HARNESSIR: HARNESSING MULTIMODAL FOUNDATION MODELS FOR UNIVERSAL REAL-WORLD IMAGE RESTORATION

Anonymous authors
Paper under double-blind review

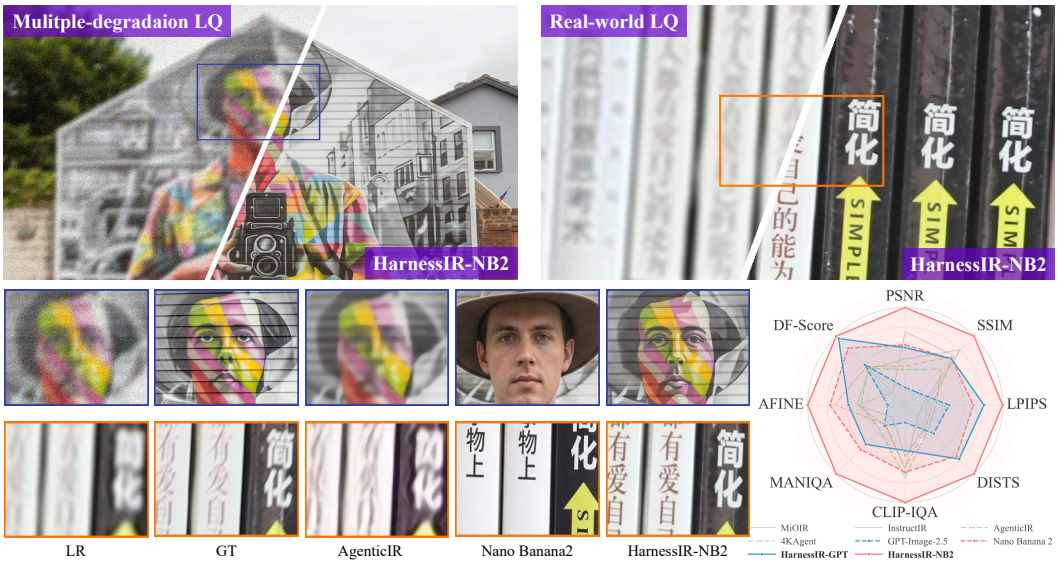

Figure 1: **Top and Bottom-left**: Visual results of HarnessIR, which can restore image details without altering their content and texts. **Bottom-right**: Radar charts comparing different metrics on real-world test sets. HarnessIR achieves much better restoration performance than Nano Banana 2, GPT-Image-2.5, previous all-in-one and agentic IR methods.

ABSTRACT

Real-world low-quality images suffer from complex mixed degradations, including but not limited to noise, blur, atmospheric effects, etc. Recent agentic methods usually model real-world image restoration (Real-IR) as a sequential tool calling problem over task-specific single-degradation restoration models.This paradigm, however, is fundamentally limited because complex real-world degradations cannot be cleanly undone degradation by degradation, and the tool used for task-specific models caps the capability of the agent system. In this work, we present HarnessIR, an agentic framework for Real-IR by harnessing a multimodal foundation model (MFM) as the executor. HarnessIR consists of five stages: perception and diagnosis, on-demand tool invocation, prompt composition, execution, and verification-driven refinement. Unlike prior agentic Real-IR methods that rely on tool chains assembled from task-specific models, HarnessIR feeds the restoration requirements, the perceptual diagnosis, and the evidence into an MFM that performs restoration in a single pass, followed by verification stages to determine whether the result warrants further processing. Under our harness, off-the-shelf MFMs handle restoration tasks remarkably well, achieving state-of-the-art results on the widely used MiO100 synthetic benchmark. More importantly, by exploiting the strong generalization ability of MFMs, HarnessIR delivers compelling restoration quality on challenging real-world scenes where previous agentic IR systems often struggle. Codes will be released.

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

1 INTRODUCTION

Image restoration (IR) is a classical yet challenging research problem. Early IR research mainly focuses on a specific type of degradation, such as denoising (Zhang et al., 2017; Zamir et al., 2022), deblurring (Cho et al., 2021; Chen et al., 2021), super-resolution (Dong et al., 2014; Chen et al., 2023), dehazing (Liu et al., 2019; Song et al., 2022), etc. Later, many all-in-one (Li et al., 2022a; Potlapalli et al., 2023) or multiple-in-one (Kong et al., 2024; Luo et al., 2024) IR methods have been proposed, aiming to train a unified model to handle different types of image degradations. However, most of these methods still assume that the input image is corrupted by one single type of degradation, and they explicitly or implicitly recognize the degradation before restoring the desired image. In practice, however, captured images in real-world are often corrupted by complex, mixed degradations that cannot be modeled by a specific degradation type.

Inspired by the recent rapid development of agentic models in multimodal understanding and generation (Achiam et al., 2023; Liu et al., 2024), agentic IR methods (Zhu et al., 2025; Zuo et al., 2025) have been proposed to address the issue of complex mixed degradations, where pretrained task-specific IR networks are exposed as *restoration tools*, and vision-language models (VLMs) are employed as the perceiver to assess image content and its degradations, map the identified degradations onto specialized IR models, and as the planner to schedule them into an execution sequence, frequently augmented with reflection, rollback, and rescheduling. However, this formulation is subject to fundamental limitations. First, it approximates coupled mixed degradations as a sequence of specialized restoration operations, whereas many real-world degradations interact and cannot be cleanly separated or reversed in sequence. Second, the planner converts fine-grained description of image content and degradation from the perceiver into a sequence of restoration tool calls, so much useful information is lost before restoration. Third, task-specified IR tools generalize poorly, limiting the overall generalization ability of the whole system. Last but not least, most agentic IR methods are developed and evaluated on synthetic mixed degradations, whereas real-world degradations often interact with each other, making them difficult to simulate. Therefore, although these paradigms have achieved great progress, their generalization to real-world scenes remains limited.

Recently, generative multimodal foundation models (MFMs) (Google, Googleb; OpenAI, 2025; Wu et al., 2025) have advanced rapidly. Having been trained for image editing, these models demonstrate great potential for IR tasks in recent works (Sun et al., 2026b; Yang et al., 2026). Basically, MFMs treat IR as an editing problem: they accept the LQ image together with an instruction to accomplish the restoration task. Unfortunately, prompting MFMs naively often yields unsatisfying results, as illustrated in Fig. 1 and Fig. 3, these models often over-generate, embellishing detail and repainting texts and image contents. However, we argue that this does not mean that MFMs cannot perform complex real-world IR tasks; the key lies in how we instruct them, together with additional tools, to execute IR tasks in appropriate steps, harnessing the strong generative and generalization capabilities of MFMs for high-quality, universal real-world image restoration (Real-IR).

To this end, we present **HarnessIR**, an agentic framework for universal real-world IR by harnessing an MFM as the executor. HarnessIR is organized into five stages: perception and diagnosis, on-demand tool invocation, prompt composition, execution, and verification-driven refinement. One key distinction between our design and previous work is that HarnessIR schedules *no restoration tools*, and it performs restoration by MFM calls; however, it invokes *auxiliary tools* — OCR (Zhang et al., 2026), face detection (Jia Guo & Gecer, 2026), depth estimation (Yang et al., 2024b), semantic segmentation (Carion et al., 2026) — which act as evidence providers for prompt composition rather than links in a restoration chain. As shown in Fig. 2, the LQ image and the restoration requests are first passed to a VLM, which returns a diagnosis of the content and its degradations, along with a recommendation on which auxiliary tools to invoke. The system executes the plan and collects the resulting evidence, which is fed back to the VLM with the LQ input and the first-stage analysis. The VLM then composes a prompt that contains rich content and degradation information, as well as the gathered evidence rendered in language. The prompt and the LQ image are passed to the MFM executor, and the VLM then examines the output to determine whether the restoration is satisfactory and how to revise the prompt for re-execution. HarnessIR binds the MFM to what the given LQ image actually needs for restoration, eliciting the MFM's restoration capability while regulating how far it can alter the input, so that degradations can be removed without changing the content.

Using advanced MFMs such as Nano Banana 2 (Google, Googleb) and GPT-Image-2.5-Sunburst (OpenAI, 2025) as executors, HarnessIR handles image restoration remarkably well. On

MiO100 (Kong et al., 2024), a widely used synthetic mixed-degradation benchmark, HarnessIR surpasses existing all-in-one and agentic IR methods on almost every metric. On the more challenging real-world images captured with both single and mixed degradations, where many all-in-one and agentic IR methods fail outright, HarnessIR outperforms prior methods by a large margin. Our ablation studies attribute these restoration gains to the harness rather than the executor alone. Our contributions are summarized as follows:

- We first identify the limitations in the previous agentic IR paradigms and propose moving the agent action primitive from restoration tool scheduling to MFM harnessing.
- We then propose **HarnessIR**, an agentic framework that harnesses a single MFM as the restorer without using task-specific restoration models as tools.
- Experiments show that HarnessIR reaches leading performance on the widely used MiO100 benchmark; more importantly, it restores real-world images with complex degradations significantly better than existing agentic IR methods.

2 Related Work and Discussion

All-in-One and Agent-based Image Restoration. Early IR research usually addresses a specific type of degradation in isolation, such as denoising (Zhang et al., 2017), deblurring (Cho et al., 2021), super-resolution (Dong et al., 2014), dehazing (Song et al., 2022), etc. Subsequently, many all-in-one and multi-in-one methods have been proposed to address different degradations using a unified network, such as contrastive degradation embeddings (Li et al., 2022a), degradation-aware learnable prompts (Potlapalli et al., 2023; Kong et al., 2024), CLIP-based degradation cues (Luo et al., 2024; Jiang et al., 2024), and natural-language instructions (Conde et al., 2024). However, most of these methods assume that the input LQ image still has a single identifiable degradation, which limits their performance in unconfined real-world scenes (Cai et al., 2019), where the captured images are often corrupted by complex and mixed degradations. To handle mixed degradations, agentic IR (Zhu et al., 2025) takes task-specific restoration networks as tools and plans a sequence of calls over them. Existing agentic IR systems differ mainly in the design of the sequence. One line of research searches the space of tool orderings and uses intermediate results to update the plan (Chen et al., 2024b; Zhu et al., 2025; Zuo et al., 2025). Another line of work constrains the plan using degradation priors, staged decomposition, or complexity-based routing (Jiang et al., 2026; Zhou et al., 2025; Li et al., 2026). The third line learns the policy directly via reinforcement learning using perceptual or no-reference quality rewards (Lu et al., 2026; Jia et al., 2026; Lin et al., 2025b; Zhu et al., 2026). In all these paradigms, the tools are task-specific restoration networks, which are also executors. Consequently, their generalization performance is bounded by these tools, which work poorly under complex, out-of-distribution real-world degradations.

Multimodal Foundation Models (MFMs). Generative MFMs (Wu et al., 2025; Labs, 2025; Xie et al., 2024) and their instruction-based editing interface allow a single model to perform an open set of pixel-level operations specified in language. Since restoration can be phrased as an editing instruction, researchers have begun to examine MFMs for IR. (Sun et al., 2026b) evaluated Nano Banana 2 (Google, Googleb) as a unified restorer. While the perceptual results are appealing, it over-synthesizes many unfaithful details. Similarly, (Yang et al., 2026) prompted MFM with fixed instructions and found that the fidelity of the image is difficult to preserve. However, we argue that this does not mean that MFMs lack the capability for performing faithful Real-IR. The key lies in how we prompt them. With an appropriate harness to elicit and utilize their capabilities, the MFMs can significantly improve perceptual quality while preserving image content.

Harness: From Foundation Models to Agentic Systems. An MFM is not by itself an agent. What turns it into an agentic system is the *harness* (Yang et al., 2024a), which includes the prompts and control flow that frame the task (Yao et al., 2022), the tools that collect evidence (Schick et al., 2023), the verification logic that decides when an output is acceptable (Shinn et al., 2023), and the memory if needed (Packer et al., 2023). With an appropriate harness, the base MFM can become a markedly strong agent depending on what is wrapped around it (Yang et al., 2024a; Wang et al., 2025). Our proposed HarnessIR is a different yet much more effective instantiation for agentic IR: it diagnoses before acting, calls auxiliary tools for evidence, explicitly constrains what should be restored, and employs a verifier to decide whether try again based on the current results.

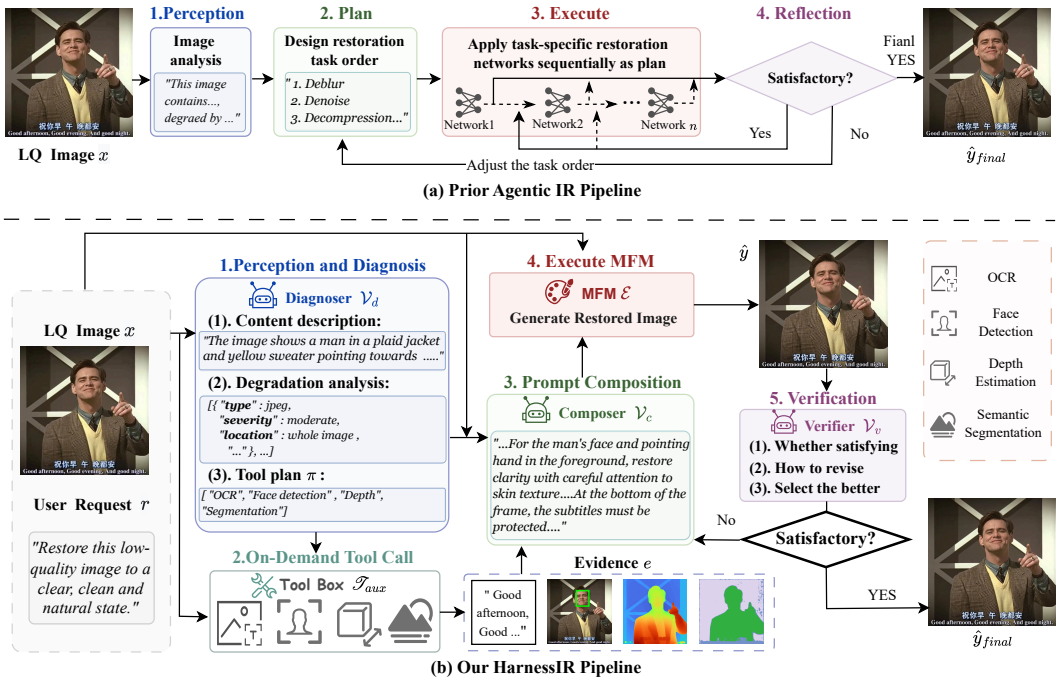

Figure 2: Comparison between (a) prior agentic IR pipelines (**top**) and (b) HarnessIR (**bottom**). Prior methods sequentially apply task-specific restoration models planned by VLMs to produce the output. HarnessIR instead uses auxiliary tools for image-specific evidence, composes a restoration instruction for MFM, and invokes verification-driven re-execution when necessary.

3 HarnessIR

3.1 Problem Definition and Overview

Given an LQ image x and a user request r, we aim to produce a restored image $\hat{y}$. The request is a short natural-language sentence provided by the user, expressing the restoration intent —*e.g.*, improve the brightness of the photo (see the **Appendix A**). As shown in Fig. 2(a), prior agentic IR methods (Zhu et al., 2025; Zuo et al., 2025) usually cast this as a composition problem over a library of task-specific restoration networks $\{f_t\}_{t\in\mathcal{T}}$: a perception step reads x into a description, a planner turns that description into a tool order $(t_1, \ldots, t_K)$, and the image is passed through the corresponding restoration networks one after another, with reflection between the steps. In contrast, our HarnessIR performs restoration with a call to a frozen MFM executor $\mathcal{E}$, driven by a prompt p:

$$\text{Prior Agentic IR:}\quad \hat{y}^{(k)} = f_{t_k}\big(\hat{y}^{(k-1)}\big), \qquad \text{HarnessIR:}\quad \hat{y}^{(k)} = \mathcal{E}\big(x,\, p^{(k)}\big), \tag{1}$$

where $k = 1, \ldots, K$ counts rounds and $\hat{y}^{(0)} = x$. Prior agentic IR methods vary the restoration executor to process intermediate outputs sequentially. Instead, HarnessIR keeps the executor fixed and varies only the prompt. In each round, it processes the original input x and therefore constitutes an alternative complete result. The verifier selects the returned candidate, $\hat{y} = \hat{y}^{(k^*)}$, and requests another round only when necessary (Sec. 3.4).

We use five stages to achieve our goal, as illustrated in Fig. 2(b). We instantiate the diagnoser, composer, and verifier using the same frozen vision-language model (VLM) (Google, Googlea) with three role-specific briefs. The *diagnoser* $\mathcal{V}_d$ produces a diagnosis d of image content and degradations, together with a plan $\pi \subseteq \mathcal{T}_{\text{aux}}$ specifying what auxiliary tools to consult. After the planned tools return evidence e, the *composer* $\mathcal{V}_c$ writes p from (x, r, d, e); the executor $\mathcal{E}$ produces $\hat{y}$; and the *verifier* $\mathcal{V}_v$ decides whether the result is accepted and which candidate is retained. Diagnosis and composition are separated because the final instruction can only be written after the diagnoser selects the tools and the requested evidence has been acquired. HarnessIR focuses on how to construct an image-specific instruction and how to verify and re-execute. In the following sections, we describe its evidence collection, prompt composition, and failure-triggered refinement in detail.

3.2 What Is Planned: Evidence to Gather, Not Restorers to Run

HarnessIR plans over a different space from previous agentic IR methods (Zhu et al., 2025; Jiang et al., 2026), which search over orderings of restoration networks to decide how the image will be processed. Instead, HarnessIR plans what the system should perceive before writing p. The set of auxiliary tools — text recognition, face detection, depth estimation, and semantic segmentation — supplies observations that the executor would not otherwise seek on its own. For a capable MFM, the simpler the prompt it is given, the more freely it follows its own prior, which may result in hallucinations for restoration. The evidence and the instructions elicit what the executor should do to reduce the possibility for hallucination.

The tools to consult are decided by the diagnoser that produces d in the same call, rather than by fixed rules, because the decision depends on the content of the image: text recognition is worth running when legible text is present, and depth estimation is worth taking when the degradation is spatially structured. For images with simple contents and degradation, the tools may not need to be called. The invoked tools return raw, image-specific observations, e.g., recognized strings, confidences, and bounding boxes for texts; bounding boxes for face; a dense depth map; and a segmentation map with semantic labels. These outputs are supplied directly to the composer in their native structured or visual forms, and they are not passed to the executor. The exact formats of tool call and a complete example are provided in the separate **Supplementary Material**.

3.3 What Is Passed to Executor: Instruction Prompt, Not Execution Order

The diagnosis is expressed in language, while the evidence from the tool may be structured or visual. The composer interprets this evidence jointly with the LQ image rather than merely transcribing it, and converts the relevant observations into a single textual instruction. The evidence is folded into the image description rather than appended as a list, each piece tied to the content it describes and the degradation it bears witness to. For example, the haze is to be cleared more aggressively in the background than in the foreground, as the depth indicates. The composer decides whether to accept the evidence or weigh the evidence against the LQ input x.

In addition, the composer's role-specific brief includes general restoration requirements that encode domain knowledge about each restoration operation. For example, deraining and desnowing should remove falling rain and snow without removing wet roads or snow cover in the scene, while brightness should remain unchanged when low-light enhancement is not requested. The composer jointly considers the provided information and these requirements, retains only the applicable requirements, and rewrites them into an image-specific prompt p. The executor receives only the LQ image x and the composed prompt p. Unlike prior agentic IR methods, which formulate their analysis as a list of task-specific restoration tools, HarnessIR embeds image-specific information into the prompt, including what degradations to treat, what content to preserve, and where these requirements apply. The complete brief of the composer is provided in the separate **Supplementary Material**.

3.4 When the Loop Fires: On Failure, Not by Design

HarnessIR is designed to succeed in one execution and invoke another round only when the verifier judges the result unsuccessful. The output is first measured using NR-IQA scores and inexpensive comparisons with x, including brightness, color statistics, and SSIM between the output and the LQ input. These measurements provide diagnostic evidence rather than fixed acceptance thresholds: they are passed to the verifier together with x and the output, and the verifier makes the final decision. The verifier carries the same restoration requirements as the composer and weighs the output against them, together with the request r and the measurements. The instruction to the executor is not shown to the verifier so that the result is judged against what the restoration task is supposed to do rather than against the instruction we ask the executor to follow.

HarnessIR starts over from x, not the output of the last round as in previous agentic IR methods. This is because if the output of the last round is verified to be unsuccessful, the content of the image can be altered, such as incorrect text, fabricated structure, etc. Such an output is the wrong place to start from, while x remains the most safe input for the executor to start from. When the verifier calls for a redo, it writes down what should be corrected. The composer then writes a new instruction for the same x, and passes it to the executor. The verifier also compares the new restoration result with

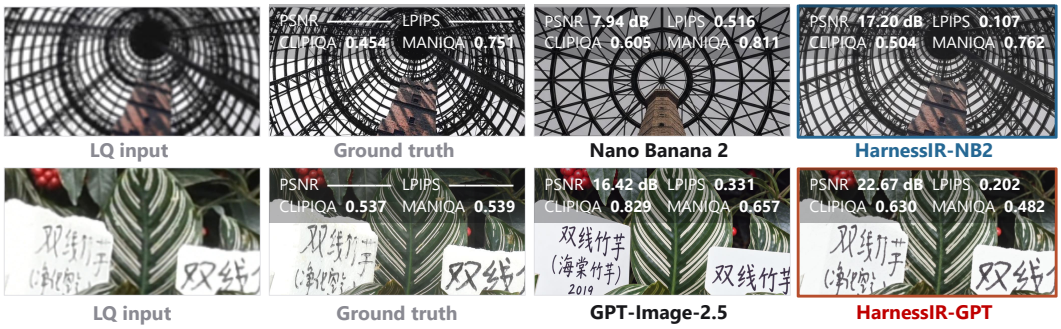

Figure 3: Direct use of MFMs may yield higher NR-IQA scores than GT, even when the image content is largely altered. NR-IQA should not be used as an independent metric for IR performance.

the best candidate obtained in previous rounds and retains the better one as $\hat{y}$. We allow at most three execution rounds; if no candidate is accepted by then, the best candidate selected by the verifier is returned. The complete verifier prompts are provided in the separate **Supplementary Material**.

4 Experiments

4.1 Experimental Setup

Implementation of HarnessIR. HarnessIR is **inference-only**: no component is trained or fine-tuned. The diagnoser, composer, and verifier are all filled by Gemini-3.7-Flash (Google, Googlea), queried once per role under different briefs (the system prompts). We run HarnessIR with two MFMs: Nano Banana 2 (NB2) (Google, Googleb) and GPT-Image-2.5-Sunburst (GPT-Image) (OpenAI, 2025), each accessed via its public interface. Four auxiliary tools are used: PP-OCRv6 (Zhang et al., 2026) for text recognition, InsightFace Project (Jia Guo & Gecer, 2026) for face detection, Depth-Anything-V2 (Yang et al., 2024b) for depth estimation, and SAM 3 (Carion et al., 2026) for semantic segmentation. We use a fixed request template for each degradation category to represent the usage of general users (see the **Appendix A**).

For each executor $\mathcal{E}$, we report two variants under exactly the same LQ image and category-specific request r: **Baseline-$\mathcal{E}$**, which directly receives the LQ image and r, representing direct MFM usage; **HarnessIR-$\mathcal{E}$**, our proposed pipeline with at most three MFM execution rounds per image. The difference between the two variants lies in only the harness, rather than the user request.

Test Sets. We evaluate HarnessIR on three test sets. One is the synthetic test set MiO100 (Kong et al., 2024; Zhu et al., 2025), which has three degradation groups (A, B, and C). Since our main concern is the model performance on real-world LQ images, following SEAR (Cui et al., 2026), which aggregates 100 real-world pairs, we collect two larger real-world test sets from public datasets: Real-Paired-200 with paired GT and Real-NoGT-200 without GT. They cover unconstrained mixed degradations, as well as haze, rain, snow, low light and so on. The composition and sources are detailed in the **Appendix B**, and both sets will be made publicly available.

Compared Methods. We compare HarnessIR against six all-in-one models (AirNet (Li et al., 2022a), PromptIR (Potlapalli et al., 2023), MiOIR (Kong et al., 2024), DA-CLIP (Luo et al., 2024), InstructIR (Conde et al., 2024), AutoDIR (Jiang et al., 2024)) and six agentic methods (AgenticIR (Zhu et al., 2025), MAIR (Jiang et al., 2026), 4KAgent (Zuo et al., 2025), TIR-Agent (Jia et al., 2026), SEAR (Cui et al., 2026), OPERA (Zhu et al., 2026)). For those methods whose codes and weights are publicly available, we reproduce their results with the official implementations, and report more NR-IQA metrics beyond the ones reported in their original papers. For those methods whose codes are not released, we only quote the MiO100 results reported in their papers.

4.2 Evaluation Protocol and Metrics

Following prior works (Zhu et al., 2025; Yang et al., 2026; Cui et al., 2026), we measure fidelity with PSNR and SSIM (Wang et al., 2004) in the Y channel, perceptual quality with LPIPS (Zhang et al., 2018) and DISTS (Ding et al., 2020), and image quality with the no-reference image quality assess-

Table 1: Group-averaged results on MiO100. The best and second-best results are marked in red and blue, respectively. Results in which HarnessIR improves on its baseline are in **bold**. Δ means the absolute deviation from the GT score; the smaller, the better. The GT and LQ rows report the raw metric values. D-Score and F-Score are given by the VLM evaluator for degradation removal and content preservation, both on a scale of 0–100; DF-Score is their per-image geometric mean.

Method	Full-reference Metrics				No-reference Metrics					VLM-based IQA		
	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	ΔMANIQA↓	ΔCLIP-IQA↓	ΔMUSIQ↓	ΔTOPIQ↓	ΔAFINE-NR↓	D-Score↑	F-Score↑	DF-Score↑
AirNet	20.30	0.65	0.46	0.25	0.22	0.28	29.00	0.32	0.27	25.1	89.7	29.78
PromptIR	20.66	0.66	0.45	0.26	0.21	0.28	28.98	0.33	0.26	26.4	91.7	30.41
MiOIR	20.76	0.66	0.43	0.25	0.22	0.29	27.29	0.33	0.26	28.5	90.6	37.09
DA-CLIP	20.33	0.64	0.46	0.26	0.22	0.29	28.37	0.33	0.25	19.4	87.7	25.47
InstructIR	20.84	0.60	0.54	0.28	0.24	0.31	31.84	0.37	0.32	21.9	90.6	29.42
AutoDIR	20.59	0.65	0.39	0.23	0.14	0.25	16.92	0.25	0.15	38.9	68.0	39.05
AgenticIR	21.15	0.67	0.35	0.21	0.14	0.18	11.73	0.19	0.07	49.4	43.3	42.97
MAIR	20.45	0.64	0.33	-	0.32	0.18	11.61	-	-	-	-	-
4KAgent	20.73	0.63	0.34	0.20	0.10	0.11	8.31	0.12	0.11	47.1	36.3	36.88
TIR-Agent	21.57	0.66	0.32	-	0.25	0.10	6.83	-	-	-	-	-
SEAR	21.51	0.67	0.34	-	0.32	0.17	10.56	-	-	-	-	-
OPERA	21.78	0.69	0.37	-	0.32	0.22	12.67	-	-	-	-	-
GPT-Image-2.5	17.95	0.54	0.36	0.20	0.02	0.10	4.64	0.06	0.01	48.8	20.9	30.38
HarnessIR-GPT	**19.70**	**0.58**	**0.30**	**0.17**	**0.01**	**0.07**	**2.81**	**0.02**	**0.01**	**67.5**	**37.7**	**48.44**
Nano Banana 2	20.81	0.63	0.24	0.13	0.00	0.03	0.70	0.04	0.00	76.7	44.9	55.87
HarnessIR-NB2	**22.21**	**0.65**	**0.21**	**0.12**	0.00	0.03	**0.25**	0.04	0.00	75.2	**51.7**	**59.94**
GT	-	-	-	-	0.64	0.65	68.79	0.64	-0.92	-	-	-
LQ	19.86	0.57	0.58	0.31	0.23	0.28	32.77	0.37	0.31	-	-	-

Table 2: Results on Real-Paired-200, which contains 200 real-world LQ images with paired GT from public real-world datasets. Marking and metrics follow Tab. 1.

Method	Full-reference Metrics				No-reference Metrics					VLM-based IQA		
	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	ΔMANIQA↓	ΔCLIP-IQA↓	ΔMUSIQ↓	ΔTOPIQ↓	ΔAFINE-NR↓	D-Score↑	F-Score↑	DF-Score↑
AirNet	22.40	0.724	0.440	0.254	0.108	0.107	13.27	0.109	0.133	16.8	85.7	25.2
PromptIR	24.08	0.743	0.416	0.241	0.098	0.099	12.78	0.105	0.114	16.7	91.3	25.3
MiOIR	24.87	0.735	0.418	0.251	0.096	0.086	12.20	0.104	0.124	10.5	93.0	18.9
DA-CLIP	23.87	0.739	0.406	0.243	0.099	0.106	11.83	0.093	0.111	17.8	88.2	26.7
InstructIR	22.92	0.747	0.411	0.236	0.108	0.116	12.34	0.104	0.124	25.4	90.2	36.9
AutoDIR	23.80	0.735	0.372	0.233	0.079	0.093	4.19	0.052	0.046	29.0	74.9	32.5
AgenticIR	22.41	0.734	0.398	0.235	0.097	0.087	7.41	0.072	0.083	33.7	68.4	38.8
4KAgent	22.18	0.708	0.389	0.231	0.088	0.064	2.89	0.030	0.131	35.2	56.1	37.4
GPT-Image-2.5	20.87	0.660	0.361	0.222	0.095	0.201	15.78	0.212	0.180	69.0	27.5	40.1
HarnessIR-GPT	**23.74**	**0.729**	**0.286**	**0.176**	**0.066**	**0.131**	**12.32**	**0.128**	**0.108**	**82.7**	**50.4**	**60.2**
Nano Banana 2	23.94	0.729	0.309	0.181	0.058	0.077	11.91	0.118	0.074	67.9	47.8	52.9
HarnessIR-NB2	**26.63**	**0.775**	**0.246**	**0.149**	**0.020**	**0.001**	**5.52**	**0.028**	**0.033**	**75.2**	**63.6**	**62.0**
GT	-	-	-	-	0.560	0.451	54.36	0.417	-0.901	-	-	-
LQ	25.27	0.736	0.436	0.258	0.098	0.092	13.08	0.107	0.115	-	-	-

ment (NR-IQA) metrics MANIQA (Yang et al., 2022), CLIP-IQA (Wang et al., 2023), MUSIQ (Ke et al., 2021), TOPIQ (Chen et al., 2024a), and A-FINE (Chen et al., 2025a). However, it should be noted that a higher NR-IQA score does not, by itself, indicate better restoration performance because their purpose is to evaluate image quality without considering whether the restored content is correct or not. As a result, hallucination can be rewarded, even when the content of the restored image deviates significantly from the input LQ image. Fig. 3 shows some examples. We see that the restored images significantly outscore their GTs on NR-IQA metrics, while the image content and text have been largely repainted. Such cases are rare for earlier discriminative restoration networks, but can be common for diffusion-based generative models. This is because the generative methods are more prone to altering image content and NR-IQA metrics only measure whether the given image is of good quality, no matter what content it has.

To solve this issue of unreasonable use of NR-IQA metrics in Real-IR method evaluation, we make two adjustments. First, when a GT exists, we report the absolute distance, denoted by Δ, between the NR-IQA scores of the restored image and the GT image. A value of Δ closer to zero means better quality since the GT is the target. Second, regardless of whether a GT exists, following recent IR and IQA works (Sun et al., 2026a; Wu et al., 2026), we use an independent VLM evaluator of the HarnessIR system to assess restoration performance directly. Given the LQ input and the restored output, it reports D-Score for degradation removal and F-Score for content preservation, both on a scale of 0–100. For each image, we compute their geometric mean, DF-Score $= \sqrt{DF}$, which is high when both degradation removal and fidelity are good. The evaluator prompt and protocol, ablations, and comparisons with human judgments are provided in the **Appendix C**.

Method	No-reference Metrics					VLM-based IQA		
	MANIQA↑	CLIP-IQA↑	MUSIQ↑	TOPIQ↑	AFINE-NR↓	D-Score↑	F-Score↑	DF-Score↑
AirNet	0.525	0.406	42.94	0.307	-0.689	13.4	86.5	21.6
PromptIR	0.531	0.413	43.17	0.310	-0.703	11.2	90.4	21.0
MiOIR	0.537	0.399	44.44	0.321	-0.695	11.4	90.2	22.5
DA-CLIP	0.529	0.395	43.50	0.311	-0.700	12.6	88.3	20.3
InstructIR	0.521	0.383	43.39	0.311	-0.689	21.4	86.0	32.0
AutoDIR	0.538	0.399	47.85	0.337	-0.732	17.9	83.7	27.0
AgenticIR	0.525	0.374	45.38	0.325	-0.697	19.1	78.1	26.9
4KAgent	0.516	0.396	48.29	0.343	-0.677	18.9	70.4	25.5
GPT-Image-2.5	0.676	0.693	72.92	0.658	-1.059	59.3	16.9	29.6
HarnessIR-GPT	0.660	0.665	70.28	0.610	-1.000	**76.4**	**36.0**	**49.3**
Nano Banana 2	0.623	0.534	66.18	0.531	-0.933	65.1	34.4	47.3
HarnessIR-NB2	0.605	0.484	62.08	0.483	-0.904	**71.3**	**64.1**	**62.9**

Table 3: Results on Real-NoGT-200. Marking and metrics follow Tab. 1. Without GT, NR-IQA alone cannot correctly reflect the restoration performance, so these columns are grayed out.

4.3 Main Results

Results on Synthetic MiO100 Test Sets. Tab. 1 summarizes the group-averaged results on MiO100 (detailed results of Group A, B and C are in **Appendix D**). Compared with the direct-use baselines (Nano Banana 2 and GPT-Image-2.5), HarnessIR-NB2 and HarnessIR-GPT gain 1.40 and 1.75 dB in PSNR, improve LPIPS by 0.03 and 0.06, and DF-Score by 4.1 and 18.1 points, respectively. These gains mainly reflect better content preservation. HarnessIR also substantially improves GPT-Image-2.5's D-Score. For NB2, D-Score scores change not too much because MiO100 is constructed by applying controlled synthetic degradations to curated GT images. NB2 can already remove these degradations effectively.

Against prior methods, HarnessIR-NB2 achieves the best results on most metrics except SSIM and F-Score. HarnessIR-GPT achieves the best result on ΔTOPIQ. Several all-in-one models achieve relatively high F-Scores but D-Scores below 30, suggesting that their fidelity comes partly from making only limited restoration changes. Although HarnessIR-GPT improves substantially over the direct use of GPT-Image-2.5, it is weak in PSNR and SSIM. This reflects the executor's capabilities rather than a limitation of the harness: the harness helps to unleash the executor's restoration potential but cannot raise its upper ceiling. GPT-Image-2.5 has a strong generative prior but tends to restore more aggressively, which makes its outputs less pixel-aligned with the GT.

Results on Real-Paired-200 and Real-NoGT-200 Test Sets. The advantage of HarnessIR is even more pronounced on real-world data. On Real-Paired-200 (Tab. 2), both HarnessIR variants substantially improve their direct-use baselines in fidelity and restoration quality. HarnessIR-NB2 achieves a 2.69 dB PSNR gain and reaches a DF-Score of 62.0, while HarnessIR-GPT improves PSNR by 2.87 dB and raises its DF-Score from 40.1 to 60.2. HarnessIR also outperforms all prior methods by a clear margin in DF-Score, with both variants exceeding the previous best of 38.8. HarnessIR-NB2 also achieves the smallest deviation from the GT on four of the five NR-IQA metrics. In particular, HarnessIR-NB2 achieves the highest PSNR, surpassing the best prior result by 1.76 dB. This is notable because the LQ input already reaches 25.27 dB; the high PSNR but low D-Scores of several all-in-one methods therefore likely reflect limited restoration, whereas HarnessIR-NB2 improves upon the input while maintaining strong degradation removal.

The results on Real-NoGT-200 (Tab. 3) lead to the same conclusion. HarnessIR substantially improves both D-Score and F-Score for the two executors, yielding DF-Scores of 62.9 for NB2 and 49.3 for GPT-Image-2.5. These scores are well above the best prior result of 32.0. The lower NR-IQA scores of HarnessIR relative to the direct use of MFM should therefore not be interpreted as a decline in performance. As discussed in Fig. 3, NR-IQA can reward hallucinated details even when the image content has been altered. We consequently use DF-Score for the no-GT setting, as it evaluates degradation removal and content preservation jointly.

Visual Results. Fig. 4 shows qualitative comparisons across the three test sets. The all-in-one methods make little visible improvement over the LQ inputs in these examples, with only PromptIR removing some of the snow in the third image. Previous agentic IR methods make the MiO100 image somewhat clearer, but produce only limited improvement on the two real-world images. In contrast, the MFM-based methods remove degradations more effectively. However, direct use of NB2 and GPT-Image-2.5 often alters image content. In the first example, HarnessIR better preserves the original color tone and scene content. In the second, both direct MFM baselines mistakenly remove the white cross already in the image, and NB2 also changes the text. In the third, both baselines turn the nighttime scene into daytime and remove the snow cover; GPT-Image-2.5 additionally makes the

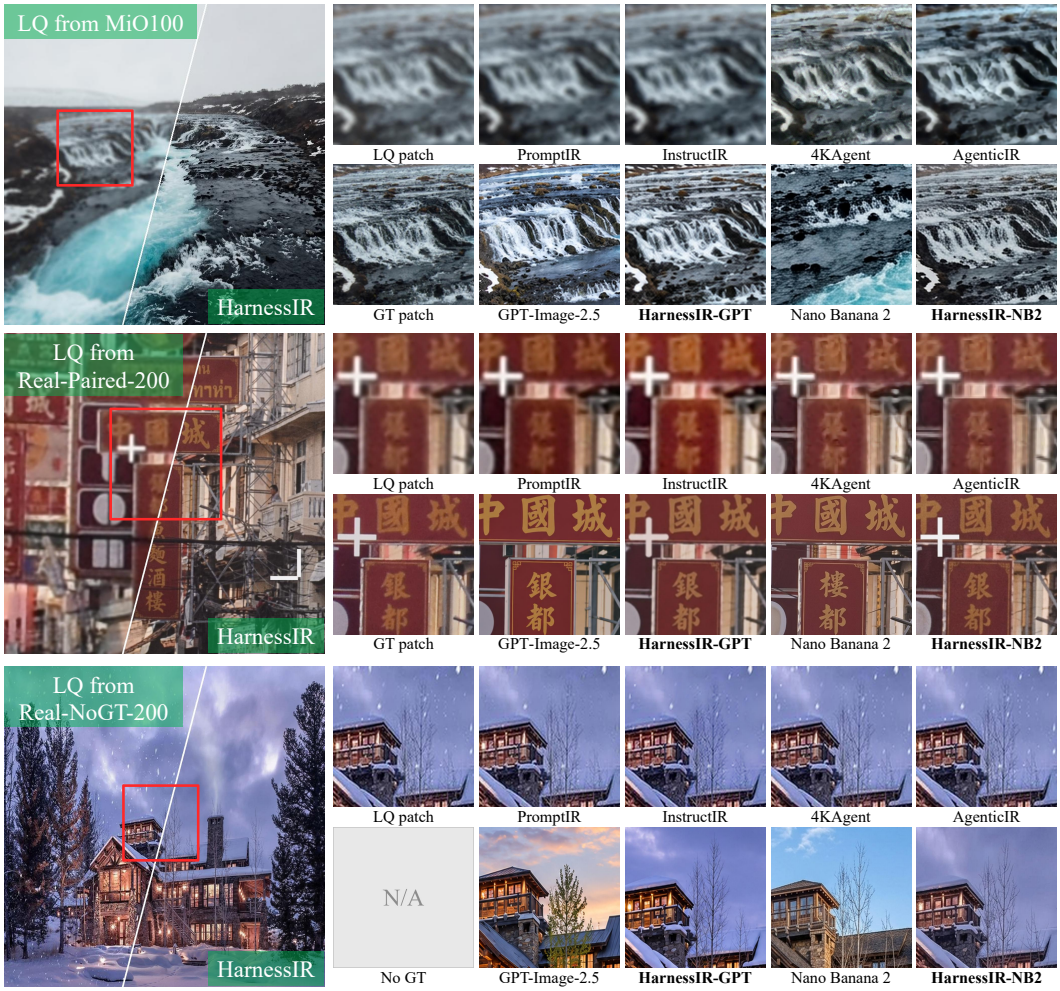

Figure 4: Visual comparisons on three test sets. HarnessIR achieves better fidelity and visual quality.

trees appear to have sprouted leaves. HarnessIR avoids these unwanted changes while still removing degradations, producing visually cleaner results that remain faithful to the original LQ images.

Due to the limit of space, we put the **Ablation Study**, **Cost Analysis**, **User Study** and **More Visual Results** in **Appendix E**, **Appendix F**, **Appendix G** and **Appendix H**, respectively.

5 Conclusion

We presented **HarnessIR**, an agentic framework that used MFMs as the executor for universal real-world IR. By combining image-specific diagnosis, auxiliary tools, prompt composition, and verification-driven re-execution, HarnessIR harnessed the MFM to remove degradations without altering image content. Across synthetic and challenging real-world benchmarks, HarnessIR consistently improved restoration performance over the direct use of MFM, achieving particularly large gains on real-world images. Our ablations further showed that these improvements came from the coordinated design of the harness at an acceptable cost. Overall, HarnessIR demonstrated that a carefully designed harness can substantially unlock the restoration potential of proper MFMs and turn them into more effective and controllable universal real-world restorers.

Limitations. HarnessIR requires access to online VLM and MFM APIs, which is a common requirement by existing agentic IR methods such as AgenticIR (Zhu et al., 2025) and 4KAgent (Zuo et al., 2025) (using GPT-4o). Moreover, an agentic harness can only elicit and regulate the capabilities of its executor but cannot raise the executor's performance ceiling. In the future, with this general IR agent, we can curate diverse and challenging restoration data and generate verification-guided training supervision to improve the capabilities of local and smaller models.

Ethics Statement

This work presents HarnessIR, an inference-only agentic framework that drives generative multimodal foundation models (MFMs) for real-world image restoration. All data used in this paper come from publicly released datasets, used under their original licenses and cited in the paper; the two real-world test sets we assemble, Real-Paired-200 and Real-NoGT-200, are re-samplings of these existing public sources rather than newly captured data, and their composition and provenance are documented in Appendix B. We did not collect personally identifiable information. Real-world photographs may nevertheless contain identifiable individuals, and we note that the face detector used by our harness serves only to localize face regions as internal evidence for prompt composition: it performs no face recognition or identity matching, and no facial embeddings or identity attributes are stored, released, or used in evaluation. The user study reported in Appendix G was minimal risk: participants only judged the perceptual quality and fidelity of anonymized restored images, with no collection of sensitive personal data; participants were informed of the purpose of the study and participated voluntarily. No IRB approval was sought, as the study falls within the minimal-risk category. We also note one intrinsic limitation with ethical relevance: because the executor is a generative model, its outputs are plausible reconstructions rather than faithful recoveries of the underlying scene, and may contain hallucinated text, faces, or fine structures. Confronting this risk is in fact a central motivation of our design — we show in Sec. 4 that no-reference IQA metrics can actively reward such hallucinations, we introduce the fidelity-oriented F-Score and the combined DF-Score to penalize them, and the verification stage of HarnessIR is explicitly designed to reject outputs that drift from the input content. Even so, restored images should not be used in forensic, medical, or evidentiary settings where content fidelity must be guaranteed, and we recommend that they be labeled as AI-processed when shared. Finally, HarnessIR builds on third-party proprietary models accessed through their public APIs in accordance with their terms of service; these models are trained on undisclosed data, so restoration quality may vary across skin tones, geographic regions, and scene types that are unevenly represented in their training distributions, and our benchmarks are not designed to audit such disparities.

Reproducibility Statement

We have made the following efforts to ensure that our results are reproducible. HarnessIR is inference-only: no component is trained or fine-tuned, so reproducing our results requires no training pipeline. Sec. 4 specifies the exact VLM and MFM versions, the four auxiliary tools, and the two variants (Baseline-$\mathcal{E}$ and HarnessIR-$\mathcal{E}$) that are compared under identical inputs and user requests. The complete briefs of the diagnoser, composer, and verifier, together with the fixed per-category request templates, are given in the separate **supplementary material**, and a full worked example tracing all five stages end-to-end is provided in the separate **supplementary material**. The construction, source datasets, sampling protocol, and preprocessing Real-Paired-200, and Real-NoGT-200 are documented in Appendix B, and both real-world test sets will be made publicly available. The evaluation protocol and metrics are described in Sec. 4, with the DF-Score evaluator prompt, evaluator-VLM replacement experiments, and alignment with human judgments detailed in Appendix C; ablations of each component of the harness are reported in Appendix E, and the API calls, runtime, and monetary cost of reproducing our runs are reported in Appendix F. Our code will be released. One caveat is intrinsic to this setting: the VLM and the MFM executors are closed-weight services with stochastic decoding, so individual outputs are not bit-wise reproducible and provider-side updates may shift absolute numbers over time; we therefore report the exact model versions used, and all reported numbers are averaged over every image in each test set rather than drawn from selected examples.

AI Use Statement

Generative AI tools are a core object of study in this work rather than an incidental writing aid, and we disclose their use as follows. A vision-language model (Gemini-3.7-Flash) is used as the diagnoser, composer, and verifier of our harness, and generative multimodal foundation models (Nano-Banana-2 and GPT-Image-2.5) are used as the restoration executors; this is the central methodological contribution of the paper and is described in full in Sec. 4, with all role-specific briefs given in the separate **supplementary material**. A vision-language model is additionally used as an independent evaluator to produce the D-Score and F-Score reported throughout our experiments; this

evaluator is separate from the models inside the harness, and its judgments are validated against human preferences through the user study reported in the separate **supplementary material** and the alignment analysis in Appendix C. We also used a large language model for grammar and phrasing improvements on the manuscript text; all research ideas, experimental designs, analyses, and conclusions are the authors' own, and every AI-assisted passage was reviewed by the authors. We take full responsibility for the final content of this work.

REFERENCES

Abdelrahman Abdelhamed, Stephen Lin, and Michael S Brown. A high-quality denoising dataset for smartphone cameras. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1692–1700, 2018.

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.

Codruta O Ancuti, Cosmin Ancuti, and Radu Timofte. Nh-haze: An image dehazing benchmark with non-homogeneous hazy and haze-free images. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1798–1805. IEEE, 2020.

Cosmin Ancuti, Codruta O Ancuti, Radu Timofte, and Christophe De Vleeschouwer. I-haze: A dehazing benchmark with real hazy and haze-free indoor images. In *International conference on advanced concepts for intelligent vision systems*, pp. 620–631. Springer, 2018.

Jianrui Cai, Hui Zeng, Hongwei Yong, Zisheng Cao, and Lei Zhang. Toward real-world single image super-resolution: A new benchmark and a new model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3086–3095, 2019.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. In *International conference on learning representations*, volume 2026, pp. 138846–138923, 2026.

Chaofeng Chen, Jiadi Mo, Jingwen Hou, Haoning Wu, Liang Liao, Wenxiu Sun, Qiong Yan, and Weisi Lin. Topiq: A top-down approach from semantics to distortions for image quality assessment. *IEEE Transactions on Image Processing*, 33:2404–2418, 2024a.

Du Chen, Tianhe Wu, Kede Ma, and Lei Zhang. Toward generalized image quality assessment: Relaxing the perfect reference quality assumption. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 12742–12752, 2025a.

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. *Advances in Neural Information Processing Systems*, 37:110643–110666, 2024b.

Jie Chen, Cheen-Hau Tan, Junhui Hou, Lap-Pui Chau, and He Li. Robust video content alignment and compensation for rain removal in a cnn framework. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6341–6349, 2018. doi: 10.1109/CVPR.2018.00658.

Junyang Chen, Jinshan Pan, and Jiangxin Dong. FaithDiff: Unleashing diffusion priors for faithful image super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2025b.

Liangyu Chen, Xin Lu, Jie Zhang, Xiaojie Chu, and Chengpeng Chen. Hinet: Half instance normalization network for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 182–192, 2021.

Xiangyu Chen, Xintao Wang, Wenlong Zhang, Xiangtao Kong, Yu Qiao, Jiantao Zhou, and Chao Dong. Hat: Hybrid attention transformer for image restoration. *arXiv preprint arXiv:2309.05239*, 2023.

Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. Rethinking coarse-to-fine approach in single image deblurring. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4641–4650, 2021.

Marcos V Conde, Gregor Geigle, and Radu Timofte. Instructir: High-quality image restoration following human instructions. In *European Conference on Computer Vision*, pp. 1–21. Springer, 2024.

Shuang Cui, Fan Ji, Guanglong Sun, Yufei Guo, Xiongxin Tang, Jiangmeng Li, and Fanjiang Xu. Self-evolving agentic image restoration via deliberate planning and intuitive execution. *arXiv preprint arXiv:2606.28971*, 2026.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

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, 2020.

Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV 13*, pp. 184–199. Springer, 2014.

Google. Gemini 3.7 flash (gemini-3.7-flash). `https://ai.google.dev/gemini-api/docs/models/gemini-3.7-flash`, Googlea.

Google. Nano banana 2 (gemini-3.1-flash-image). `https://ai.google.dev/gemini-api/docs/models/gemini-3.1-flash-image`, Googleb.

Qiyuan Guan, Xiang Chen, Guiyue Jin, Jiyu Jin, Shumin Fan, Tianyu Song, and Jinshan Pan. Rethinking nighttime image deraining via learnable color space transformation. *Advances in Neural Information Processing Systems*, 38:3189–3225, 2026.

Guoli Jia, Yisheng Zhang, Haote Hu, Shanxu Zhao, Kaikai Zhao, Long Sun, Xinwei Long, Kai Tian, Che Jiang, Zhaoxiang Liu, et al. Tir-agent: Training an explorative and efficient agent for image restoration. In *European Conference on Computer Vision*, pp. 523–540. Springer, 2026.

Xiang An Jack Yu Jia Guo, Jiankang Deng and Baris Gecer. Insightface: 2d and 3d face analysis project. `https://github.com/deepinsight/insightface`, 2026.

Xu Jiang, Gehui Li, Bin Chen, and Jian Zhang. Multi-agent image restoration. *International Journal of Computer Vision*, 134(5):205, 2026.

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*, pp. 340–359. Springer, 2024.

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*, pp. 5148–5157, 2021.

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.

Black Forest Labs. FLUX.2: Frontier Visual Intelligence. `https://bfl.ai/blog/flux-2`, 2025.

Bingchen Li, Xin Li, Yiting Lu, and Zhibo Chen. Hybrid agents for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22636–22647, 2026.

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, 2019.

Boyun Li, Xiao Liu, Peng Hu, Zhongqin Wu, Jiancheng Lv, and Xi Peng. All-In-One Image Restoration for Unknown Corruption. In *IEEE Conference on Computer Vision and Pattern Recognition*, New Orleans, LA, June 2022a.

Wei Li, Qiming Zhang, Jing Zhang, Zhen Huang, Xinmei Tian, and Dacheng Tao. Toward real-world single image deraining: A new benchmark and beyond. *arXiv preprint arXiv:2206.05514*, 2022b.

Jie Liang, Radu Timofte, Qiaosi Yi, Zhengqiang Zhang, Shuaizheng Liu, Lingchen Sun, Rongyuan Wu, Xindong Zhang, Hui Zeng, and Lei Zhang. Ntire 2025 the 2nd restore any image model (raim) in the wild challenge, 2025. URL https://arxiv.org/abs/2506.01394.

Beibei Lin, Yeying Jin, Yan Wending, Wei Ye, Yuan Yuan, and Robby T Tan. Nighthaze: Nighttime image dehazing via self-prior learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 5209–5217, 2025a.

Yunlong Lin, Zixu Lin, Haoyu Chen, Panwang Pan, Chenxin Li, Sixiang Chen, Kairun Wen, Yeying Jin, Wenbo Li, and Xinghao Ding. Jarvisir: Elevating autonomous driving perception with intelligent image restoration. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22369–22380. IEEE, 2025b.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 26296–26306, 2024.

Xiaohong Liu, Yongrui Ma, Zhihao Shi, and Jun Chen. Griddehazenet: Attention-based multi-scale network for image dehazing. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7314–7323, 2019.

Yun-Fu Liu, Da-Wei Jaw, Shih-Chia Huang, and Jenq-Neng Hwang. Desnownet: Context-aware deep network for snow removal. *IEEE Transactions on Image Processing*, 27(6):3064–3073, 2018.

Yuen Peng Loh and Chee Seng Chan. Getting to know low-light images with the exclusively dark dataset. *Computer Vision and Image Understanding*, 178:30–42, 2019. doi: https://doi.org/10.1016/j.cviu.2018.10.010.

Jianglin Lu, Yuanwei Wu, Ziyi Zhao, Hongcheng Wang, Felix Jimenez, Abrar Majeedi, and Yun Fu. Restore-r1: Efficient image restoration agents via reinforcement learning with multimodal llm perceptual feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8629–8639, 2026.

Ziwei Luo, Fredrik K Gustafsson, Zheng Zhao, Jens Sjölund, and Thomas B Schön. Photo-realistic image restoration in the wild with controlled vision-language models. *arXiv preprint arXiv:2404.09732*, 2024.

OpenAI. Gpt-image-2 model documentation. https://developers.openai.com/api/docs/models/gpt-image-2.5-sunburst, 2025.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G Patil, Ion Stoica, and Joseph E Gonzalez. Memgpt: Towards llms as operating systems. *arXiv preprint arXiv:2310.08560*, 2023.

Vaishnav Potlapalli, Syed Waqas Zamir, Salman Khan, and Fahad Shahbaz Khan. Promptir: Prompting for all-in-one blind image restoration. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Ruijie Quan, Xin Yu, Yuanzhi Liang, and Yi Yang. Removing raindrops and rain streaks in one go. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9147–9156, 2021.

Jaesung Rim, Haeyun Lee, Jucheol Won, and Sunghyun Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *European conference on computer vision*, pp. 184–201. Springer, 2020.

Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10745–10755. IEEE, 2021.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in neural information processing systems*, 36:68539–68551, 2023.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36:8634–8652, 2023.

Yuda Song, Zhuqing He, Hui Qian, and Xin Du. Vision transformers for single image dehazing. *arXiv preprint arXiv:2204.03883*, 2022.

Lingchen Sun, Rongyuan Wu, Xiangtao Kong, Jixin Zhao, Qiaosi Yi, Yujing Sun, Shuaizheng Liu, Zhengqiang Zhang, and Lei Zhang. Pixrestore: Unified image restoration via pixel diffusion transformer. *arXiv preprint arXiv:2608.16793*, 2026a.

Weixiong Sun, Xiang Yin, and Chao Dong. Can nano banana 2 replace traditional image restoration models? an evaluation of its performance on image restoration tasks. *arXiv preprint arXiv:2604.03061*, 2026b.

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*, volume 37, pp. 2555–2563, 2023.

Xingyao Wang, Boxuan Li, Yufan Song, Frank F Xu, Xiangru Tang, Mingchen Zhuge, Jiayi Pan, Yueqi Song, Bowen Li, Jaskirat Singh, et al. Openhands: An open platform for ai software developers as generalist agents. In *International Conference on Learning Representations*, volume 2025, pp. 65882–65919, 2025.

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.

Pengxu Wei, Ziwei Xie, Hannan Lu, Zongyuan Zhan, Qixiang Ye, Wangmeng Zuo, and Liang Lin. Component divide-and-conquer for real-world image super-resolution. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16*, pp. 101–117. Springer, 2020.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng, Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. Qwen-image technical report, 2025. URL https://arxiv.org/abs/2508.02324.

Tianhe Wu, Jian Zou, Jie Liang, Lei Zhang, and Kede Ma. Visualquality-r1: Reasoning-induced image quality assessment via reinforcement learning to rank. *Advances in Neural Information Processing Systems*, 38:88167–88190, 2026.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024.

J Xu, H Li, Z Liang, D Zhang, and L Zhang. Real-world noisy image denoising: A new benchmark. arxiv 2018. *arXiv preprint arXiv:1804.02603*.

John Yang, Carlos Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. Swe-agent: Agent-computer interfaces enable automated software engineering. *Advances in Neural Information Processing Systems*, 37:50528–50652, 2024a.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in neural information processing systems*, 37:21875–21911, 2024b.

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*, pp. 1191–1200, 2022.

Yufeng Yang, Xianfang Zeng, Zhangqi Jiang, Fukun Yin, Jianzhuang Liu, Wei Cheng, Shiyu Liu, Yuqi Peng, Gang YU, Shifeng Chen, et al. Realrestorer: Towards generalizable real-world image restoration with large-scale image editing models. *arXiv preprint arXiv:2603.25502*, 2026.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

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*, pp. 5728–5739, 2022.

Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing*, 26 (7):3142–3155, 2017.

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*, pp. 586–595, 2018.

Yubo Zhang, Xueqing Wang, Manhui Lin, Yue Zhang, Penglongyi Deng, Ting Sun, Tingquan Gao, Zelun Zhang, Jiaxuan Liu, Changda Zhou, et al. Pp-ocrv6: From 1.5 m to 34.5 m parameters, surpassing billion-scale vlms on ocr tasks. *arXiv preprint arXiv:2606.13108*, 2026.

Yingjie Zhou, Jiezhang Cao, Farong Wen, Zicheng Zhang, Yu Zhou, Yue Shi, Xiaohong Liu, Radu Timofte, Luc Van Gool, and Guangtao Zhai. Q-agent: Quality-driven chain-of-thought image restoration agent through robust multimodal large language model. *arXiv preprint arXiv:2504.07148*, 2025.

Feng Zhu, Shuyang Xie, Yihan Zeng, Ming Liu, and Wangmeng Zuo. Opera: An agent for image restoration with end-to-end joint planning-execution optimization. *arXiv preprint arXiv:2605.22104*, 2026.

Kaiwen Zhu, Jinjin Gu, Zhiyuan You, Yu Qiao, and Chao Dong. An intelligent agentic system for complex image restoration problems. In *International Conference on Learning Representations*, volume 2025, pp. 57985–58013, 2025.

Yushen Zuo, Qi Zheng, Mingyang Wu, Xinrui Jiang, Renjie Li, Jian Wang, Yide Zhang, Gengchen Mai, Lihong V Wang, James Zou, et al. 4kagent: agentic any image to 4k super-resolution. *arXiv preprint arXiv:2507.07105*, 2025.

Appendix

In this appendix, we provide the following materials:

- **A. Input Request**: The fixed request template for each degradation category to represent the usage of general users.
- **B. Real Test Set Composition**: Detailed construction and data sources of Real-Paired-200 and Real-NoGT-200.
- **C. DF-Score Evaluation Protocol**: The evaluator prompt, alignment with human judgments, and evaluator-VLM replacement experiments.
- **D. Detailed Results on MiO100**: Per-group results for MiO100 Groups A, B, and C.
- **E. Ablation Studies**: Ablations of executor inputs, refinement strategy, and the VLM used by HarnessIR.
- **F. Cost Analysis**: API calls, runtime, local tool invocations, and monetary cost for HarnessIR and the compared agentic methods.
- **G. User Study**: A user study in which participants evaluate the visual quality of different methods.
- **H. More Visual Results**: Additional qualitative comparisons between HarnessIR and other representative methods.

A Input Request

We use short, fixed requests to represent the intent of a general user. The request for *mix* serves as the default. When one or more degradation types are specified, we prepend a removal instruction to the same default request. For mixed degradations, the names of all applicable degradations are joined with "and" or followed in the end. For example, an image containing both haze and rain needs to be brightened is assigned the request "Please remove the haze and rain from the image and restore this low-quality image to a clear, clean, and natural state. Meanwhile, recover its normal brightness". The basic requests are as follows:

```
### [[mix]] ###
Restore this low-quality image to a clear, clean and natural state.

### [[haze]] ###
Please remove the haze from the image and restore this low-quality image to a clear, clean
    ↪ and natural state.

### [[rain]] ###
Please remove the rain from the image and restore this low-quality image to a clear, clean
    ↪ and natural state.

### [[snow]] ###
Please remove the snow from the image and restore this low-quality image to a clear, clean
    ↪ and natural state.

### [[lowlight]] ###
Restore this low-quality image to a clear, clean and natural state and recover its normal
    ↪ brightness.
```

B Real Test Set Composition

We construct two real-world test sets from publicly available datasets: Real-Paired-200, which contains 200 paired low-quality and ground-truth images, and Real-NoGT-200, which contains 200 real-world low-quality images without ground-truth. The source datasets and the number of selected images are summarized in Tab. B.1. We randomly sample images from each source dataset; the number selected from each is given in parentheses in Tab. B.1. For consistency across datasets, images whose longer side is shorter than 1024 pixels are isotropically resized so that their longer side becomes 1024 pixels. Images whose longer side exceeds 1024 pixels are center-cropped along the longer dimension until it is 1024 pixels, while the shorter dimension is left unchanged; images

Table B.1: Source datasets of the two real-world test sets, with the number of images contributed by each in parentheses.

Degradation	Source datasets (images)	Total
Real-Paired-200 — 200 LQ/GT pairs		
Low resolution	DRealSR (20) (Wei et al., 2020), RealSR (20) (Cai et al., 2019)	40
Haze	I-HAZE (20) (Ancuti et al., 2018), NH-HAZE (20) (Ancuti et al., 2020)	40
Rain	RealRain-1k-H (20) (Li et al., 2022b), RainDS-real (20) (Quan et al., 2021)	40
Blur	RealBlur (20) (Rim et al., 2020)	20
Sensor noise	SIDD (20) (Abdelhamed et al., 2018), PolyU (20) (Xu et al.)	40
Mixed, in the wild	RAIM (20) (Liang et al., 2025)	20
Real-NoGT-200 — 200 LQ images, no reference		
Mixed, no named weather	RealDeg (classic film (25),old photo(25) and social media(25)) (Chen et al., 2025b), ImageNet-worst (25) (Deng et al., 2009)	100
Haze	ACDC-fog (7) (Sakaridis et al., 2021), JarvisIR-fog (6) (Lin et al., 2025b), RESIDE-RTTS (6) (Li et al., 2019), NightHaze (6) (Lin et al., 2025a)	25
Rain	ACDC-rain (7) (Sakaridis et al., 2021), HQ-NightRain-Real (6) (Guan et al., 2026), JarvisIR-rain (6) (Lin et al., 2025b), NTURain-real (6) (Chen et al., 2018)	25
Snow	ACDC-snow (9) (Sakaridis et al., 2021), JarvisIR-snow (8) (Lin et al., 2025b), Snow100K-realistic (8) (Liu et al., 2018)	25
Low light	ExDark (9) (Loh & Chan, 2019), ACDC-night (8) (Sakaridis et al., 2021), JarvisIR-night (8) (Lin et al., 2025b)	25

whose longer side is already 1024 pixels are left unchanged. For paired samples, the corresponding ground-truth image undergoes the matching resize or center-crop operation so that it has the same final spatial resolution and remains pixel-aligned with the low-quality image.

C DF-SCORE EVALUATION PROTOCOL

This section describes the evaluation protocol and reliability analysis of DF-Score. We first provide the complete system prompt used by the independent VLM evaluator to assign D-Score and F-Score. We then assess the alignment of DF-Score and representative no-reference IQA metrics with human judgments through a pairwise study. Finally, we test whether D-Score, F-Score, and DF-Score remain consistent across VLM evaluators from different model families.

System Prompt for DF-Score Evaluator. Since the system prompt is several pages long, we have placed it in the separate **supplementary material**.

Alignment with Human Judgments. We evaluate how well DF-Score and commonly used NR-IQA metrics agree with human judgments of restoration quality. We construct 300 comparison pairs from the Real-NoGT-200 test set. Each pair contains one LQ image and two restorations of that image produced by distinct methods. The method pool includes PromptIR (Potlapalli et al., 2023), InstructIR (Conde et al., 2024), 4KAgent (Zuo et al., 2025), AgenticIR (Zhu et al., 2025), GPT-Image 2.5 (OpenAI, 2025), Nano Banana 2 (Google, Googleb), HarnessIR-GPT, and HarnessIR-NB2. For each pair, two methods are sampled uniformly from this pool. The left–right order of the restorations is randomized, and method identities and metric scores are hidden from the annotators.

We recruit 30 annotators. For each pair, annotators are shown the LQ image together with the two restorations and asked to select the better result. The instructions emphasize that a preferable restoration should remove the target degradation, preserve the content and structure of the input image, and avoid visible artifacts. Each annotator provides one forced-choice judgment per pair.

We aggregate the annotations by majority vote to obtain the human preference for each pair. Pairs without a strict majority are discarded. For a metric m, we measure its agreement with human judgments using the alignment ratio:

$$\mathrm{Align}(m) = \frac{\#\{\text{valid pairs whose metric ordering agrees with the majority vote}\}}{\#\{\text{valid pairs}\}}.$$

A random ranking gives an expected alignment of 50%. We report the alignment of DF-Score and four representative NR-IQA metrics—MUSIQ, AFINE-NR, TOPIQ, and MANIQA—in Fig. C.1. DF-Score agrees with human preference in 87% of valid comparisons, whereas the NR-IQA metrics achieve lower alignment rates, ranging from 50.8% to 67.1%. This suggests that, when ground-truth images are unavailable, DF-Score may better reflect human preferences than the NR-IQA metrics.

Consistency Across Evaluator Families. We examine whether D-Score, F-Score, and their geometric mean, DF-Score, depend strongly on the choice of VLM evaluator. Using the restored outputs for the 400 images in Real-Paired-200 and Real-NoGT-200, we obtain scores from GPT-5.6-Sol and Claude-Opus-5 under the same evaluation protocol and compute the Pearson linear correlation coefficient (PLCC) between each evaluator's scores and those of Gemini-3.7-Flash. Tab. C.2 reports the correlations for all three scores. Despite the presence of some outliers, the correlations are consis-

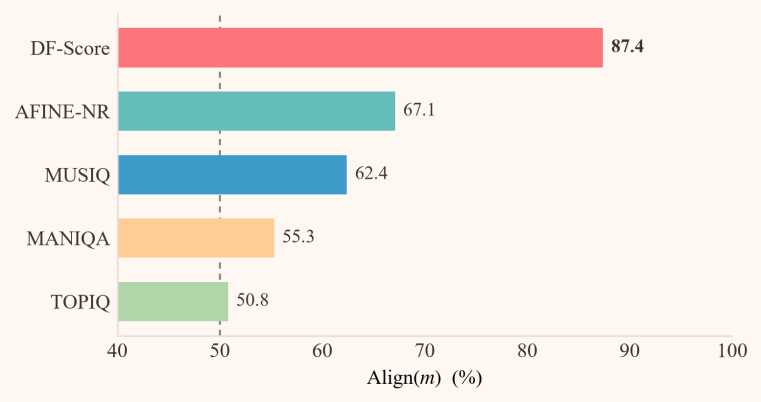

Figure C.1: Pairwise $\mathrm{Align}(m)$ with majority human preference on the real no-GT test set. Dashed line: chance (50%).

Table C.2: PLCC of D-Score, F-Score, and DF-Score between each alternative VLM evaluator and Gemini-3.7-Flash on restored outputs for the 400 images in Real-Paired-200 and Real-NoGT-200.

Evaluator	PLCC vs. Gemini-3.7-Flash		
	D-Score	F-Score	DF-Score
GPT-5.6-Sol	0.85	0.87	0.94
Claude-Opus-5	0.87	0.89	0.92

tently high across all three measures, ranging from 0.85 to 0.94. This result suggests that DF-Score is robust to the choice of VLM evaluator. Assessing whether degradations have been removed and whether the original image content has been preserved is comparatively straightforward, and different VLMs appear capable of making these judgments consistently, even though their absolute scores may differ.

D Detailed Results on MiO100

Tab. D.3 shows the detailed quantitative results for each group of MiO100. Restoration becomes more challenging as the simulated degradations become more complex. Each of Groups A and B combines two degradations, while Group C combines three, resulting in more severe and interacting distortions. Accordingly, most methods show lower PSNR, SSIM, and DF-Score for Group C. For example, when using Nano Banana 2 directly, the DF-Score falls from 61.75 in Group A and 61.00 in Group B to 44.85 in Group C.

HarnessIR improves both executors in all three groups. For NB2, it raises the DF-Score to 65.47, 65.22, and 49.12 for Groups A, B, and C, respectively. The gains over direct GPT-Image-2.5 are larger for Groups A and B, at 27.29 and 20.85 points, and smaller for Group C, at 6.05 points. For all the three groups, both HarnessIR variants achieve higher DF-Scores than the previous methods. Overall, the results show that HarnessIR's advantage persists across different degradation levels.

E Ablation Studies

E.1 Ablation of Executor Inputs

We first explore how information is prepared for and supplied to the MFM executor on Real-Paired-200. The settings vary depending on the user request, image-specific diagnosis, auxiliary-tool evidence, and whether that evidence is supplied as visual outputs or expressed in text. We use *executor inputs* to refer to this information; the LQ image is provided in every setting. To isolate their effects, all variants use a single execution round, without verification and re-execution. Tab. E.4 compares five settings. In the baseline, the executor receives only the LQ image x and the user's request r.

Table D.3: Per-group results on MiO100, for which Tab. 1 reports the average. Marking and metrics follow Tab. 1.

Datasets	Method	Full-reference Metrics				No-reference Metrics					VLM-based IQA		
		PSNR↑	SSIM↑	LPIPS↓	DISTS↓	ΔMANIQA↓	ΔCLIP-IQA↓	ΔMUSIQ↓	ΔTOPIQ↓	ΔAFINE-NR↓	D-Score↑	F-Score↑	DF-Score↑
Group A	AirNet	20.70	0.66	0.43	0.24	0.19	0.27	26.37	0.31	0.25	20.6	88.3	22.42
	PromptIR	21.29	0.67	0.43	0.25	0.19	0.26	26.14	0.31	0.24	21.6	92.9	24.57
	MiOIR	21.74	0.69	0.39	0.23	0.19	0.27	23.93	0.30	0.23	27.5	89.1	35.04
	DA-CLIP	20.87	0.65	0.42	0.24	0.19	0.26	24.94	0.30	0.24	20.4	92.9	25.87
	InstructIR	21.52	0.64	0.47	0.25	0.21	0.31	27.86	0.34	0.27	24.4	87.9	31.96
	AutoDIR	21.04	0.67	0.37	0.22	0.12	0.24	15.18	0.24	0.14	40.1	66.2	38.11
	AgenticIR	21.53	0.70	0.31	0.19	0.11	0.17	8.80	0.17	0.07	54.8	47.1	48.09
	MAIR	21.02	0.67	0.30	-	0.31	0.17	9.47	-	-	-	-	-
	4KAgent	21.48	0.67	0.30	0.18	0.08	0.10	5.47	0.11	0.08	49.4	32.7	36.78
	TIR-Agent	22.07	0.69	0.29	-	0.25	0.10	4.97	-	-	-	-	-
	SEAR	21.80	0.70	0.30	-	0.30	0.15	7.37	-	-	-	-	-
	OPERA	22.32	0.71	0.33	-	0.29	0.20	9.32	-	-	-	-	-
	GPT-Image-2.5	17.48	0.53	0.37	0.20	0.02	0.11	4.78	0.07	0.00	42.8	20.1	27.52
	HarnessIR-GPT	**20.19**	**0.61**	**0.28**	**0.16**	**0.02**	**0.06**	**2.87**	**0.02**	**0.00**	**71.2**	**45.5**	**54.81**
	Nano Banana 2	21.38	0.66	0.22	0.13	0.00	0.04	0.35	0.04	0.01	80.2	51.4	61.75
	HarnessIR-NB2	**22.45**	**0.67**	**0.20**	**0.12**	**0.00**	**0.03**	**0.31**	**0.04**	**0.00**	78.8	**58.5**	**65.47**
	GT	-	-	-	-	0.64	0.65	68.66	0.64	-0.91	-	-	-
	LQ	20.20	0.61	0.50	0.28	0.21	0.28	29.06	0.34	0.27			
Group B	AirNet	19.97	0.70	0.38	0.22	0.16	0.23	21.46	0.25	0.20	28.8	83.1	33.91
	PromptIR	20.38	0.71	0.36	0.22	0.15	0.23	21.42	0.25	0.19	31.6	88.1	35.52
	MiOIR	21.19	0.72	0.34	0.21	0.17	0.22	20.66	0.27	0.21	39.0	90.5	48.97
	DA-CLIP	20.76	0.69	0.38	0.23	0.16	0.26	20.91	0.27	0.16	23.8	80.2	29.74
	InstructIR	20.37	0.62	0.51	0.27	0.19	0.27	26.93	0.32	0.27	13.2	90.4	22.54
	AutoDIR	21.66	0.72	0.32	0.20	0.10	0.21	13.30	0.22	0.13	42.5	76.1	45.03
	AgenticIR	21.72	0.72	0.30	0.19	0.11	0.16	9.28	0.16	0.05	53.4	47.2	46.96
	MAIR	20.92	0.70	0.28	-	0.29	0.14	7.87	-	-	-	-	-
	4KAgent	20.95	0.67	0.30	0.18	0.07	0.10	6.16	0.11	0.09	51.9	45.2	44.09
	TIR-Agent	22.80	0.73	0.27	-	0.24	0.10	3.61	-	-	-	-	-
	SEAR	22.13	0.73	0.29	-	0.30	0.14	8.20	-	-	-	-	-
	OPERA	22.28	0.75	0.31	-	0.31	0.19	9.75	-	-	-	-	-
	GPT-Image-2.5	18.42	0.56	0.34	0.19	0.02	0.11	4.65	0.07	0.02	45.4	22.1	30.42
	HarnessIR-GPT	**19.96**	**0.60**	**0.28**	**0.16**	**0.02**	**0.08**	**3.36**	**0.03**	**0.01**	**69.6**	**40.2**	**51.27**
	Nano Banana 2	21.23	0.66	0.22	0.12	0.00	0.03	0.67	0.04	0.01	79.9	50.2	61.00
	HarnessIR-NB2	**22.94**	**0.68**	**0.19**	**0.11**	**0.00**	**0.03**	**0.14**	**0.04**	**0.00**	78.1	**58.2**	**65.22**
	GT	-	-	-	-	0.64	0.65	68.85	0.64	-0.92	-	-	-
	LQ	20.64	0.60	0.54	0.29	0.19	0.26	27.62	0.33	0.25			
Group C	AirNet	20.22	0.59	0.57	0.30	0.30	0.35	39.18	0.41	0.35	26.0	97.6	33.00
	PromptIR	20.30	0.59	0.57	0.30	0.30	0.35	39.39	0.42	0.35	26.0	94.1	31.15
	MiOIR	19.36	0.58	0.56	0.30	0.30	0.37	37.29	0.43	0.34	18.9	92.2	27.25
	DA-CLIP	19.35	0.57	0.58	0.31	0.31	0.36	39.27	0.43	0.34	14.1	89.9	20.81
	InstructIR	20.63	0.55	0.64	0.32	0.31	0.35	40.73	0.44	0.41	28.2	93.4	33.76
	AutoDIR	19.06	0.57	0.47	0.27	0.19	0.30	22.29	0.30	0.19	34.2	61.6	34.00
	AgenticIR	20.20	0.59	0.43	0.25	0.20	0.20	17.10	0.24	0.10	39.9	35.6	33.87
	MAIR	19.42	0.55	0.41	-	0.36	0.23	17.49	-	-	-	-	-
	4KAgent	19.77	0.56	0.43	0.24	0.14	0.13	13.29	0.15	0.16	40.1	31.1	29.78
	TIR-Agent	19.83	0.56	0.41	-	0.27	0.10	11.91	-	-	-	-	-
	SEAR	20.61	0.59	0.42	-	0.37	0.21	16.12	-	-	-	-	-
	OPERA	20.75	0.61	0.47	-	0.36	0.28	18.94	-	-	-	-	-
	GPT-Image-2.5	17.96	0.52	0.36	0.20	0.02	0.09	4.48	0.05	0.02	58.2	20.5	33.20
	HarnessIR-GPT	**18.95**	**0.54**	**0.34**	**0.18**	**0.00**	**0.07**	**2.21**	**0.02**	0.03	**61.8**	**27.4**	**39.25**
	Nano Banana 2	19.81	0.58	0.28	0.15	0.00	0.03	1.07	0.04	0.01	70.0	33.0	44.85
	HarnessIR-NB2	**21.23**	**0.60**	**0.25**	**0.14**	0.01	**0.03**	**0.29**	**0.04**	**0.00**	68.8	**38.4**	**49.12**
	GT	-	-	-	-	0.64	0.65	68.85	0.64	-0.92	-	-	-
	LQ	18.73	0.50	0.69	0.35	0.29	0.31	41.64	0.44	0.41			

Detailed User Request replaces the short request with a more explicit paragraph that a user could plausibly provide, but adds no image-specific analysis. *Single-VLM Instruction* uses one VLM to diagnose the image and compose the instruction without using any auxiliary tools. The final two settings follow HarnessIR's use of separate VLM stages and auxiliary tools. In *Visual Evidence*, the executor also receives the tools' original outputs, including the depth map and semantic segmentation map. In *Text Evidence*, the relevant evidence is composed into language by the composer, which matches the executor-input setting used by HarnessIR.

The results show a steady improvement as progressively more information is supplied to the executor. Expanding the user request alone raises PSNR from 23.94 to 24.75 dB and DF-Score from 52.9 to 54.5. Image-specific analysis by a single VLM further increases the DF-Score to 56.6. Adding auxiliary tools yields further gains, particularly in degradation removal: the visual-evidence setting achieves a D-Score of 74.4 and a DF-Score of 57.8. However, giving the executor visual maps directly can introduce artifacts, such as semantic-label colors appearing in the restored image. Expressing the relevant evidence in text avoids this risk and suits the executor's language-based interface. This setting achieves the strongest one-round result, with a PSNR of 26.01 dB and a DF-Score of 59.6. Overall, the results suggest that richer requests, image-specific analysis, auxil-

Table E.4: Ablation of executor inputs on Real-Paired-200. All variants use one execution round, without re-execution to isolate the effect of how information is supplied to the executor. The **bold** row uses the executor-input setting adopted by HarnessIR. Metrics follow Tab. 1; the best and second-best results in each column are marked in red and blue.

Executor-input configuration	Full-reference Metrics				No-reference Metrics					VLM-based IQA		
	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	ΔMANIQA↓	ΔCLIP-IQA↓	ΔMUSIQ↓	ΔTOPIQ↓	ΔAFINE-NR↓	D-Score↑	F-Score↑	DF-Score↑
Direct MFM (Baseline)	23.94	0.729	0.309	0.181	0.058	0.077	11.91	0.118	0.074	67.9	47.8	52.9
Detailed User Request	24.75	0.733	0.298	0.177	0.054	0.069	10.88	0.105	0.065	71.4	50.9	54.5
Single-VLM Instruction	25.49	0.761	0.263	0.161	0.042	0.044	9.30	0.080	0.057	72.5	54.2	56.6
Two VLMs + Tools (Visual Evidence)	25.62	0.758	0.267	0.161	0.026	0.032	7.39	0.057	0.049	74.4	55.3	57.8
Two VLMs + Tools (Text Evidence)	26.01	0.763	0.261	0.158	0.032	0.024	7.72	0.062	0.045	73.0	58.5	59.6
GT	-	-	-	-	0.560	0.451	54.36	0.417	-0.901	-	-	-

Table E.5: Ablation of refinement strategies on Real-Paired-200. We compare three settings: *HarnessIR (Single-pass)*, which performs a single MFM execution without re-execution; *HarnessIR (In-place refinement)*, which applies a revised instruction to the output of the previous round; and *HarnessIR (Redo from LQ)* which applies a revised instruction to original LQ input. Metrics follow Tab. 1, the best and second-best results are marked in red and blue.

Refinement strategy	Full-reference Metrics				No-reference Metrics					VLM-based IQA		
	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	ΔMANIQA↓	ΔCLIP-IQA↓	ΔMUSIQ↓	ΔTOPIQ↓	ΔAFINE-NR↓	D-Score↑	F-Score↑	DF-Score↑
HarnessIR (Single-pass)	26.01	0.763	0.261	0.158	0.032	0.024	7.72	0.062	0.045	73.0	58.5	59.6
HarnessIR (In-place refinement)	25.64	0.760	0.265	0.159	0.020	0.016	6.79	0.052	0.032	71.8	55.6	57.2
HarnessIR (Redo from LQ)	26.63	0.775	0.246	0.149	0.020	0.001	5.52	0.028	0.033	75.2	63.6	62.0
GT	-	-	-	-	0.560	0.451	54.36	0.417	-0.901	-	-	-

iary evidence, and evidence-grounded instruction writing each contribute to better restoration and content preservation.

These gains require only a modest increase in inference cost. As shown in Tab. F.7, adding one VLM call raises the cost from $0.070 for direct MFM usage to $0.077. Using two VLM calls and auxiliary tools costs $0.085. Thus, the improvements are achieved without a prohibitively expensive control loop, offering a favorable trade-off between restoration quality and cost.

E.2 Redo vs. In-place Repair During Re-execution

We compare two refinement strategies with the single-pass version in Tab. E.5. *HarnessIR (Single-pass)* performs one execution without re-execution and serves as the reference setting. *HarnessIR (In-place refinement)* feeds the output of the previous execution back to the executor for further correction, following the sequential refinement strategy of previous agent IR works (Zhu et al., 2025; Jiang et al., 2026). In contrast, *HarnessIR (Redo from LQ)* returns to the original LQ image when refinement is triggered, composes a revised instruction, and generates a new candidate from scratch, which corresponds to the final HarnessIR design described in Sec. 3.4.

The results support restarting from the original input rather than refining an already altered image. In-place refinement is inferior even to the single-pass setting, with the DF-Score decreasing from 59.6 to 57.2. In contrast, redo from the LQ input improves PSNR from 26.01 to 26.63 dB and raises the D-Score/F-Score from 73.0/58.5 to 75.2/63.6, resulting in a DF-Score of 62.0. This suggests that an unsuccessful intermediate result may contain hallucinated or distorted content that should not be propagated into the next round. Restarting from the LQ input allows the revised instruction to correct the failure without accumulating errors. Importantly, this improvement is not uniform across all samples: only 37% of the 200 test images undergo re-execution, and the gains come from this subset. Images that already produce satisfactory results in the first round remain unchanged. Thus, redo primarily repairs difficult tail cases rather than improving every image uniformly. Detailed statistics on the re-execution frequency and its associated cost are provided in Appendix F.

E.3 VLM Choice

We compare different VLMs for HarnessIR's diagnoser and composer on Real-Paired-200. Verification-driven re-execution is disabled in all variants, allowing us to compare the VLMs without gains from additional execution rounds.

Table E.6: Effect of the VLMs used for diagnoser and composer on Real-Paired-200. Verification-driven re-execution is disabled in all variants. Metrics follow Tab. 1; the best and second-best results in each column are marked in red and blue.

VLM	Full-reference Metrics				No-reference Metrics					VLM-based IQA		
	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	ΔMANIQA↓	ΔCLIP-IQA↓	ΔMUSIQ↓	ΔTOPIQ↓	ΔAFINE-NR↓	D-Score↑	F-Score↑	DF-Score↑
HarnessIR (Qwen3-VL-32B)	25.68	0.760	0.267	0.161	0.034	0.024	7.82	0.067	0.050	71.2	56.1	57.5
HarnessIR (GPT-5.6-Sol)	26.18	0.768	0.253	0.151	0.029	0.017	6.48	0.058	0.048	74.4	59.7	60.5
HarnessIR (GPT-5.6-Luna)	26.05	0.762	0.259	0.155	0.034	0.027	7.78	0.066	0.059	72.7	58.7	59.1
HarnessIR (Gemini-3.7-Flash)	26.01	0.763	0.261	0.158	0.032	0.024	7.72	0.062	0.045	73.0	58.5	59.6
GT	-	-	-	-	0.560	0.451	54.36	0.417	-0.901	-	-	-

Table F.7: Average per-image runtime, call counts, restoration performance, and cost on Real-Paired-200. VLM and MFM denote API calls. Tool / Perception: Tool reports local model used as restoration or auxiliary tools, perception reports local VLM model used for perception.

Method	VLM	MFM	Tool / Perception	Time (s)	PSNR↑	DF-Score↑	Cost ($)
AgenticIR	0.70	–	5.94 / 9.87	88.8	22.41	38.8	0.137
4KAgent	0.53	–	7.47 / 10.08	125.3	22.18	37.4	0.190
Baseline (MFM only)	–	1.00	–	14.7	23.94	52.9	0.070
HarnessIR (One VLM call)	1.00	1.00	–	27.2	25.49	56.6	0.077
HarnessIR (Two VLM calls + Tools)	2.00	1.00	2.40 / –	41.8	26.01	59.6	0.085
HarnessIR (Two VLM calls + Tools, GPT-5.6-Sol)	2.00	1.00	2.40 / –	45.5	26.18	60.5	0.183
HarnessIR (Final, Three VLM calls + Tools + redo)	3.90	1.45	2.40 / –	64.3	26.63	62.0	0.129

The results suggest that HarnessIR does not depend on a specific VLM, provided the model is sufficiently capable. The locally deployed Qwen3-VL-32B performs somewhat worse, reaching a DF-Score of 57.5. GPT-5.6-Luna and Gemini-3.7-Flash perform similarly, with DF-Scores of 59.1 and 59.6, respectively. GPT-5.6-Sol achieves the highest PSNR and DF-scores, but its gains over Gemini-3.7-Flash are modest: 0.17 dB in PSNR and 0.9 points in DF-Score. We therefore use Gemini-3.7-Flash in the final system: its restoration performance is comparable, while GPT-5.6-Sol incurs substantially higher API cost (see Appendix F).

F Cost Analysis

As shown in Tab. F.7, we report the per-image runtime, call counts, restoration performance, and cost of different HarnessIR variants and two previous agentic IR methods (Zhu et al., 2025; Zuo et al., 2025) on Real-Paired-200. Runtime and VLM call counts come from actual runs. For local computation, including the prior methods' functional modules and HarnessIR's auxiliary tools, we convert measured runtime into cost using on-demand A100 GPU rental rates; API costs are estimated from observed token usage and public list prices. We exclude costs that are difficult to assign to individual images, such as parallel waiting, environment setup, and deployment overhead. Rental rates provide the fairest method-independent basis for comparison, whereas GPU purchase costs depend on assumptions about depreciation, utilization, service lifetime, and allocation, making per-image costs ill-defined and incomparable. The resulting values should therefore be viewed as **idealized references** for computational and monetary efficiency. We discuss the results first and detail the calculations afterward.

Results Analysis. Compared with the two previous agentic IR methods, the final HarnessIR system achieves a higher DF-Score on Real-Paired-200 at a lower estimated per-image cost: $0.129, compared with $0.137 for AgenticIR and $0.190 for 4KAgent. For the HarnessIR variants, starting from direct MFM usage, adding a single VLM call raises the DF-Score from 52.9 to 56.6 and the PSNR from 23.94 to 25.49 dB, while increasing the cost only from $0.070 to $0.077 per image. This one-call variant requires no local GPU computation, making it a particularly attractive option when simplicity and deployment cost are priorities. Adding a second VLM call and local auxiliary tools raises the cost modestly to $0.085, while further improving the DF-Score to 59.6 and the PSNR to 26.01 dB. Thus, the diagnosis-and-composition pipeline provides a favorable additional quality gain for a small increase in cost.

The complete HarnessIR system adds verification and selective re-execution, reaching a DF-Score of 62.0 and a PSNR of 26.63 dB at an average cost of $0.129 per image. This average gain is driven by the subset of difficult cases that trigger re-execution rather than by uniformly improving every image: 37% of images undergo at least one redo, and the verifier retains the better candidate. The increased cost, therefore, supports targeted correction of cases that need it. Finally, replacing Gemini-3.7-Flash with GPT-5.6-Sol in the two-VLM configuration increases the cost from $0.085 to $0.183 per image. Although GPT-5.6-Sol yields some performance improvement, the gain is modest relative to the more than twofold cost increase. We therefore use Gemini-3.7-Flash in the final system to balance performance and cost.

Calculation Details. The GPU price is the mean on-demand A100 rate of the three major cloud providers, Google Cloud [1], Microsoft Azure [2], and Amazon Web Services[3], normalized to a per-GPU hourly rate: $2.671 per GPU-hour. All locally executed calls, including local restoration, perception, and auxiliary-tool calls, are converted to GPU-rental cost using this rate. Following the deployment setup of 4KAgent, both AgenticIR and 4KAgent use two GPUs: one GPU is dedicated to DepictQA, while the other runs the largest model so that it remains within the available GPU memory. For AgenticIR and 4KAgent, call counts and runtimes are averaged over Real-Paired-200.

We calculate API costs using public list prices, retrieved in September 2026, without cached-input or batch discounts.[4] The relevant prices are reported as input/output rates per million tokens: Nano Banana 2 uses in/out rates of $0.50/$60 for image output and $0.50/$3 for text output; Gemini-3.7-Flash uses an in/out rate of $0.75/$3.75; and GPT-5.6-Sol uses an in/out rate of $4/$20. Thus, GPT-5.6-Sol is more than five times as expensive as Gemini-3.7-Flash for both input and output tokens. For HarnessIR, API costs are calculated from the actual input and output usage of the VLM and MFM calls, while local auxiliary tools are accounted for through GPU-rental time. For the HarnessIR-NB2 configurations, one Gemini-3.7-Flash VLM call costs approximately $0.007 (different under different token counts), one MFM call costs approximately $0.070, and one complete auxiliary-tool suite costs approximately $0.0048. In the final system, every image undergoes diagnosis, composition, execution, and verification. Of the 200 images, 29% trigger one redo and 8% trigger two redos; thus, 37% trigger at least one redo, corresponding to an average of 0.45 redo rounds per image. Each redo adds two VLM calls and one MFM call, yielding 3.90 VLM calls and 1.45 MFM calls per image on average. The costs reported in Tab. F.7 are computed from unrounded per-image token usage and runtime measurements.

G User Study

We conduct a user study to compare restoration quality on real images. For each test case, annotators see one low-quality (LQ) image together with the eight restorations of that image. The eight methods are the same set used in the DF-Score comparison: PromptIR (Potlapalli et al., 2023), InstructIR (Conde et al., 2024), 4KAgent (Zuo et al., 2025), AgenticIR (Zhu et al., 2025), GPT-Image 2.5 (OpenAI, 2025), Nano Banana 2 (Google, Googleb), HarnessIR-GPT and HarnessIR-NB2. The eight results are shown in random order. Method names and metric values are withheld as well. Annotators are required to select the single restoration they judge to be the best. The instruction asks them to prefer a result that removes the target degradation, preserves the main scene content, and avoids obvious artifacts.

We draw 100 images at random from the Real-Paired-200 and 100 images at random from Real-NoGT-200, giving 200 groups in total. Each of the 25 participants annotates all 200 groups, so each group receives 25 independent votes and the study contains $25 \times 200 = 5000$ votes.

The selection rate of a method is the percentage of the 5000 votes in which annotators choose that method as the best restoration. Fig. G.2 shows a clear preference for the harnessed restorations.

[1] GCP a2-ultragpu-1g: https://cloud.google.com/products/compute/pricing/accelerator-optimized#accelerator-optimized-machine-type-family

[2] Azure ND96asr_A100_v4: https://azure.microsoft.com/pricing/details/virtual-machines/linux/

[3] AWS p4de.24xlarge: https://aws.amazon.com/cn/savingsplans/compute-pricing/

[4] Gemini and Nano Banana 2 pricing: https://ai.google.dev/gemini-api/docs/pricing. GPT-5.6-Sol pricing: https://openai.com/api/pricing/.

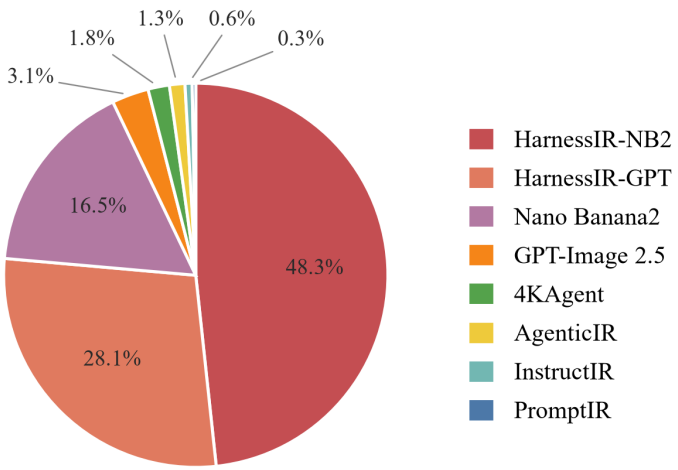

Figure G.2: Human selection rates among our methods and baselines.

HarnessIR-NB2 receives 48.3% of the votes and HarnessIR-GPT receives 28.1%, so the two HarnessIR variants together account for 76.4%. These human preferences are consistent with the quantitative results and qualitative comparisons presented above. They further confirm that HarnessIR substantially improves restoration quality while maintaining high perceptual quality and faithfully preserving the original content of the LQ images.

H More Visual Results

To further demonstrate the visual quality of HarnessIR, Fig. H.3 presents additional comparisons on the three test sets. Compared with direct MFM usage, HarnessIR restores facial details more faithfully and maintains a consistent color appearance, as illustrated by the first and fifth examples. It also recovers text while avoiding the spatial shifts and content changes observed in the direct MFM results, particularly in the second and third examples. For rainy and snowy scenes, HarnessIR removes the rain or snow that obstructs visibility while preserving the structure and appearance of other scene elements. For instance, it does not alter the shape of trees or hallucinate leaves in the fourth example of Fig. H.3.

The comparisons also reveal several limitations of previous methods on real-world images. Existing all-in-one and agentic methods often provide only a limited visible improvement on such inputs. InstructIR frequently brightens scenes even when low-light enhancement is not requested, as shown in the third and fifth examples. 4KAgent also suffers from a severe failure in the third example, producing a substantially corrupted result. This behavior illustrates a limitation of sequentially applying single-degradation restoration models: processing one degradation can change the distribution of the remaining degradations, causing the next model in the sequence to encounter an out-of-distribution input and fail. HarnessIR avoids this error-propagation problem by handling mixed degradations in a unified MFM execution while applying a redo refinement strategy.

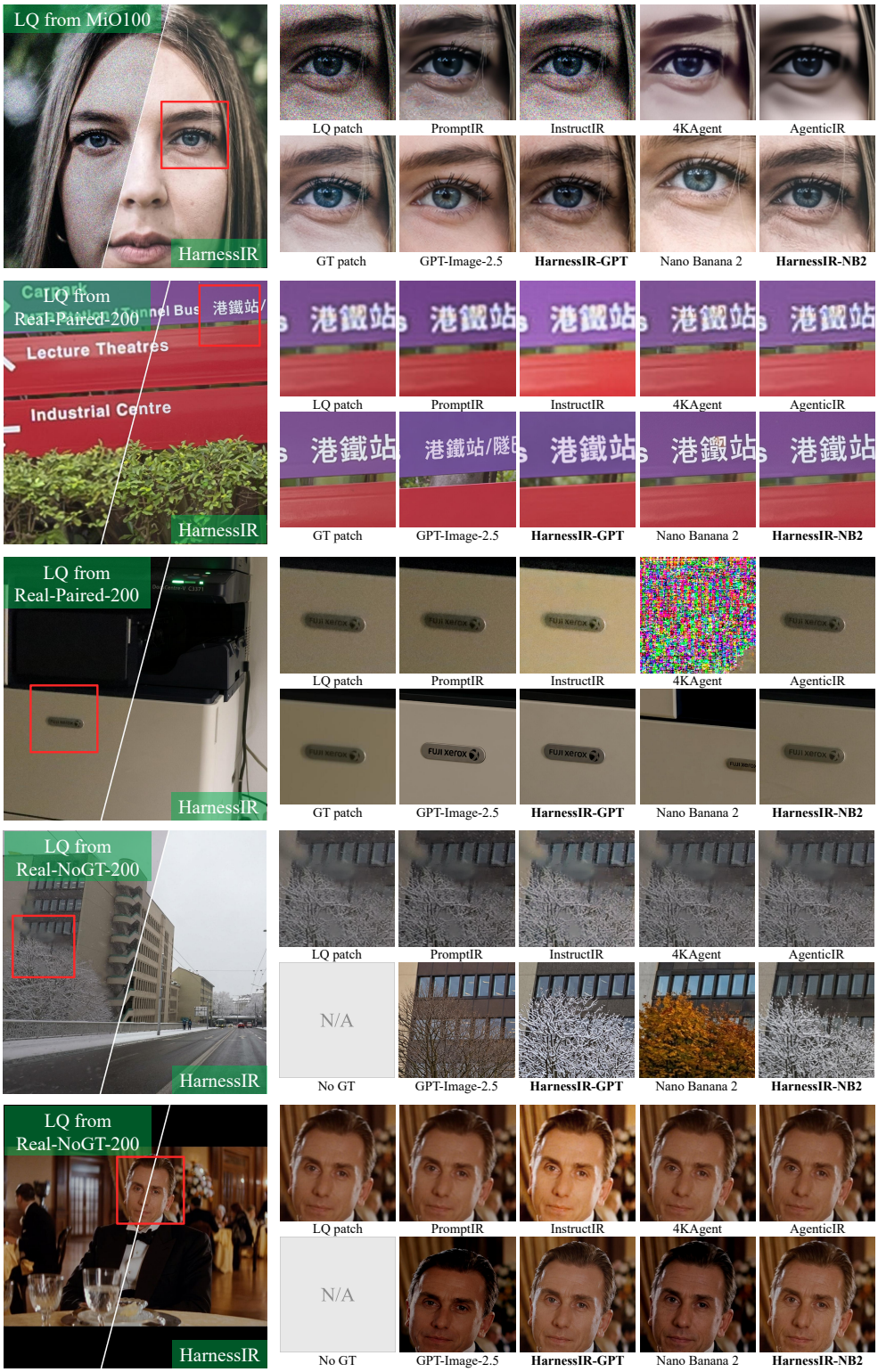

Figure H.3: Additional visual comparisons of different methods. HarnessIR more effectively removes degradations while preserving image content. Zoom in for a better view.