UNLOCKING ALL-IN-ONE RESTORATION FROM PRE-TRAINED DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Pre-trained text-to-image and text-to-video diffusion models (PTDMs) have enabled significant advancements in restoration by serving as powerful clean-image priors. Existing restoration methods typically exploit these priors through backbone fine-tuning, ControlNet-style modules, explicit degradation operators, or inversion. In this work, we investigate whether a frozen PTDM intrinsically possesses restoration capabilities that can be elicited through its native conditioning pathway to directly steer noisy degraded states toward clean outputs. We first probe the conditioning pathway of PTDMs through text prompts and text-token optimization, and find that neither effectively enables restoration. Interestingly, optimizing the conditioning prompt directly at the text-encoder output unlocks strong restoration behavior from the PTDM. However, naively learning this prompt to denoise noisy degraded states to a clean output is unstable due to a misalignment with the reverse sampling trajectory. To resolve this, we learn prompts within a diffusion bridge formulation that aligns training and inference dynamics, enforcing a coherent denoising path from noisy degraded states to clean images. Furthermore, we introduce a sampling-time partial inversion strategy to improve restoration performance and a prompt composition scheme for handling mixed degradations. Instantiated on the WAN text-to-video and FLUX text-to-image models, our approach elegantly enables these PTDMs to achieve strong and competitive restoration performance. The code will be made publicly available.

1 INTRODUCTION

Restoration aims to remove degradations such as haze, rain, snow, motion blur, and low light from images or videos. Recent methods leverage pre-trained text-to-image or video diffusion models (PTDMs) such as Stable Diffusion (Rombach et al., 2022; Podell et al., 2023), FLUX (Labs, 2024), and WAN (Wan et al., 2025). Trained on massive data, these models possess strong priors over the natural image distribution and can generate high-quality content, making them attractive backbones for restoration. Most existing approaches treat PTDMs as a *clean-image prior*, using their generative capability to produce a clean output consistent with the degraded input (see Fig. 1). As a result, they envisage the PTDM's role as a clean-signal denoiser and add external mechanisms to enforce consistency with the degraded input. For instance, some methods (Jiang et al., 2023; Lin et al., 2024b) leverage the PTDM by conditioning on the degraded input via fine-tuning or by using ControlNet-style modules (Mandal et al., 2025; Zhang et al., 2025) that inject information from the degraded input into the denoising process. Alternatively, plug-and-play methods (Martin et al., 2024; Fei et al., 2023) use explicit degradation operators to enforce consistency with the degraded input during sampling, while diffusion-bridge approaches leverage PTDMs as a clean-image prior (Wang et al., 2025; HOU et al., 2026) and fine-tune them for operating on degraded inputs.

Interestingly, editing tasks have shown that PTDM sampling trajectories can be effectively *steered* through language while keeping the model frozen, leveraging its priors to enable strong semantic edits. However, directly transferring this paradigm to low-level restoration tasks is largely ineffective. Inversion-based editing methods (Hertz et al., 2022; Mokady et al., 2023) alter image details, while partial-noising approaches such as SDEdit (Meng et al., 2021) preserve the input structure but fail to remove degradations from noisy degraded states under natural-language guidance (e.g., "remove haze"), by simply suppressing the noise (see Appendix B.2). This motivates a key question:

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.

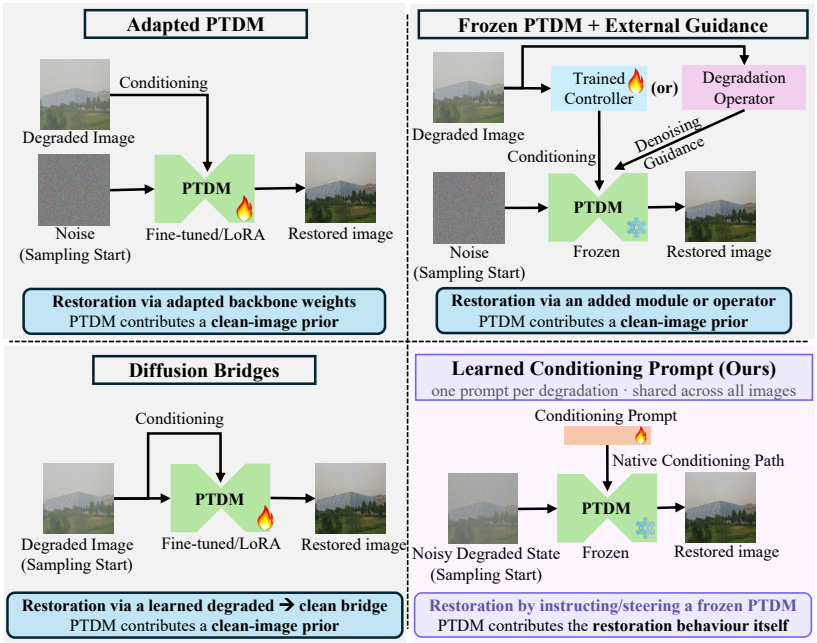
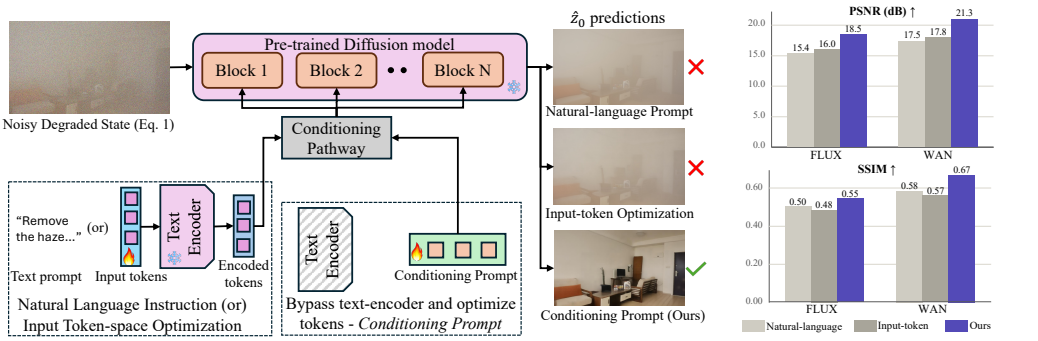


Figure 1: Prior methods use PTDMs primarily as clean-image priors and rely on additional mechanisms to enable effective restoration. In contrast, we learn an instruction in the native conditioning space that directly steers a frozen PTDM from noisy degraded states to clean outputs.

Do PTDMs intrinsically possess restoration capabilities that can be elicited through their native conditioning pathway so as to directly steer noisy degraded states toward clean outputs?

In this paper, we investigate this by probing the native conditioning pathway of frozen PTDMs. Concretely, we construct noisy intermediate states $x_t = \alpha_t x_{\text{deg}} + \sigma_t \epsilon$ from the degraded input, and ask whether a conditioning signal can guide the PTDM to denoise these states toward clean outputs. We first find that both natural-language prompts and learned input token embeddings (e.g. textual inversion) fail to effectively elicit restoration behavior (see Fig. 2). In contrast, bypassing this input token-space and directly optimizing the conditioning signal at the output of the text encoder, which we call the *conditioning prompt*, unlocks strong restoration performance (see Fig. 2), i.e. the frozen PTDM maps noisy degraded inputs toward clean targets, suggesting that restoration capabilities exist in the PTDM but are not accessible through the input token-space. However, naively learning the conditioning prompt to denoise a noisy degraded state leads to a mismatch between the state distributions visited during training and inference. We call this *trajectory mismatch* and address it by learning conditioning prompts within a *diffusion-bridge* formulation that aligns trajectories during training and inference-time sampling. We further introduce a partial inversion strategy to improve sampling-time performance and a prompt composition approach for handling mixed degradations. We instantiate our framework on the WAN 2.1 (Wan et al., 2025) and the FLUX.1-dev (Labs, 2024) models to tackle the challenging all-in-one restoration (AiOR) task. Analogous to instruction-guided non-blind AiOR methods (Conde et al., 2025; Lin et al., 2024b; Liu et al., 2024; Tian et al., 2025), we learn a conditioning prompt for each degradation, enabling the PTDM to function as a prompt-conditioned AiOR framework. To summarize, our contributions are as follows:

1. We reveal that PTDMs *intrinsically* possess restoration capabilities that are largely inaccessible through the input token-space but can be elicited by optimizing the conditioning signal at the text-encoder output.
2. We identify a train-test trajectory mismatch that makes naive conditioning prompt learning unstable, and instead optimize the prompt within a diffusion-bridge formulation that aligns states seen during training and inference.
3. We introduce sampling-time partial inversion that significantly improves conditioning-prompt performance, together with a prompt composition strategy for mixed degradations.
4. We instantiate our framework on WAN and FLUX for AiOR, showing that frozen PTDMs guided by conditioning prompts achieve strong and competitive restoration performance.



(a) Schematic and qualitative $\hat{z}_0$ predictions

(b) Quantitative results for $\hat{z}_0$ predictions

Figure 2: **Prompting a frozen PTDM for restoration.** (a) Natural-language and optimized input-token prompts fail to remove degradations (in this case haze) from noisy degraded states, while our *conditioning prompt*, optimized at the text-encoder output, enables restoration at the same parameter budget. All variants are trained to predict the clean target from noisy degraded states (Eq. 2). (b) Quantitative comparison averaged across multiple degradations.

2 Related Works

2.1 Pre-trained Diffusion Models for Editing

PTDMs have been widely explored for editing tasks. Inversion-based editing methods aim to recover a noise representation that reconstructs the input under the PTDM (Brack et al.; Cao et al., 2023; Mokady et al., 2023) and then apply structural or language-driven controls during denoising to realize edits (Hertz et al., 2022; Ju et al., 2023; Tumanyan et al., 2023). While effective for semantic manipulation, these mechanisms struggle to preserve fine-grained fidelity due to imperfect inversion, which is particularly problematic for restoration. SDEdit (Meng et al., 2021) avoids inversion by partially noising the input and denoising from that state, offering a trade-off between fidelity and generative edits via the chosen noise level. However, replacing the editing instruction with a restoration instruction (e.g., "remove haze") performs poorly, as the model suppresses noise without removing degradations. Motivated by this gap, we investigate whether specific conditioning signals can steer a frozen PTDM toward clean targets during denoising.

2.2 All-in-One Restoration

AiOR aims to tackle multiple degradations using a single model. Early works such as All-in-One (Li et al., 2020) and TransWeather (Valanarasu et al., 2022) explored unified architectures and training strategies. PromptIR (Potlapalli et al., 2024) used learned prompts to encode degradation-specific information, AWRaCLe (Rajagopalan & Patel, 2024) leverages visual in-context cues, and instruction-guided methods such as InstructIR (Conde et al., 2025), DFPIR (Tian et al., 2025), and PixWizard (Lin et al., 2024b) use language to select restoration behavior. These condition-driven AiOR methods reuse a shared restoration backbone across degradations, with prompts or instructions specifying the desired task. With the rise of PTDMs, several methods exploit their clean-image priors to improve restoration (Jiang et al., 2023; Liu et al., 2024; Rajagopalan et al., 2025). AutoDIR (Jiang et al., 2023), UniLDiff (Cheng et al., 2025), and PixWizard (Lin et al., 2024b) adapt PTDMs to condition on degraded inputs using full or parameter-efficient fine-tuning. Other methods keep the PTDM frozen but train ControlNet-style branches (Mandal et al., 2025; Zhang et al., 2025) or task-plugins (Liu et al., 2024) to inject guidance derived from the degraded input during the sampling process. Plug-and-play methods (Martin et al., 2024; Fei et al., 2023) require degradation operators to enforce input fidelity while sampling using the PTDM's clean-image prior. Inversion-based methods (Xiao et al., 2024; Chihaoui & Favaro, 2025) invert the degraded input into the diffusion trajectory and modify sampling along the PTDM's clean-image prior. The above methods treat the PTDM as a clean-signal denoiser and use specific mechanisms to enforce consistency with the degraded input. Recent diffusion-bridge methods leverage PTDMs as clean-image priors for diffusion bridges (Zhou et al., 2023) and adapt them using fine-tuning (HOU et al., 2026) or ControlNet modules (Wang et al., 2025) to follow bridge dynamics. Unlike these approaches, we elicit restoration from a frozen PTDM through its native text-conditioning pathway.

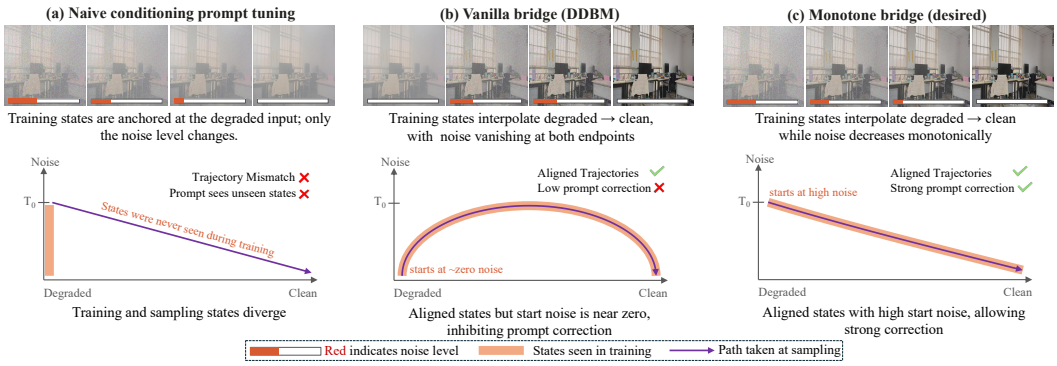


Figure 3: **Top:** Training states for the three forward processes from higher (left) to lower (right) timesteps. In all cases, the *conditioning prompt* is learned to predict clean endpoint; only intermediate training states differ. **Bottom:** Trajectory mismatch illustration. Thick lines denote training states and thin lines sampling trajectories. Naive prompt tuning remains anchored to the degraded signal during training, so sampling queries the prompt on unseen, progressively cleaner states. Both bridges align training and sampling. However, DDBM starts at near-zero noise, limiting prompt correction strength, while the monotone bridge starts at high noise and enables stronger correction.

3 Proposed Method

In this section, we explain the core innovations of our method.

3.1 Guiding pre-trained diffusion models for restoration

Given paired data $(x_{\text{deg}}, x_{\text{clean}})$, our goal is to produce a restored result, $x_{\text{pred}} \approx x_{\text{clean}}$. We leverage pre-trained text-to-image/video diffusion models (PTDMs) that operate in the latent space of a VAE whose encoder is $\mathcal{E}$. Let $z_{\text{deg}} = \mathcal{E}(x_{\text{deg}})$ and $z_{\text{clean}} = \mathcal{E}(x_{\text{clean}})$ denote the corresponding VAE latents, and let $\epsilon \sim \mathcal{N}(0, I)$. Following the flow-matching parameterization used by WAN/FLUX where the forward process typically noises a clean latent, we construct an analogous *noisy degraded input* as

$$z_t^{\text{deg}} = \big(1 - t\big)\, z_{\text{deg}} + t\, \epsilon, \qquad t \in [0, T_0], \qquad 0 < T_0 \leq 1. \tag{1}$$

Since PTDMs possess strong clean-image priors and Eq. 1 maps z_{deg} to a state compatible with reverse-time sampling, we ask whether z_t^{deg} can be restored by steering the PTDM's reverse time-sampling with a restoration instruction. Concretely, we condition the PTDM on a text prompt c (e.g., "Remove the haze."), for which we obtain token embeddings $y = \tau(c)$ and context embeddings $e = \text{TE}(y)$, where τ denotes tokenization/embedding lookup, and TE denotes the frozen text encoder. e guides the PTDM through its conditioning pathway (e.g. cross-attention). As shown in Fig. 2, such natural-language prompts fail to remove degradations. To test whether this failure is due to limited language expressivity, we optimize input token embeddings U before the frozen text encoder, producing $u = \text{TE}(U)$ while optimizing to reconstruct the clean latent

$$\mathcal{L}_{\text{restore}} = \mathbb{E}_{(z_{\text{deg}}, z_{\text{clean}}),\, t,\, \epsilon}\left[\left\|\hat{z}_{0,\theta}(z_t^{\text{deg}}, t; u) - z_{\text{clean}}\right\|_2^2\right], \tag{2}$$

where $\hat{z}_{0,\theta} = z_t^{\text{deg}} - t v_\theta(z_t^{\text{deg}}, t; u)$ denotes the backbone's clean-latent prediction. However, the optimized prompt continues to produce a degraded output (Fig. 2).

In contrast, bypassing the input token-space and directly optimizing the conditioning at the text encoder's output space, which we call the *conditioning prompt* p, elicits a strong restoration response. This conditioning prompt optimization (using p in Eq. 2) drives $\hat{z}_{0,\theta}$ toward z_{clean} and produces visibly restored predictions (Fig. 2). This suggests restoration capability is present in the frozen PTDM but is not effectively elicited through the input token-space. However, learning the conditioning prompt this way creates a train-test *trajectory mismatch*. Specifically, Eq. 1 anchors training states at z_{deg}, exposing p to samples $\{z_t^{\text{deg}}\}_{t\in[0,T_0]}$ along the "degraded → degraded+noise" path (see Fig. 3 (a)). During sampling, the states at each timestep progressively move from z_{deg} toward z_{clean}, exposing p to unseen states no longer anchored at z_{deg}, leading to instability and artifacts (see Sec. 4.4).

3.2 Addressing Trajectory Mismatch

The above mismatch motivates constructing training trajectories whose intermediate states smoothly transition from z_{deg} to z_{clean}, i.e., a *diffusion bridge* between the paired endpoints. A bridge yields intermediate states that (i) interpolate from z_{deg} to z_{clean} and (ii) can be used in both training and inference, thereby mitigating the train-test trajectory mismatch (see Fig. 3). Thus, a bridge ensures that the conditioning prompt p encounters similar intermediate states during training and inference.

Denoising diffusion bridge models (DDBMs) (Zhou et al., 2023) construct Gaussian distributions

$$q(z_t \mid z_{\text{clean}}, z_{\text{deg}}) = \mathcal{N}\Big(z_t;\ a_t\, z_{\text{deg}} + b_t\, z_{\text{clean}},\ s_t^2 I\Big), \tag{3}$$

with (a_t, b_t, s_t) defined by the chosen bridge schedule at time t. This yields a closed-form expression for sampling an input from the forward process as $z_t^{\text{br}} = a_t z_{\text{deg}} + b_t z_{\text{clean}} + s_t \epsilon$. To match the flow-matching input form expected by WAN/FLUX models (Eq. 1), we re-parametrize the bridge input by viewing $a_t\, z_{\text{deg}} + b_t\, z_{\text{clean}}$ as the signal component of the input and ϵ as the sampled noise. Setting $\sigma_t = \frac{s_t}{1+s_t}$, where σ_t is the desired noise coefficient for WAN/FLUX, we get

$$y_t^{\text{br}} = (1-\sigma_t)(a_t\, z_{\text{deg}} + b_t\, z_{\text{clean}}) + \sigma_t\, \epsilon, \tag{4}$$

which is in the form of Eq. 1. We then optimize the prompt p to predict the clean endpoint from y_t^{br}:

$$\min_p\ \mathbb{E}_{(z_{\text{deg}}, z_{\text{clean}}),\, t,\, \epsilon}\Big[\big\|\hat{z}_{0,\theta}(y_t^{\text{br}}, t; p) - z_{\text{clean}}\big\|_2^2\Big], \tag{5}$$

where $\hat{z}_{0,\theta}$ is the clean prediction of the PTDM (Sec. 3.1). Intuitively, the above process can be viewed as p observing various intermediate signals of the form $a_t\, z_{\text{deg}} + b_t\, z_{\text{clean}}$ and providing the necessary guidance to restore them. During sampling, $\hat{z}_{0,\theta}$ is used to step through the reverse trajectory (Zhou et al., 2023), ensuring p encounters similar intermediate samples as in training (Fig. 3 (b)).

Despite aligning training and sampling, a DDBM-like bridge has a key limitation: its variance s_t^2 is *small near both endpoints* and peaks in between (top row of Fig. 3 (b)). Thus, states near the degraded endpoint are nearly deterministic, giving the frozen generative prior little stochasticity to redirect the trajectory toward z_{clean} and causes under-restoration (Sec. 4.4). Thus, we require a bridge which *starts noisy* near the degraded input and whose noise *decreases monotonically* as the signal approaches the clean endpoint (see Fig. 3 (c)). Concretely, we want intermediate states of the form

$$z_t^{\text{br}} = \mu(t) + s(t)\, \epsilon, \qquad \mu(t) = \lambda(t)\, z_{\text{deg}} + \big(1-\lambda(t)\big)\, z_{\text{clean}}, \qquad \epsilon \sim \mathcal{N}(0, I), \tag{6}$$

where $t \in [0, T_0]$ (with $T_0 \le 1$), $\lambda(0) = 0$, $\lambda(T_0) = 1$, and $s(t)$ *increases* with t. This exposes the conditioning prompt early on to noisy degraded states allowing strong restoration, while gradually moving to cleaner states, avoiding the train-test trajectory mismatch. For this purpose, we adopt the energy-oriented diffusion bridge (EBR) (HOU et al., 2026) which constructs bridge states as

$$z_t^{\text{ebr}} = (1-t)\Big[\Big(1 - \frac{t}{T_0}\Big)\, z_{\text{clean}} + \Big(\frac{t}{T_0}\Big)\, z_{\text{deg}}\Big] + t\, \epsilon, \tag{7}$$

which satisfies our desired endpoint behaviour. Importantly, our contribution is *not* the bridge schedule itself, but showing that this monotone noisy-to-clean bridge is the missing ingredient that makes our prompt-only restoration feasible on frozen PTDMs. We train p with the objective

$$\min_p\ \mathbb{E}_{(z_{\text{deg}}, z_{\text{clean}}),\, t,\, \epsilon}\Big[\big\|\hat{z}_{0,\theta}(z_t^{\text{ebr}}, t; p) - z_{\text{clean}}\big\|_2^2\Big], \tag{8}$$

where $\hat{z}_{0,\theta}$ is the clean estimate from the PTDM conditioned using p. At inference, we run a DDIM-like (Song et al., 2020) sampler using the same bridge schedule to align intermediate states visited during training and sampling, mitigating trajectory mismatch and enabling successful prompt training.

3.3 Improving Sampling-time Performance

We now introduce two sampling-time techniques that improve restoration performance.

Image-specific start state. At inference, sampling begins from $z_{T_0}^{\text{ebr}} = (1-T_0)\, z_{\text{deg}} + T_0\, \epsilon$ where $\epsilon \sim \mathcal{N}(0, I)$ is drawn independently of the input z_{deg}. Instead, we obtain a starting noise specific to z_{deg} through *partial inversion*. We run the deterministic sampler of Sec. 3.2 in reverse from z_{deg} for K steps under a *null condition*, rather than the learned prompt p, yielding $\hat{\epsilon} = \text{Inv}_K(z_{\text{deg}}; \phi)$. The start state is then constructed from this inverted noise as

$$\hat{z}_{T_0}^{\text{ebr}} = (1-T_0)\, z_{\text{deg}} + T_0\, \hat{\epsilon}. \tag{9}$$

Table 1: Quantitative results of our prompt learning approach on the FLUX model and SOTA AiOR approaches for images from OOD, mixed and unseen degradations. H-Haze, R-Rain, S-Snow, B-Blur, L-Low-light, and U-Unseen. Red: best and blue: second best.

Methods	HazeRD (H)						LHP (R)						WeatherBench (S)					
	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑
DCPT (ICLR'25)	15.23	0.798	0.185	0.123	0.672	68.55	27.56	0.831	0.192	0.134	0.481	58.89	21.75	0.762	0.234	0.159	0.327	44.56
DFPIR (CVPR'25)	16.09	0.811	0.190	0.131	0.641	67.65	28.98	0.876	0.133	0.093	0.484	58.93	21.41	0.758	0.241	0.164	0.345	45.30
AutoDIR (ECCV'24)	15.58	0.812	0.224	0.174	0.636	67.49	29.62	0.859	0.169	0.121	0.520	58.06	22.44	0.761	0.226	0.152	0.372	46.29
PixWizard (ICLR'25)	16.53	0.636	0.288	0.177	0.682	67.54	22.14	0.763	0.209	0.144	0.564	56.12	16.47	0.504	0.450	0.277	0.465	49.51
RestoreVAR (ICLR'26)	15.21	0.679	0.213	0.130	0.680	68.87	25.69	0.771	0.197	0.137	0.534	57.65	21.69	0.740	0.234	0.159	0.371	46.02
SLAIR (ECCV'26)	16.19	0.726	0.210	0.131	0.667	66.77	24.78	0.780	0.188	0.136	0.535	56.10	21.20	0.731	0.245	0.167	0.364	45.69
FLUX (Ours)	17.02	0.813	0.195	0.143	0.684	69.46	27.82	0.877	0.136	0.096	0.544	58.27	22.23	0.762	0.265	0.182	0.386	46.47

Methods	4KRD (B)						SICE (L)						TOLED (U)					
	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑
DCPT (ICLR'25)	27.10	0.854	0.165	0.129	0.510	58.90	13.54	0.574	0.315	0.197	0.589	64.17	20.68	0.748	0.258	0.193	0.427	48.22
DFPIR (CVPR'25)	27.54	0.861	0.157	0.124	0.518	60.34	15.25	0.706	0.270	0.175	0.538	64.51	12.90	0.604	0.322	0.219	0.386	43.01
AutoDIR (ECCV'24)	27.03	0.842	0.170	0.133	0.521	58.43	13.91	0.617	0.283	0.186	0.594	64.71	20.11	0.676	0.265	0.203	0.451	51.01
PixWizard (ICLR'25)	19.87	0.600	0.295	0.183	0.515	55.14	15.31	0.490	0.587	0.359	0.493	51.98	19.23	0.659	0.316	0.217	0.416	43.32
RestoreVAR (ICLR'26)	23.49	0.736	0.160	0.111	0.561	64.66	15.76	0.610	0.276	0.166	0.607	67.34	23.54	0.733	0.218	0.168	0.448	52.41
SLAIR (ECCV'26)	23.88	0.743	0.197	0.128	0.552	54.57	13.37	0.553	0.316	0.199	0.588	62.17	22.51	0.742	0.230	0.165	0.462	51.26
FLUX (Ours)	24.61	0.797	0.106	0.079	0.673	72.81	16.04	0.659	0.270	0.175	0.608	67.13	19.70	0.754	0.199	0.161	0.527	62.31

Methods	POLED (U)						CDD (S+H)						LOLBlur (L+B)		Average					
	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑
DCPT (ICLR'25)	8.69	0.346	0.665	0.471	0.296	37.50	18.16	0.816	0.233	0.153	0.640	64.68	0.383	41.20	19.08	0.716	0.280	0.194	0.480	54.07
DFPIR (CVPR'25)	10.27	0.363	0.687	0.493	0.228	35.13	15.17	0.733	0.277	0.206	0.550	61.42	0.333	37.31	18.45	0.714	0.284	0.200	0.447	52.62
AutoDIR (ECCV'24)	13.27	0.429	0.582	0.427	0.328	46.67	16.54	0.753	0.200	0.178	0.627	67.00	0.363	39.27	19.81	0.718	0.264	0.196	0.490	55.43
PixWizard (ICLR'25)	15.35	0.422	0.713	0.479	0.301	36.79	14.96	0.472	0.403	0.248	0.602	61.59	0.366	39.32	17.48	0.568	0.407	0.260	0.489	51.25
RestoreVAR (ICLR'26)	10.60	0.370	0.595	0.452	0.309	48.25	16.51	0.589	0.281	0.161	0.620	65.35	0.418	46.97	19.06	0.653	0.271	0.185	0.505	57.50
SLAIR (ECCV'26)	14.71	0.478	0.515	0.373	0.353	47.42	17.40	0.632	0.264	0.169	0.596	59.63	0.440	44.13	19.26	0.673	0.271	0.183	0.506	54.19
FLUX (Ours)	19.90	0.575	0.522	0.400	0.283	45.48	17.49	0.745	0.191	0.155	0.666	67.87	0.440	51.43	20.60	0.748	0.235	0.174	0.534	60.14

The noise *level* is unchanged, keeping the training and sampling distributions aligned. This technique significantly improves restoration performance (see Sec. 4.4). Note that, unlike inversion-based restoration methods (Xiao et al., 2024), our partial inversion only constructs an intermediate start state; the restoration mapping is still performed by the PTDM using our conditioning prompt.

Composing prompts for mixed degradations. We learn one conditioning prompt $p^{(m)}$ per degradation m, analogous to instruction-guided AiOR models (Conde et al., 2025; Lin et al., 2024b; Tian et al., 2025; Liu et al., 2024; Rajagopalan & Patel, 2024) that reuse a single trained backbone with task-specific conditioning. Here, the frozen PTDM acts as the all-in-one restorer as it contains the restoration capability for multiple degradations. An input exhibiting M *mixed* degradations has no single corresponding prompt. Since each prompt induces a velocity, we compose them at every denoiser call. A simple composition is the mean of velocities $v_{\text{avg}} = \frac{1}{M}\sum_m v_m$, which allocates direction in proportion to magnitude. However, velocity magnitude reflects correction scale rather than degradation relevance, underweighting degradations requiring smaller-norm corrections. We therefore normalize the directional contributions of each prompt while retaining its magnitude:

$$d = \frac{1}{M}\sum_{m=1}^{M}\frac{v_m}{\|v_m\|}, \qquad v_{\text{mix}} = \frac{d}{\|d\|}\,\|v_{\text{avg}}\|. \tag{10}$$

This approach enables superior performance while handling mixed degradations (see Sec. 4.4).

4 Experiments

We present implementation details, experimental setup, evaluations and ablations. Differences from parameter-efficient fine-tuning and soft prompt tuning, along with limitations, are in Appendix B.

4.1 Implementation Details

We apply our conditioning prompt approach for the challenging all-in-one restoration (AiOR) task using WAN 2.1 (Wan et al., 2025) text-to-video (1.3B parameters) and FLUX.1-dev (Labs, 2024) text-to-image (12B parameters) models. Each conditioning prompt has 226 tokens for WAN (~ 0.34M parameters) and 512 tokens for FLUX (~ 1.57M parameters). Prompts are learned as residuals over the null-text context with zero-initialized scalar gates at attention layers (see Appendix D). We use a start noising level $T_0 = 0.4$ (Eq. 7) for training and inference, number of sampling steps $N = 6$, and number of inversion steps $K = 6$ for WAN and $K = 10$ for FLUX (see Appendix C).

Table 2: Quantitative results of our prompt learning approach on the WAN model alongside SOTA image and video restoration approaches (ViWS-Net and AverNet) for the task of video AiOR on OOD datasets. H-Haze, R-Rain, S-Snow, B-Blur, and L-Low-light. Red: best and blue: second best.

Methods	RHVD (H)			LasVR (R)							NTURain (R)			BSD (B)						
	C↑	M↑	Do↑	P↑	S↑	L↓	D↓	C↑	M↑	Do↑	C↑	M↑	Do↑	P↑	S↑	L↓	D↓	C↑	M↑	Do↑
DCPT (ICLR'25)	0.571	60.39	0.532	29.59	0.853	0.187	0.155	0.553	62.81	0.607	0.600	63.69	0.554	22.73	0.790	0.206	0.155	0.351	37.63	0.361
DFPIR (CVPR'25)	0.567	59.65	0.535	30.50	0.860	0.183	0.147	0.534	61.73	0.639	0.567	65.24	0.549	27.76	0.860	0.174	0.142	0.372	39.62	0.382
AutoDIR (ECCV'24)	0.567	60.36	0.556	29.33	0.827	0.168	0.128	0.571	63.33	0.653	0.601	64.94	0.603	27.21	0.840	0.175	0.141	0.365	37.93	0.398
PixWizard (ICLR'25)	0.577	58.62	0.543	21.61	0.598	0.251	0.174	0.587	58.72	0.515	0.596	60.00	0.475	19.71	0.657	0.329	0.212	0.366	35.44	0.274
RestoreVAR (ICLR'26)	0.586	59.27	0.580	25.11	0.688	0.237	0.159	0.547	60.85	0.618	0.594	62.33	0.552	21.93	0.744	0.207	0.143	0.401	43.19	0.392
SLAIR (ECCV'26)	0.568	56.76	0.559	27.78	0.768	0.156	0.113	0.532	54.34	0.582	0.593	57.13	0.504	21.46	0.744	0.205	0.148	0.399	38.58	0.391
ViWS-Net (ICCV'23)	0.529	58.72	0.546	29.79	0.847	0.135	0.118	0.527	61.50	0.651	0.513	58.21	0.584	21.82	0.712	0.299	0.250	0.316	32.59	0.314
AverNet (NeurIPS'24)	0.488	50.62	0.551	30.80	0.863	0.176	0.149	0.472	50.43	0.584	0.533	52.98	0.537	26.99	0.825	0.222	0.166	0.299	30.08	0.362
WAN (Ours)	0.618	64.91	0.694	30.59	0.862	0.136	0.100	0.559	62.09	0.657	0.599	63.57	0.640	24.33	0.780	0.202	0.158	0.410	47.74	0.487

Methods	AAURainSnow (S)			BVI (L)							Average						
	C↑	M↑	Do↑	P↑	S↑	L↓	D↓	C↑	M↑	Do↑	P↑	S↑	L↓	D↓	C↑	M↑	Do↑
DCPT (ICLR'25)	0.435	48.90	0.325	19.87	0.806	0.188	0.111	0.553	69.90	0.603	24.06	0.816	0.193	0.140	0.510	57.22	0.497
DFPIR (CVPR'25)	0.429	47.03	0.296	21.98	0.793	0.200	0.119	0.533	67.59	0.581	26.74	0.837	0.185	0.136	0.500	56.81	0.497
AutoDIR (ECCV'24)	0.438	47.16	0.312	16.44	0.627	0.237	0.149	0.562	68.59	0.667	24.32	0.764	0.193	0.139	0.517	57.05	0.531
PixWizard (ICLR'25)	0.471	47.22	0.304	16.04	0.517	0.417	0.239	0.527	63.80	0.666	19.12	0.590	0.332	0.208	0.520	53.96	0.463
RestoreVAR (ICLR'26)	0.442	48.57	0.370	18.52	0.675	0.216	0.120	0.552	70.98	0.687	21.85	0.702	0.220	0.140	0.520	57.53	0.533
SLAIR (ECCV'26)	0.450	45.72	0.354	19.65	0.698	0.181	0.124	0.588	69.40	0.686	22.96	0.737	0.181	0.129	0.522	53.65	0.513
ViWS-Net (ICCV'23)	0.420	48.13	0.351	14.09	0.459	0.482	0.411	0.268	32.05	0.260	21.90	0.672	0.305	0.259	0.428	48.53	0.451
AverNet (NeurIPS'24)	0.416	43.62	0.334	16.87	0.633	0.377	0.271	0.244	28.91	0.311	24.88	0.773	0.258	0.195	0.408	42.77	0.446
WAN (Ours)	0.464	49.87	0.393	19.92	0.724	0.235	0.163	0.599	72.58	0.724	24.94	0.788	0.191	0.140	0.541	60.12	0.599

4.2 Setup

We train conditioning prompts on in-domain (ID) datasets, and report performance on (i) out-of-distribution (OOD) datasets, (ii) mixed and unseen degradations, and (iii) ID test sets.

Image restoration (FLUX). We train prompts for dehazing, desnowing, deraining, low-light enhancement, and deblurring using: RESIDE (Li et al., 2019) (haze), Snow100K (Liu et al., 2018) (snow), Rain13K (Zamir et al., 2021) (rain), LOLv1 (Wei et al., 2018) (low-light), and GoPro (Nah et al., 2017) (blur). For OOD testing, we evaluate on HazeRD (Zhang et al., 2017) (haze), LHP (Guo et al., 2023) (rain), WeatherBench (Guan et al., 2025) (snow), 4KRD (Deng et al., 2021) (blur), and SICE (Cai et al., 2018) (low-light). For mixed degradations, we use the haze+snow split of CDD (Guo et al., 2024), and LOLBlur (Zhou et al., 2022b) for low-light+blur. To further test the generalization of our prompts, we evaluate on the unseen Under-Display Camera (UDC) restoration task (following (Jiang et al., 2023; Lin et al., 2024b)) using the TOLED and POLED datasets (Zhou et al., 2021). The UDC task resembles a combination of low-light enhancement, deblurring and denoising (Zhou et al., 2021). We therefore condition both TOLED and POLED with our low-light+blur prompts.

Video restoration (WAN). We train prompts for dehazing, deraining, desnowing, low-light enhancement, and deblurring using: REVIDE (Zhang et al., 2021) (haze), NTU-Rain (Chen et al., 2018) (rain), RVSD (Chen et al., 2023) (snow), SDSD (Wang et al., 2021) (low-light), and GoPro (Nah et al., 2017) (blur). For OOD testing, we use RHVD (Chu et al., 2021) (haze), NTU-Rain (Chen et al., 2018) and LasVR (Liu et al., 2019) (rain), AAURainSnow (Bahnsen & Moeslund, 2018) (snow), BVI (Lin et al., 2024a) (low-light), and BSD (Zhong et al., 2023) (blur).

Notably, our evaluation includes a substantial set of real-world degradation datasets (see Appendix G), allowing us to assess the robustness of our conditioning prompts beyond synthetic degradations. We provide the full per-dataset sample counts and details in Appendix G.

4.3 Evaluation

We evaluate our conditioning prompt-instantiated FLUX and WAN backbones on the tasks discussed in Sec. 4.2. Our goal is to demonstrate that the restoration behavior elicited by our prompts enables strong restoration performance. We also compare with state-of-the-art (SOTA) AiOR methods, and backbone-matched variants primarily using them as a reference for judging restoration quality. We mainly showcase results on OOD, mixed and unseen degradations as it provides a stronger test of the real-world robustness of our conditioning prompts. Detailed ID results are given in Appendix H.

Image restoration. We evaluate the prompt-instantiated FLUX model using distortion and perceptual metrics: PSNR (P) in dB, SSIM (S), LPIPS (L) (Zhang et al., 2018), DISTS (D) (Ding et al., 2020), CLIPIQA (C) (Wang et al., 2023), and MUSIQ (M) (Ke et al., 2021). We compare with recent AiOR

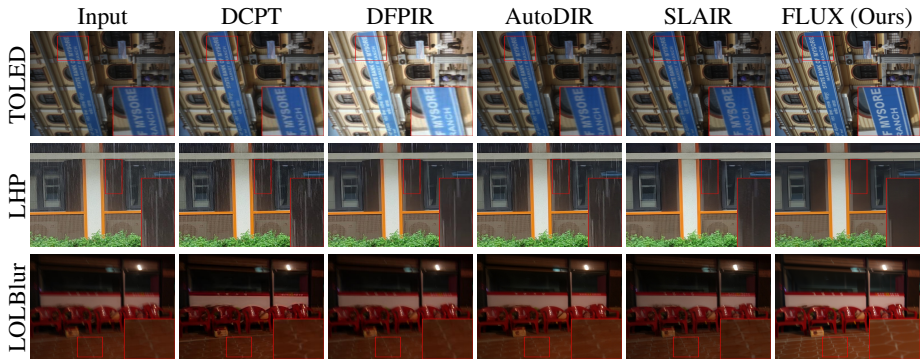


Figure 4: Qualitative results of the prompt-instantiated FLUX model and SOTA AiOR methods on OOD, mixed and unseen degradations. Our prompts enable FLUX to achieve remarkable restoration.

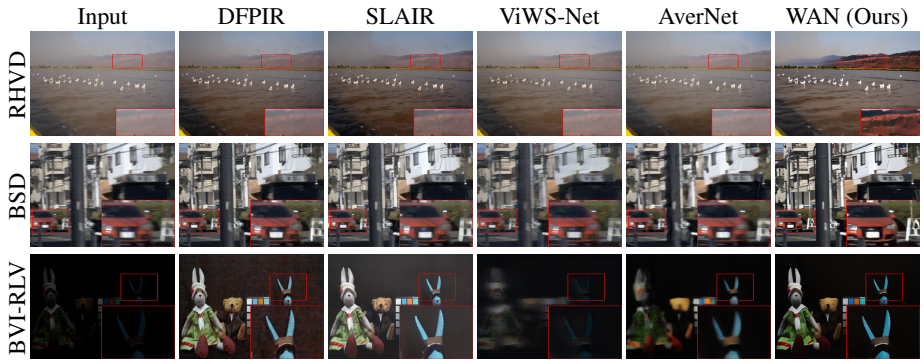


Figure 5: Qualitative results of the prompt-instantiated WAN model and SOTA AiOR approaches for OOD degradations. ViWS-Net and AverNet are video restoration approaches while others are proposed for image restoration. Our prompts elicit strong restoration from the WAN model.

methods, including DCPT (Hu et al., 2025), DFPIR (Tian et al., 2025), AutoDIR (Jiang et al., 2023), PixWizard (Lin et al., 2024b), RestoreVAR (Rajagopalan et al., 2025), and SLAIR (Liang et al., 2026), with all models retrained on the same datasets for fairness. Since latent diffusion methods are affected by the VAE reconstruction bottleneck (Lin et al., 2024b; Rajagopalan et al., 2025), PSNR/SSIM for our approach may be lower even for high-quality restorations. As shown in Table 1, our method remains highly competitive, even outperforming SOTA approaches on average.

Video restoration. For videos, we evaluate our WAN-based instantiation on the additional DOVER (Do) (Wu et al., 2023) metric to gauge video quality and compare with the video AiOR approaches ViWS-Net (Yang et al., 2023) and AverNet (Zhao et al., 2024) (re-trained on our training data). The OOD results are reported in Table 2 with image-based methods included for reference. The WAN model achieves highly competitive performance on OOD degradations. Furthermore, although non-generative methods such as DFPIR and AverNet score higher on some distortion metrics, the qualitative results (Fig. 5) show that they fail to remove distortions as effectively as our approach.

Qualitative evaluation. We visualize results on OOD, mixed, and unseen degradations in Fig. 4 for image AiOR and Fig. 5 for video AiOR. In the interest of space, we show qualitative results for select OOD, mixed, and unseen degradations; remaining datasets, ID results, and other methods' results are given in the appendix. As seen from the figures, our prompt-instantiated FLUX and WAN models showcase strong degradation removal capabilities that are on-par with SOTA approaches. In the supplementary, we include video clips for the video AiOR task. Together, these results demonstrate that a frozen PTDM, guided by our learned conditioning prompts, can achieve strong and competitive restoration performance even on real-world degradations.

Backbone matched comparisons. In addition to comparisons with SOTA, we benchmark against two low-rank adaptation (LoRA) (Hu et al., 2021) variants of the same PTDM backbone: channel concatenation of the degraded input ("Concat") and a LoRA-trained diffusion bridge ("Bridge"). "Concat" is similar to concatenation-based restoration approaches such as AutoDIR (Jiang et al., 2023) while "Bridge" is analogous to directly fine-tuning on the EBR-bridge without our prompt. Table 4



Figure 6: Naive prompt training enhances the image, but produces artifacts due to trajectory mismatch. Using DDBM causes under-enhancement. EBR provides the best results.

Table 3: Quantitative comparison of naive prompt tuning, DDBM bridge, and EBR bridge (desired). Red: best.

Training type	P↑	S↑	L↓	D↓	C↑	M↑
Naive	17.39	0.590	0.351	0.233	0.407	45.91
DDBM	20.57	0.676	0.205	0.139	0.544	62.13
EBR	21.86	0.721	0.172	0.116	0.551	64.13

Table 4: Quantitative comparisons of backbone-matched variants. Trainable parameters are the total across all our conditioning prompts. Red is best while blue is second best within each backbone.

Method	Parameters (M)		In-distribution (mean)						Out-of-distribution (mean)					
	Total	Train.	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑
FLUX + Concat (LoRA)	13,419.7	1,434.46	17.22	0.388	0.485	0.247	0.458	56.69	15.53	0.373	0.545	0.297	0.471	55.66
FLUX + Bridge (LoRA)	13,419.7	1,434.45	23.65	0.762	0.158	0.105	0.455	65.75	19.64	0.655	0.274	0.180	0.471	60.20
FLUX (Ours)	11,994.7	7.87	25.39	0.843	0.118	0.090	0.571	64.91	20.60	0.748	0.235	0.174	0.534	60.14
WAN + Concat (LoRA)	1,633.4	87.50	15.40	0.446	0.416	0.189	0.472	57.59	16.52	0.499	0.434	0.246	0.491	57.52
WAN + Bridge (LoRA)	1,633.4	87.49	21.68	0.738	0.163	0.092	0.512	59.45	21.00	0.744	0.204	0.133	0.473	54.52
WAN (Ours)	1,548.0	1.74	23.30	0.783	0.148	0.105	0.560	64.26	24.94	0.788	0.191	0.140	0.541	60.12

compares the performance and trainable parameters. Despite training $50\times$ (WAN) and $180\times$ (FLUX) more parameters and having direct capacity to modify the backbone, these variants underperform our conditioning prompts on nearly all metrics. This highlights the effectiveness of our elicited restoration behavior through conditioning, which also preserves the original pre-trained backbone priors.

4.4 Ablations

We now ablate key components of our framework.

Bridge resolves trajectory mismatch. We compare the three training state constructions for conditioning prompt learning: (i) Naive prompt tuning (Eq. 1), (ii) DDBM-style bridge (Eq. 4), and (iii) monotone noisy-degraded $\rightarrow$ clean bridge (EBR-style; Eq. 7). Table 3 and Fig. 6 show that performance increases progressively from (i) to (iii). Thus, the monotonic bridge works best.

Image-specific start state. At inference, we obtain a starting noise specific to the degraded input via partial inversion, and use it to construct the noisy degraded start state. This significantly improves the performance of our approach across all metrics. Quantitative results averaged over ID test sets is given in Table 5.

Table 5: Effect of our partial inversion strategy (sampling steps $N = 6$). Red indicates best.

Backbone	Start state	P↑	S↑	L↓	D↓	C↑	M↑
FLUX	Random noise	23.58	0.753	0.187	0.127	0.550	63.62
	Inversion (ours)	25.39	0.843	0.118	0.090	0.571	64.91
WAN	Random noise	21.86	0.721	0.172	0.116	0.551	64.13
	Inversion (ours)	23.30	0.783	0.148	0.105	0.560	64.26

Prompt composition approach. For handling mixed degradations, we use the prompt composition approach proposed in Sec. 3.3 which enables balanced contributions for each prompt. Table 6 shows this approach yields superior results on the TOLED dataset containing blur and low-light.

Table 6: Ablation of our proposed velocity composition for TOLED (low-light + blur). Red: best.

Composition	P↑	S↑	L↓	D↓	C↑	M↑
Mean of velocities	17.85	0.720	0.218	0.174	0.499	58.15
Ours (Eq. 10)	19.70	0.754	0.199	0.161	0.527	62.31

5 Conclusion

In this work, we showed that pre-trained text-to-image and text-to-video diffusion models (PTDMs) intrinsically possess restoration capabilities that can be elicited through their native conditioning pathway. We found that natural-language prompting or input token-space prompt optimization are ineffective for eliciting this behaviour while directly optimizing a conditioning prompt at the text-encoder output space unlocks it. We further identified a train-test trajectory mismatch that makes naive conditioning prompt learning unstable, and addressed it by training within a diffusion bridge formulation that aligns intermediate states across training and inference. Building on these insights, we instantiated our framework on the WAN text-to-video model and FLUX text-to-image model, demonstrating that frozen PTDMs guided only by our learned conditioning prompts can achieve strong and competitive restoration performance to state-of-the-art approaches.

AI USE STATEMENT

We have not used generative AI tools for design or providing feedback on research methodology or experiments, implementing methods and interpreting results. Generating synthetic data sets, developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, assisting with translation, cleaning and reformatting datasets, supporting qualitative and thematic data analysis are not applicable to this work. Additionally, we used generative AI tools for creating or modifying figures or images and editing a research paper to improve readability. We have reviewed all AI-assisted work. For example, we proof-read the paper and verified AI modified figures do not have factual errors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Sec. 3 and Appendix D describe the conditioning-prompt architecture in detail. All hyper-parameters, T_0, the number of sampling and inversion steps, and optimizer settings are in Sec. 4.1 and Appendix C, and dataset details are in Appendix G. We will release code upon publication.

REFERENCES

Chris H. Bahnsen and Thomas B. Moeslund. Aau rainsnow traffic surveillance dataset, 2018. URL https://www.kaggle.com/dsv/105294.

M Brack, F Friedrich, K Kornmeier, L Tsaban, P Schramowski, K Kersting, and A Ledits+ Passos. Limitless image editing using text-to-image models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8861–8870.

Jianrui Cai, Shuhang Gu, and Lei Zhang. Learning a deep single image contrast enhancer from multi-exposure images. *IEEE Transactions on Image Processing*, 27(4):2049–2062, 2018.

Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 22560–22570, 2023.

Haoyu Chen, Jingjing Ren, Jinjin Gu, Hongtao Wu, Xuequan Lu, Haoming Cai, and Lei Zhu. Snow removal in video: A new dataset and a novel method. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13165–13176. IEEE, 2023.

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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6286–6295, 2018.

Zihan Cheng, Liangtai Zhou, Dian Chen, Ni Tang, Xiaotong Luo, and Yanyun Qu. Unildiff: Unlocking the power of diffusion priors for all-in-one image restoration. *arXiv preprint arXiv:2507.23685*, 2025.

Hamadi Chihaoui and Paolo Favaro. Invert2restore: Zero-shot degradation-blind image restoration. *arXiv preprint arXiv:2503.21486*, 2025.

Ying Chu, Guoxing Luo, and Fan Chen. A real haze video database for haze level evaluation. In *2021 13th International Conference on Quality of Multimedia Experience (QoMEX)*, pp. 69–72, 2021. doi: 10.1109/QoMEX51781.2021.9465461.

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, 2025.

Senyou Deng, Wenqi Ren, Yanyang Yan, Tao Wang, Fenglong Song, and Xiaochun Cao. Multi-scale separable network for ultra-high-definition video deblurring. In *the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14030–14039, 2021.

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.

Ben Fei, Zhaoyang Lyu, Liang Pan, Junzhe Zhang, Weidong Yang, Tianyue Luo, Bo Zhang, and Bo Dai. Generative diffusion prior for unified image restoration and enhancement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9935–9946, 2023.

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022.

Qiyuan Guan, Qianfeng Yang, Xiang Chen, Tianyu Song, Guiyue Jin, and Jiyu Jin. Weatherbench: A real-world benchmark dataset for all-in-one adverse weather image restoration. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 12607–12613, 2025.

Yu Guo, Yuan Gao, Yuxu Lu, Huilin Zhu, Ryan Wen Liu, and Shengfeng He. Onerestore: A universal restoration framework for composite degradation. In *European conference on computer vision*, pp. 255–272. Springer, 2024.

Yun Guo, Xueyao Xiao, Yi Chang, Shumin Deng, and Luxin Yan. From sky to the ground: A large-scale benchmark and simple baseline towards real rain removal. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12097–12107, October 2023.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

Jinhui HOU, Zhiyu Zhu, and Junhui Hou. Energy-oriented diffusion bridge for image restoration with foundational diffusion models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=B9JHSksyox.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pp. 2790–2799. PMLR, 2019.

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.

JiaKui Hu, Lujia Jin, Zhengjian Yao, and Yanye Lu. Universal image restoration pre-training via degradation classification. *arXiv preprint arXiv:2501.15510*, 2025.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European conference on computer vision*, pp. 709–727. Springer, 2022.

Yitong Jiang, Zhaoyang Zhang, Tianfan Xue, and Jinwei Gu. Autodir: Automatic all-in-one image restoration with latent diffusion. *arXiv preprint arXiv:2310.10123*, 2023.

Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Pnp inversion: Boosting diffusion-based editing with 3 lines of code. In *The Twelfth International Conference on Learning Representations*, 2023.

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.

Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 3045–3059, 2021.

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. doi: 10.1109/TIP.2018.2867951.

Hao Li, Xiang Chen, Jiangxin Dong, Jinhui Tang, and Jinshan Pan. Foundir: Unleashing million-scale training data to advance foundation models for image restoration. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 12626–12636, 2025.

Ruoteng Li, Robby T. Tan, and Loong-Fah Cheong. All in one bad weather removal using architectural search. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3172–3182, 2020. doi: 10.1109/CVPR42600.2020.00324.

Shuyi Liang, Yixin Yang, Hanyue Lou, Yuning Cui, and Boxin Shi. SLAIR: Structured latent flow matching for all-in-one image restoration. In *European Conference on Computer Vision*. Springer, 2026.

Ruirui Lin, Guoxi Huang, Joanne Lin, Qi Sun, Alexandra Malyugina, David R Bull, and Nantheera Anantrasirichai. Bvi-rlv: A fully registered dataset for low-light video enhancement. *arXiv preprint arXiv:2407.03535*, 2024a.

Weifeng Lin, Xinyu Wei, Renrui Zhang, Le Zhuo, Shitian Zhao, Siyuan Huang, Huan Teng, Junlin Xie, Yu Qiao, Peng Gao, et al. Pixwizard: Versatile image-to-image visual assistant with open-language instructions. *arXiv preprint arXiv:2409.15278*, 2024b.

Tie Liu, Mai Xu, and Zulin Wang. Removing rain in videos: a large-scale database and a two-stream convlstm approach. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 664–669. IEEE, 2019.

Yuhao Liu, Zhanghan Ke, Fang Liu, Nanxuan Zhao, and Rynson WH Lau. Diff-plugin: Revitalizing details for diffusion-based low-level tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4197–4208, 2024.

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. doi: 10.1109/TIP.2018.2806202.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Debabrata Mandal, Soumitri Chattopadhyay, Guansen Tong, and Praneeth Chakravarthula. Unicorn: Latent diffusion-based unified controllable image restoration network across multiple degradations. *arXiv preprint arXiv:2503.15868*, 2025.

Ségolene Martin, Anne Gagneux, Paul Hagemann, and Gabriele Steidl. Pnp-flow: Plug-and-play image restoration with flow matching. *arXiv preprint arXiv:2410.02423*, 2024.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6038–6047, 2023.

Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3883–3891, 2017.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

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, 2024.

Sudarshan Rajagopalan and Vishal M Patel. Awracle: All-weather image restoration using visual in-context learning. *arXiv preprint arXiv:2409.00263*, 2024.

Sudarshan Rajagopalan, Kartik Narayan, and Vishal M Patel. Restorevar: Visual autoregressive generation for all-in-one image restoration. *arXiv preprint arXiv:2505.18047*, 2025.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

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

Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1921–1930, 2023.

J. Jose Valanarasu, R. Yasarla, and V. M. Patel. Transweather: Transformer-based restoration of images degraded by adverse weather conditions. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2343–2353, 2022.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Hanting Wang, Tao Jin, Wang Lin, Shulei Wang, Hai Huang, Shengpeng Ji, and Zhou Zhao. Irbridge: Solving image restoration bridge with pre-trained generative diffusion models. *arXiv preprint arXiv:2505.24406*, 2025.

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.

Ruixing Wang, Xiaogang Xu, Chi-Wing Fu, Jiangbo Lu, Bei Yu, and Jiaya Jia. Seeing dynamic scene in the dark: A high-quality video dataset with mechatronic alignment. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9700–9709, 2021.

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

Haoning Wu, Erli Zhang, Liang Liao, Chaofeng Chen, Jingwen Hou, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. Exploring video quality assessment on user generated contents from aesthetic and technical perspectives. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 20144–20154, 2023.

Jie Xiao, Ruili Feng, Han Zhang, Zhiheng Liu, Zhantao Yang, Yurui Zhu, Xueyang Fu, Kai Zhu, Yu Liu, and Zheng-Jun Zha. Dreamclean: Restoring clean image using deep diffusion prior. In *The Twelfth International Conference on Learning Representations*, 2024.

Yijun Yang, Angelica I Aviles-Rivero, Huazhu Fu, Ye Liu, Weiming Wang, and Lei Zhu. Video adverse-weather-component suppression network via weather messenger and adversarial backpropagation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13200–13210, 2023.

Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *CVPR*, 2021.

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.

Xinyi Zhang, Hang Dong, Jinshan Pan, Chao Zhu, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Fei Wang. Learning to restore hazy video: A new real-world dataset and a new method. In *CVPR*, pp. 9239–9248, 2021.

Yanfu Zhang, Li Ding, and Gaurav Sharma. Hazerd: an outdoor scene dataset and benchmark for single image dehazing. In *2017 IEEE international conference on image processing (ICIP)*, pp. 3205–3209. IEEE, 2017.

Yuhong Zhang, Hengsheng Zhang, Xinning Chai, Zhengxue Cheng, Rong Xie, Li Song, and Wenjun Zhang. Diff-restorer: Unleashing visual prompts for diffusion-based universal image restoration. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

Haiyu Zhao, Lei Tian, Xinyan Xiao, Peng Hu, Yuanbiao Gou, and Xi Peng. Avernet: All-in-one video restoration for time-varying unknown degradations. *Advances in Neural Information Processing Systems*, 37:127296–127316, 2024.

Zhihang Zhong, Ye Gao, Yinqiang Zheng, Bo Zheng, and Imari Sato. Real-world video deblurring: A benchmark dataset and an efficient recurrent neural network. *International Journal of Computer Vision*, 131(1):284–301, 2023.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International journal of computer vision*, 130(9):2337–2348, 2022a.

Linqi Zhou, Aaron Lou, Samar Khanna, and Stefano Ermon. Denoising diffusion bridge models. *arXiv preprint arXiv:2309.16948*, 2023.

Shangchen Zhou, Chongyi Li, and Chen Change Loy. Lednet: Joint low-light enhancement and deblurring in the dark. In *European conference on computer vision*, pp. 573–589. Springer, 2022b.

Yuqian Zhou, David Ren, Neil Emerton, Sehoon Lim, and Timothy Large. Image restoration for under-display camera. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pp. 9179–9188, 2021.

A Appendix Overview

In this supplementary, we first provide discussions regarding differences with soft prompt-tuning and parameter-efficient fine-tuning (PEFT) followed by an evaluation of existing high-level editing approaches for restoration tasks and a discussion of the limitations of our work. We then provide the full procedure and ablations for obtaining the optimal hyper-parameters described in Sec. 4.1. Subsequently, we provide additional architectural details of our conditioning prompt-learning framework. We then present an ablation on our residual prompt-learning approach. Next, we provide computational complexity comparisons with state-of-the-art approaches. We then describe the evaluation datasets in detail. Finally, we provide detailed quantitative results on in-distribution (ID) datasets and additional qualitative results on ID, out-of-distribution (OOD), mixed, and unseen degradations. For video AiOR, we also provide side-by-side video clips to better assess temporal behavior.

To summarize, the supplementary discusses the following:

1. Discussions and Limitations (Sec. B)
2. Implementation Details and Hyperparameter Selection (Sec. C)
3. Additional Architectural Details (Sec. D)
4. Residual Prompt Ablation (Sec. E)
5. Computational Complexity Comparisons (Sec. F)
6. Evaluation Dataset Details (Sec. G)
7. Additional Quantitative Results (Sec. H)
8. Additional Qualitative Results (Sec. I)

B Discussion and Limitations

B.1 Differences with soft prompt-tuning and PEFT

Our approach is related to soft prompt-tuning (Lester et al., 2021; Zhou et al., 2022a; Jia et al., 2022; Gal et al., 2022) in that both optimize a small set of learned embeddings while keeping the pre-trained backbone frozen. However, the role of prompt optimization in our setting is different. Rather than treating prompts primarily as a parameter-efficient mechanism for adapting a model to a downstream task, we use them to investigate whether *restoration-relevant behavior* is already present in a pre-trained text-to-image or text-to-video diffusion model (PTDM), and how such behavior can be elicited through its native conditioning pathway. This distinction is reflected in our experimental findings. Optimizing natural-language prompts or embeddings in the input-token space, corresponding to the setting in Eq. 2, does not effectively recover restoration behavior (Fig. 2). In contrast, optimizing directly in the conditioning space of the PTDM succeeds at the same parameter budget. This suggests that the effectiveness of our method is not explained solely by the generic adaptability of a frozen backbone through learned embeddings and rather shows that where and how the learned instruction is introduced into the model plays a critical role. We further show that directly applying the learned conditioning prompt is insufficient unless the training and inference trajectories are aligned, motivating the solving of *trajectory mismatch*.

Our approach is also conceptually distinct from parameter-efficient fine-tuning (PEFT) methods such as adapters and LoRA (Hu et al., 2021; Houlsby et al., 2019). PEFT aims to adapt a pre-trained model to a downstream task by introducing a small number of trainable parameters within or alongside the backbone. In this setting, the learned parameters are allowed to modify the model's backbone so that it acquires the desired task behavior. In contrast, our objective is not to alter the backbone to make it a restoration model, but to determine whether restoration behavior can be accessed from the unchanged PTDM by learning an instruction expressed through its existing conditioning mechanism. Thus, all backbone parameters remain fixed, and the learned prompt acts only as a conditioning signal that guides the native denoising process. The distinction is therefore of *adaptation versus elicitation*, where PEFT modifies the model to perform the task, while our approach investigates how to invoke the desired behavior from the frozen PTDM.

Thus, prompt learning in our framework serves as a mechanism for probing and eliciting restoration capabilities from the frozen PTDM. The paired restoration data is used to learn an effective conditioning instruction, while the backbone itself remains unchanged. Combined with our analysis of conditioning-space optimization and trajectory alignment, this allows us to investigate both *whether* restoration behavior is present in the PTDM and *how* it can be reliably elicited.

B.2 Editing approaches for restoration

We further evaluate whether existing editing strategies can directly leverage a frozen PTDM for restoration. Specifically, we consider Prompt-to-Prompt (Hertz et al., 2022) with Null-text inversion (Mokady et al., 2023) (P2P+NTI), which first inverts the degraded image and then applies language-guided editing, and SDEdit (Meng et al., 2021), which partially noises the input before denoising it under a restoration instruction such as "remove haze." As shown in Fig. 7, neither approach effectively performs low-level restoration. P2P+NTI substantially alters image details due to imperfect inversion and semantic editing, while SDEdit preserves the input structure but largely retains the degradation after denoising.



Figure 7: Editing-based approaches such as SDEdit (Meng et al., 2021), and Prompt-to-Prompt (Hertz et al., 2022) with Null-text inversion (Mokady et al., 2023) (P2P+NTI) work well for high-level editing but perform poorly for restoration.

These results highlight that the ability of PTDMs to follow natural-language instructions for high-level semantic editing does not directly translate to restoration. In contrast, our approach learns the restoration instruction directly in the PTDM's conditioning space and couples it with a denoising trajectory designed to steer noisy degraded states toward clean outputs, enabling restoration while keeping the generative backbone frozen.

B.3 LIMITATIONS

While our results show that prompts learned in an appropriate conditioning space on a diffusion bridge can elicit strong restoration performance from frozen PTDMs, several directions remain for future work. Similar to instruction-guided AiOR methods (e.g., InstructIR, PixWizard, DFPIR), our method requires selecting the conditioning prompt at inference, making it non-blind to degradation type. This provides controllability but limits flexibility for truly unknown or unseen degradations. POLED is one such case: as shown in Fig. 10, using low-light+blur prompts enhances the image and removes blur, but does not address the veil effect and color shift (Zhou et al., 2021), suggesting that these artifacts fall outside the scope of our learned prompts. We also note POLED is challenging for other AiOR approaches such as SLAIR and AutoDIR (top-performers from Table 1), with our approach recovering better details. A promising direction is to learn unified or compositional prompts that can infer and combine degradation cues, enabling blind restoration of unknown degradations as mixtures of learned prompt components. In addition, our approach uses a fixed start noise level T_0 for all degradations. Although this worked well in our experiments, adapting T_0 based on degradation-type or intensity could improve fidelity for mild cases, while allowing strong corrections for severe inputs. Finally, enhanced prompt injection schemes such as block-wise or time-step-conditioned prompts, may further improve restoration. Overall, these directions suggest that our approach offers a strong and flexible foundation for native conditioning pathway-based restoration using frozen PTDMs.

C IMPLEMENTATION DETAILS AND HYPERPARAMETER SELECTION

We optimize conditioning prompts using AdamW (Loshchilov & Hutter, 2017) with learning rate 5×10^{-4}, batch size 2, for 20k iterations on WAN and 40k iterations on FLUX. We now detail the procedure for selecting the starting noise level $T_0 = 0.4$ used during training, the number of sampling steps $N = 6$ and number of inversion steps $K = 6$ for WAN and $K = 10$ for FLUX. To select these parameters, we first define a score which yields a balanced contribution from all perception and distortion metrics used for evaluation. We now show the procedure to calculate the score using T_0 selection as an example.

C.1 COMPUTING SCORE FOR T_0 SELECTION

We aggregate multiple distortion and perceptual metrics into a single scalar score with equal weighting. For candidate T_0 values, we evaluate performance using PSNR, SSIM, LPIPS, DISTS, CLIP-IQA, and MUSIQ on in-domain datasets. Since these metrics have different units and directions (higher is better for PSNR/SSIM/CLIP-IQA/MUSIQ; lower is better for LPIPS/DISTS), we first apply min-max normalization across the set of tested T_0 values independently for each metric. Concretely, for a higher-is-better metric m with score s_m we use

$$\tilde{s}_m(T_0) = \frac{s_m(T_0) - \min_{T_0'} s_m(T_0')}{\max_{T_0'} s_m(T_0') - \min_{T_0'} s_m(T_0')},$$

and for a lower-is-better metric we invert the normalization,

$$\tilde{s}_m(T_0) = \frac{\max_{T_0'} s_m(T_0') - s_m(T_0)}{\max_{T_0'} s_m(T_0') - \min_{T_0'} s_m(T_0')}.$$

The final selection score is the unweighted mean over metrics,

$$S(T_0) = \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} \tilde{s}_m(T_0),$$

where $\mathcal{M}$ denotes the metric set. We choose the T_0 that maximizes $S(T_0)$. We set the maximum noising level T_0 used in training (Eq. 7) based on degradation severity. Milder degradations require lower noise to restore, whereas severe degradations benefit from a larger T_0 as the PTDM has sufficient flexibility to correct the artifact. We tuned T_0 from 0.1 to 0.6 on the most severe degradation in our training set, i.e. low-light. As shown in Fig. 8 (a), $T_0 = 0.4$ yields the best score for both WAN and FLUX. For robustness, we use this T_0 value for all degradations, ensuring adequate headroom for challenging cases.

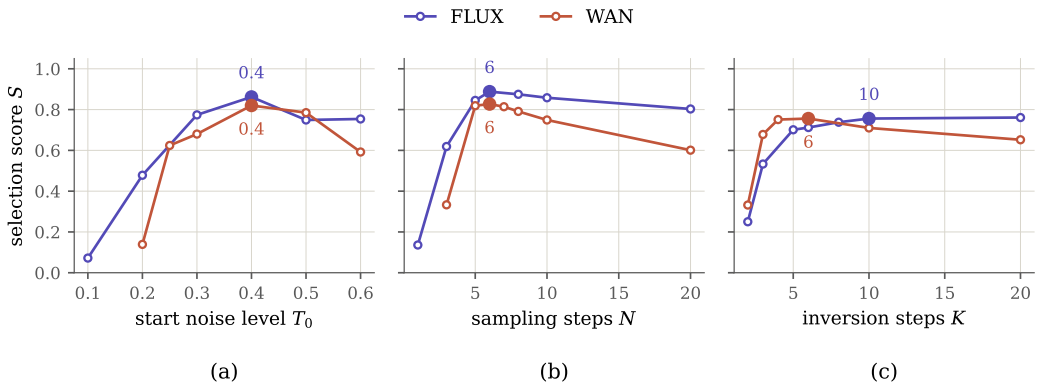


Figure 8: Selection score S for (a) the start noise level T_0 (Sec. C.1), (b) the number of sampling steps N (Sec. C.2), and (c) the number of inversion steps K (Sec. C.2). Markings indicate the selected value corresponding to the highest score for each sweep: $T_0 = 0.4$ and $N = 6$ for both backbones, $K = 10$ for FLUX and $K = 6$ for WAN. The score was calculated over in-distribution data.

C.2 Sampling step and inversion step selection

Using the above score formulation, we calculate the scores while sweeping sampling steps from 2 to 20 (no partial inversion used). From Fig. 8 (b), we can observe that $N = 6$ yields the best scores for both FLUX and WAN models. Subsequently, we fix $N = 6$ and vary the inversion steps, K, from 2 to 20. Fig. 8 (c) shows that $K = 6$ yields the best results for WAN while $K = 10$ and $K = 20$ yield near identical results for the FLUX model. Since the gain is minimal for double the neural function evaluations (NFEs), we select $K = 10$ for the FLUX model.

D Additional Architectural Details

As mentioned in Sec. 4.1 of the main paper, we found that a residual prompt learning strategy for the *conditioning prompt* is more effective and parameter-efficient than directly optimizing the conditioning prompt in text-encoder output space (see Sec. E). Note that our primary findings that learning conditioning prompts in the text-encoder output space unlocks restoration behaviour still holds. The residual prompt-learning strategy is only an improvement over this finding (see Sec. E). Instead of replacing the conditioning features used by attention, we learn the prompt tokens and inject them as a residual perturbation on top of the conditioning features obtained from the null-text embedding. Furthermore, a learnable scalar gate initialized at zero is used to gate the contribution of the prompt. This design preserves the pretrained model behavior at initialization and lets the network gradually learn the prompt embeddings. Let $e_{\text{null}} \in \mathbb{R}^{L \times D_{\text{a}}}$ denote the conditioning hidden states (or features) provided to the cross-attention or FLUX attention (Labs, 2024) layer, where L is the conditioning sequence length and D_{a} is the hidden dimension of the diffusion transformer.

We now learn the conditioning prompt $p_{\text{a}} \in \mathbb{R}^{L \times D_{\text{a}}}$, together with a scalar gate α^l, where l denotes the lth transformer block. Note that the prompt p_{a} is still *shared across all blocks*. α^l is initialized to zero, so the model matches the frozen pretrained backbone at the start of training. Depending on the architecture, the learned prompts are injected as a residual into the conditioning states before attention, or through an additional attention contribution derived from the learned prompt conditioning. In all cases, the backbone attention weights remain frozen; only the prompt tokens and the scalar gates are optimized, with the prompt tokens shared across all blocks.

D.1 WAN: residual prompting in cross-attention

For WAN (Wan et al., 2025), prompt injection is performed at the cross-attention layer (see Fig. 9(a)). Let $\text{Attn}(h, e_{\text{null}})$ denote the cross-attention output for image hidden states h^l and null-text conditioning states e_{null}, where l is the lth transformer block. We first compute the cross-attention

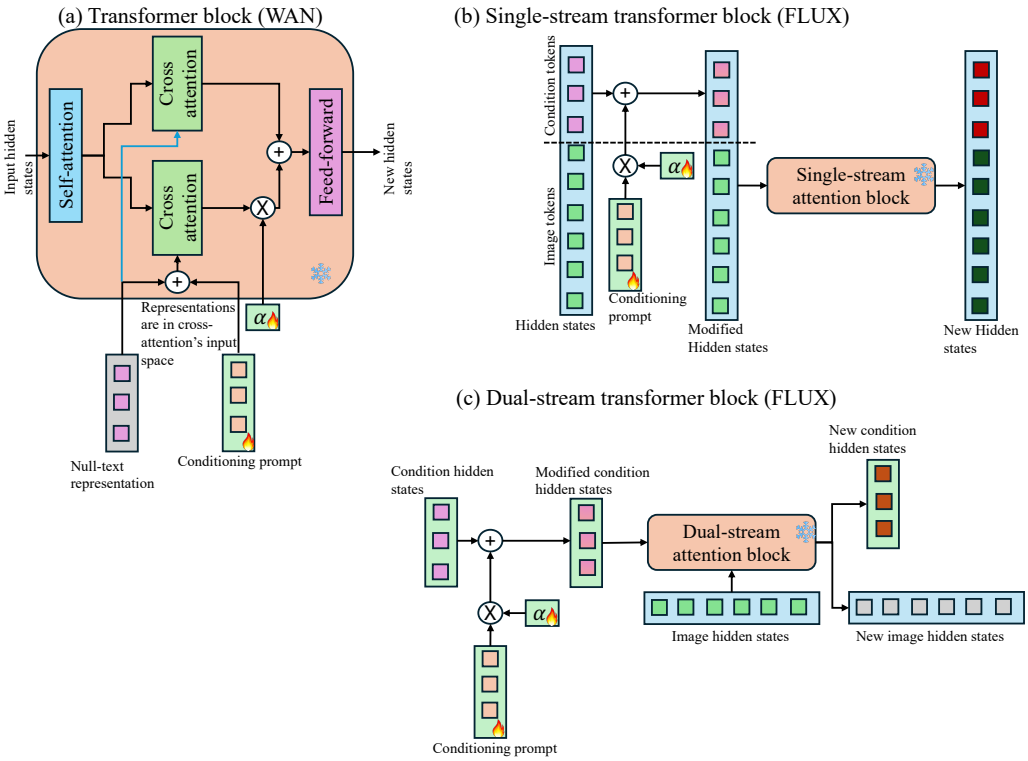


Figure 9: Illustration of our residual prompt fusion approach in the WAN and FLUX transformer architectures.

output

$$o^l_{\text{null}} = \text{Attn}^l(h^l, e_{\text{null}}).$$

We then form a prompt-conditioned version of the attention input. In our residual formulation, the learned prompt is added to the original conditioning states:

$$e_{\text{prompt}} = e_{\text{null}} + p_{\text{a}}.$$

Applying the same frozen cross-attention module to this perturbed conditioning yields

$$o^l_{\text{prompt}} = \text{Attn}^l(h^l, e_{\text{prompt}}).$$

The final output is obtained by gating o_{prompt} and adding it as a residual to o^l_{null}:

$$o^l = o^l_{\text{null}} + \alpha^l o^l_{\text{prompt}}.$$

When $\alpha^l = 0$, the block reduces exactly to the pretrained model with null-text conditioning. As training progresses, the learned gate controls how strongly the prompt modifies the attention output.

D.2 FLUX: RESIDUAL PROMPTING FOR SINGLE-STREAM AND DUAL-STREAM BLOCKS

FLUX (Labs, 2024) contains both single-stream and dual-stream transformer blocks, and the prompt injection is adapted accordingly.

Dual-stream block. In the dual-stream block, image hidden states and condition hidden states are processed as separate streams. In FLUX, the condition hidden states continuously evolve through the blocks. Let e^l_{null} denote the condition hidden states originating from the null-text prompt, entering the block l. We inject the learned prompt as a gated residual added directly to the condition states:

$$\tilde{e}^l_{\text{null}} = e^l_{\text{null}} + \alpha^l p_{\text{a}}.$$

The modified condition states $\tilde{e}^l_{\text{null}}$ are then passed into the original frozen dual-stream attention block together with the image hidden states. Thus, for FLUX, only the input conditioning stream is perturbed, while the pretrained block structure is unchanged. The above process is illustrated in Fig. 9(c).

Single-stream block. In the single-stream block, image and condition tokens are concatenated into a single sequence. Let the first L tokens correspond to the conditioning portion of the hidden states:

$$h^l = [h^l_{\text{null}}; h^l_{\text{img}}],$$

where l denotes the lth transformer block. We inject the learned prompt only into the conditioning segment:

$$\tilde{h}^l_{\text{null}} = h^l_{\text{null}} + \alpha^l p_{\text{a}},$$

and construct the modified hidden state sequence as

$$\tilde{h}^l = [\tilde{h}^l_{\text{null}}; h^l_{\text{img}}].$$

The resulting sequence is then processed by the original frozen single-stream attention block. Hence, even in the single-stream case, the prompt acts only on the condition tokens and does not directly alter the image-token portion. The above process is illustrated in Fig. 9(b).

This residual design improves over directly learning the conditioning prompt at the text-encoder output space while using fewer parameters (Sec. E).

E Residual Prompt Ablation

The residual conditioning prompt-learning approach explained in Sec. D, improves over directly learning the conditioning prompt at the text-encoder output space. We show this for the WAN model in Table 7. It can be observed that the residual approach achieves better performance than learning the prompt directly at the output space of the text-encoder (Table 7), while using fewer parameters ($\sim$340K (residual) vs. $\sim$900K (TE-output space)).

Table 7: Ablation on learning the conditioning prompt in the text-encoder (TE) output space vs. a residual prompt injected in the attention context space. Results are on the REVIDE (Zhang et al., 2021) video dehazing dataset.

Conditioning	P↑	S↑	L↓	D↓	C↑	M↑
TE output-space	20.41	0.758	0.207	0.143	0.545	64.96
Residual prompt	22.20	0.774	0.161	0.122	0.624	72.76

F Computation Complexity Comparisons

Table 8 gives sampling steps, inference times, and efficiency comparisons for our WAN and FLUX instantiations against other state-of-the-art approaches on a single A6000 GPU at resolution 512×512 with the identified optimal sampling-time parameters (Sec. C). For our approach, the entirety of the inference cost is that of the PTDM, which is higher than that of other AiOR approaches for FLUX. Nevertheless, the core contribution of our work is to provide new insights into the restoration capabilities of frozen PTDMs for which we used the FLUX backbone due to its strong priors. We would also like to mention that our approach has the fewest trainable parameters since we only need to learn the prompt. Furthermore, the backbone matched comparisons in Table 4 of the main paper show that our approach achieves the best performance compared to low-rank adaptation-based restoration variants of the FLUX and WAN backbones, despite using signficantly fewer trainable parameters.

G Evaluation Dataset Details

In this section, we provide additional details about the test datasets used.

G.1 Image Restoration

For the all-in-one image restoration task, we evaluate: DCPT (Hu et al., 2025), DFPIR (Tian et al., 2025), AutoDIR (Jiang et al., 2023), PixWizard (Lin et al., 2024b), RestoreVAR (Rajagopalan et al.,

Table 8: Computational cost comparison with state-of-the-art approaches at 512×512 on a single RTX A6000 for single degradation removal. Trainable parameters for our approach are the total across the five degradation-specific conditioning prompts. Our inference cost is mostly that of the pre-trained backbone and is higher than that of other AiOR approaches. For backbone-matched comparisons with LoRA-based restoration variants of FLUX and WAN, see Table 4.

Method	Params (M)	Trainable (M)	TFLOPs/img	s/img
DFPIR (Tian et al., 2025)	31.1	31.10	1.32	0.14
DCPT (Hu et al., 2025)	35.4	35.40	1.38	0.14
FoundIR (Li et al., 2025)	36.3	36.30	3.84	0.32
RestoreVAR (Rajagopalan et al., 2025)	405.9	405.90	5.55	0.44
SLAIR (Liang et al., 2026)	780.1	780.10	10.40	0.26
AutoDIR (Jiang et al., 2023)	943.2	943.20	244.64	4.82
PixWizard (Lin et al., 2024b)	2,647.4	2,647.40	683.09	14.61
WAN (Ours)	1,548.0	1.74	43.75	1.11
FLUX (Ours)	11,994.7	7.87	347.67	5.51

2025), SLAIR (Liang et al., 2026), and our approach instantiated on FLUX (Labs, 2024) on the following datasets for in-distribution (ID) and out-of-distribution (OOD):

1. RESIDE (Li et al., 2019): 500 synthetic test images for dehazing (ID)
2. Snow100k-L (Liu et al., 2018): 16801 synthetic test images for desnowing (ID)
3. Rain13K (Zamir et al., 2021): We use the Rain100L and Rain100H synthetic testsets for deraining (ID) with 100 images each.
4. GoPro (Nah et al., 2017): 1111 synthetic test images for deblurring (ID)
5. LOLv1 (Wei et al., 2018): 15 real test images for low-light enhancement (ID)
6. HazeRD (Zhang et al., 2017): 75 synthetic test images for dehazing (OOD)
7. WeatherBenchSnow (Guan et al., 2025): Real snow subset of the WeatherBench dataset with 200 test images (OOD)
8. LHP (Guo et al., 2023): 1000 real rainy images (OOD)
9. 4KRD (Deng et al., 2021): 300 synthetic images for deblurring (OOD)
10. SICE (Cai et al., 2018): 565 images for enhancement task (OOD) of which we use real low-light images (images with index less than 4)
11. LOLBlur (Zhou et al., 2022b): 482 test images containing a mix of real low-light and blur degradations
12. CDD (Guo et al., 2024): 200 images from the haze+snow split of CDD, containing a synthetic mix of haze and snow degradations
13. TOLED and POLED (Zhou et al., 2021): 30 real images each, containing unseen task of under-display camera restoration

G.2 Video Restoration

For the all-in-one video restoration task, we evaluate methods: DCPT (Hu et al., 2025), DFPIR (Tian et al., 2025), AutoDIR (Jiang et al., 2023), PixWizard (Lin et al., 2024b), RestoreVAR (Rajagopalan et al., 2025), SLAIR (Liang et al., 2026), ViWS-Net (Yang et al., 2023), AverNet (Zhao et al., 2024) and our approach instantiated on WAN (Wan et al., 2025) on the following datasets for in-distribution (ID) and out-of-distribution (OOD). We evaluate on the first 33 frames of each video sequence due to compute constraints:

1. REVIDE (Zhang et al., 2021): 6 real sequences for dehazing (ID)
2. RVSD (Chen et al., 2023): 30 synthetic sequences for desnowing (ID)
3. SDSD (Wang et al., 2021): 12 real sequences for low-light enhancement (ID)
4. GoPro (Nah et al., 2017): 11 synthetic sequences for deblurring (ID)
5. NTURain-Synthetic (Chen et al., 2018): 11 synthetic sequences for deraining (ID)

Table 9: Quantitative results of our prompt-instantiated FLUX model along with state-of-the-art image AiOR approaches over the in-distribution image restoration datasets RESIDE (Li et al., 2019) (haze), Rain13K (Zamir et al., 2021) (rain), Snow100k (Liu et al., 2018) (snow), GoPro (Nah et al., 2017) (blur), and LOLv1 (Wei et al., 2018) (low-light). H-Haze, R-Rain, S-Snow, L-Low-light, and B-Blur. Red is best while blue is second best, and they are computed between our approach and state-of-the-art approaches.

Methods	RESIDE (H)						Rain13K (R)						Snow100K (S)					
	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑
DCPT	28.58	0.970	0.016	0.020	0.677	67.82	24.21	0.766	0.285	0.182	0.534	63.56	28.51	0.875	0.118	0.096	0.564	63.88
DFPIR	27.91	0.962	0.026	0.032	0.665	67.36	27.25	0.834	0.217	0.148	0.516	63.34	19.31	0.672	0.320	0.196	0.547	57.06
AutoDIR	22.89	0.914	0.037	0.047	0.665	67.57	28.84	0.856	0.123	0.092	0.589	66.02	27.80	0.851	0.111	0.093	0.577	62.91
PixWizard	21.55	0.760	0.164	0.126	0.612	63.01	18.65	0.550	0.369	0.199	0.634	65.06	16.65	0.469	0.383	0.206	0.572	63.31
RestoreVAR	24.67	0.821	0.074	0.048	0.664	66.47	23.12	0.670	0.231	0.147	0.579	65.83	24.00	0.710	0.156	0.105	0.576	63.74
SLAIR	25.58	0.843	0.086	0.054	0.657	62.22	24.89	0.731	0.220	0.137	0.562	57.17	24.82	0.727	0.184	0.116	0.551	55.65
FLUX + Concat	18.80	0.486	0.406	0.219	0.452	56.37	16.77	0.308	0.530	0.289	0.484	57.41	17.03	0.354	0.468	0.228	0.468	58.27
FLUX + Bridge	25.41	0.838	0.089	0.069	0.491	67.11	24.25	0.757	0.172	0.120	0.447	65.91	24.55	0.764	0.132	0.087	0.437	65.46
FLUX (Ours)	27.32	0.945	0.037	0.042	0.673	67.16	26.09	0.838	0.173	0.125	0.590	66.55	27.20	0.844	0.108	0.087	0.574	63.25

Methods	GoPro (B)						LOLv1 (L)						Average					
	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑
DCPT	27.89	0.874	0.172	0.108	0.416	48.57	22.98	0.840	0.134	0.115	0.541	67.74	26.43	0.865	0.145	0.104	0.546	62.31
DFPIR	27.51	0.851	0.203	0.127	0.386	44.19	22.73	0.819	0.161	0.141	0.503	62.95	24.94	0.827	0.185	0.128	0.523	58.98
AutoDIR	26.31	0.809	0.209	0.139	0.386	41.25	19.20	0.770	0.161	0.137	0.588	69.27	25.00	0.840	0.128	0.101	0.561	61.40
PixWizard	14.93	0.332	0.552	0.271	0.460	51.88	20.81	0.736	0.188	0.135	0.587	67.85	18.51	0.569	0.331	0.187	0.573	62.22
RestoreVAR	23.96	0.737	0.166	0.098	0.461	55.32	21.72	0.782	0.126	0.086	0.595	71.50	23.49	0.744	0.150	0.096	0.575	64.57
SLAIR	24.84	0.757	0.188	0.109	0.456	47.44	19.74	0.786	0.177	0.132	0.606	68.83	23.97	0.768	0.171	0.109	0.566	58.26
FLUX + Concat	17.87	0.378	0.407	0.197	0.427	55.94	15.63	0.414	0.616	0.304	0.461	55.47	17.22	0.388	0.485	0.247	0.458	56.69
FLUX + Bridge	23.53	0.718	0.172	0.101	0.395	60.70	20.53	0.736	0.225	0.149	0.506	69.59	23.65	0.762	0.158	0.105	0.455	65.75
FLUX (Ours)	25.25	0.791	0.114	0.075	0.511	62.42	21.11	0.801	0.162	0.122	0.511	65.21	25.39	0.843	0.118	0.090	0.571	64.91

6. RHVD (Chu et al., 2021): 20 randomly selected real video sequences for dehazing (OOD)
7. AAURainSnow (Bahnsen & Moeslund, 2018): 22 real sequences for desnowing (OOD), also contains rain in some instances
8. LasVR (Liu et al., 2019): 28 synthetic rain sequences (OOD)
9. NTU-Rain (Chen et al., 2018): 7 real-rain sequences (OOD)
10. BVI (Lin et al., 2024a): 40 real low-light sequences (OOD)
11. BSD (Zhong et al., 2023): 20 real motion-blurred sequences (OOD)

H Additional Quantitative Results

In-distribution results. Table 9 provides the detailed quantitative results for AiOR over the image datasets discussed in Sec. 4.2. Table 10 provides the detailed quantitative results for AiOR over the video datasets discussed in Sec. 4.2. From the tables, it can be inferred that our prompt-learning approach enables the WAN and FLUX models to achieve competitive ID performance to state-of-the-art approaches.

Out-of-distribution results for backbone-matched comparisons. Tables 11 and 12 provide the detailed quantitative results for OOD, mixed and unseen degradations for backbone-matched variants introduced in Sec. 4.3. Our approach achieves the best performance in most instances.

I Additional Qualitative Results

In this section, we provide additional qualitative results to illustrate the restoration performance of our prompt-instantiated FLUX and WAN models. We additionally provide video clips in the supplementary material to view temporal dynamics.

I.1 Additional results on OOD, mixed and unseen degradations

Image restoration. Figs. 10, 11 and 12 provide qualitative results for state-of-the-art AiOR methods for image restoration and our prompt-instantiated pre-trained FLUX model on OOD, mixed and unseen degradations. In most instances, our approach achieves strong restoration performance for these challenging scenarios.

Table 10: Quantitative results of our prompt-instantiated WAN model along with state-of-the-art image and video AiOR approaches over the in-distribution video restoration datasets REVIDE (Zhang et al., 2021) (haze), GoPro (Nah et al., 2017) (blur), RVSD (Chen et al., 2023) (snow), SDSD (Wang et al., 2021) (low-light), and NTURain-Synthetic (Chen et al., 2018) (rain). H-Haze, B-Blur, S-Snow, L-Low-light, R-Rain. Red is best while blue is second best, and they are computed between our approach and state-of-the-art approaches.

Methods	REVIDE (H)							GoPro (B)						
	P↑	S↑	L↓	D↓	C↑	M↑	Do↑	P↑	S↑	L↓	D↓	C↑	M↑	Do↑
DCPT	15.21	0.765	0.274	0.195	0.539	64.56	0.464	27.57	0.869	0.178	0.113	0.417	49.05	0.405
DFPIR	14.62	0.740	0.295	0.213	0.511	63.36	0.443	26.84	0.840	0.213	0.134	0.380	43.89	0.367
AutoDIR	17.60	0.768	0.292	0.263	0.509	64.58	0.413	25.74	0.795	0.209	0.142	0.384	41.71	0.403
PixWizard	11.86	0.409	0.643	0.311	0.603	60.22	0.356	19.75	0.573	0.362	0.198	0.399	40.73	0.288
RestoreVAR	15.59	0.737	0.277	0.176	0.568	66.92	0.538	23.68	0.724	0.170	0.102	0.455	55.34	0.445
SLAIR	21.79	0.849	0.145	0.105	0.629	71.04	0.645	24.43	0.744	0.196	0.115	0.450	46.74	0.407
ViWS-Net	17.85	0.785	0.285	0.230	0.468	59.99	0.420	24.12	0.739	0.265	0.184	0.373	37.38	0.382
AverNet	19.29	0.828	0.276	0.184	0.408	48.17	0.351	26.40	0.835	0.242	0.171	0.298	28.38	0.354
WAN + Concat	16.10	0.523	0.411	0.193	0.528	67.97	0.654	15.39	0.360	0.418	0.179	0.393	47.71	0.486
WAN + Bridge	22.33	0.801	0.148	0.097	0.598	71.47	0.686	23.08	0.720	0.151	0.090	0.417	49.14	0.505
WAN (Ours)	23.24	0.833	0.134	0.107	0.642	73.49	0.662	24.13	0.751	0.128	0.090	0.493	59.08	0.528

Methods	RVSD (S)							SDSD (L)						
	P↑	S↑	L↓	D↓	C↑	M↑	Do↑	P↑	S↑	L↓	D↓	C↑	M↑	Do↑
DCPT	22.27	0.798	0.236	0.126	0.572	59.54	0.550	11.85	0.449	0.395	0.218	0.410	53.08	0.313
DFPIR	18.35	0.775	0.221	0.140	0.551	57.74	0.459	16.65	0.552	0.325	0.191	0.399	51.83	0.309
AutoDIR	19.85	0.811	0.149	0.113	0.530	60.21	0.525	15.07	0.536	0.293	0.177	0.502	63.26	0.486
PixWizard	17.18	0.537	0.317	0.165	0.561	58.87	0.428	14.62	0.538	0.438	0.261	0.454	52.42	0.327
RestoreVAR	19.10	0.694	0.203	0.121	0.553	60.02	0.554	16.37	0.568	0.284	0.158	0.461	62.72	0.450
SLAIR	22.60	0.769	0.167	0.102	0.531	54.64	0.532	19.18	0.644	0.246	0.144	0.497	54.47	0.457
ViWS-Net	22.98	0.837	0.137	0.112	0.488	57.43	0.532	14.12	0.506	0.556	0.440	0.195	28.69	0.137
AverNet	23.23	0.874	0.133	0.102	0.490	45.77	0.476	18.09	0.660	0.351	0.223	0.233	33.22	0.224
WAN + Concat	16.00	0.418	0.451	0.208	0.477	58.22	0.618	14.14	0.544	0.430	0.191	0.479	58.29	0.557
WAN + Bridge	21.80	0.742	0.153	0.091	0.529	60.28	0.685	17.15	0.642	0.269	0.118	0.523	60.57	0.612
WAN (Ours)	23.05	0.817	0.120	0.090	0.569	62.59	0.651	17.84	0.644	0.275	0.175	0.508	64.29	0.596

Methods	NTURain-Syn (R)							Average						
	P↑	S↑	L↓	D↓	C↑	M↑	Do↑	P↑	S↑	L↓	D↓	C↑	M↑	Do↑
DCPT	28.82	0.906	0.133	0.097	0.617	63.98	0.638	21.14	0.757	0.243	0.149	0.511	58.04	0.474
DFPIR	30.11	0.929	0.098	0.080	0.597	65.23	0.657	21.31	0.767	0.230	0.151	0.487	56.41	0.447
AutoDIR	31.46	0.943	0.054	0.051	0.625	63.79	0.688	21.94	0.770	0.199	0.149	0.510	58.71	0.503
PixWizard	20.05	0.663	0.244	0.131	0.637	59.96	0.589	16.69	0.544	0.400	0.213	0.530	54.44	0.397
RestoreVAR	24.79	0.752	0.164	0.105	0.588	61.78	0.644	19.90	0.695	0.219	0.132	0.525	61.35	0.526
SLAIR	26.93	0.810	0.121	0.070	0.573	54.70	0.595	22.98	0.763	0.175	0.107	0.536	56.31	0.527
ViWS-Net	31.31	0.934	0.046	0.047	0.552	62.54	0.658	22.07	0.760	0.257	0.202	0.415	49.20	0.425
AverNet	33.12	0.943	0.078	0.074	0.556	55.75	0.630	24.02	0.828	0.216	0.150	0.397	42.25	0.407
WAN + Concat	15.37	0.388	0.374	0.174	0.487	55.77	0.713	15.40	0.446	0.416	0.189	0.472	57.59	0.605
WAN + Bridge	24.07	0.787	0.094	0.064	0.496	55.80	0.710	21.68	0.738	0.163	0.092	0.512	59.45	0.639
WAN (Ours)	28.25	0.872	0.083	0.063	0.592	61.89	0.701	23.30	0.783	0.148	0.105	0.560	64.26	0.627

Video restoration. Qualitative results for state-of-the-art AiOR methods for both image and video restoration, and our prompt-instantiated pre-trained WAN model on OOD datasets are provided in Figs. 13 and 14. Our approach achieves strong restoration performance. To view temporal dynamics, we provide the video clips for the scenes in Figs. 13, 14, 17, and 18, and Fig. 5 of the main paper. To view these clips, open the "supplementary_videos" folder (obtained after unzipping the supplementary file). For each dataset, videos containing the clips for all the evaluated AiOR approaches for image and video restoration, along with our prompt instantiated pre-trained WAN model are provided. Files are named as "#sequence_name.mp4". Each video contains the results of all the methods side-by-side for a particular sequence. From the videos, it can be observed that our approach also yields strong temporal stability.

Table 11: Per-dataset results for the backbone-matched variants on the OOD, mixed and unseen image restoration datasets. Red indicates best while blue is second best.

Methods	HazeRD (H)						LHP (R)						WeatherBench (S)					
	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑
FLUX + Concat	16.06	0.434	0.509	0.280	0.512	61.15	18.19	0.417	0.510	0.282	0.482	53.83	17.36	0.453	0.463	0.269	0.413	47.82
FLUX + Bridge	16.54	0.713	0.198	0.126	0.654	73.45	25.39	0.783	0.193	0.129	0.413	60.10	22.96	0.749	0.227	0.162	0.302	47.50
FLUX (Ours)	17.02	0.813	0.195	0.143	0.684	69.46	27.82	0.877	0.136	0.096	0.544	58.27	22.23	0.762	0.265	0.182	0.386	46.47

Methods	4KRD (B)						SICE (L)						TOLED (U)					
	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑
FLUX + Concat	17.19	0.347	0.431	0.222	0.513	64.78	12.54	0.340	0.567	0.293	0.486	58.25	15.94	0.404	0.463	0.267	0.507	58.97
FLUX + Bridge	23.36	0.728	0.170	0.116	0.501	66.50	11.49	0.424	0.375	0.226	0.551	64.68	21.89	0.761	0.230	0.169	0.412	60.72
FLUX (Ours)	24.61	0.797	0.106	0.079	0.673	72.81	16.04	0.659	0.270	0.175	0.608	67.13	19.70	0.754	0.199	0.161	0.527	62.31

Methods	POLED (U)						CDD (S+H)						LOLBlur (L+B)		Average					
	P↑	S↑	L↓	D↓	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑	C↑	M↑	P↑	S↑	L↓	D↓	C↑	M↑
FLUX + Concat	12.51	0.263	0.853	0.481	0.396	45.21	14.48	0.327	0.562	0.284	0.522	63.12	0.412	47.85	15.53	0.373	0.545	0.297	0.471	55.66
FLUX + Bridge	16.35	0.462	0.616	0.401	0.416	50.15	19.13	0.623	0.187	0.112	0.631	71.39	0.361	47.36	19.64	0.655	0.274	0.180	0.471	60.20
FLUX (Ours)	19.90	0.575	0.522	0.400	0.283	45.48	17.49	0.745	0.191	0.155	0.666	67.87	0.440	51.43	20.60	0.748	0.235	0.174	0.534	60.14

Table 12: Per-dataset results for the backbone-matched variants on the OOD video restoration datasets. Red indicates best while blue is second best.

Methods	RHVD (H)			LasVR (R)							NTURain (R)			AAURainSnow (S)		
	C↑	M↑	Do↑	P↑	S↑	L↓	D↓	C↑	M↑	Do↑	C↑	M↑	Do↑	C↑	M↑	Do↑
WAN + Concat	0.554	62.25	0.620	18.55	0.505	0.370	0.212	0.514	60.18	0.659	0.497	57.85	0.642	0.515	60.24	0.671
WAN + Bridge	0.551	61.66	0.622	25.70	0.782	0.160	0.112	0.546	60.65	0.697	0.513	55.52	0.631	0.366	44.21	0.394
WAN (Ours)	0.618	64.91	0.694	30.59	0.862	0.136	0.100	0.559	62.09	0.657	0.599	63.57	0.640	0.464	49.87	0.393

Methods	BSD (B)							BVI (L)							Average						
	P↑	S↑	L↓	D↓	C↑	M↑	Do↑	P↑	S↑	L↓	D↓	C↑	M↑	Do↑	P↑	S↑	L↓	D↓	C↑	M↑	Do↑
WAN + Concat	16.68	0.489	0.469	0.259	0.338	38.54	0.424	14.35	0.505	0.464	0.269	0.534	66.42	0.696	16.52	0.499	0.434	0.246	0.491	57.52	0.618
WAN + Bridge	20.20	0.734	0.239	0.164	0.323	36.56	0.459	17.11	0.717	0.213	0.124	0.580	69.55	0.723	21.00	0.744	0.204	0.133	0.473	54.52	0.588
WAN (Ours)	24.33	0.780	0.202	0.158	0.410	47.74	0.487	19.92	0.724	0.235	0.163	0.599	72.58	0.724	24.94	0.788	0.191	0.140	0.541	60.12	0.599

I.2 Visual results on ID datasets

Image restoration. Figs. 15 and 16 provide qualitative results for state-of-the-art AiOR methods for image restoration and our prompt-instantiated pre-trained FLUX model on in-distribution test sets. We find that our method restores degradations well and is competitive to the other approaches.

Video restoration. Figs. 17 and 18 provide qualitative results for state-of-the-art AiOR methods for both image and video restoration, and our prompt-instantiated pre-trained WAN model on in-distribution test sets. We find that our method restores degradations well and is competitive to the other approaches. Similar to OOD datasets, video clips are provided under the folder "supplementary_videos".

I.3 Results for methods excluded in Figs. 4 and 5 of main paper

For completeness, we provide qualitative results for the methods that were excluded in Figs. 4 and 5 of the main paper due to space constraints.

Image Restoration. Due to space constraints, we excluded the RestoreVAR (Rajagopalan et al., 2025) and PixWizard (Lin et al., 2024b) methods from Fig. 4 of the main paper. In Fig. 19, we provide qualitative results for our prompt-instantiated FLUX model, RestoreVAR and PixWizard for the samples from Fig. 4 of the main paper. Our approach yields strong and competitive restoration performance.

Video Restoration. Due to space constraints in Fig. 5 of the main paper, we excluded the DCPT, AutoDIR, PixWizard and RestoreVAR methods. In Fig. 20, we provide qualitative results for our prompt-instantiated WAN model and these approaches for the samples from Fig. 5 of the main paper. Our approach yields strong and competitive restoration performance.

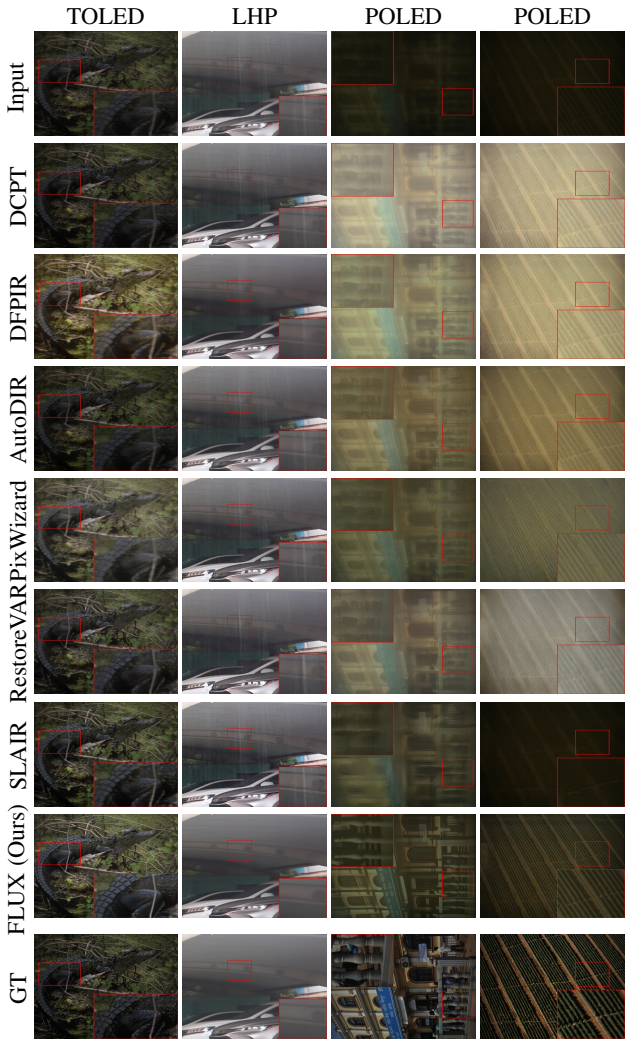


Figure 10: Additional qualitative results for state-of-the-art AiOR approaches for image restoration and the pre-trained FLUX model instantiated with our learned prompts on the TOLED (Zhou et al., 2021) (Unseen), POLED (Zhou et al., 2021) (Unseen) and LHP (Guo et al., 2023) (Rain, OOD) datasets.

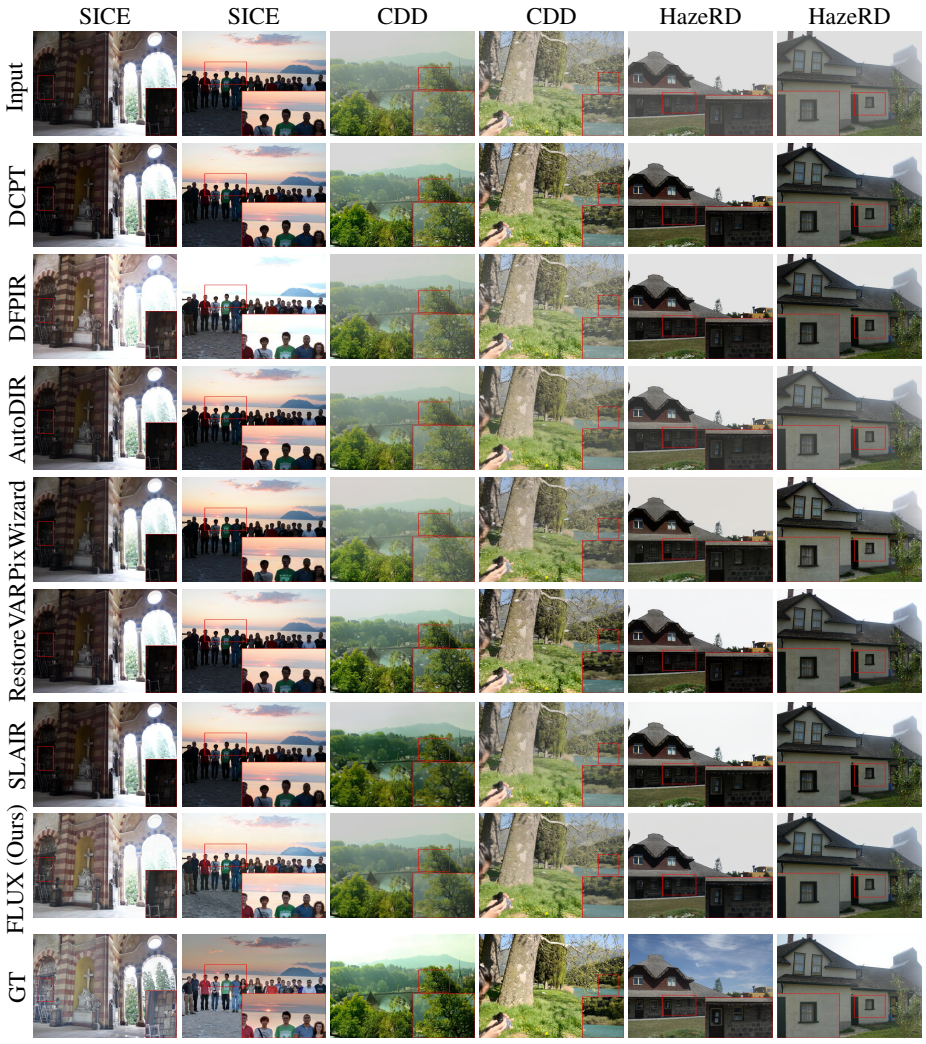


Figure 11: Additional qualitative results for state-of-the-art AiOR approaches for image restoration and the pre-trained FLUX model instantiated with our learned prompts on the SICE (Cai et al., 2018) (low-light, OOD), CDD (Guo et al., 2024) (haze+snow, mixed), and HazeRD (Haze, OOD) datasets.

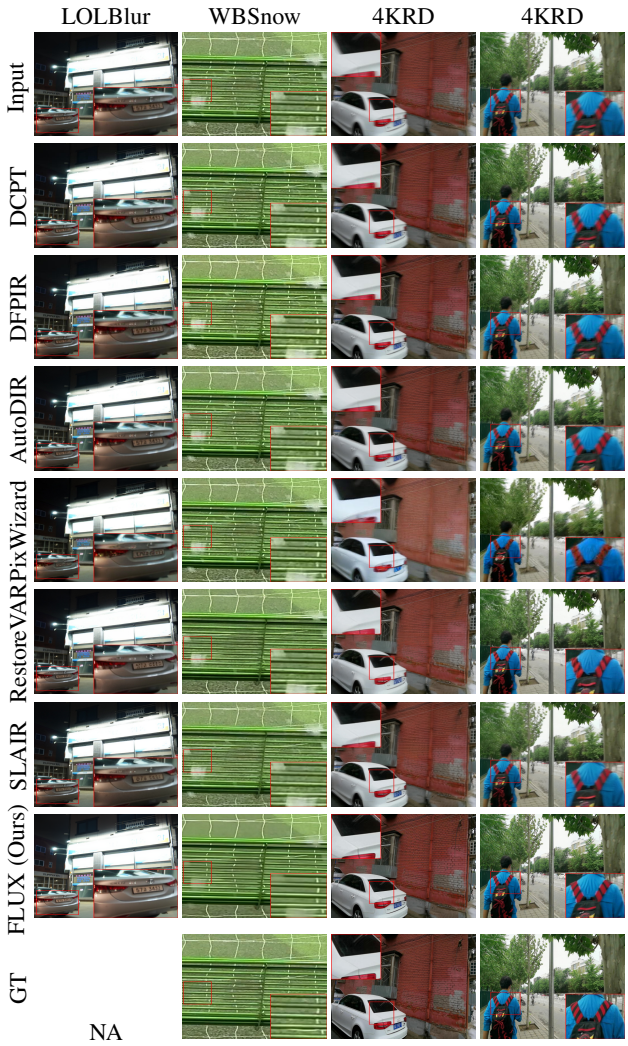


Figure 12: Additional qualitative results for state-of-the-art AiOR approaches for image restoration and the pre-trained FLUX model instantiated with our learned prompts on the LOLBlur (Zhou et al., 2022b) (low-light+blur, mixed), WBSnow (Guan et al., 2025) (snow, OOD), and 4KRD (Deng et al., 2021) (blur, OOD) datasets. WBSnow denotes the snow subset of WeatherBench.

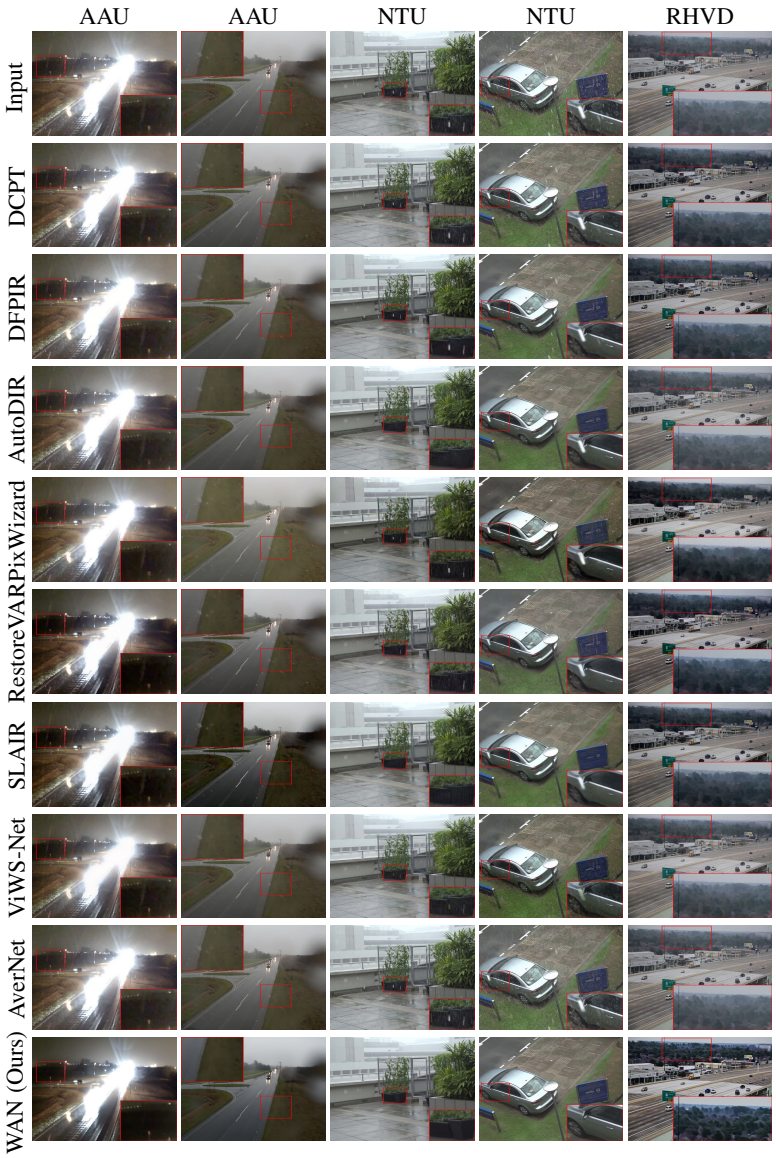


Figure 13: Qualitative results for state-of-the-art AiOR approaches (for images and videos), and the pre-trained WAN model instantiated with our learned prompts on the OOD datasets AAURain-Snow (Bahnsen & Moeslund, 2018) (rain, snow), NTU-Rain-Real (Chen et al., 2018) (rain), and RHVD (Chu et al., 2021) (haze). AAU: AAURainSnow, NTU: NTU-Rain.

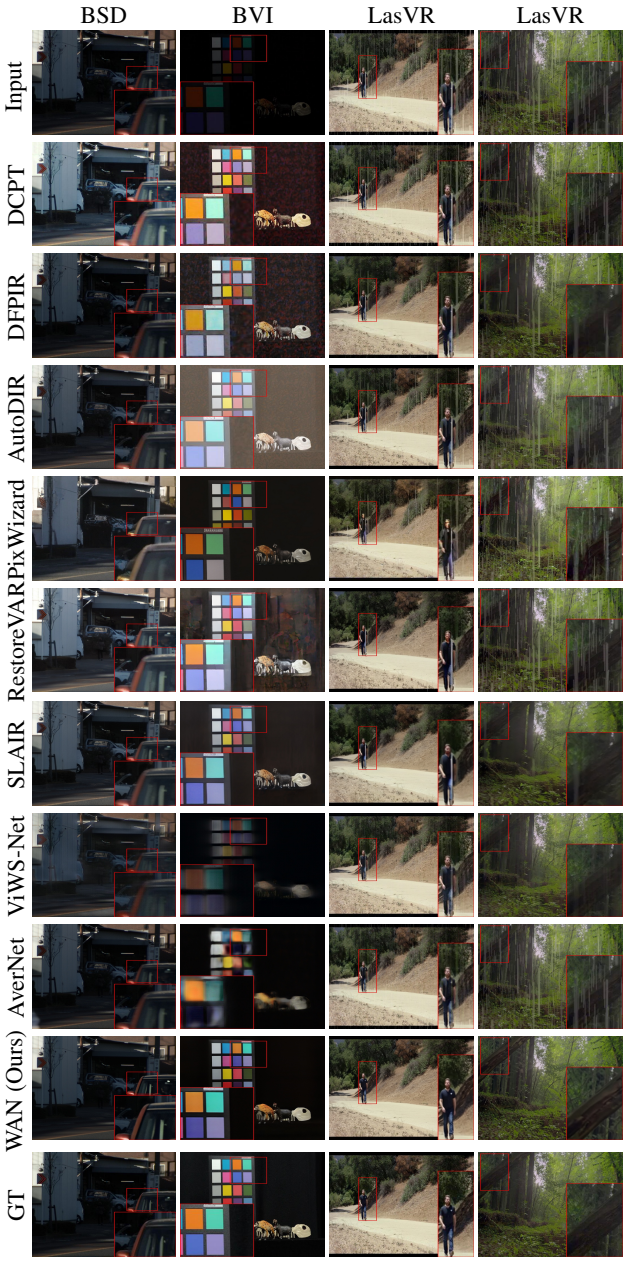


Figure 14: Qualitative results for state-of-the-art AiOR approaches (for images and videos), and the pre-trained WAN model instantiated with our learned prompts on the OOD datasets BSD (Zhong et al., 2023) (blur), LasVR (Liu et al., 2019) (rain), and BVI (Lin et al., 2024a) (low-light).

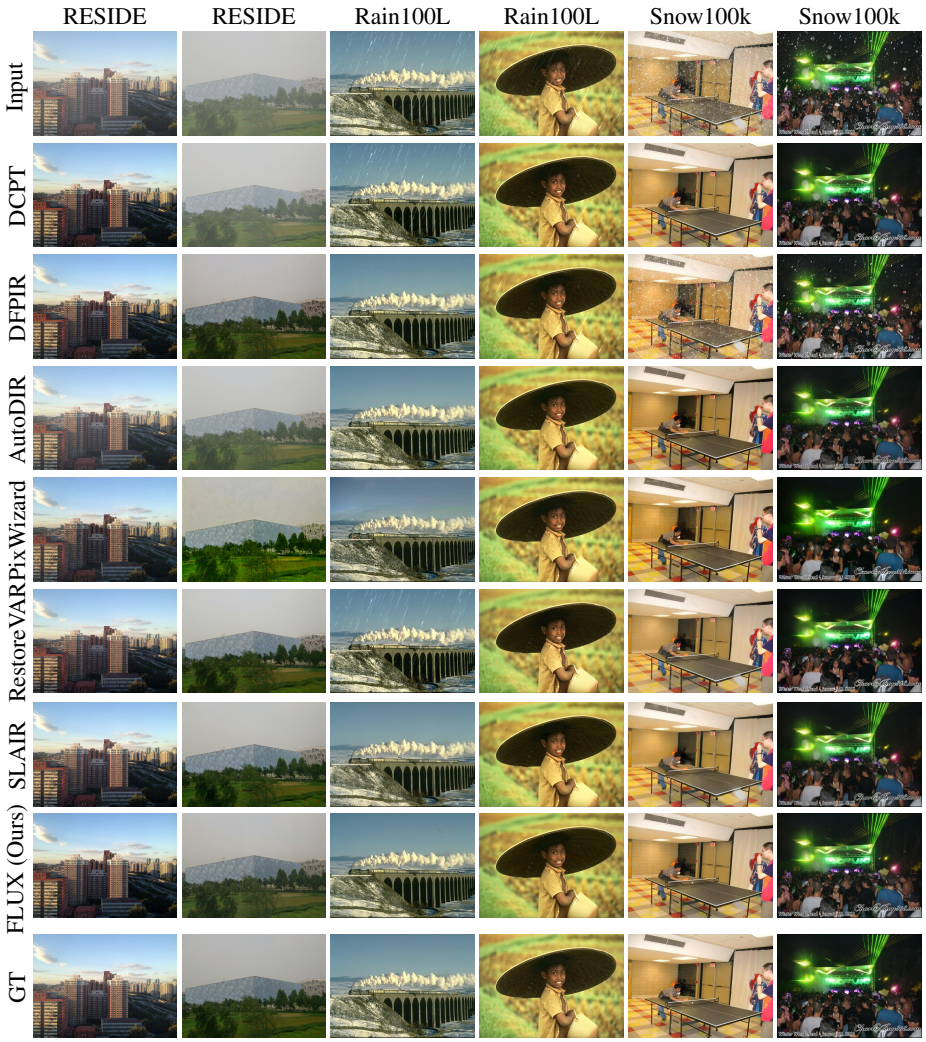


Figure 15: Qualitative results for state-of-the-art AiOR approaches for image restoration and the pre-trained FLUX model instantiated with our learned prompts on the in-distribution datasets RESIDE (Li et al., 2019) (haze), Rain13K (Zamir et al., 2021) (rain), and Snow100k (Liu et al., 2018) (snow).

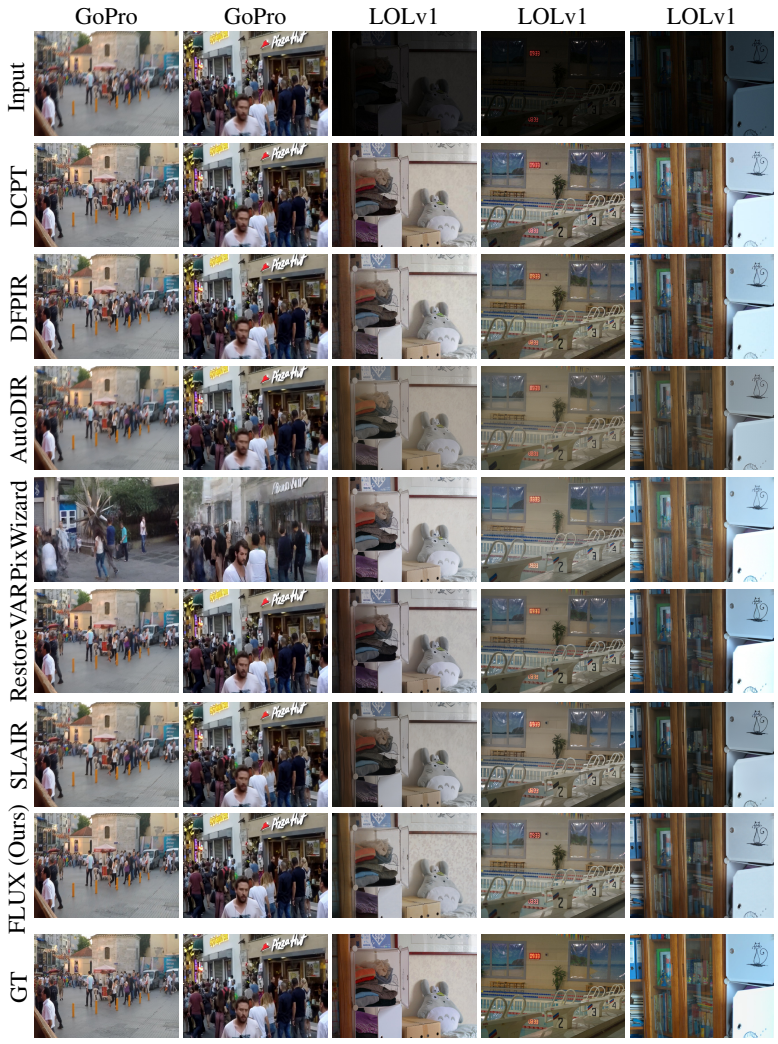


Figure 16: Qualitative results for state-of-the-art AiOR approaches for image restoration and the pre-trained FLUX model instantiated with our learned prompts on the in-distribution datasets GoPro (Nah et al., 2017) (blur) and LOLv1 (Wei et al., 2018) (low-light).



Figure 17: Qualitative results for state-of-the-art AiOR approaches (for images and videos), and the pre-trained WAN model instantiated with our learned prompts on the in-distribution datasets GoPro (Nah et al., 2017) (blur), REVIDE (Nah et al., 2017) (haze) and SDSD (Wang et al., 2021) (low-light).

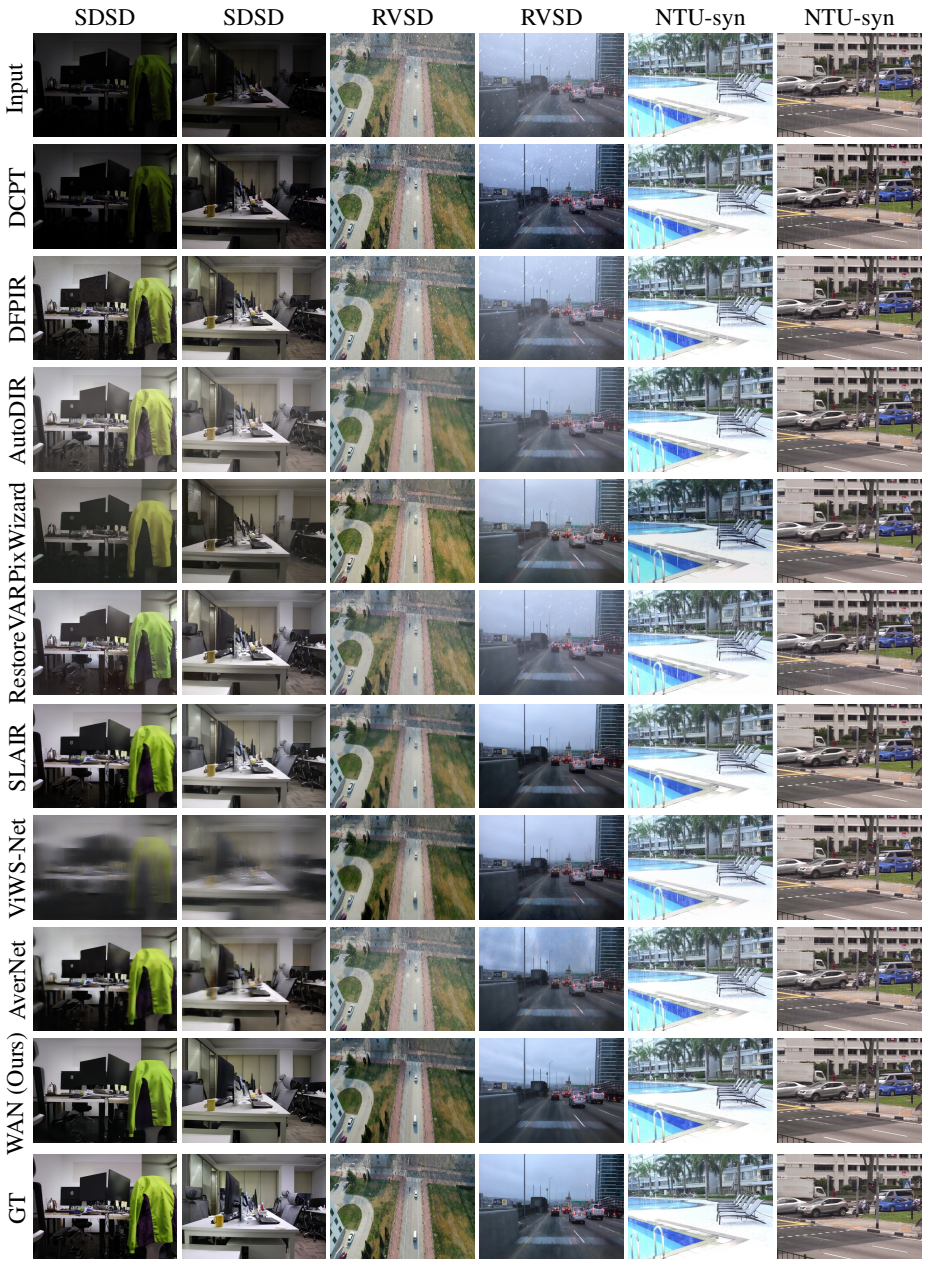


Figure 18: Qualitative results for state-of-the-art AiOR approaches (for images and videos), and the pre-trained WAN model instantiated with our learned prompts on the in-distribution datasets SDSD (Wang et al., 2021) (low-light), RVSD (Chen et al., 2023) (snow), and NTURain-Synthetic (Chen et al., 2018) (rain).

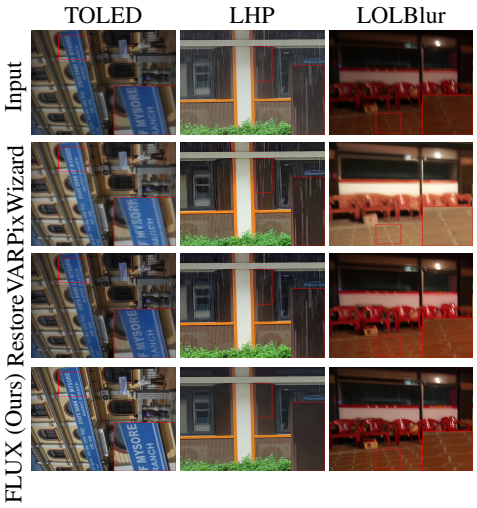


Figure 19: Qualitative results for the pre-trained FLUX model using our learned prompts, PixWizard (Lin et al., 2024b) and RestoreVAR (Rajagopalan et al., 2025) on OOD, mixed and unseen degradation samples from Fig. 4 of the main paper. WBSnow denotes the snow subset of WeatherBench (Guan et al., 2025).



Figure 20: Qualitative results for the pre-trained WAN model using our learned prompts along with DCPT (Hu et al., 2025), AutoDIR (Jiang et al., 2023), PixWizard (Lin et al., 2024b) and RestoreVAR (Rajagopalan et al., 2025) on OOD degradation samples from Fig. 5 of the main paper. AAU: AAURainSnow (Bahnsen & Moeslund, 2018), NTU: real test set of NTU-Rain (Chen et al., 2018).