaches [3, 35, 38, 66], which often rely on predefined task labels, heuristic rules, or supervised sequence annotation, we learn a policy that autonomously selects both restoration tools and their execution order in a label-free environment, guided solely by perceptual feedback from MLLMs.

3. Restore-R1 Framework

3.1. Problem Definition

In complex image restoration, we are given a library of n restoration tools $\mathcal{F} = \{f_1, f_2, \ldots, f_n\}$, where each tool can handle one or more specific degradations. Given a degraded input x_L, the objective is to recover a clean image $\widetilde{x}_H$ by executing a sequence of restoration tools:

$$\widetilde{x}_H = (f_k \circ f_{k-1} \circ \cdots \circ f_0)(x_L), \tag{1}$$

where f_t denotes the tool selected at step t. The core challenge lies in identifying the optimal tool-calling sequence:

$$\tau^* := f_k^* \circ f_{k-1}^* \circ \ldots \circ f_0^* \tag{2}$$

that minimizes the discrepancy between $\widetilde{x}_H$ and the clean image x_H. Existing agent-based methods either (i) finetune a large vision-language model to directly predict the tool sequence [4] or (ii) recognize degradations and then search the action space via exhaustive, greedy, or prior knowledge-based strategies [14, 23, 73, 74]. As discussed in Sec. 2.1, these approaches incur substantial inference latency due to exhaustive tool searching and frequent rollbacks, and their performance depends critically on supervised degradation labels for recognition model training. Our goal is to replace such search-heavy pipelines with a policy optimization-driven agent, named SimpeCall, trained in a label-free environment using feedback from MLLMs. The overall SimpeCall architecture is illustrated in Figure 2, where an restoration agent interacts with an environment that executes restoration actions, receives MLLM-based qualitative feedback, and learns an efficient restoration policy.

3.2. Image Restoration Agent

Our Restore-R1 agent performs two tasks: network update and action prediction. Let a restoration trajectory be:

$$\zeta := \{s_0, f_0, s_1, f_1, \ldots, s_T\}, \tag{3}$$

where $s_0 = x_L$, $s_{t+1} = f_t(s_t)$, s_t and T denotes the maximum allowable length of the restoration sequence. Let $\pi_\theta(f|s)$ denote a policy (actor) network parameterized by θ, which governs the agent's behavior by predicting a state-conditioned action distribution. Our objective is to optimize the expected return $R(\zeta)$:

$$\max_\theta \mathcal{L}(\theta) = \mathbb{E}_{\zeta\sim\pi_\theta}[R(\zeta)] = \sum_\zeta P(\zeta \mid \theta)\, R(\zeta), \tag{4}$$

where $R(\zeta) := \sum_{t=0}^{T} R(s_t, f_t)$ is the cumulative rewards, $R(s_t, f_t)$ denotes the immediate reward associated with executing action f_t in state s_t, and $P(\zeta \mid \theta)$ is the probability of experiencing trajectory ζ under the policy π_θ.

Network Update. To optimize the policy network π_θ, we let the agent interact with the environment to collect trajectories, receive reward signals from an evaluator (see Sec. 3.3), and update the network parameters through policy optimization. A naïve policy objective function would be [43]:

$$\mathcal{L}(\theta) = \mathbb{E}_t \left[\log \pi_\theta(f_t \mid s_t)\hat{A}_t\right], \tag{5}$$

where $\pi_\theta(f_t \mid s_t)$ denotes the likelihood that the policy selects action f_t given state s_t, and $\hat{A}_t$ is the corresponding advantage, where $\hat{A}_t > 0$ means the action is better than other action possible at that state. To stabilize learning, we adopt the clipped surrogate objective function [43]:

$$\mathcal{L}_{\text{PO}}(\theta) = \mathbb{E}_t \left[\min\left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right],$$

where the scalar ϵ limits the magnitude of policy updates, and $r_t(\theta) = \frac{\pi_\theta(f_t|s_t)}{\pi_{\theta_{\text{old}}}(f_t|s_t)}$ is the importance sampling ratio, which measures how the action probability at state s_t changes from the old policy to the current one. A positive value of $r_t(\theta)$ indicates that action f_t is favored more strongly by the current policy compared to the earlier one. Clipping the ratio prevents overly large policy deviations, enforcing a soft trust region analogous to TRPO [41] while remaining computationally simple.

To evaluate the quality of the chosen action, we follow [41, 43, 44] and maintain a critic network V_ϕ trained via:

$$\mathcal{L}_{\text{VF}}(\phi) = \mathbb{E}_t \left[\left(V_\phi(s_t) - \hat{R}_t\right)^2\right], \tag{6}$$

where $\hat{R}_t$ is an estimate of the return. We then compute $\hat{A}_t$ using generalized advantage estimation (GAE) [42]:

$$\hat{A}_t = \sum_{l=0}^{\infty} (\gamma\lambda)^l \delta_{t+l},\ \delta_t = R(s_t, f_t) + \gamma V(s_{t+1}) - V(s_t),$$

where γ discounts future rewards and λ controls the GAE weighting. To encourage exploration, we measure the uncertainty of the policy distribution $\mathcal{H}[\cdot]$ with an entropy bonus term:

$$\mathcal{L}_{\text{EB}}(\theta) = \mathbb{E}_t \left[\mathcal{H}\left[\pi_\theta(\cdot \mid s_t)\right]\right], \tag{7}$$

Then the final objective can be written as:

$$\mathcal{L} = \mathcal{L}_{\text{PO}} - c_1 \cdot \mathcal{L}_{\text{VF}} + c_2 \cdot \mathcal{L}_{\text{EB}}, \tag{8}$$

where c_1 and c_2 are balancing coefficients.

Action Prediction. During training, actions are sampled from the stochastic policy, i.e., $f_t \sim \pi_\theta(\cdot|s_t)$, to encourage exploration and collect diverse trajectories. In contrast, during inference, the agent adopts a deterministic strategy by selecting the most probable action, i.e., $f_t = \arg\max_f \pi_\theta(f|s_t)$. The policy network comprises a vision encoder followed by a lightweight multilayer perceptron (MLP). Each state s_t is represented by concatenating the visual features extracted from the current input with an action record vector that summarizes the policy's past decisions. Given this state representation, the actor predicts the next restoration tool, and this process is repeated iteratively until the image converges to a visually satisfactory result.

3.3. Label-Free Environment

The environment is responsible for action execution and evaluation. It applies a selected restoration tool to the input

Table 2. Comparison with SOTA methods on Full-reference metrics: F1 (PSNR), F2 (SSIM), F3 (LPIPS), and No-reference metrics: N1 (MANIQA), N2 (CLIP-IQA), N3 (MUSIQ), N4 (DeQA-Score) across all settings. "GT" denotes whether ground truth supervision is required for model training. Remarkably, **without using any labels**, our method matches SOTA performance on full-reference metrics and outperforms compared methods on no-reference metrics.

Method	GT	Full-Reference			No-Reference				Full-Reference			No-Reference			
		F1 ↑	F2 ↑	F3 ↓	N1 ↑	N2 ↑	N3 ↑	N4 ↑	F1 ↑	F2 ↑	F3 ↓	N1 ↑	N2 ↑	N3 ↑	N4 ↑
		Setting I							Setting II						
AirNet [22]	✓	18.489	0.608	0.374	0.295	0.399	48.850	2.911	17.196	0.642	0.326	0.323	0.439	53.794	3.098
PromptIR [37]	✓	19.198	0.609	0.371	0.300	0.400	49.189	2.907	17.485	0.660	0.301	0.324	0.441	53.823	3.154
InstructIR [6]	✓	18.394	0.586	0.416	0.268	0.390	46.047	2.844	17.508	0.593	0.412	0.286	0.433	48.147	2.994
MiOIR(R) [17]	✓	19.970	0.659	0.337	0.282	0.430	50.269	2.998	18.045	0.671	0.304	0.288	0.479	53.303	3.173
MiOIR(U) [17]	✓	19.920	0.672	0.332	0.289	0.436	51.217	3.011	17.786	0.676	0.278	0.299	0.466	55.143	3.191
DA-CLIP [29]	✓	19.252	0.614	0.370	0.292	0.417	50.320	2.974	17.695	0.637	0.349	0.308	0.414	53.729	3.071
AutoDIR [15]	✓	19.405	0.644	0.346	0.306	0.431	53.580	3.186	18.540	0.664	0.319	0.318	0.448	55.683	3.257
AgenticIR [74]	✓	20.923	0.698	0.306	0.315	0.441	58.590	3.423	20.418	0.719	0.300	0.310	0.448	58.643	3.434
Ours	✗	19.513	0.647	0.365	0.349	0.525	63.003	3.657	17.958	0.636	0.385	0.343	0.530	63.344	3.599
		Setting III							Setting IV						
AirNet [22]	✓	16.469	0.517	0.504	0.271	0.349	45.650	2.540	16.128	0.472	0.606	0.155	0.197	24.761	1.941
PromptIR [37]	✓	16.513	0.511	0.515	0.262	0.343	44.741	2.536	16.692	0.492	0.594	0.168	0.212	24.830	1.987
InstructIR [6]	✓	15.912	0.414	0.664	0.192	0.343	34.712	2.331	15.702	0.488	0.632	0.178	0.192	25.373	2.008
MiOIR(R) [17]	✓	16.575	0.514	0.512	0.215	0.394	41.312	2.608	15.608	0.487	0.633	0.178	0.225	26.632	2.030
MiOIR(U) [17]	✓	16.599	0.529	0.505	0.237	0.376	43.873	2.586	15.596	0.488	0.643	0.174	0.254	26.902	2.022
DA-CLIP [29]	✓	16.388	0.470	0.562	0.233	0.342	41.088	2.488	15.491	0.482	0.626	0.181	0.236	26.862	2.048
AutoDIR [15]	✓	16.870	0.540	0.451	0.263	0.396	49.707	2.867	15.651	0.464	0.582	0.203	0.231	32.172	2.199
AgenticIR [74]	✓	18.600	0.601	0.465	0.235	0.337	47.811	2.836	16.879	0.498	0.573	0.165	0.259	37.100	2.207
Ours	✗	17.650	0.563	0.475	0.295	0.473	57.756	3.247	16.180	0.473	0.564	0.233	0.389	49.810	2.864

image, and then evaluates the output to provide feedback signal for reward computation.

Action Evaluation. Existing restoration agents [4, 14, 23, 73, 74] adopt Llava-Llama3 [48], DepictQA [64], Coinstruct [60], and Qwen2-VL [55] to recognize degradation types and assess restoration quality. However, these models requires extensive labeled data for supervised training. In our label-free environment, neither clean images nor ground-truth optimal tool sequences are available, which renders supervised quality prediction infeasible. To overcome this challenge, we leverage the perceptual capability of MLLMs [8, 26, 64, 65] as a source of reward feedback. A key difficulty is that most MLLMs generate discrete textual tokens, which do not align with the continuous nature of restoration quality and thus are unsuitable for reward computation. Rather than forcing MLLMs to output discretized scores, we seek a model capable of producing a continuous quality distribution that can serve as numerical feedback for policy optimization. To achieve this, we adopt the recently proposed DeQA-Score [65] as the evaluator, which produces distributional quality estimates compatible with reward shaping. During training, the reward at step t is defined as the improvement in DeQA-Score:

$$R(s_t, f_t) = \mathrm{DS}(s_{t+1}) - \mathrm{DS}(s_t), \tag{9}$$

where $\mathrm{DS}(\cdot)$ denotes the DeQA-Score evaluator. Positive values indicate that the selected tool improves perceptual quality, while negative values penalize harmful or unnecessary actions. This formulation converts MLLM perception into a practical and stable reward signal, enabling fully label-free policy learning.

Action Execution. We consider seven degradation types commonly found in real-world CIR, including haze, defocus blur, motion blur, rain, noise, dark, and JPEG compression artifact. For these degradations, we construct a tool set for the agent to use, which consist of multiple specialized models for each type of degradation. Following [14, 74], we collect three to six publicly available restoration models for each degradation. For example, we take MAXIM [49], X-Restormer [5]; RIDCP [61], DehazeFormer [47] as the available tools for Dehazing. A detailed description of the tool set can be seen in the Supplementary Material. This tool set forms the full discrete action space for the policy.

4. Experiments

4.1. Experimental Settings

Data Construction. To train a restoration agent capable of handling diverse degradations, we construct a dataset that includes mixed-degradation images with varying complexity. Specifically, we design 15 degradation combinations, grouped into four settings, including Setting I (5 combi-

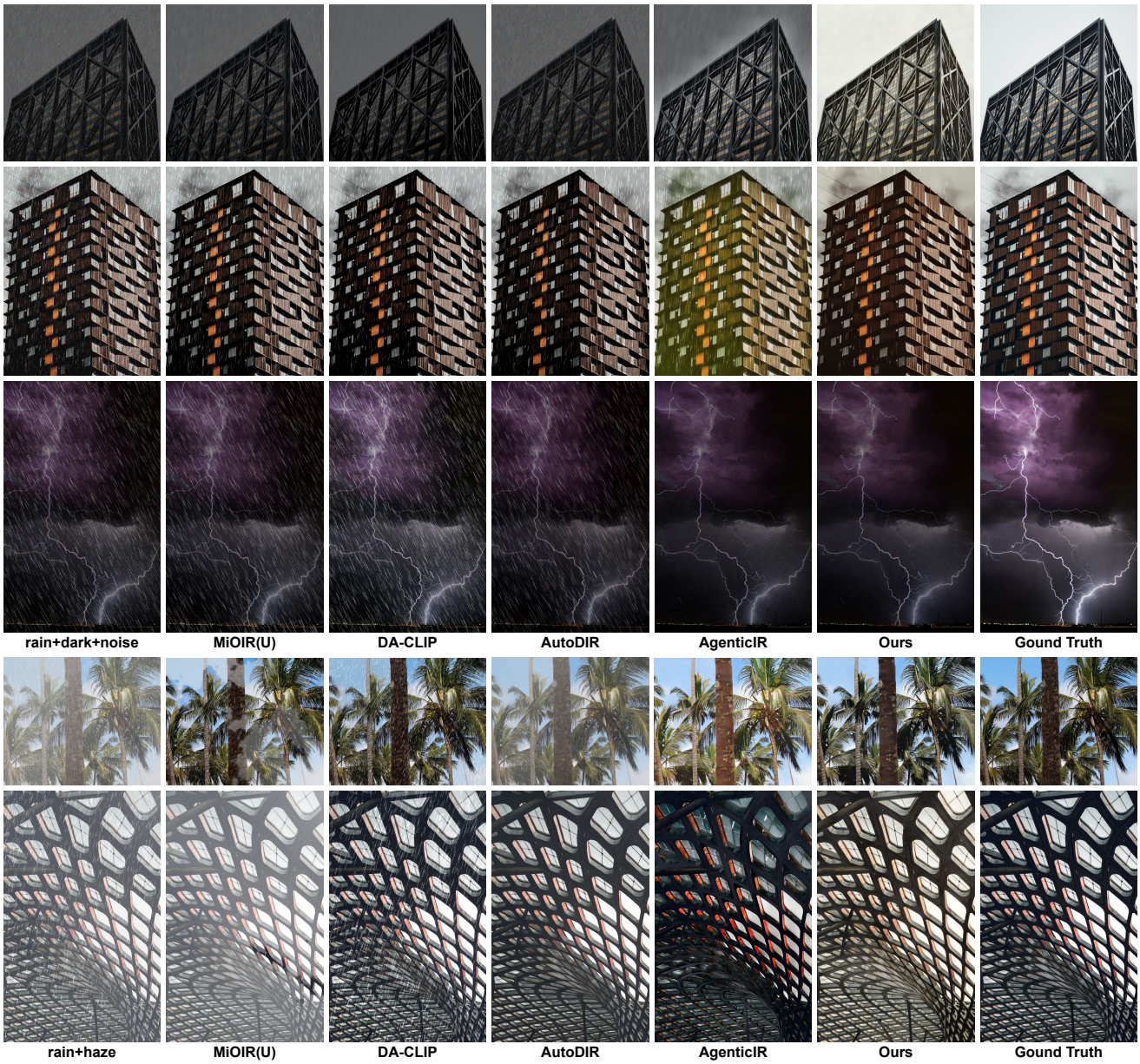


Figure 3. Qualitative comparison between our method and SOTA restoration baselines (for other baselines see the supplementary material).

nations of 2 degradations), Setting II (3 combinations of 2 degradations), Setting III (4 combinations of 3 degradations), and Setting IV (3 combinations of 4 or 5 degradations). Details of all combinations are provided in the supplementary material. Following previous agentic studies [14, 74], we use clean images from the MiO100 dataset [17, 18] to generate degraded images. For training, we only use images from Setting I, sampling 20 images per combination to construct the training set, while all remaining images, including all images from Setting II-IV, are reserved for testing. This ensures that out agent is evaluated on unseen degradation mixtures, enabling a rigorous assessment of its generalization capability.

Baselines. We compare our method against both all-in-one and agentic approaches. All-in-one methods include AirNet [22], PromptIR [37], InstructIR [6], MiOIR(R) [17] (with Restormer [69]), MiOIR(U) [17] (with Uformer [57]), DA-CLIP [29], and AutoDIR [15]. For agentic baselines, we consider the SOTA method AgenticIR [74]. Other agentic approaches [4, 14, 23, 73] have not released code or checkpoints, but from our analysis in Sec. 2.1 and summarization in Table 1, they share similar workflow and are expected to perform comparably to AgenticIR. Notably, all these methods require ground-truths for model training.

Table 3. Performance comparison across different reward strategies. "Ours (N_i)" denotes the use of the i-th no-reference quality assessment model for reward computation. Results are reported on Full-reference metrics: F1 (PSNR), F2 (SSIM), F3 (LPIPS), and No-reference metrics: N1 (MANIQA), N2 (CLIP-IQA), N3 (MUSIQ), and N4 (DeQA-Score).

	Rewards	Full-Reference			No-Reference			
		F1 ↑	F2 ↑	F3 ↓	N1 ↑	N2 ↑	N3 ↑	N4 ↑
Setting I	Ours (N1)	18.955	0.602	0.448	0.383	0.476	61.652	3.316
	Ours (N2)	18.340	0.610	0.411	0.332	0.523	60.129	3.230
	Ours (N3)	17.970	0.602	0.415	0.370	0.503	62.754	3.380
	Ours (N4)	19.513	0.647	0.365	0.349	0.525	63.003	3.657
Setting II	Ours (N1)	17.364	0.578	0.456	0.371	0.464	61.578	3.234
	Ours (N2)	16.991	0.580	0.432	0.332	0.536	62.300	3.252
	Ours (N3)	16.762	0.584	0.422	0.331	0.466	60.711	3.220
	Ours (N4)	17.958	0.636	0.385	0.343	0.530	63.344	3.599
Setting III	Ours (N1)	17.418	0.538	0.517	0.343	0.444	58.988	3.048
	Ours (N2)	16.571	0.462	0.549	0.328	0.570	57.727	2.824
	Ours (N3)	16.989	0.515	0.526	0.311	0.460	55.076	2.857
	Ours (N4)	17.650	0.563	0.475	0.295	0.473	57.756	3.247
Setting IV	Ours (N1)	15.813	0.475	0.599	0.271	0.342	49.895	2.592
	Ours (N2)	15.091	0.445	0.563	0.213	0.371	48.036	2.471
	Ours (N3)	15.449	0.455	0.570	0.227	0.261	44.688	2.279
	Ours (N4)	16.180	0.473	0.564	0.233	0.389	49.810	2.864

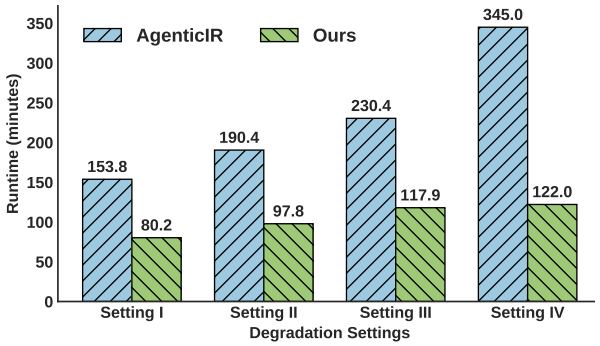


Figure 4. Runtime comparison between ours and AgenticIR [74].

Figure 5. Illustration of tool effects. Left: images with dark degradation (tiger: motion blur+dark, panda: dark+noise; skyscraper: rain+dark). Right: outputs from the dehazing model RIDCP [61].

Implementation Details. For model construction, the actor and critic share a frozen CLIP (ViT-L/14) vision encoder [39], while each uses its own two-layer MLP with LayerNorm and ReLU. The actor predicts a probability distribution across the set of available actions, whereas the critic produces a scalar value estimate. Both networks use a hidden dimension of 128. The state representation is formed by concatenating visual features with a binary action-record vector derived from the previous step's action distribution. For model training, we set the learning rate, critic coefficient, entropy coefficient, entropy decay factor, discount factor, GAE parameter, and clipping factor to 0.01, 0.5, 0.05, 0.99, 0.99, 0.95, and 0.2, respectively. The complete restoration tool set is described in supplementary material.

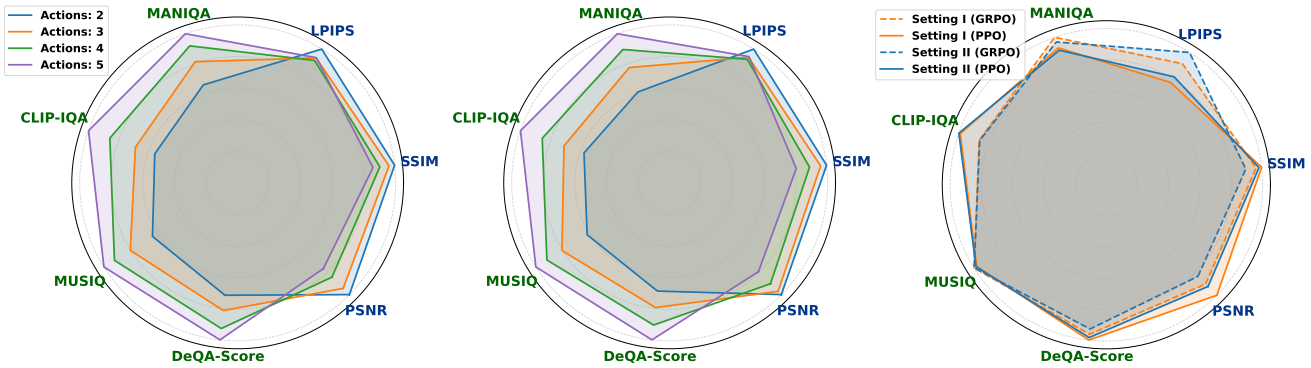
Evaluation Metrics. We evaluate models using seven quality metrics, including three full-reference metrics: PSNR, SSIM [56], LPIPS [72], and four no-reference metrics: MANIQA [62], CLIP-IQA [54], MUSIQ [16], and DeQA-Score [65]. Detailed definitions and properties of these metrics are provided in the supplementary material.

4.2. Comparison with SOTA

Quantitative Comparison. Table 2 presents the performance comparison between our method and compared baselines across all degradation settings. Despite not relying no any ground-truth supervision, our method achieves highly competitive performance on full-reference metrics across all degradation settings. This demonstrates the ability of our agent to perform faithful structural restoration guided only by perception feedback. More notably, as degradation complexity increases (Settings III and IV), our method demonstrates superior robustness and consistently surpasses all baselines on no-reference metrics, achieving the highest perceptual quality. This suggests that our agent effectively learns restoration strategies that generalize well to unseen mixtures and emphasize perceptual coherence.

Qualitative Comparison. Figure 3 presents visual comparisons under rain+haze and rain+dark+noise degradations. The comparison with other baselines are provided in the supplementary. Despite the absence of degradation annotations, our method learns to remove multiple corruptions from degraded images and achieves comparable or superior visual quality than the supervised baselines. This shows the effectiveness of our proposed MLLM perceptual feedback.

Efficiency Comparison. Figure 4 presents the runtime comparison between our method and AgenticIR [74] under all settings. As seen, AgenticIR's inference time grows rapidly from Setting I (2 degradations) to Setting IV (4 or 5 degradations), This trend is expected because AgenticIR conducts tool searching for each degradation type, causing runtime to scale with the number of degradations present in the input. Moreover, its reliance on reflection and rollback introduces additional tool invocations, further compounding



(a) 3 degradations (b) 5 degradations (c) GRPO vs. PPO

Figure 6. Illustration of the distortion-perception tradeoff for (a) 3 degradations and (b) 5 degradations. As the number of actions increases, the circles in the radar plots gradually shift leftward. (c) Performance comparison between PPO and GRPO on two different settings.

the overall computational overhead. In contrast, our inference time remains almost constant across all settings. The efficiency originates from our one-pass design: the agent outputs a complete tool-calling sequence in a single policy execution, without any iterative search or backtracking.

4.3. Ablation Studies

Selection on Reward Functions. Different restoration tools contribute unevenly depending on the evaluation perspective, which directly influences reward computation and policy learning. To investigate this effect, we present ablation results under various reward configurations in Table 3. While MANIQA and CLIP-IQA rewards offer strong learning signals and yield competitive results, they show preference toward specific scenarios and lead to performance imbalance. For instance, MANIQA performs well on Settings III-IV but poorly on Settings I-II. In contrast, DeQA-Score rewards consistently yield the best or near-best results under full-reference and no-reference metrics.

Effect of Tools. Prior agents [4, 14, 74] rely heavily on degradation recognition and strictly map each degradation type to a specific group of tools. However, they largely ignore the effects of tool interactions and degradation stacking. In practice, tools designed for a particular degradation type may also produce unexpected enhancement effects on other corruption types. As shown in Figure 5, the dehazing model RIDCP [61], although originally designed for haze removal, significantly improves visibility and brightness in low-light images. This observation indicates that the functional scope of a tool may extend beyond its intended degradation domain, due to inherent model capacity, interactions of tools, or degradation stacking effects. Therefore, the optimal tool sequence might not be inferred from a simple type-to-tool mapping. This motivates the need for our policy-based approach that learns tool sequencing holistically rather than relying on predefined associations.

Perception-Distortion Tradeoff. Figures 6a and 6b show that as the number of execution steps increases, the no-reference metrics continue to improve, whereas the full-reference metrics gradually decline (additional results are provided in the supplementary material). The rise in no-reference scores is expected because our agent is optimized to maximize the accumulated final-stage rewards, which is tied to perceptual quality. The concurrent drop in full-reference metrics aligns with observations in prior work [2], which established the theory, termed the perception-distortion tradeoff. This theory demonstrates that no image restoration algorithm can simultaneously achieve both minimal distortion and the maximal perceptual quality; improving one inevitably degrades the other due to the intrinsic ambiguity of the inverse imaging process.

GRPO Optimization. Our framework is compatible with various policy optimization methods. To demonstrate this flexibility, we additionally train the model using GRPO [44]. Figure 6c compares the results of GRPO and PPO, showing that both training strategies lead to similar performance. This confirms that our framework generalizes well across different policy optimization algorithms. Due to space limitation, we provide additional empirical results in the supplementary material.

5. Conclusion

In this work, we propose a policy optimization-based framework for complex image restoration, in which a lightweight agent learns to select restoration tools and determine their execution order through interaction with multimodal perceptual feedback. Compared with existing restoration approaches, our method eliminates the reliance on reflection, rollback, and tool searching, improving inference efficiency while maintaining high restoration quality. We believe this work represents a promising step toward autonomous, perception-driven restoration systems that can adapt to general conditions without explicit supervision.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 2

[2] 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

[3] Miaomiao Cai, Simiao Li, Wei Li, Xudong Huang, Hanting Chen, Jie Hu, and Yunhe Wang. Dspo: Direct semantic preference optimization for real-world image super-resolution. *arXiv preprint arXiv:2504.15176*, 2025. 3

[4] Haoyu Chen, Wenbo Li, Jinjin Gu, Jingjing Ren, Sixiang Chen, Tian Ye, Renjing Pei, Kaiwen Zhou, Fenglong Song, and Lei Zhu. Restoreagent: Autonomous image restoration agent via multimodal large language models. In *Advances in Neural Information Processing Systems*, pages 110643–110666. Curran Associates, Inc., 2024. 1, 2, 3, 4, 5, 6, 8

[5] Xiangyu Chen, Zheyuan Li, Yuandong Pu, Yihao Liu, Jiantao Zhou, Yu Qiao, and Chao Dong. A comparative study of image restoration networks for general backbone network design. In *European Conference on Computer Vision*, pages 74–91. Springer, 2024. 5, 1

[6] Marcos V Conde, Gregor Geigle, and Radu Timofte. Instructir: High-quality image restoration following human instructions. In *European Conference on Computer Vision*, pages 1–21. Springer, 2024. 2, 3, 5, 6, 1, 4

[7] Qihua Dong, Luis Figueroa, Handong Zhao, Kushal Kafle, Jason Kuen, Zhihong Ding, Scott Cohen, and Yun Fu. Cot referring: Improving referring expression tasks with grounded reasoning. *arXiv preprint arXiv:2510.06243*, 2025. 2

[8] Qihua Dong, Kuo Yang, Lin Ju, Handong Zhao, Yitian Zhang, Yizhou Wang, Huimin Zeng, Jianglin Lu, and Yun Fu. Ref-adv: Exploring MLLM visual reasoning in referring expression tasks. In *The Fourteenth International Conference on Learning Representations*, 2026. 2, 5

[9] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407, 2024. 2

[10] Yuanbin Fu, Jie Ying, Houlei Lv, and Xiaojie Guo. Semi-supervised camouflaged object detection from noisy data. In *Proceedings of the 32nd ACM International Conference on Multimedia*, page 4766–4775, 2024. 1

[11] Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *IEEE transactions on pattern analysis and machine intelligence*, 33(12):2341–2353, 2010. 1

[12] Ruozhen He, Qihua Dong, Jiaying Lin, and Rynson WH Lau. Weakly-supervised camouflaged object detection with scribble annotations. In *Proceedings of the AAAI conference on artificial intelligence*, pages 781–789, 2023. 1

[13] Jiaxi Jiang, Kai Zhang, and Radu Timofte. Towards flexible blind jpeg artifacts removal. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4997–5006, 2021. 1

[14] Xu Jiang, Gehui Li, Bin Chen, and Jian Zhang. Multi-agent image restoration. *arXiv preprint arXiv:2503.09403*, 2025. 1, 2, 3, 4, 5, 6, 8

[15] Yitong Jiang, Zhaoyang Zhang, Tianfan Xue, and Jinwei Gu. Autodir: Automatic all-in-one image restoration with latent diffusion. In *European Conference on Computer Vision*, pages 340–359. Springer, 2024. 1, 2, 3, 5, 6, 4

[16] 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. 7, 1

[17] Xiangtao Kong, Chao Dong, and Lei Zhang. Towards effective multiple-in-one image restoration: A sequential and prompt learning strategy. *arXiv preprint arXiv:2401.03379*, 2024. 1, 2, 5, 6, 4

[18] Xiangtao Kong, Jinjin Gu, Yihao Liu, Wenlong Zhang, Xiangyu Chen, Yu Qiao, and Chao Dong. A preliminary exploration towards general image restoration. *arXiv preprint arXiv:2408.15143*, 2024. 6

[19] Zhenglun Kong, Yize Li, Fanhu Zeng, Lei Xin, Shvat Messica, Xue Lin, Pu Zhao, Manolis Kellis, Hao Tang, and Marinka Zitnik. Token reduction should go beyond efficiency in generative models–from vision, language to multimodality. *arXiv preprint arXiv:2505.18227*, 2025. 2

[20] Junyong Lee, Hyeongseok Son, Jaesung Rim, Sunghyun Cho, and Seungyong Lee. Iterative filter adaptive network for single image defocus deblurring. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2034–2042, 2021. 1

[21] Boyi Li, Wenqi Ren, Dengpan Fu, Dacheng Tao, Dan Feng, Wenjun Zeng, and Zhangyang Wang. Benchmarking single-image dehazing and beyond. *IEEE transactions on image processing*, 28(1):492–505, 2018. 1

[22] Boyun Li, Xiao Liu, Peng Hu, Zhongqin Wu, Jiancheng Lv, and Xi Peng. All-in-one image restoration for unknown corruption. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17452–17462, 2022. 1, 2, 3, 5, 6, 4

[23] Bingchen Li, Xin Li, Yiting Lu, and Zhibo Chen. Hybrid agents for image restoration. *arXiv preprint arXiv:2503.10120*, 2025. 1, 2, 3, 4, 5, 6

[24] Mingjia Li, Yuanbin Fu, Xinhui Li, and Xiaojie Guo. Deep flexible structure preserving image smoothing. In *Proceedings of the 30th ACM International Conference on Multimedia*, page 1875–1883, 2022. 1

[25] 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*, pages 1833–1844, 2021. 1

[26] Jianglin Lu, Hailing Wang, Yi Xu, Yizhou Wang, Kuo Yang, and Yun Fu. Representation potentials of foundation models for multimodal alignment: A survey. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 16669–16684. Association for Computational Linguistics, 2025. 2, 5

[27] Jianglin Lu, Hailing Wang, Kuo Yang, Yitian Zhang, Simon Jenni, and Yun Fu. The indra representation hypothesis for multimodal alignment. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. 2

[28] Jianglin Lu, Simon Jenni, Kushal Kafle, Jing Shi, Handong Zhao, and Yun Fu. Seeing through words: Controlling visual retrieval quality with language models. In *The Fourteenth International Conference on Learning Representations*, 2026. 1, 2

[29] Ziwei Luo, Fredrik K Gustafsson, Zheng Zhao, Jens Sjölund, and Thomas B Schön. Controlling vision-language models for multi-task image restoration. In *ICLR*, 2024. 2, 3, 5, 6, 4

[30] Claudio Michaelis, Benjamin Mitzkus, Robert Geirhos, Evgenia Rusak, Oliver Bringmann, Alexander S Ecker, Matthias Bethge, and Wieland Brendel. Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484*, 2019. 1

[31] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013. 3

[32] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[33] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022. 3

[34] Dongwon Park, Byung Hyun Lee, and Se Young Chun. All-in-one image restoration for unknown degradations using adaptive discriminative filters for specific degradations. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5815–5824. IEEE, 2023. 1, 2

[35] Jongchan Park, Joon-Young Lee, Donggeun Yoo, and In So Kweon. Distort-and-recover: Color enhancement using deep reinforcement learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5928–5936, 2018. 3

[36] Yohan Poirier-Ginter and Jean-François Lalonde. Robust unsupervised stylegan image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22292–22301, 2023. 1

[37] Vaishnav Potlapalli, Syed Waqas Zamir, Salman H Khan, and Fahad Shahbaz Khan. Promptir: Prompting for all-in-one image restoration. *Advances in Neural Information Processing Systems*, 36:71275–71293, 2023. 2, 3, 5, 6, 1, 4

[38] Junbo Qiao, Miaomiao Cai, Wei Li, Yutong Liu, Xudong Huang, Gaoqi He, Jiao Xie, Jie Hu, Xinghao Chen, and Shaohui Lin. Realsr-r1: Reinforcement learning for real-world image super-resolution with vision-language chain-of-thought. *arXiv preprint arXiv:2506.16796*, 2025. 3

[39] 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. 7

[40] Lingyan Ruan, Bin Chen, Jizhou Li, and Miuling Lam. Learning to deblur using light field generated and real defocus images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16304–16313, 2022. 1

[41] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR, 2015. 3, 4

[42] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015. 4

[43] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 2, 3, 4

[44] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. 2, 3, 4, 8

[45] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016. 3

[46] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676): 354–359, 2017. 3

[47] Yuda Song, Zhuqing He, Hui Qian, and Xin Du. Vision transformers for single image dehazing. *IEEE Transactions on Image Processing*, 32:1927–1941, 2023. 5, 1

[48] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 2, 5

[49] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxim: Multi-axis mlp for image processing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5769–5780, 2022. 5, 1

[50] Hailing Wang, Qiaoyu Tian, Liang Li, and Xiaojie Guo. Image demoiréing with a dual-domain distilling network. In *2021 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2021. 1, 2

[51] Hailing Wang, Wei Li, Yuanyuan Xi, Jie Hu, Hanting Chen, Longyu Li, and Yunhe Wang. Ift: Image fusion transformer for ghost-free high dynamic range imaging. *arXiv preprint arXiv:2309.15019*, 2023. 1, 3

[52] Hailing Wang, Jianglin Lu, Yitian Zhang, and Yun Fu. Outlier-aware post-training quantization for image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16175–16184, 2025. 1, 2, 3

[53] Hongru Wang, Cheng Qian, Wanjun Zhong, Xiusi Chen, Jiahao Qiu, Shijue Huang, Bowen Jin, Mengdi Wang, Kam-Fai Wong, and Heng Ji. Otc: Optimal tool calls via reinforcement learning. *arXiv e-prints*, pages arXiv–2504, 2025. 3

[54] Jianyi Wang, Kelvin CK Chan, and Chen Change Loy. Exploring clip for assessing the look and feel of images. In *Proceedings of the AAAI conference on artificial intelligence*, pages 2555–2563, 2023. 7, 1

[55] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 2, 5

[56] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 7, 1

[57] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17683–17693, 2022. 1, 6

[58] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8(3):279–292, 1992. 3

[59] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992. 3

[60] Haoning Wu, Hanwei Zhu, Zicheng Zhang, Erli Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Annan Wang, Wenxiu Sun, Qiong Yan, et al. Towards open-ended visual quality comparison. In *European Conference on Computer Vision*, pages 360–377. Springer, 2024. 2, 5

[61] Rui-Qi Wu, Zheng-Peng Duan, Chun-Le Guo, Zhi Chai, and Chongyi Li. Ridcp: Revitalizing real image dehazing via high-quality codebook priors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22282–22291, 2023. 5, 7, 8, 1

[62] Sidi Yang, Tianhe Wu, Shuwei Shi, Shanshan Lao, Yuan Gong, Mingdeng Cao, Jiahao Wang, and Yujiu Yang. Maniqa: Multi-dimension attention network for no-reference image quality assessment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1191–1200, 2022. 7, 1

[63] Zhiwen Yang, Haowei Chen, Ziniu Qian, Yang Yi, Hui Zhang, Dan Zhao, Bingzheng Wei, and Yan Xu. All-in-one medical image restoration via task-adaptive routing. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 67–77. Springer, 2024. 1, 2

[64] Zhiyuan You, Zheyuan Li, Jinjin Gu, Zhenfei Yin, Tianfan Xue, and Chao Dong. Depicting beyond scores: Advancing image quality assessment through multi-modal language models. In *European Conference on Computer Vision*, pages 259–276. Springer, 2024. 2, 5

[65] 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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14483–14494, 2025. 5, 7, 1

[66] Ke Yu, Chao Dong, Liang Lin, and Chen Change Loy. Crafting a toolchain for image restoration by deep reinforcement learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2443–2452, 2018. 3

[67] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Learning enriched features for real image restoration and enhancement. In *European conference on computer vision*, pages 492–511. Springer, 2020. 1

[68] 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. 1

[69] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5728–5739, 2022. 1, 6

[70] Haichao Zhang and Yun Fu. Vqtoken: Neural discrete token representation learning for extreme token reduction in video large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. 2

[71] Haichao Zhang, Yao Lu, Lichen Wang, Yunzhe Li, Daiwei Chen, Yunpeng Xu, and Yun Fu. Linkedout: Linking world knowledge representation out of video llm for next-generation video recommendation. *arXiv preprint arXiv:2512.16891*, 2025. 2

[72] 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. 7, 1

[73] Yingjie Zhou, Jiezhang Cao, Zicheng Zhang, Farong Wen, Yanwei Jiang, Jun Jia, Xiaohong Liu, Xiongkuo Min, and Guangtao Zhai. Qagent: Quality-driven chain-of-thought image restoration agent through robust multimodal large language model. *arXiv preprint arXiv:2504.07148*, 2025. 2, 3, 4, 5, 6

[74] Kaiwen Zhu, Jinjin Gu, Zhiyuan You, Yu Qiao, and Chao Dong. An intelligent agentic system for complex image restoration problems. In *The Thirteenth International Conference on Learning Representations*, 2025. 1, 2, 3, 4, 5, 6, 7, 8

[75] Karel Zuiderveld. *Contrast limited adaptive histogram equalization*, page 474–485. Academic Press Professional, Inc., USA, 1994. 1

Restore-R1: Efficient Image Restoration Agents via Reinforcement Learning with Multimodal LLM Perceptual Feedback

Supplementary Material

6. Experimental Details

6.1. Data

In this section, we show how to synthesize degraded images following existing work [74]. For dark images, the V channel value of the images in the HSV color space will be randomly decreased by one of the following strategies: linear mapping, Gamma correction, and subtracting a constant. For defocus blur, the images will be filtered with circular kernels with random radius as in [30]. For JPEG compression artifact, images are compressed with random quality factor, such as 5, 40, and 90. For noise, images are added with Poisson or Gaussian noise with random scale. For rain, images are first added with noise and filtered the noise with linear kernels with random directions as in [17]. For motion blur, images are filtered with linear kernels with random direction and radius as in [30]. For haze generation, we simulate degraded images using the atmospheric scattering model, where the global atmospheric light and scattering coefficient are randomly sampled following the settings in prior work [11, 21]. Following [74], we consider degradation combinations that are common in real-world scenarios, e.g., dark+noise, rain+haze, rain+dark, rain+haze+defocus blur + JPEG compression artifact. Table 4 shows the 15 degradation combinations used in our experiments.

6.2. Tool Set

For each degradation, we follow previous restoration agents [14, 74] and consider several open-sourced restoration models as the callable tools. These candidate tools are:

- *Brightening*: CLAHE [75], Gamma correction ($\gamma = 2/3$), constant shift (adding a constant 40).
- *Defocus deblurring*: DRBNet [40], IFAN [20], Restormer [69].
- *JPEG compression artifact removal*: SwinIR [25] (quality factor 40), FBCNN [13] (quality factor 90), FBCNN [13] (quality factor 5), FBCNN [13] (blind to quality factor).
- *Denoising*: SwinIR [25] (noise level 15), SwinIR [25] (noise level 50), MAXIM [49], MPRNet [68], Restormer [69], X-Restormer [5].
- *Deraining*: MAXIM [49], MPRNet [68], Restormer [69], X-Restormer [5].
- *Motion deblurring*: MAXIM [49], MPRNet [68], Restormer [69], X-Restormer [5].
- *Dehazing*: MAXIM [49], X-Restormer [5]; RIDCP [61], DehazeFormer [47].

6.3. Evaluation Metrics

We assess model performance using three full-reference metrics: PSNR, SSIM [56], LPIPS [72], and four no-reference metrics: MANIQA [62], CLIP-IQA [54], MUSIQ [16], and DeQA-Score [65]. We briefly introduce these metrics below:

- *PSNR*: A pixel-level metric that measures the mean squared error between the restored and reference images. Higher PSNR indicates better fidelity.
- *SSIM* [56]: Evaluates perceptual similarity by assessing luminance, contrast, and structural components between image pairs. It better aligns with human visual perception than PSNR.
- *LPIPS* [72]: A perceptual metric that uses deep neural network features to compare image similarity, capturing differences that are visually meaningful but not captured by PSNR or SSIM.
- *MANIQA* [62]: A no-reference image quality assessment model that leverages transformer-based architecture and multi-level semantic features to predict perceptual quality without needing a ground-truth image.
- *CLIP-IQA* [54]: A no-reference quality assessment model that uses CLIP embeddings to assess image quality based on its alignment with natural image statistics learned from large-scale vision-language pretraining.
- *MUSIQ* [16]: A transformer-based non-reference image quality assessment model that adapts to various resolutions and content types by processing image patches, offering strong generalization across diverse datasets.
- *DeQA-Score* [65]: The model DeQA-Score computes a continuous image-quality score by first having a MLLM process one or more input images, then outputting a soft-label distribution over discrete quality levels (rather than a simple one-hot label). It treats the human quality ratings as approximately Gaussian and trains the MLLM with a KL-divergence loss to match that soft label. At inference time, the predicted distribution over levels is combined (by a weighted sum over discrete rating tokens) to produce a final quality score that more accurately reflects continuous human judgments.

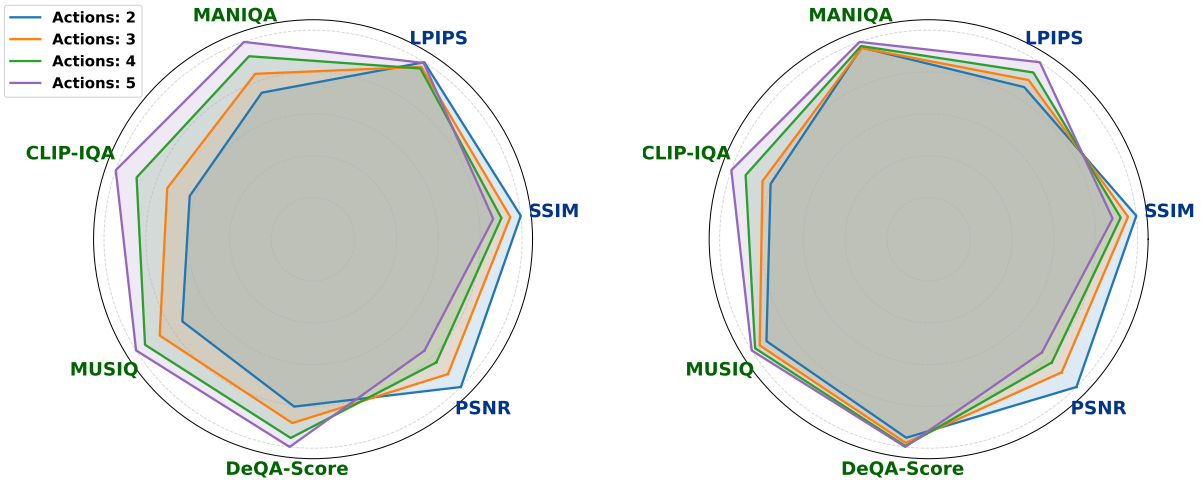
7. More Results

7.1. Qualitative Comparison

Figure 8 shows visual comparisons between our method and AirNet [22], PromptIR [37], InstructIR [6], and MiOIR(R) [17] under rain+haze and rain+dark+noise degradation

Table 4. Degradation data construction

Settings	# of Degradations	Case Number	Combinations
I	2	Case 1	dark+noise
		Case 2	defocus blur+JPEG compression artifact
		Case 3	motion blur + dark
		Case 4	noise+JPEG compression artifact
		Case 5	rain+haze
II	2	Case 6	haze+noise
		Case 7	motion blur+JPEG compression artifact
		Case 8	rain+dark
III	3	Case 9	dark+defocus blur+JPEG compression artifact
		Case 10	motion blur+defocus blur+noise
		Case 11	rain+dark+noise
		Case 12	rain+haze+noise
IV	>3	Case 13	haze+dark+motion blur+JPEG compression artifact
		Case 14	rain+haze+defocus blur+JPEG compression artifact
		Case 15	rain+motion blur+defocus blur+noise+JPEG compression artifact



(a) defocus blur+JPEG compression artifact

(b) noise+JPEG compression artifact

Figure 7. Illustration of distortion-perception tradeoff on (a) noise+jpeg compression artifact and (b) motion blur+defocus blur+noise.

cases. The results further demonstrate that our method effectively removes multiple co-occurring corruptions from degraded images and produces visual quality that is comparable to, or even exceeds, these supervised baselines.

7.2. Quantitative Comparison

Tabls 5, 6, 7 present the performance comparison between our method and the competing baselines across all degradation cases. As shown, even without access to ground-truth supervision, our method achieves competitive results on full-reference metrics and consistently outperforms all baselines on no-reference metrics, further demonstrating the effectiveness and robustness of our approach.

7.3. Perception-Distortion Tradeoff

Figures 7a and 7b illustrate the perception–distortion tradeoff under the defocus blur + JPEG compression and noise + JPEG compression degradation cases. These results lead to the same conclusion as discussed in Section 4.3.

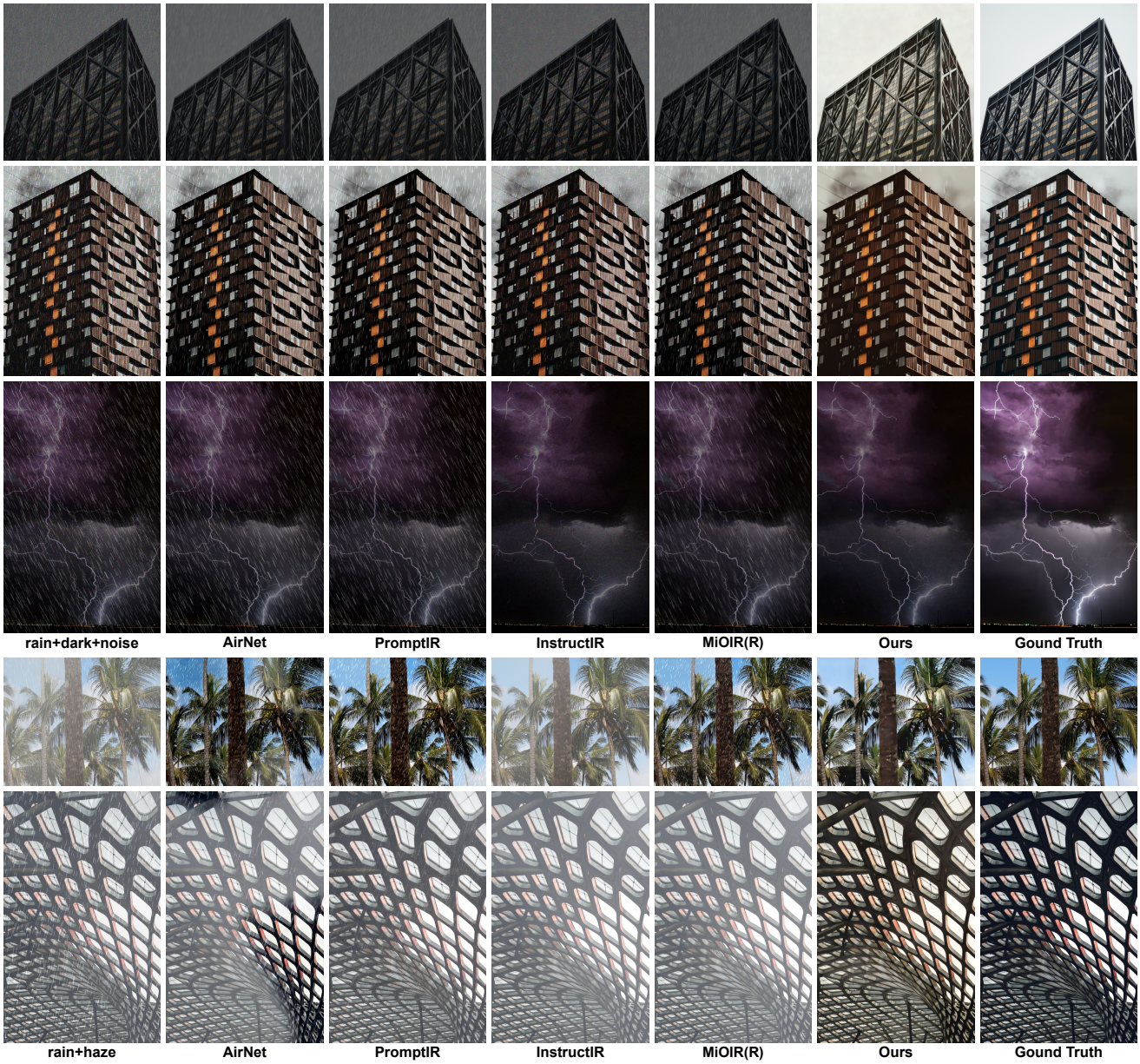


Figure 8. Qualitative comparison between our method and SOTA restoration baselines.

Table 5. Quantitative comparison across multiple mixed-degradation conditions. Arrows indicate the desired direction of improvement.

	Method	Full-Reference			No-Reference			
		PSNR ↑	SSIM ↑	LPIPS ↓	MANIQA ↑	CLIP-IQA ↑	MUSIQ ↑	DeQA-Score) ↑
Case 1	AirNet [22]	16.272	0.633	0.221	0.423	0.572	64.253	3.539
	PromptIR [37]	16.217	0.615	0.242	0.407	0.540	64.364	3.479
	InstructIR [6]	15.305	0.469	0.496	0.260	0.507	49.016	3.010
	MiOIR(R) [17]	16.563	0.632	0.233	0.312	0.612	59.093	3.561
	MiOIR(U) [17]	16.696	0.675	0.210	0.348	0.598	62.262	3.628
	DA-CLIP [29]	16.343	0.552	0.340	0.330	0.494	57.732	3.289
	AutoDIR [15]	16.771	0.658	0.257	0.362	0.590	63.672	3.625
	AgenticIR [74]	19.692	0.708	0.338	0.346	0.444	59.594	3.388
	Ours	17.995	0.664	0.362	0.399	0.631	68.240	3.871
Case 2	AirNet [22]	22.282	0.629	0.501	0.194	0.232	28.489	2.282
	PromptIR [37]	23.110	0.632	0.486	0.203	0.247	28.715	2.298
	InstructIR [6]	23.727	0.648	0.519	0.211	0.216	29.441	2.324
	MiOIR(R) [17]	23.960	0.653	0.503	0.209	0.256	30.628	2.353
	MiOIR(U) [17]	23.965	0.653	0.511	0.203	0.278	30.848	2.338
	DA-CLIP [29]	23.580	0.647	0.490	0.207	0.272	30.073	2.354
	AutoDIR [15]	23.915	0.654	0.439	0.215	0.269	36.215	2.571
	AgenticIR [74]	22.755	0.658	0.435	0.216	0.303	43.866	2.688
	Ours	22.338	0.634	0.456	0.225	0.302	44.950	2.832
Case 3	AirNet [22]	14.083	0.450	0.411	0.175	0.281	42.572	2.429
	PromptIR [37]	14.061	0.433	0.406	0.183	0.290	42.710	2.434
	InstructIR [6]	14.489	0.461	0.413	0.182	0.278	42.521	2.481
	MiOIR(R) [17]	17.294	0.561	0.371	0.176	0.281	43.473	2.497
	MiOIR(U) [17]	16.989	0.574	0.388	0.170	0.276	43.659	2.471
	DA-CLIP [29]	15.520	0.517	0.356	0.201	0.299	48.516	2.728
	AutoDIR [15]	15.984	0.567	0.319	0.254	0.364	56.248	3.340
	AgenticIR [74]	17.306	0.519	0.314	0.269	0.406	59.304	3.325
	Ours	15.691	0.482	0.393	0.345	0.582	66.178	3.797
Case 4	AirNet [22]	23.522	0.613	0.458	0.287	0.379	44.700	2.958
	PromptIR [37]	24.249	0.620	0.448	0.293	0.408	45.182	2.982
	InstructIR [6]	24.238	0.626	0.439	0.294	0.413	45.629	3.028
	MiOIR(R) [17]	25.453	0.671	0.381	0.320	0.464	52.704	3.137
	MiOIR(U) [17]	25.628	0.673	0.373	0.324	0.486	53.445	3.086
	DA-CLIP [29]	24.592	0.632	0.403	0.316	0.506	49.354	3.123
	AutoDIR [15]	24.221	0.622	0.436	0.295	0.437	45.982	3.030
	AgenticIR [74]	25.489	0.800	0.258	0.344	0.462	61.722	3.720
	Ours	24.834	0.770	0.281	0.374	0.487	63.852	3.774
Case 5	AirNet [22]	16.288	0.716	0.279	0.396	0.531	64.235	3.349
	PromptIR [37]	18.351	0.745	0.275	0.414	0.515	64.975	3.341
	InstructIR [6]	14.213	0.727	0.215	0.392	0.538	63.626	3.377
	MiOIR(R) [17]	16.578	0.776	0.197	0.395	0.539	65.445	3.444
	MiOIR(U) [17]	16.322	0.784	0.177	0.401	0.541	65.869	3.533
	DA-CLIP [29]	16.225	0.722	0.262	0.408	0.515	65.923	3.374
	AutoDIR [15]	16.136	0.721	0.280	0.402	0.497	65.782	3.365
	AgenticIR [74]	19.373	0.805	0.184	0.400	0.592	68.463	3.994
	Ours	16.705	0.687	0.333	0.403	0.624	71.796	4.010

Table 6. Quantitative comparison across multiple mixed-degradation conditions. Arrows indicate the desired direction of improvement.

	Method	Full-Reference			No-Reference			
		PSNR ↑	SSIM ↑	LPIPS ↓	MANIQA ↑	CLIP-IQA ↑	MUSIQ ↑	DeQA-Score) ↑
Case 6	AirNet [22]	14.781	0.657	0.313	0.373	0.492	63.081	3.293
	PromptIR [37]	14.783	0.661	0.310	0.355	0.467	62.443	3.320
	InstructIR [6]	14.456	0.446	0.646	0.248	0.469	43.749	2.770
	MiOIR(R) [17]	14.672	0.632	0.320	0.267	0.562	55.223	3.390
	MiOIR(U) [17]	14.830	0.665	0.285	0.302	0.487	60.079	3.361
	DA-CLIP [29]	14.503	0.602	0.397	0.313	0.399	56.516	3.134
	AutoDIR [15]	17.249	0.699	0.268	0.322	0.518	62.531	3.567
	AgenticIR [74]	18.725	0.699	0.349	0.315	0.433	59.596	3.580
	Ours	15.999	0.623	0.389	0.393	0.623	68.938	3.869
Case 7	AirNet [22]	20.800	0.623	0.401	0.198	0.249	34.574	2.591
	PromptIR [37]	21.494	0.632	0.385	0.207	0.271	34.707	2.630
	InstructIR [6]	21.738	0.640	0.400	0.210	0.255	36.812	2.665
	MiOIR(R) [17]	21.941	0.648	0.389	0.193	0.302	39.335	2.640
	MiOIR(U) [17]	21.949	0.648	0.393	0.186	0.326	39.704	2.619
	DA-CLIP [29]	21.664	0.640	0.384	0.193	0.295	38.169	2.641
	AutoDIR [15]	21.679	0.634	0.374	0.222	0.296	38.146	2.822
	AgenticIR [74]	20.397	0.625	0.383	0.212	0.349	48.192	2.922
	Ours	20.370	0.603	0.411	0.245	0.350	50.361	3.061
Case 8	AirNet [22]	16.006	0.647	0.263	0.399	0.576	63.726	3.409
	PromptIR [37]	16.178	0.687	0.207	0.411	0.585	64.320	3.512
	InstructIR [6]	16.329	0.692	0.190	0.400	0.574	63.881	3.548
	MiOIR(R) [17]	17.522	0.732	0.202	0.404	0.573	65.350	3.489
	MiOIR(U) [17]	16.580	0.715	0.156	0.409	0.585	65.647	3.592
	DA-CLIP [29]	16.917	0.669	0.266	0.419	0.548	66.502	3.438
	AutoDIR [15]	16.691	0.660	0.314	0.410	0.530	66.372	3.382
	AgenticIR [74]	22.132	0.832	0.169	0.402	0.561	68.141	3.800
	Ours	17.506	0.682	0.356	0.390	0.618	70.732	3.866
Case 9	AirNet [22]	14.814	0.458	0.544	0.189	0.192	28.382	2.131
	PromptIR [37]	15.014	0.446	0.534	0.196	0.202	28.532	2.128
	InstructIR [6]	15.434	0.483	0.550	0.203	0.182	29.241	2.165
	MiOIR(R) [17]	15.491	0.490	0.550	0.206	0.233	30.287	2.197
	MiOIR(U) [17]	15.483	0.489	0.555	0.201	0.260	30.538	2.184
	DA-CLIP [29]	15.778	0.496	0.543	0.206	0.244	30.166	2.224
	AutoDIR [15]	15.973	0.514	0.488	0.196	0.219	34.427	2.338
	AgenticIR [74]	18.798	0.584	0.495	0.196	0.262	41.299	2.504
	Ours	16.287	0.535	0.482	0.304	0.491	60.227	3.438
Case 10	AirNet [22]	21.250	0.559	0.536	0.196	0.277	29.854	2.096
	PromptIR [37]	21.269	0.562	0.562	0.196	0.254	28.904	2.086
	InstructIR [6]	19.104	0.347	0.780	0.103	0.305	22.685	1.882
	MiOIR(R) [17]	20.923	0.530	0.569	0.116	0.345	25.380	2.034
	MiOIR(U) [17]	21.225	0.566	0.526	0.153	0.278	26.593	2.090
	DA-CLIP [29]	20.474	0.486	0.624	0.163	0.295	25.290	2.019
	AutoDIR [15]	18.855	0.479	0.520	0.241	0.335	41.623	2.528
	AgenticIR [74]	20.034	0.546	0.525	0.195	0.292	38.196	2.401
	Ours	20.202	0.530	0.531	0.186	0.288	37.379	2.368

Table 7. Quantitative comparison across multiple mixed-degradation conditions. Arrows indicate the desired direction of improvement.

	Method	Full-Reference			No-Reference			
		PSNR ↑	SSIM ↑	LPIPS ↓	MANIQA ↑	CLIP-IQA ↑	MUSIQ ↑	DeQA-Score) ↑
Case 11	AirNet [22]	16.650	0.521	0.452	0.371	0.499	62.832	3.060
	PromptIR [37]	16.601	0.511	0.460	0.354	0.493	62.143	3.065
	InstructIR [6]	15.790	0.455	0.560	0.236	0.455	47.557	2.826
	MiOIR(R) [17]	16.734	0.529	0.438	0.285	0.525	56.743	3.200
	MiOIR(U) [17]	16.487	0.532	0.451	0.319	0.537	60.476	3.142
	DA-CLIP [29]	16.241	0.443	0.514	0.296	0.443	55.044	2.971
	AutoDIR [15]	16.776	0.575	0.387	0.324	0.554	62.473	3.373
	AgenticIR [74]	18.674	0.661	0.404	0.287	0.394	57.342	3.254
	Ours	17.464	0.596	0.440	0.343	0.569	66.904	3.582
Case 12	AirNet [22]	13.162	0.528	0.483	0.327	0.429	61.533	2.871
	PromptIR [37]	13.167	0.523	0.502	0.300	0.422	59.385	2.866
	InstructIR [6]	13.320	0.370	0.767	0.227	0.428	39.365	2.451
	MiOIR(R) [17]	13.151	0.508	0.490	0.252	0.474	52.839	3.000
	MiOIR(U) [17]	13.202	0.527	0.489	0.273	0.430	57.886	2.927
	DA-CLIP [29]	13.058	0.453	0.567	0.265	0.386	53.851	2.739
	AutoDIR [15]	15.875	0.592	0.410	0.292	0.474	60.303	3.228
	AgenticIR [74]	16.895	0.612	0.435	0.263	0.399	54.407	3.184
	Ours	16.648	0.591	0.447	0.345	0.542	66.515	3.600
Case 13	AirNet [22]	14.032	0.467	0.542	0.174	0.183	30.106	2.185
	PromptIR [37]	15.082	0.506	0.521	0.191	0.194	30.544	2.237
	InstructIR [6]	14.738	0.508	0.548	0.199	0.177	32.111	2.264
	MiOIR(R) [17]	14.787	0.513	0.544	0.187	0.245	34.559	2.285
	MiOIR(U) [17]	14.769	0.513	0.551	0.178	0.272	35.271	2.263
	DA-CLIP [29]	14.590	0.505	0.539	0.192	0.231	33.721	2.322
	AutoDIR [15]	14.821	0.509	0.525	0.207	0.216	33.228	2.372
	AgenticIR [74]	15.514	0.520	0.513	0.181	0.277	41.847	2.405
	Ours	14.671	0.476	0.517	0.276	0.448	56.298	3.168
Case 14	AirNet [22]	15.499	0.465	0.635	0.152	0.195	23.437	1.842
	PromptIR [37]	15.397	0.475	0.634	0.171	0.207	23.194	1.924
	InstructIR [6]	12.947	0.452	0.682	0.186	0.195	23.107	1.937
	MiOIR(R) [17]	12.791	0.449	0.693	0.188	0.213	23.952	1.967
	MiOIR(U) [17]	12.775	0.450	0.704	0.185	0.241	24.009	1.967
	DA-CLIP [29]	12.772	0.442	0.689	0.194	0.239	24.610	1.967
	AutoDIR [15]	13.324	0.428	0.645	0.205	0.229	30.584	2.069
	AgenticIR [74]	16.214	0.491	0.602	0.159	0.250	35.394	2.175
	Ours	15.120	0.447	0.570	0.266	0.472	57.365	3.217
Case 15	AirNet [22]	18.854	0.484	0.640	0.138	0.213	20.739	1.797
	PromptIR [37]	19.598	0.495	0.627	0.141	0.234	20.752	1.800
	InstructIR [6]	19.420	0.504	0.667	0.150	0.204	20.900	1.823
	MiOIR(R) [17]	19.245	0.500	0.662	0.159	0.218	21.385	1.838
	MiOIR(U) [17]	19.244	0.502	0.674	0.159	0.248	21.425	1.835
	DA-CLIP [29]	19.110	0.498	0.649	0.158	0.237	22.256	1.854
	AutoDIR [15]	18.808	0.456	0.576	0.196	0.248	32.703	2.155
	AgenticIR [74]	18.910	0.484	0.604	0.156	0.250	34.060	2.042
	Ours	18.750	0.497	0.605	0.156	0.248	35.768	2.207