RestoreAgent: Autonomous Image Restoration Agent via Multimodal Large Language Models

Haoyu Chen[1], **Wenbo Li**[2], **Jinjin Gu**[3], **Jingjing Ren**[1], **Sixiang Chen**[1], **Tian Ye**[1], **Renjing Pei**[2], **Kaiwen Zhou**[2], **Fenglong Song**[2], **Lei Zhu**[1,4*]

[1]The Hong Kong University of Science and Technology (Guangzhou) [2]Huawei Noah's Ark Lab
[3]The University of Sydney [4]The Hong Kong University of Science and Technology
`Project page: https://haoyuchen.com/RestoreAgent`

NeurIPS 2024

Presenter: Hao Wang

Advisor: Prof. Chia-Wen Lin

Outline

- Introduction
- Framework
- Method
- Experiment
- Conclusion

Outline

- Introduction
- Framework
- Method
- Experiment
- Conclusion

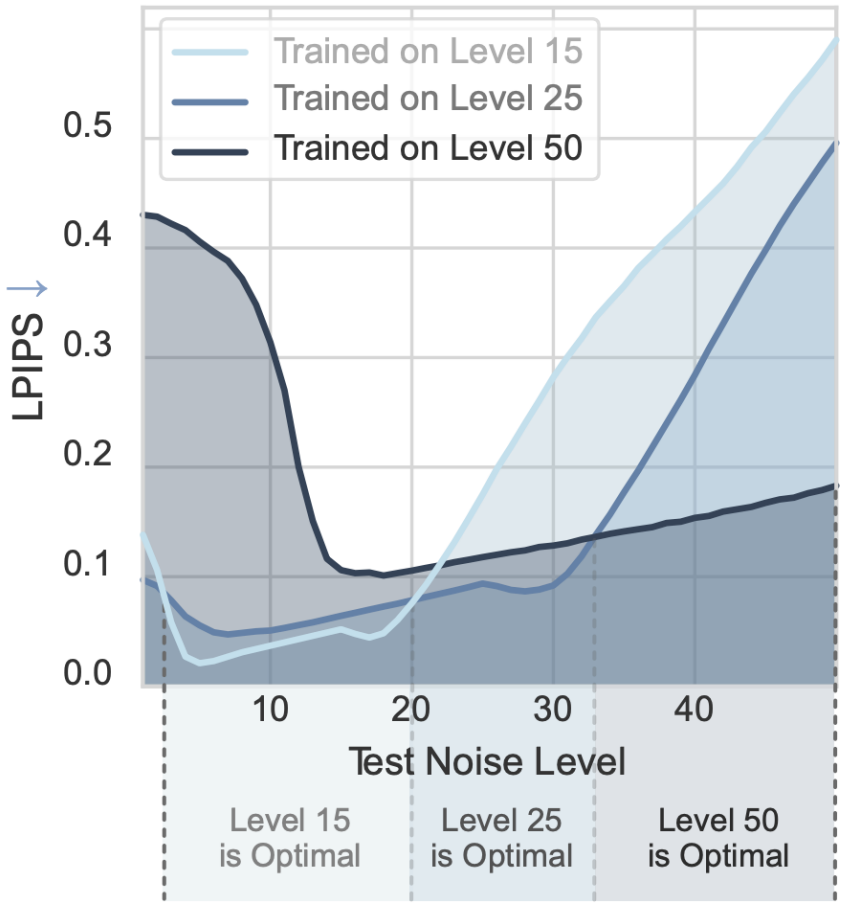
Introduction

- All-in-one models, though capable of handling multiple tasks, typically support only a **limited range** and often produce **overly smooth, low-fidelity** outcomes due to their **broad data distribution fitting**.

- Leveraging **multimodal large language models**, RestoreAgent autonomously assesses the type and extent of degradation in input images and performs restoration through determining the appropriate **restoration tasks**, optimizing the **task sequence**, selecting the most **suitable models**, and executing the restoration.

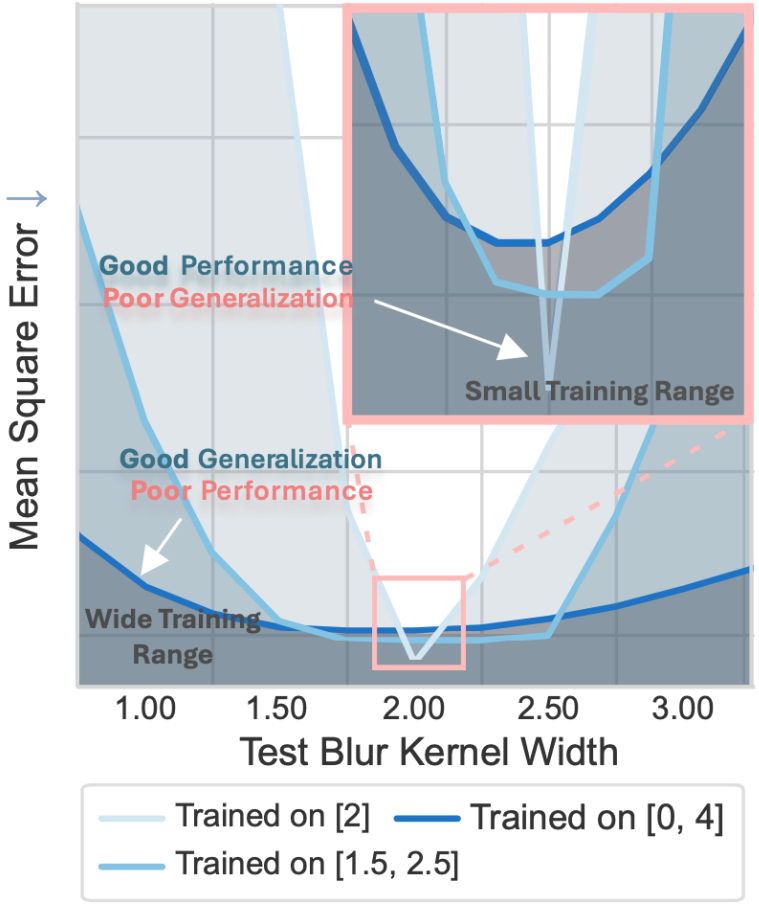
- System's modular design facilitates the fast integration of new tasks and models, enhancing its flexibility and scalability for various applications.

Introduction



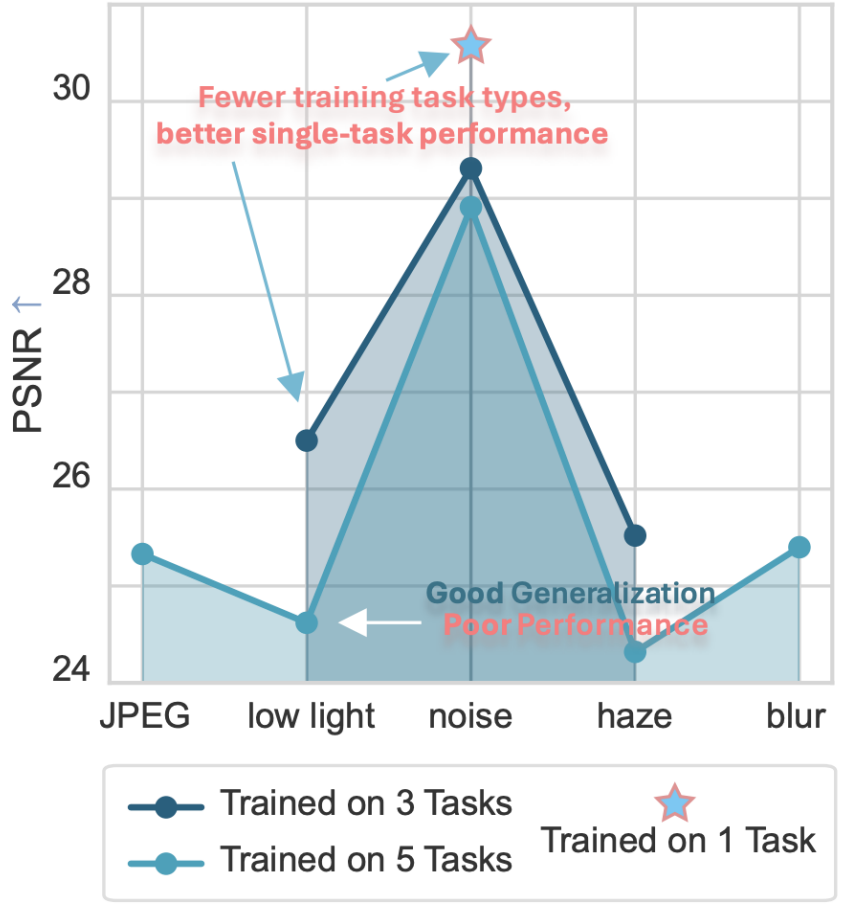
a. Trained on 3 different noise levels
Tested across varying noise levels

Each model has a specific strength range



b. Trained on 3 blur kernel ranges
Tested across varying blur kernels

Generalization vs. performance Trade-off: Improving one worsens the other

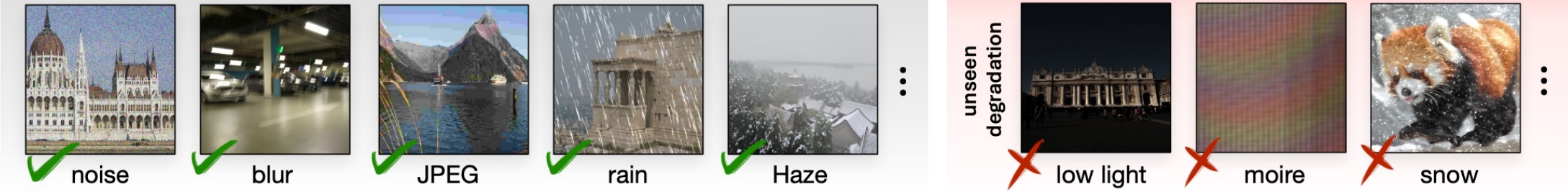


c. Trained on different numbers of tasks
Tested on various tasks

All-in-one models underperform single-task models

Why not all-in-one?

a1. Not truly "all": Models still fail on unseen degradation types



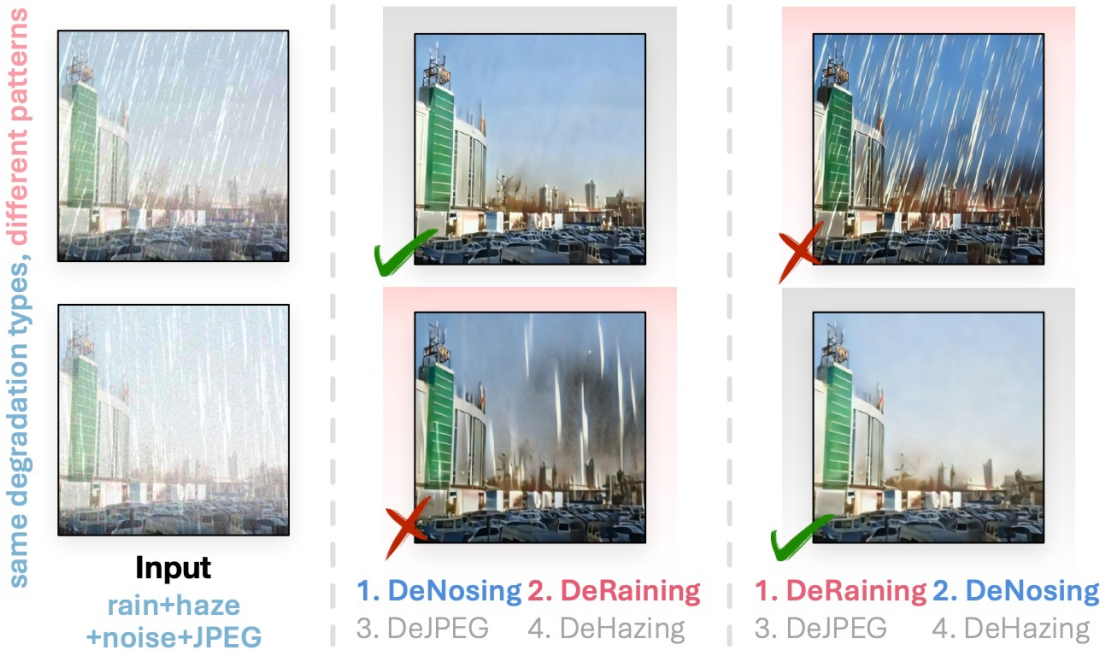
a2. Limited performance: Specialized models outperform generalists

a3. Single Task + All-in-One > All-in-One only

Why not use a fixed or random task execution order?

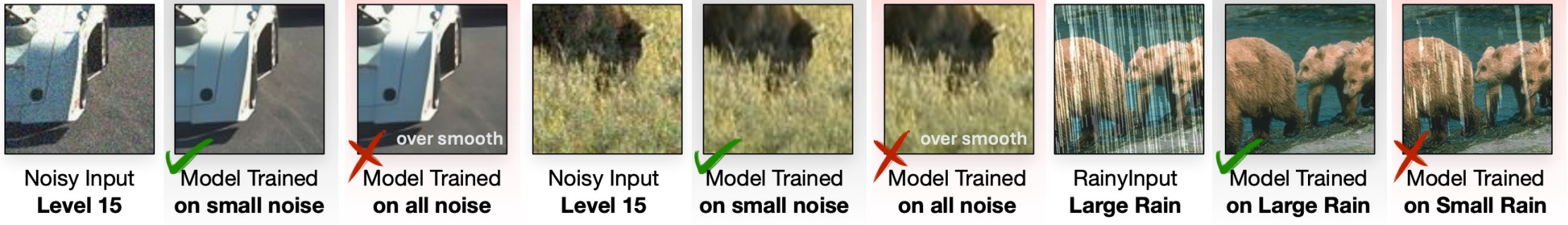
b.1 Wrong execution order causes wrong results.

b.2 Same degradation types with different patterns require distinct execution orders



Why not use a single fixed model for a task?

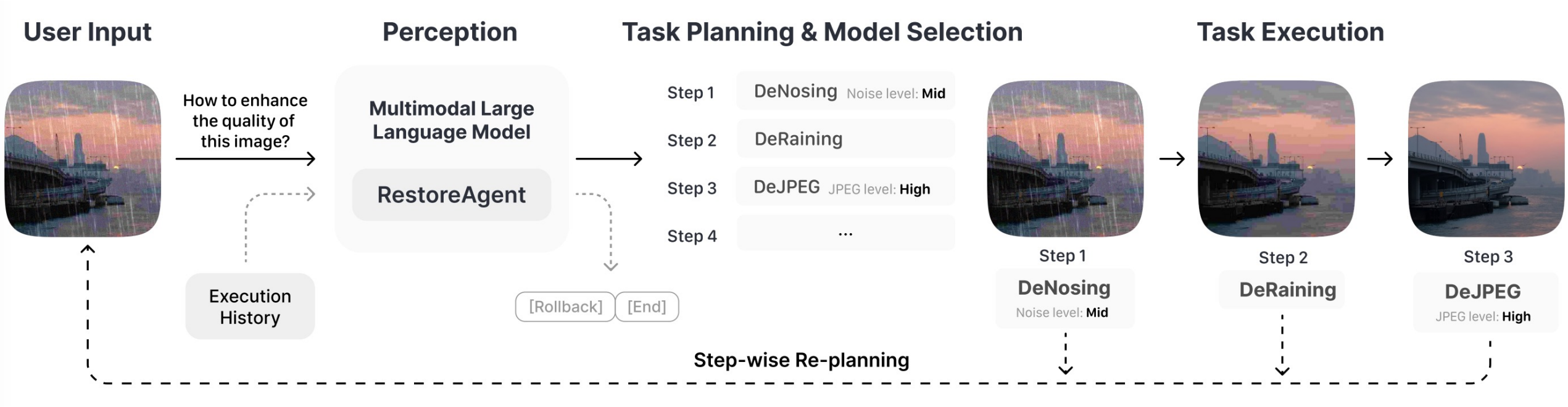
c. Inflexible models limit optimal performance



Outline

- Introduction
- Framework
- Method
- Experiment
- Conclusion

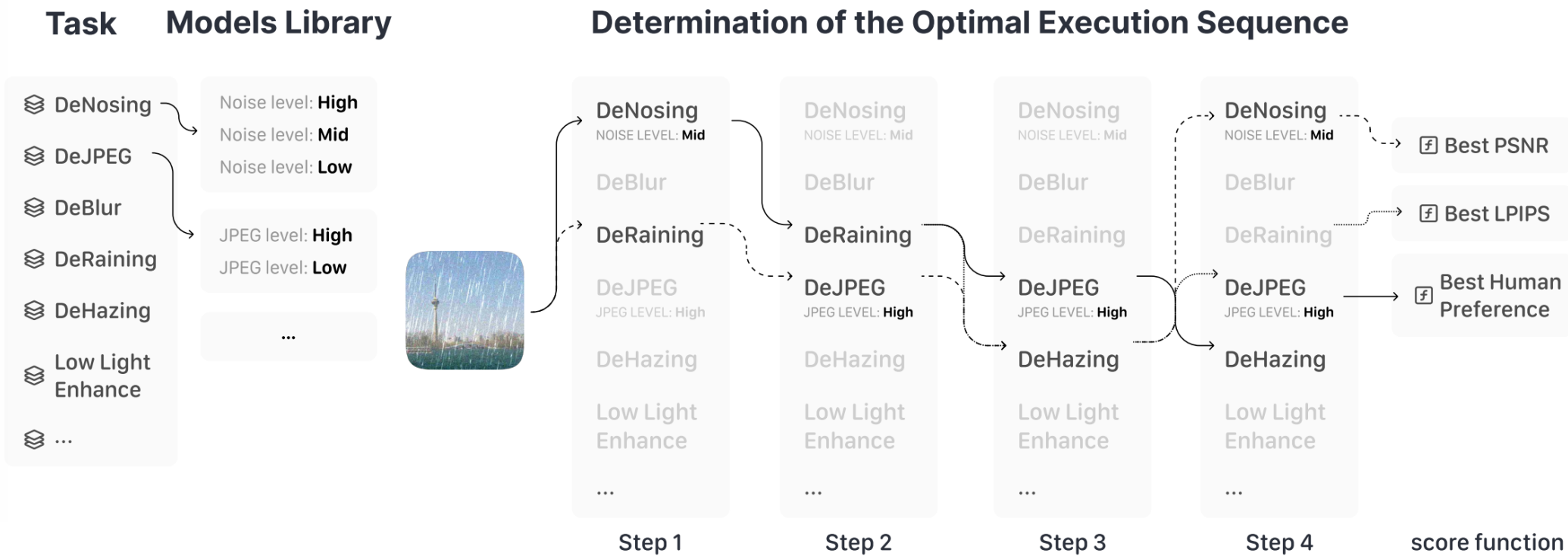
Framework


User Input
Perception
Task Planning & Model Selection
Task Execution
How to enhance the quality of this image?
Multimodal Large Language Model
RestoreAgent
Execution History
Step 1 DeNosing Noise level: Mid
Step 2 DeRaining
Step 3 DeJPEG JPEG level: High
Step 4 ...
[Rollback] [End]
Step 1
DeNosing
Noise level: Mid
Step 2
DeRaining
Step 3
DeJPEG
JPEG level: High
Step-wise Re-planning

Outline

- Introduction
- Framework
- Method
- Experiment
- Conclusion

Data Construction



Problem Definition

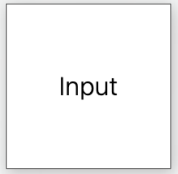
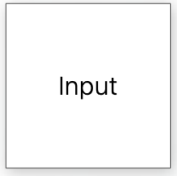
$$\mathcal{D} = \{d_1, d_2, \ldots, d_n\}$$

$$\{M_{d_i}^1, M_{d_i}^2, \ldots\}$$

$$\sigma = (M_{a_1}^{b_1}, M_{a_2}^{b_2}, \ldots, M_{a_m}^{b_m})$$

$$\sigma^* = \arg \max_{\sigma \in \mathfrak{S}(\mathcal{D}, \mathcal{M})} S(I, \sigma)$$

Scenarios

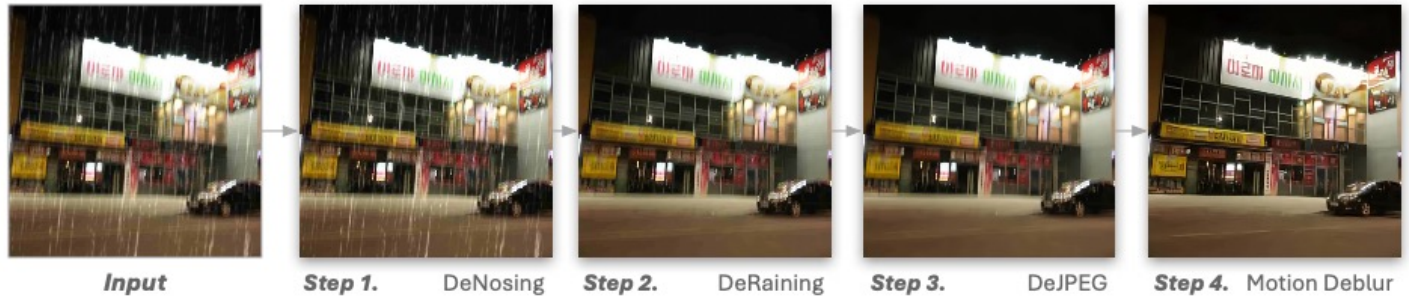
		Input	Answer	Function
1 (Primary)		How to enhance the quality of this image? Execution history: None.	1. Denoising, low noise level, 2. Dehazing, 3. DeJPEG, high quality factor.	Initiate full enhancement sequences for degraded images.
2		How to enhance the quality of this image? Execution history: 1. Denoising, low noise level.	1. Denoising, low noise level, 2. Dehazing, 3. DeJPEG, high quality factor	Dynamically adjust strategies based on intermediate results.
3		How to enhance the quality of this image? Execution history: 1. Denoising, low noise level, 2. Dehazing	Rollback.	Identify and correct suboptimal steps through rollback mechanisms.
4		How to enhance the quality of this image? Execution history: 1. Denoising, low noise level. Rollback from Dehazing.	1. Denoising, low noise level, 2. DeJPEG, high quality factor, 3. Dehazing.	Avoid repetition of ineffective procedures post-rollback.
5		How to enhance the quality of this image? Execution history: 1. Denoising, low noise level, 2. DeJPEG, high quality factor, 3. Dehazing.	Stop.	Recognize when image quality has reached its optimal state.

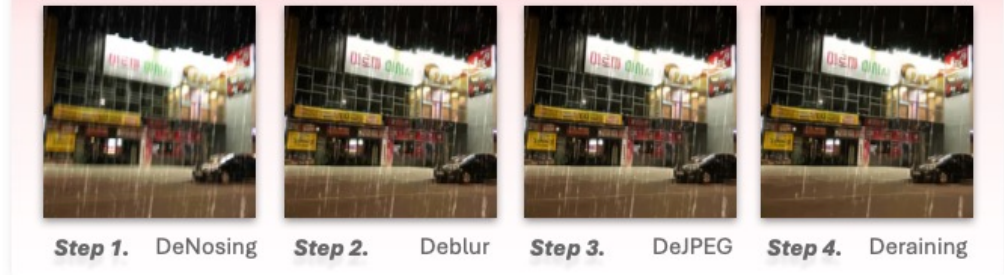
Outline

- Introduction
- Framework
- Method
- Experiment
- Conclusion


Input
Step 1. DeNosing
Step 2. DeJPEG
Step 3. Deraining
Step 4. Dehazing

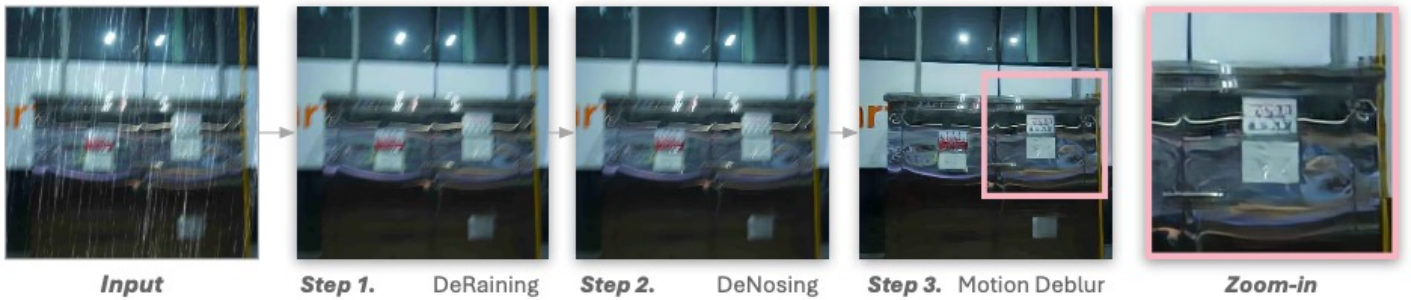

Step 1. Dehazing
Step 2. DeJPEG
Step 3. DeNosing
Step 4. Deraining


Input
Step 1. DeNosing
Step 2. DeRaining
Step 3. DeJPEG
Step 4. Motion Deblur


Step 1. DeNosing
Step 2. Deblur
Step 3. DeJPEG
Step 4. Deraining

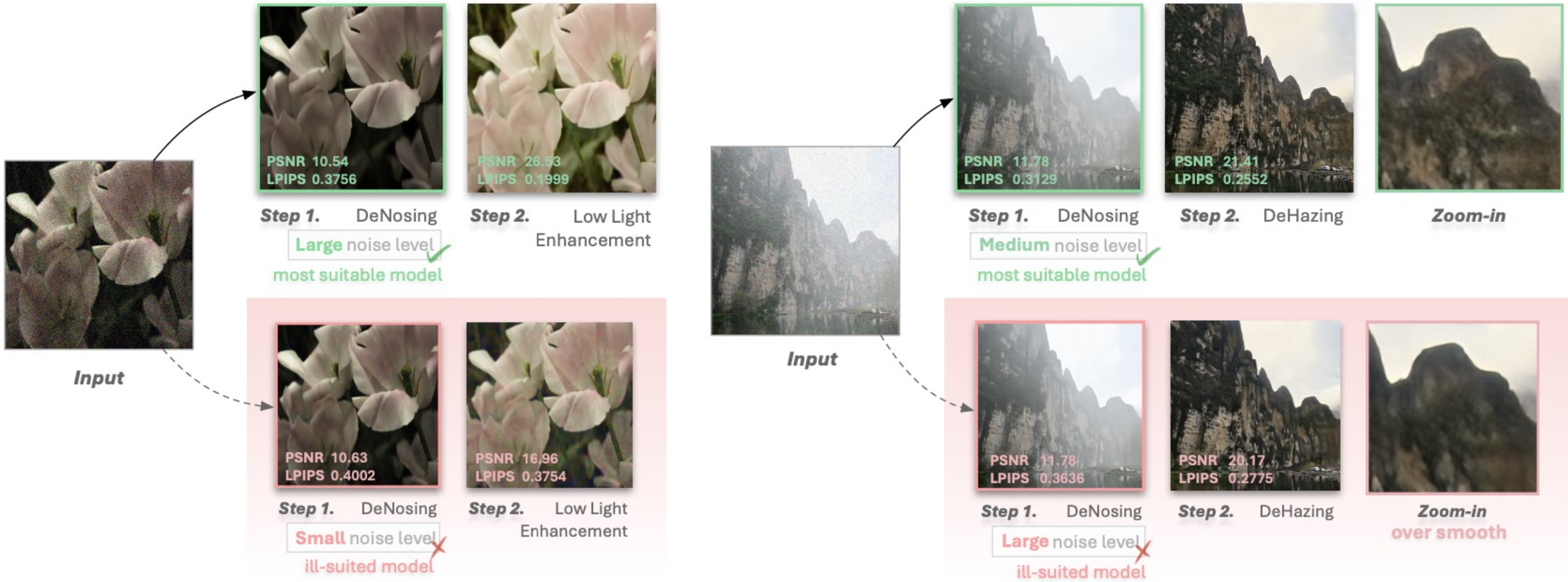

Input
Step 1. DeNosing
Step 2. Deraining
Step 3. DeHazing
Zoom-in


Step 1. DeNosing
Step 2. DeJPEG
Step 3. Deraining
Zoom-in


Input
Step 1. DeRaining
Step 2. DeNosing
Step 3. Motion Deblur
Zoom-in


Step 1. DeNosing
Step 2. Deblur
Step 3. Deraining
Zoom-in

Results


PSNR 10.54
LPIPS 0.3756
PSNR 26.53
LPIPS 0.1999
Step 1. DeNosing
Large noise level
most suitable model
Step 2. Low Light Enhancement
Input
PSNR 10.63
LPIPS 0.4002
PSNR 16.96
LPIPS 0.3754
Step 1. DeNosing
Small noise level
ill-suited model
Step 2. Low Light Enhancement
PSNR 11.78
LPIPS 0.3129
PSNR 21.41
LPIPS 0.2552
Step 1. DeNosing
Medium noise level
most suitable model
Step 2. DeHazing
Zoom-in
Input
PSNR 11.78
LPIPS 0.3636
PSNR 20.17
LPIPS 0.2775
Step 1. DeNosing
Large noise level
ill-suited model
Step 2. DeHazing
Zoom-in
over smooth

Results

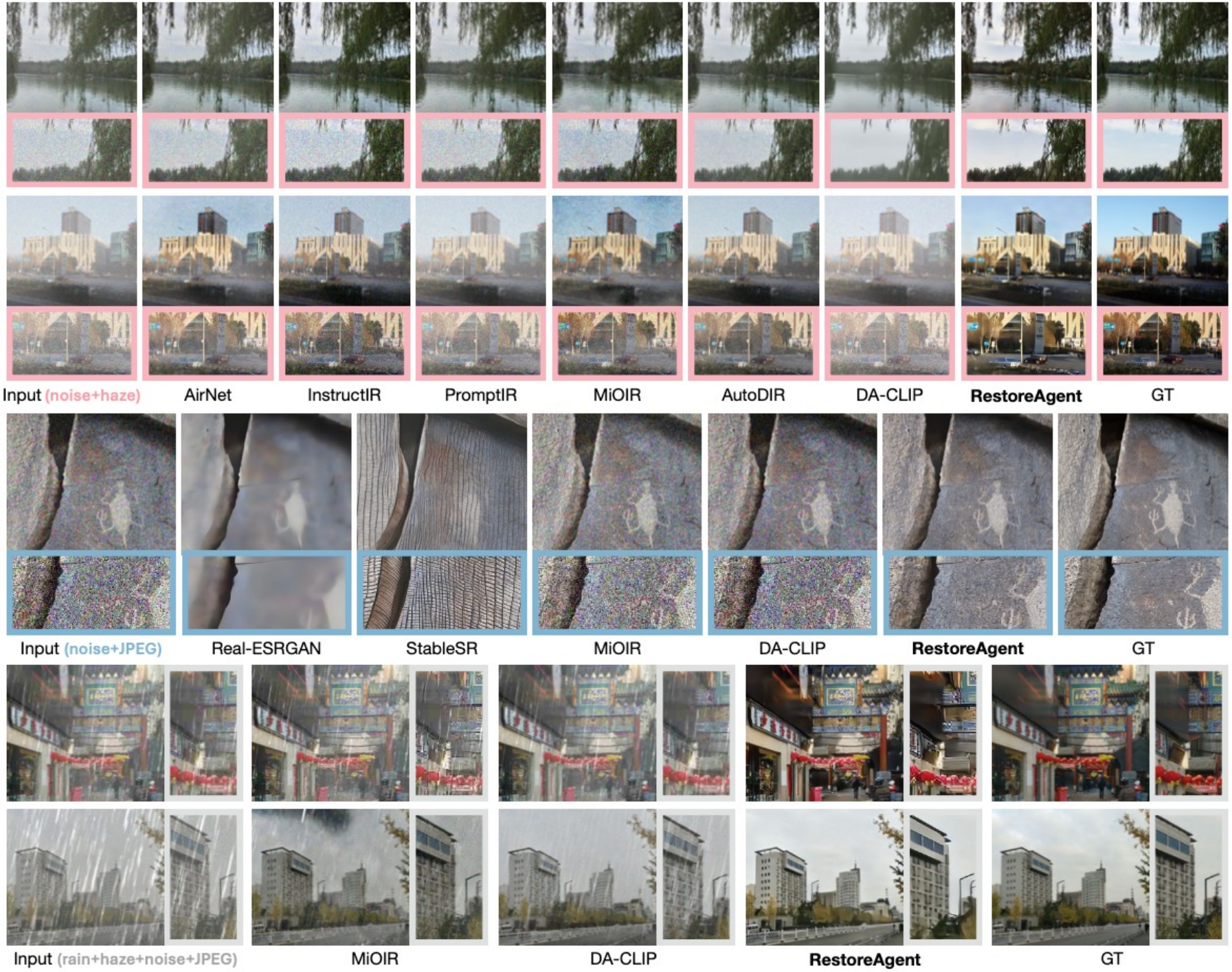
	Noise + JPEG						Low Light + Noise					
	PSNR ↑	SSIM ↑	LPIPS ↓	DISTS ↓	balanced ↑	ranking /17	PSNR ↑	SSIM ↑	LPIPS ↓	DISTS ↓	balanced ↑	ranking /10
Random Order & Model	24.52	0.7273	0.2889	0.2212	1.47	6.7	15.57	0.6541	0.4351	0.2588	1.98	3.9
Random Oder + Predict Model	25.24	0.7765	0.2327	0.1960	3.07	4.2	15.62	0.6887	0.3651	0.2283	3.03	3.0
Random Model + Predict Order	24.90	0.7568	0.2597	0.2132	2.03	6.0	17.57	0.7044	0.3685	0.2324	3.75	2.3
Pre-defined Oder and Model	25.29	0.7828	0.2366	0.2037	2.47	5.3	17.75	0.7098	0.3385	0.2260	3.93	2.1
Human Expert	25.06	0.7588	0.2551	0.2121	2.25	5.5	18.05	0.7239	0.3278	0.2220	4.29	1.9
RestoreAgent	25.32	0.7806	0.2308	0.1958	3.17	3.9 ↑1.6	17.80	0.7226	0.3259	0.2138	4.39	1.7 ↑0.2

	Motion Blur + Noise + JPEG						Rain + Noise + JPEG					
	PSNR ↑	SSIM ↑	LPIPS ↓	DISTS ↓	balanced ↑	ranking /64	PSNR ↑	SSIM ↑	LPIPS ↓	DISTS ↓	balanced ↑	ranking /64
Random Order & Model	24.81	0.7816	0.2381	0.1747	2.32	19.5	25.64	0.7970	0.2412	0.2020	2.90	16.1
Random Oder + Predict Model	24.73	0.7787	0.2261	0.1684	2.69	16.1	25.67	0.8008	0.2368	0.1956	3.11	15.0
Random Model + Predict Order	24.95	0.7912	0.2263	0.1647	3.18	13.6	26.14	0.8074	0.2314	0.1996	3.49	13.3
Pre-defined Oder and Model	24.84	0.7895	0.2305	0.1662	2.97	15.0	25.80	0.7981	0.2360	0.2041	2.83	16.7
Human Expert	25.20	0.795	0.2205	0.1646	3.82	9.0	25.99	0.8063	0.2258	0.1992	3.58	12.6
RestoreAgent	25.16	0.7939	0.2042	0.1546	4.35	4.6 ↑4.4	26.38	0.8136	0.2200	0.1891	4.67	6.4 ↑6.2

	Haze + Noise + JPEG						Haze + Rain + Noise + JPEG					
	PSNR ↑	SSIM ↑	LPIPS ↓	DISTS ↓	balanced ↑	ranking /64	PSNR ↑	SSIM ↑	LPIPS ↓	DISTS ↓	balanced ↑	ranking /287
Random Order & Model	18.98	0.7156	0.3267	0.2212	1.52	23.4	15.13	0.6300	0.4464	0.2800	1.28	102.5
Random Oder + Predict Model	19.00	0.7235	0.3133	0.2081	2.03	20.4	17.45	0.6897	0.3692	0.2400	2.86	72.8
Random Model + Predict Order	19.67	0.7653	0.2778	0.2010	2.95	15.9	19.79	0.7833	0.2815	0.1991	5.66	16.9
Pre-defined Oder and Model	19.47	0.7803	0.2641	0.1912	3.51	12.4	19.29	0.7785	0.2815	0.1974	5.502	26.1
Human Expert	19.50	0.7753	0.2703	0.1982	3.36	12.7	19.39	0.7802	0.2928	0.2043	5.503	21.3
RestoreAgent	19.55	0.7794	0.25663	0.1863	3.93	8.4 ↑4.3	19.72	0.7816	0.2741	0.1903	5.86	9.7 ↑11.6

	Motion Blur + Rain + Noise + JPEG						Average Result Across All Datasets					
	PSNR ↑	SSIM ↑	LPIPS ↓	DISTS ↓	balanced ↑	ranking /287	PSNR ↑	SSIM ↑	LPIPS ↓	DISTS ↓	balanced ↑	ranking (%)
Random Order & Model	21.96	0.6672	0.3366	0.2239	2.57	85.3	21.31	0.7139	0.3246	0.2241	1.92	34.7
Random Oder + Predict Model	22.11	0.6667	0.3038	0.2122	3.66	58.8	21.74	0.7385	0.2848	0.2045	2.89	26.1
Random Model + Predict Order	22.74	0.6996	0.2794	0.1979	5.39	24.7	22.42	0.7574	0.2750	0.2027	3.44	22.7
Pre-defined Oder and Model	22.35	0.6862	0.2858	0.1997	4.65	35.7	22.38	0.7639	0.2644	0.1986	3.48	22.1
Human Expert	22.96	0.7092	0.2861	0.2031	5.42	21.2	22.51	0.7634	0.2670	0.2014	3.73	19.5
RestoreAgent	22.95	0.7097	0.2615	0.1887	6.35	5.7 ↑15.5	22.61	0.7700	0.2513	0.1890	4.38	12.9 ↑6.6

Results


Input (noise+haze)
AirNet
InstructIR
PromptIR
MiOIR
AutoDIR
DA-CLIP
RestoreAgent
GT
Input (noise+JPEG)
Real-ESRGAN
StableSR
MiOIR
DA-CLIP
RestoreAgent
GT
Input (rain+haze+noise+JPEG)
MiOIR
DA-CLIP
RestoreAgent
GT

Results

	noise + JPEG				haze + noise				rain + haze + noise				rain + haze + noise + JPEG			
	PSNR ↑	SSIM ↑	LPIPS ↓	DISTS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	DISTS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	DISTS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	DISTS ↓
Real-ESRGAN [48]	23.43	0.7242	0.3022	0.2106	-	-	-	-	-	-	-	-	-	-	-	-
StableSR [47]	17.61	0.4464	0.3705	0.2124	-	-	-	-	-	-	-	-	-	-	-	-
AirNet [31]	-	-	-	-	17.56	0.5897	0.5569	0.2964	18.22	0.6767	0.4314	0.2336	-	-	-	-
PromptIR [40]	-	-	-	-	16.13	0.5428	0.6696	0.3544	17.81	0.7099	0.4506	0.2317	-	-	-	-
MiOIR [27]	23.98	0.6961	0.3266	0.2325	15.79	0.4790	0.7118	0.3628	16.22	0.6388	0.4719	0.2771	13.80	0.6410	0.4875	0.2939
InstructIR [14]	-	-	-	-	17.36	0.4288	0.7696	0.3646	19.45	0.6897	0.3994	0.2170	-	-	-	-
DA-CLIP [37]	22.47	0.6128	0.3525	0.2287	16.98	0.7061	0.3901	0.2737	15.44	0.6011	0.4597	0.2754	15.30	0.6863	0.3871	0.2627
AutoDIR [24]	-	-	-	-	17.51	0.6942	0.4248	0.2444	19.22	0.7705	0.3043	0.1802	-	-	-	-
RestoreAgent	25.32	0.7806	0.2308	0.1958	20.47	0.8053	0.2193	0.1758	19.53	0.8237	0.2166	0.1638	19.72	0.7816	0.2741	0.1903

Outline

- Introduction
- Framework
- Method
- Experiment
- Conclusion

Conclusion

- First identifies several critical factors in processing multi-degraded images. such as the **sequence of task execution**, the importance of **model selection**, and the **limitations of the all-in-one** approach.

- Introduce RestoreAgent, an agent model capable of making **intelligent processing** decisions based on the degradation characteristics of the input image.

- Experimental results demonstrate that our pipeline for handling multi-degraded images **outperforms the all-in-one approach**. Furthermore, the performance of decision-making results significantly **exceeds those made by human experts**.