Hybrid Agents for Image Restoration
Supplementary Material

Bingchen Li Xin Li[†] Yiting Lu Zhibo Chen[†]
University of Science and Technology of China
{lbc31415926, luyt31415}@mail.ustc.edu.cn
{xin.li, chenzhibo}@ustc.edu.cn

1. More Details about Restoration Tools

In this section, we provide more details to support our claims in Section 3.2 of the main paper.

1.1. Network Architecture of Restoration Model

We propose a three-stage training recipe for the construction of restoration tools. In the first stage, we aim to learn a general-purpose restoration model that encompasses common knowledge across various tasks. Therefore, we follow promising prompt learning-based works [12] and adopt an enhanced version [8] to further improve the representative abilities of the restoration model. The detailed model architecture is demonstrated in Figure 1.

1.2. LoRA Fine-tuning of Restoration Tools

We employ LoRA [6] to effectively build task-specific restoration tools based on the well-trained model from Stage I. We follow the general implementation and add low-rank matrices to the attention block layers that generate Q, K, and V, respectively. Moreover, the LoRA is added to the Linear layer in the feedforward block. The similar implementation is adopted for RHAG [3]. The details are demonstrated in Figure 2.

1.3. The Generation of Hybrid Distortion

To tailor the hybrid restoration tool, we further fine-tune the restoration model on online-generated mix-degraded image pairs. As the degradation pipeline in Real-ESRGAN [17] and BSRGAN [21] shows promising results on hybrid and real-world distortion removal, we follow such design and build a degradation pipeline that includes our 10 distortions in Figure 3. Considering that rainstreak, raindrop, haze and low light are intrinsically real-world distortions, we only add noise/JPEG on these four types of distortions to form the hybrid distortion. Among the 10 distortions, we use existing datasets for rainstreak, raindrop, haze, and low-light. For the other six distortions, we synthesize image pairs. The specific synthesis methods are as follows:

- Gaussian blur: We add Gaussian blur with sigma value ranging from 0.2 to 4.
- Motion blur: We follow the implementation in [4].
- Gaussian noise: We add Gaussian noise with intensity value ranging from 15 to 50.
- JPEG compression: We add JPEG compression with quality factor between 10 and 40.
- HEVC compression: We add HEVC compression (HM-18.0) with three quality factor 32, 37, and 42, where a higher value demonstrates a worse image quality.
- VVC compression: Similar as HEVC, we add VVC vompression (VTM-21.0) with three quality factor 32, 37, and 42.

1.4. The Construction of Instruction Tuning Dataset

In this section, we provide more details about the construction of our instruction tuning dataset for SlowAgent and FeedbackAgent, respectively. Furthermore, we provide examples for a more intuitive illustration of the constructed instruction tuning dataset.

1.4.1. Instruction Tuning Dataset for SlowAgent

Once the FastAgent categories the user prompt as the ambiguous one, it will invoke the SlowAgent to automatically finish the restoration process. Consequently, the task of the SlowAgent is to detect distortions and select the appropriate restoration tool. Therefore, we build the instruction tuning dataset based on the distortion type of the image and the according restoration tool. As described in Sections 3.3 and 4.1 of the main paper, we construct the instruction-tuning dataset based on 11 types of distortions (*i.e.*, 10 single distortion types and one hybrid distortion type). For each image-text pairs, we use the following format: "[User: <Question><image>.], [Assistant: DISTORTION: <type>. CALL: de-<type>tool.]", where "Question" is randomly chosen from Table 1, and "type" denotes for distortion type. Notably, to prevent the multi-

[†]Corresponding authors.

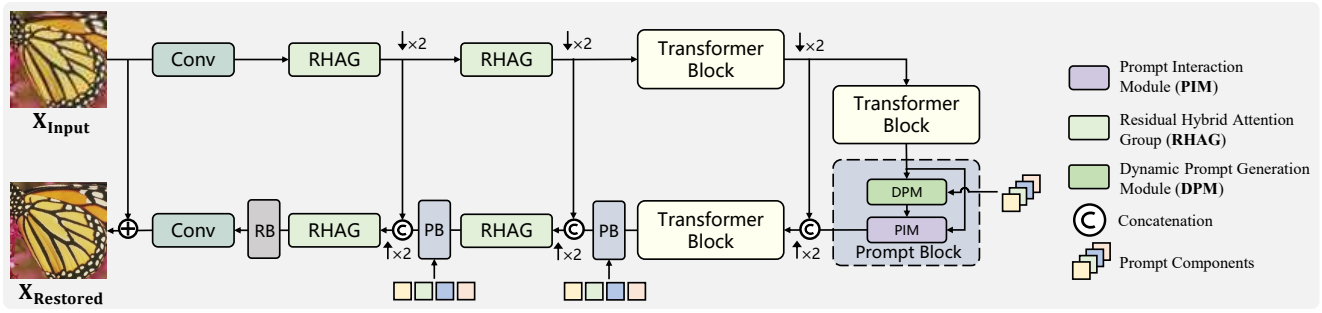

Figure 1. The illustration of network architecture of restoration model. We adopt the enhanced version of PromptIR [12] from [8]. "RB" denotes for refinement block, "PB" denotes for prompt block, respectively.

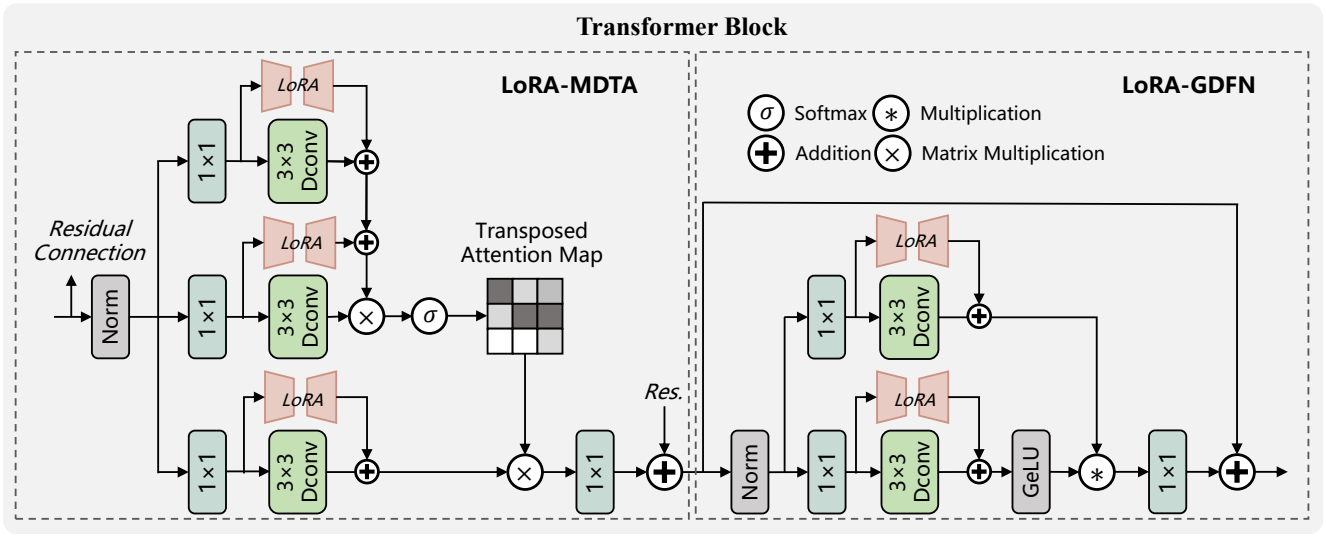

Figure 2. The implementation details of LoRA fine-tuning. We take the transformer block [12] here for an example. As multi-DConv head transposed self-attention (MDTA) leverages depth-wise convolution to generate Q, K, V matrices, we add LoRA on these layers. For residual hybrid attention group (RHAG), we add LoRA on linear layers that related to Q, K, V matrices generation. Additionally, following common implementation, we add LoRA layers in the feedforward module.

modal large language model (MLLM) from overfitting to the questions rather than the image distortion identification, we used GPT to randomly generate 20 different questions.

Additionally, to enable the SlowAgent to robustly identify distortion types in images, we constructed a dataset containing 70k samples. We generate 5k image-text pairs for each single distortion and 20k pairs for hybrid distortions. For the hybrid distortions, we include 2k image-text pairs for rainstreak, raindrop, haze, and low-light with noise/JPEG, respectively. We use distortion pipeline described in Section 1.3 to generate another 12k hybrid distortions image-text pairs.

1.4.2. Instruction Tuning Dataset for FeedbackAgent

In our HybridAgent system, the SlowAgent is responsible for recognizing distortions in images. However, it lacks the ability to determine whether the restoration process should be terminated or continued. Therefore, we develop a FeedbackAgent to determine whether an image is clean, providing support for the restoration process. For each image-text pairs, we use the following format: "[User: This is a restored image.<image>RESTORATION HISTORY: de-<type>. Is it clean now?], [Assistant: Yes. CALL: END.] or [Assistant: No. CALL: SlowAgent.]", where "END" indicates that the restoration process is finished. Follow the description in Section 3.3 of the main paper, we construct 30k image-text pairs for "clean" and 33k image-text pairs for "not clean". For "clean" samples, we generate 25k image-text pairs for (i) single-distorted images restored using the correct tool (2.5k pairs for each single distortion), 2.5k pairs for (ii) hybrid-distorted images restored with the hybrid restoration tool, and 2.5k pairs for (iii) single-distorted images restored by the hybrid restoration tool. For "not clean" samples, we generate 8k image-text pairs for

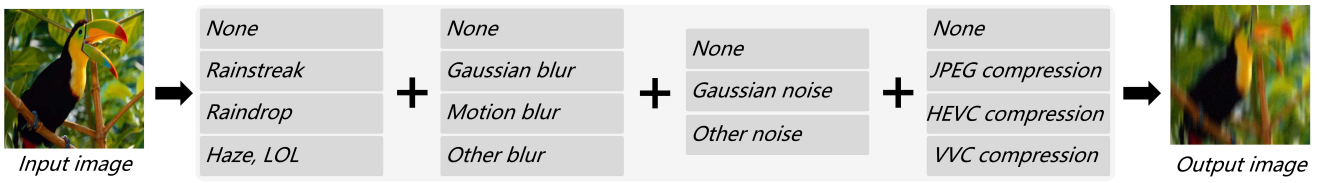

Figure 3. The distortion pipeline we used to synthesize hybrid degradations. Notably, for rainstreak, raindrop, haze, and low-light images, we only add Gaussian noise or JPEG distortions. "Other blur" and "other noise" refer to additional types of blur and noise implemented in Real-ESRGAN [17]. During synthesis, we ensure that each image contains at least two types of distortions.

Table 1. List of Questions we utilized to construct the instruction tuning dataset for SlowAgent.

Questions
What is the distortion type of this image?
What kind of distortion is present in this image?
What type of image distortion can be observed here?
What distortion effect is visible in this image?
Can you identify the distortion in this image?
What is the nature of the distortion in this image?
What type of distortion has affected this image?
What form of distortion is evident in this image?
How is this image distorted?
What kind of image distortion does this show?
What kind of visual distortion is in this image?
What distortion does this image exhibit?
What is the specific distortion type in this image?
How is this image distorted visually?
What kind of alteration or distortion appears in this image?
What type of distortion can be seen in this image?
What image distortion effect is noticeable here?
What is the distortion pattern in this image?
Can you describe the distortion present in this image?
What distortion characteristic is evident in this image?

(i) single-distorted images restored with an incorrect tool, and 25k image-text pairs for (ii) hybrid-distorted images restored with a single restoration tool (2.5k pairs for each task-specific restoration tool).

2. More Implementation Details

2.1. Instruction Tuning for HybridAgent

Current studies on MLLMs generally focus on building models that excel in diverse tasks [1, 9, 11]. However, this generalization often limits their performance on specialized tasks requiring expert knowledge, such as IR. To adapt MLLMs for the roles of SlowAgent and FeedbackAgent, we follow previous works [9, 19] and adopt instruction tuning.

SlowAgent. Since the primary task of the SlowAgent is to detect distortions and select the appropriate restoration tool, the instruction-tuning dataset must encompass a broad range of distortion types, with corresponding text outputs designed as invocation commands for restoration tools. Specifically, to cover the majority of real application scenarios, we construct our dataset with 10 distortions: noise, gaussian blur, motion blur, JPEG, HEVC [16], VVC [2], rainstreak, raindrop, haze, low light. Additionally, the combination of these distortions is considered the 11th type. We provide details about the combination in the **supplementary.** Since SlowAgent's distortion recognition should be independent of image content and resolution, we apply random rotations and flips as augmentation. We randomly crop the image to a resolution ranging between 224×224 and 784×784. Our instruction tuning dataset for the SlowAgent includes 70k image-text pairs, enabling the tuned model to robustly identify image distortion types. More details are given in the **supplementary**.

FeedbackAgent. On the other hand, since no existing MLLM can reliably assess whether a restored image is clean, instruction tuning is essential for the FeedbackAgent. To address this, we constructed an additional dataset with approximately 60k image-text pairs, where images are labeled as "clean" or "not clean". We define "clean" as: (i) single-distorted images restored using the correct tool, (ii) hybrid-distorted images restored with the hybrid restoration tool, or (iii) single-distorted images restored by the hybrid restoration tool. Conversely, "not clean" includes: (i) single-distorted images restored with an incorrect tool, or (ii) hybrid-distorted images restored with a single restoration tool.

2.2. Training Details

The training of HybridAgent mainly contains two parts: (i) The construction of restoration tools, and (ii) The instruction tuning of SlowAgent and FeedbackAgent. To build restoration tools, we first train a prompt learning-based all-in-one model [8] across 10 distortions, with an initial learning rate of 2e-4 and gradually decay to 1e-6 with cosine annealing. AdamW optimizer is adopted to train the model for 600k iterations. During training, the image pairs are cropped into 128×128 patches with random horizontal and vertical flips as the data augmentation. Subsequently,

Table 2. Training consumptions of the restoration tools. Notably, we generate the mixed-degraded samples online in Stage III for storage saving purpose, where the compression processes of HEVC and VVC consume a significant amount of time.

	Trainable Params. (M)	Training Hours
Stage I	34.79	87.71
Stage II	6.49	8.98
Stage III	6.36	31.47

we leverage LoRA with a rank of 8 to tailor our task-specific restoration tools. For each tool, the model with LoRA weights is further optimized for 100k iterations with a fixed learning rate of 5e-5. Consequently, to build a hybrid restoration tool, we further fine-tune a new set of LoRA weights with mixed degradations for 200k iterations, where a fixed learning rate of 1e-4 is adopted. For Stage I, we utilized 8 RTX 3090 GPUs for training, with a total batch size of 32. For Stage II and III, we utilized 4 4090D with a total batch size of 16. We provide the training times in Table 2. Notably, we generate the mixed-degraded samples online in Stage III, where the compression processes of HEVC [16] and VVC [2] consume a significant amount of time.

We instruct tuning Co-Instruct [19] model for our HybridAgent, as it excels at distortion-related question-answering tasks. For both agents, we fine-tune the MLLM using 4 RTX 4090D GPUs with a total batch size of 256 using LoRA [6]. The initial learning rate is set to 1e-4 and gradually decayed to 0 using cosine annealing. Each agent is trained for 2 epochs, with SlowAgent taking approximately 2.5 hours and FeedbackAgent about 1.5 hours. Deepspeed [15] is utilized to accelerate the training process.

2.3. More Details about Testset

Beyond single distortion removal, it is crucial to evaluate the hybrid distortion removal capabilities of HybridAgent. To this end, we generate a total of 200 mix-degraded images, with details provided in Table 3. For the first six rows, we select 20 images from the combined datasets of CBSD68 [10], Urban100 [7], Kodak24 [5], and McMaster [22] as ground truth. For the remaining rows, we select 10 distorted images from Rain100H [20], RainDrop [13], RESIDE-6k [14], and LOL [18], and further add noise or JPEG artifacts.

2.4. More details about User Prompt

Users can provide various textual prompts to HybridAgent. To synthesize such prompts and evaluate the distinguishing capabilities of FastAgent, we use GPT-4 to generate a total of 220 diverse user prompts. We provide some samples in Table 4. *Notably, we assume the user has precise knowledge of the distortion type to carry out direct prompts.*

Table 3. Number of samples for the hybrid distortion testset. In total we generate 200 images for the evaluation of hybrid distortion removal.

Hybrid Distortion	**Number of Samples**
Blur + JPEG	20
Blur + Noise	20
Blur + Noise + JPEG	20
Motionblur + JPEG	20
Motionblur + Noise	20
Motionblur + Noise + JPEG	20
Rainstreak + JPEG	10
Rainstreak + Noise	10
Raindrop + JPEG	10
Raindrop + Noise	10
Haze + JPEG	10
Haze + Noise	10
Low light + JPEG	10
Low light + Noise	10

Table 4. Examples of textual user prompts generated by GPT-4. Notably, we assume the user has precise knowledge of the distortion type to carry out direct prompts.

Distortion	**Textual Prompts**
Noise	Please remove the grain from this image. The speckles in this photo need to be cleared up. Please fix the random spots in this image.
HEVC	Can you reduce the H.265 artifacts to improve the picture's clarity? The HM compression makes the image look rough; can you fix it? Can you remove the HEVC artifacts for a clearer image?
Haze	Please reduce the haze that blurs the scene. I'd prefer the photo to be haze-free for better contrast. I'd like the image to look vibrant, free from the dull haze.
Ambiguous	Please fix this image. This image does not look good, please help me. Can you help me enhance this image?

3. Analysis of Three-Stage Training

3.1. Effectiveness of Stage II

To construct restoration tools that not only share common knowledge across different distortion removal tasks but also possess task-specific expertise, we propose a three-stage training recipe in Section 3.2 of the main paper. To demonstrate the effectiveness of Stage II, we compare the performance between base model and task-specific model on 10 distortions. As shown in Table 6, task-specific restoration tools achieve performance improvements over the base model. Notably, task-specific tools outperform the base model, especially for rain, haze, and low-light distortions. This further highlights the importance of task-specific tools in enabling HybridAgent to effectively handle various distortions. As demonstrated in Figure 4, although the base model learns common knowledge across various distortions, its ability to handle specific distortions may be compromised. For instance, in low-light enhancement, the image becomes overly bright. Therefore, it is crucial for HybridAgent to leverage task-specific restoration tools to effectively handle various distortions.

On the other hand, to highlight the importance of common knowledge shared across different tasks, we compare step-by-step removal of three hybrid distortions using Stage II's restoration tools with task-specific models trained from scratch. As demonstrated in Table 5, Stage II's restoration tools achieve higher performance, indicating the effectiveness of task-common knowledge learning in Stage I. Additionally, we only have to storage light-weight LoRA weights for restoration tools, which is resource-friendly towards numerous of distortions.

Table 5. Comparisons of performance on step-by-step removal of hybrid distortions. We evaluate the PSNR↑/SSIM↑/LPIPS↓.

	Rainstreak + Noise	Rainstreak + JPEG	Noise + JPEG
From Scratch	23.13/0.617/0.372	22.04/0.685/0.331	30.19/0.813/0.185
Stage II	23.36/0.622/0.351	22.18/0.692/0.300	30.23/0.816/0.163

3.2. Effectiveness of Stage III

To demonstrate the advantage of Stage III training for the hybrid restoration tool, we compare its performance with models trained from scratch with the same mix-degradation pipeline. As shown in Table 7, the hybrid restoration tool trained in Stage III outperforms the model trained from scratch, especially for complex distortions (*e.g.*, rainstreak).

3.3. Effectiveness of Parameter Reinitialization

Prompt components serve as conditional information, enabling the restoration model to identify distortions and execute the appropriate removal process. Therefore, reinitializing the parameters of prompt components in Stage II is essential to prevent the model from generating inaccurate conditional information. To validate this, we conduct experiments on three settings: (i) As demonstrated in the main paper, we reinitialize the parameters of the prompt components; (ii) We initialize the prompt components with the respective parameters from Stage I; (iii) We initialize the prompt components with the respective parameters from Stage I and keep them fixed during LoRA fine-tuning. The results are demonstrated in Table 8. Comparing between (i) and (ii), we conclude that reinitialize prompt components are essential for the construction of task-specific restoration tools, especially for difficult distortions such as raindrop and low-light. *Furthermore, we conclude that LoRA fine-tuning is efficient than full fine-tuning the model.* By comparing (ii) and (iii), we further validate that the prompt components from Stage I are not suitable for task-specific learning, as they generate inaccurate distortion guidance, leading to suboptimal results.

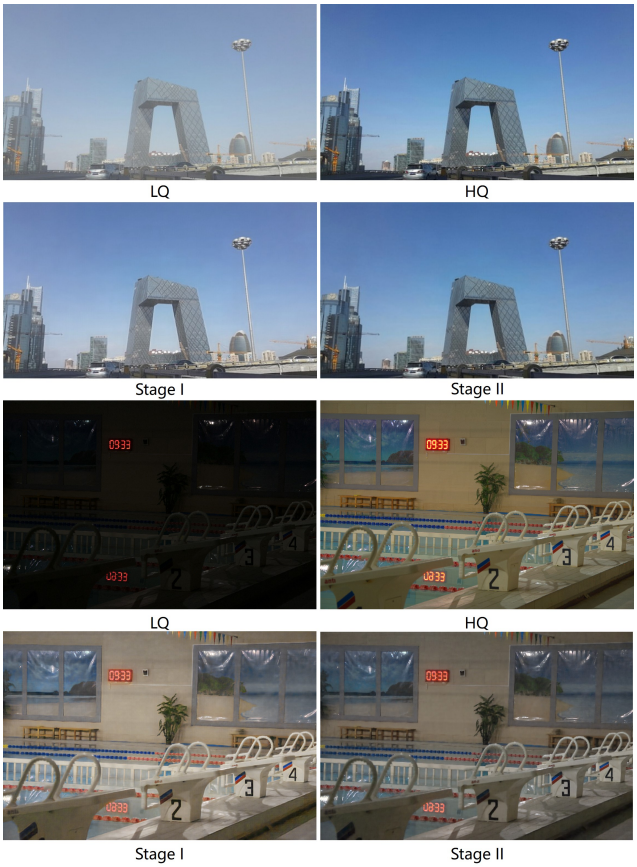

Figure 4. Qualitative comparisons between Stage I's and Stage II's restoration results on de-haze (top) and low light enhancement (bottom).

Table 6. Comparisons of performance on single distortion removal between Stage I's base model and Stage II's task-specific restoration tools. We evaluate the PSNR↑/SSIM↑.

	De-noise	De-blur	De-motionblur	De-jpeg	De-HEVC	De-VVC	De-rainstreak	De-raindrop	De-haze	De-low light
Stage I	30.63/0.874	28.97/0.834	23.28/0.709	29.17/0.862	26.73/0.779	27.09/0.792	28.33/0.867	26.05/0.893	16.14/0.777	21.49/0.810
Stage II	30.76/0.877	30.65/0.853	23.80/0.720	30.21/0.876	27.58/0.785	27.69/0.794	30.05/0.894	30.35/0.914	29.93/0.960	22.61/0.828

Table 7. Comparisons of performance on hybrid removal of mixed distortions. We evaluate the PSNR↑/SSIM↑/LPIPS↓.

	Rainstreak + Noise	Rainstreak + JPEG	Noise + JPEG
From Scratch	26.31/0.760/0.288	25.35/0.755/0.266	30.60/0.879/0.102
Stage III	26.49/0.766/0.203	25.44/0.764/0.242	30.64/0.882/0.096

Table 8. Comparisons of performance on whether the parameters of prompt components are reinitialized in Stage II. We evaluate the PSNR↑/SSIM↑.

	De-Noise	De-Raindrop	De-Low light
(i)	30.76/0.877	30.35/0.914	22.61/0.828
(ii)	30.72/0.876	29.12/0.907	22.37/0.827
(iii)	30.70/0.876	27.89/0.889	21.88/0.816

4. Qualitative Results of All-in-One Methods

In this section, we provide the qualitative comparisons between HybridAgent and other all-in-one methods to support the results in Table 4 of the main paper. As demonstrated in Figure 5, HybridAgent achieves better restoration qualities than other methods, demonstrate the effectiveness of the collaboration of hybrid restoration tool and task-specific restoration tools.

References

[1] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023. 3

[2] Benjamin Bross, Ye-Kui Wang, Yan Ye, Shan Liu, Jianle Chen, Gary J Sullivan, and Jens-Rainer Ohm. Overview of the versatile video coding (vvc) standard and its applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(10):3736–3764, 2021. 3, 4

[3] Xiangyu Chen, Xintao Wang, Jiantao Zhou, Yu Qiao, and Chao Dong. Activating more pixels in image super-resolution transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22367–22377, 2023. 1

[4] Huiyu Duan, Xiongkuo Min, Sijing Wu, Wei Shen, and Guangtao Zhai. Uniprocessor: a text-induced unified low-level image processor. In *European Conference on Computer Vision*, pages 180–199. Springer, 2025. 1

[5] Rich Franzen. Kodak lossless true color image suite. http://r0k.us/graphics/kodak/, 1999. Online accessed 24 Oct 2021. 4

[6] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 1, 4

[7] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5197–5206, 2015. 4

[8] Bingchen Li, Xin Li, Yiting Lu, Ruoyu Feng, Mengxi Guo, Shijie Zhao, Li Zhang, and Zhibo Chen. Promptcir: Blind compressed image restoration with prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6442–6452, 2024. 1, 2, 3

[9] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 3

[10] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings eighth IEEE international conference on computer vision. ICCV 2001*, pages 416–423. IEEE, 2001. 4

[11] OpenAI. Gpt-4 technical report, 2023. 3

[12] Vaishnav Potlapalli, Syed Waqas Zamir, Salman H Khan, and Fahad Shahbaz Khan. Promptir: Prompting for all-in-one image restoration. *Advances in neural information processing systems*, 36:71275–71293, 2023. 1, 2

[13] Rui Qian, Robby T Tan, Wenhan Yang, Jiajun Su, and Jiaying Liu. Attentive generative adversarial network for raindrop removal from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2482–2491, 2018. 4

[14] Xu Qin, Zhilin Wang, Yuanchao Bai, Xiaodong Xie, and Huizhu Jia. Ffa-net: Feature fusion attention network for single image dehazing. In *Proceedings of the AAAI conference on artificial intelligence*, pages 11908–11915, 2020. 4

[15] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506, 2020. 4

[16] Gary J Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand. Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on circuits and systems for video technology*, 22(12):1649–1668, 2012. 3, 4

[17] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with

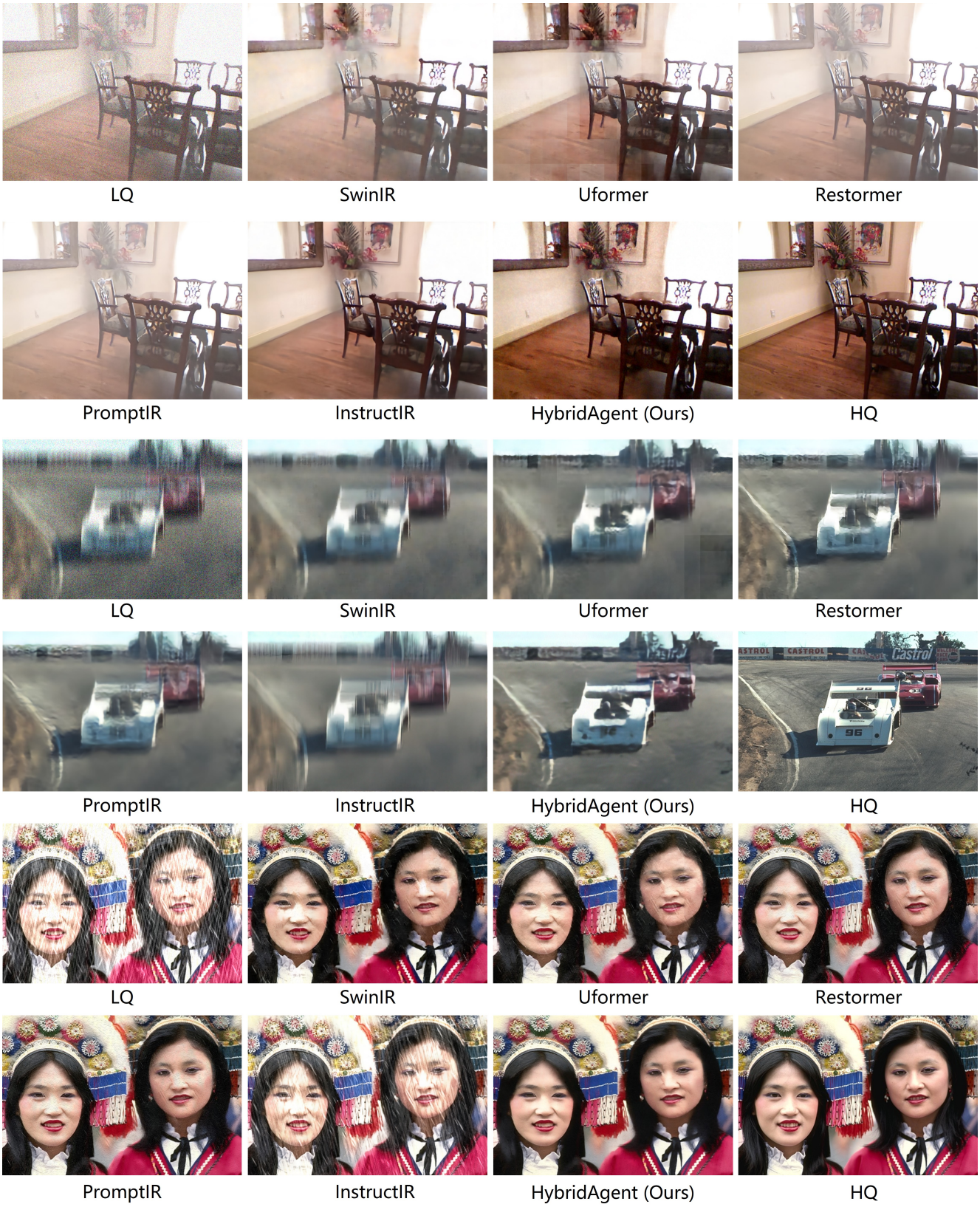

Figure 5. Qualitative comparisons between HybridAgent and other all-in-one methods. From top to bottom: Haze + Noise, Motionblur + Noise, Rainstreak + Noise.

pure synthetic data. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1905–1914, 2021. 1, 3

[18] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. *arXiv preprint arXiv:1808.04560*, 2018. 4

[19] Haoning Wu, Hanwei Zhu, Zicheng Zhang, Erli Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Annan Wang, Wenxiu Sun, Qiong Yan, et al. Towards open-ended visual quality comparison. In *European Conference on Computer Vision*, pages 360–377. Springer, 2024. 3, 4

[20] Wenhan Yang, Robby T Tan, Jiashi Feng, Jiaying Liu, Zongming Guo, and Shuicheng Yan. Deep joint rain detection and removal from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1357–1366, 2017. 4

[21] Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4791–4800, 2021. 1

[22] Lei Zhang, Xiaolin Wu, Antoni Buades, and Xin Li. Color demosaicking by local directional interpolation and nonlocal adaptive thresholding. *Journal of Electronic imaging*, 20(2): 023016–023016, 2011. 4