NOISE-CONTROLLED IMAGE RESTORATION VIA PLUG-AND-PLAY CHANNEL-WISE MODULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Real-world weather conditions often involve combinations of rain, snow, haze, and poor illumination, yet image restoration networks are typically trained separately for individual degradations. Although these degradations may share common representation at certain feature levels, incorporating degradation-specific processing without redesigning the restoration backbone remains challenging. We propose NCRestoration, a plug-and-play conditioning framework for all-in-one image restoration that applies channel-wise affine modulation after complete backbone blocks. Fixed random Fourier features encode degradation categories, and independent lightweight controllers adapt features at different depths while preserving each block's internal computation. Multi-hot conditions represent mixed degradations; an optional jointly trained image-based predictor supplies soft conditions for label-free inference. The same interface supports convolutional and Transformer backbones with sequential or U-shaped structures. With known degradation labels, NC improves average single-weather PSNR over matched plain backbones by 0.47 dB for Restormer and 1.03 dB for PromptIR, and improves all seven CDD-11 weather mixtures for both backbones (+0.84 and +0.83 dB). The same modules accept soft conditions from a jointly trained CLIP predictor without modification: this blind variant improves the Transformer backbones by 0.82–1.40 dB on single weather and 0.17–0.30 dB on CDD-11 mixtures, and the predictor can be replaced without changing the restorer. Encoding ablations show a 0.76 dB gain from degradation-specific conditioning over constant control. These results support outside-block modulation as a plug-and-play conditioning interface for all-in-one restoration.

1 INTRODUCTION

Image restoration aims to recover clear images from inputs affected by degradations such as rain, haze, snow, and low light. Although task-specific networks perform well on individual degradations, they are usually trained separately, so deployment across several tasks requires maintaining multiple models and selecting an appropriate one for each input. Real scenes also contain overlapping conditions: rain or snow may occur together with haze, and any of these combinations may be further affected by low light. A restoration system should therefore handle both individual degradations and their combinations, rather than assume that each image belongs to one isolated task.

All-in-one restoration addresses this need by training one model on several degradation types. The challenge is to share useful image features while allowing different restoration tasks to use them differently. Existing methods guide this process through learned degradation representations, input-dependent prompts, and adaptive feature processing (Li et al., 2022; Potlapalli et al., 2023; Cui et al., 2025; Tian et al., 2025). Reusable prompt modules and degradation-classification pre-training also show that established backbones can be adapted for all-in-one restoration (Potlapalli et al., 2023; Hu et al., 2025). We study a specific design question within this direction: can the same block-wise modulation mechanism condition different restoration backbones without changing their internal operations?

We propose NCRestoration, a common conditioning interface for restoration backbones. Building on Fourier-conditioned affine modulation (Hou & Liu, 2026), we encode degradation categories with fixed random Fourier features and use an independent multilayer perceptron at each block to

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.

predict channel-wise scale and shift parameters. Applying these transformations after complete blocks preserves their internal residual computation and accommodates both convolutional and Transformer features. The same interface fits sequential networks and U-shaped encoder–decoders without changing their downsampling, upsampling, or skip connections. Here, plug-and-play refers to architectural integration; the backbone and controllers are trained jointly.

A multi-hot condition combines the codes of co-occurring degradations without defining a new category for each mixture. For blind restoration, a jointly trained image-based predictor supplies soft condition weights, so inference requires only the degraded image. The predictor is optional: the modulation mechanism does not depend on how the condition is obtained.

We evaluate NC on convolutional and Transformer backbones for single and mixed weather degradations. Oracle conditioning improves average single-weather PSNR for every tested backbone (Table 1) and improves all seven CDD-11 mixtures for both Restormer and PromptIR (Table 2). Encoding ablations and fixed-checkpoint condition interventions show that the restorer relies on degradation-specific control. The oracle comparisons measure the modulation interface itself, while the blind variant evaluates one practical source of the condition.

Our contributions are threefold:

- We develop a common outside-block modulation mechanism for convolutional and Transformer restoration backbones, including sequential and U-shaped networks, without changing their internal block operations.
- We support single and mixed degradations through a shared condition representation, with an optional jointly trained predictor for restoration from an image alone.
- We show gains over matched plain backbones on single and mixed weather benchmarks, and analyze the conditioning path through a parameter-matched control, task specialists, encoding ablations, and condition interventions.

2 Related Work

Task-specific and all-in-one restoration. Convolutional and Transformer networks provide strong backbones for task-specific restoration. NAFNet uses a simple nonlinear-activation-free convolutional design (Chen et al., 2022), while Restormer uses efficient self-attention for high-resolution restoration (Zamir et al., 2022). Both are commonly trained separately for different tasks. All-in-one methods instead share a model across degradation types: AirNet learns a contrastive degradation representation to guide restoration (Li et al., 2022), and PromptIR uses input-dependent prompts through a reusable plug-in module (Potlapalli et al., 2023). IDR decomposes degradations into shared underlying ingredients (Zhang et al., 2023), and InstructIR conditions restoration on natural-language instructions (Conde et al., 2024). DCPT takes a different approach, pre-training restoration encoders through degradation classification and demonstrating improvements for both CNNs and Transformers (Hu et al., 2025). These methods establish that all-in-one restoration need not require an entirely new backbone. Our focus is the conditioning operation itself: independent channel-wise scale and shift parameters applied after complete blocks, learned jointly with the restoration network rather than through a separate degradation-classification pre-training stage.

Degradation conditioning and feature modulation. Conditional feature processing allows a shared network to respond differently to different inputs or tasks. FiLM predicts condition-dependent affine transformations of feature channels (Perez et al., 2018), and noise-conditional score networks use layer-specific conditional parameters to handle different noise levels (Song & Ermon, 2019b;a). In restoration, SFT layers predict spatial affine parameters from semantic priors (Wang et al., 2018), AdaFM adds modulation layers to adjust a network to continuous degradation levels (He et al., 2019), and CResMD maps a vector of degradation levels to weights that scale the residual branches of each block (He et al., 2020). AdaFM and CResMD control the strength of known degradations inside the residual branches of convolutional networks. TCN combines Fourier embeddings with channel-wise affine modulation to condition a model on adversarial threat levels (Hou & Liu, 2026). We build on this design for restoration, conditioning on degradation types and their combinations, and applying channel-wise modulation after complete convolutional or Transformer blocks.

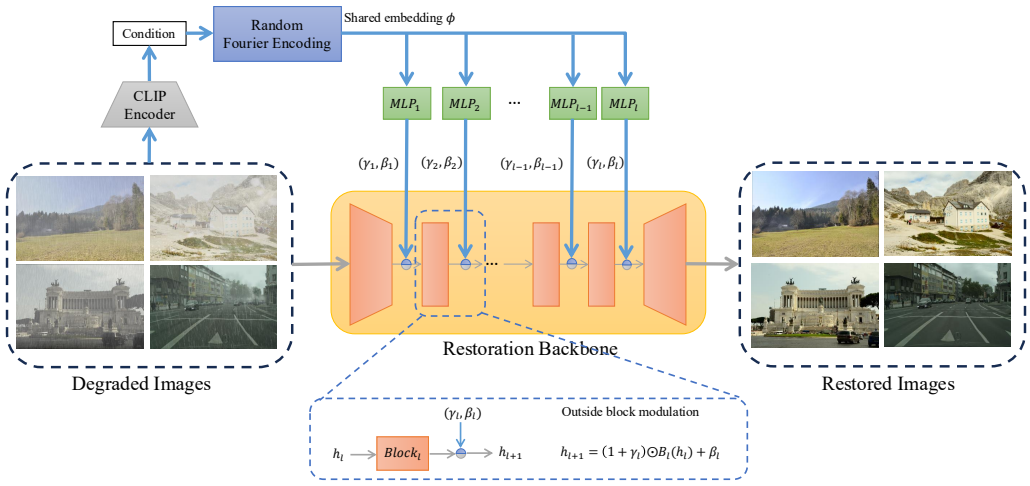

Figure 1: Overview of NCRestoration. In the blind setting shown here, a CLIP-based predictor estimates degradation scores from the input image. These scores weight fixed random Fourier category embeddings to form a shared condition code. Independent block-specific MLPs predict channel-wise scale offsets and shifts, which are applied after complete backbone blocks. The inset illustrates this outside-block modulation. In the oracle setting, known degradation labels replace the predicted scores. The image examples illustrate different inputs; each image is restored independently.

Related restoration methods also adapt intermediate features to the degradation. MWFormer combines predicted weather representations, hypernetworks, and feature-wise linear modulation (Zhu et al., 2024). AdaIR adapts the interaction of degradation-dependent frequency components (Cui et al., 2025), while DFPIR uses degradation-guided feature perturbations to reduce interference among tasks (Tian et al., 2025). Our method uses a different placement rule: the modulation follows each complete block, including its internal residual computation, rather than modifying the block's attention, frequency processing, or other internal operations. This rule provides the same conditioning interface for the convolutional and Transformer backbones considered in our experiments.

Blind restoration and mixed degradations. Blind restoration estimates the information needed for restoration from the degraded image itself. TransWeather uses weather-type queries in a Transformer encoder–decoder (Valanarasu et al., 2022); DA-CLIP learns content and degradation representations (Luo et al., 2024); and MPerceiver uses CLIP-based degradation predictions to guide textual and visual prompts (Ai et al., 2024). OneRestore targets composite degradations with scene-descriptor-guided restoration and introduces the CDD-11 benchmark (Guo et al., 2024). These approaches motivate image-based conditioning, but estimating a condition and applying it to a restorer are separate design choices. In our blind variant, a supervised CLIP-based predictor supplies degradation scores to the same Fourier-conditioned modulation modules used with known labels. Multiple degradation types can be active together, allowing the condition to describe mixed inputs rather than a single mutually exclusive class. We evaluate both single and mixed degradations.

3 METHOD

Given a degraded image and its degradation condition, our method encodes the condition and uses independent controllers to modulate the output of each backbone block (Figure 1). When the condition is unknown, an image-based predictor estimates it from the degraded image. We first define the condition representation, then describe block-wise modulation, blind prediction, and training.

3.1 PROBLEM FORMULATION

Let $x \in \mathbb{R}^{3\times H\times W}$ denote a degraded RGB image and y its paired clean target. We represent the known condition by a binary vector $\mathbf{a} \in \{0,1\}^K$, where K is the number of condition categories. A single degradation is represented by a one-hot vector; mixed degradations activate multiple entries, giving a multi-hot vector. A clean category may also be included for identity restoration, where $x = y$.

We learn a restoration model R_θ by adding modulation modules after backbone blocks. The parameters θ include the backbone and its controllers. In the *oracle setting*, the ground-truth condition is supplied to the model. In the *blind setting*, an image-based predictor estimates the condition from the degraded input. Both settings use the same condition encoding and modulation mechanism.

3.2 RANDOM FOURIER DEGRADATION CODE

To provide a shared code for all controllers, we map each condition category $c \in \{0,\dots,K-1\}$, with $K \geq 2$, to the scalar $\epsilon_c = c/(K-1)$. Following the random Fourier feature encoding used in TCN (Tancik et al., 2020; Hou & Liu, 2026), we construct a fixed embedding for each category as

$$\phi_c = \Phi(\epsilon_c) = [\sin(2\pi\boldsymbol{\omega}\epsilon_c), \cos(2\pi\boldsymbol{\omega}\epsilon_c)] \in \mathbb{R}^d, \qquad \omega_j \sim \mathcal{N}(0,\sigma^2), \tag{1}$$

where d is even and $\boldsymbol{\omega} \in \mathbb{R}^{d/2}$ is sampled once and kept fixed throughout training. The indices identify condition categories rather than severity levels; their assignment is fixed within each experiment. The encoding depends on this assignment and is not invariant to reordering the categories.

We combine the category embeddings using their condition weights. The same expression applies to binary labels and, more generally, nonnegative weights $\mathbf{a} \in \mathbb{R}^K_{\geq 0}$:

$$\phi(\mathbf{a}) = \sum_{c=0}^{K-1} a_c \phi_c. \tag{2}$$

For a single degradation, $\phi(\mathbf{e}_c) = \phi_c$, where $\mathbf{e}_c$ is the one-hot vector for category c. For mixed degradations, we sum the active embeddings without dividing by the number of active categories. This represents a combination without introducing a new category. We sum embeddings, not their scalar indices.

3.3 INDEPENDENT OUTSIDE-BLOCK MODULATION

We condition the backbone at block outputs so that its internal operations remain unchanged. Let B_ℓ map input features h_ℓ to $z_\ell = B_\ell(h_\ell) \in \mathbb{R}^{C_\ell\times H_\ell\times W_\ell}$. Given the shared condition code $\phi \in \mathbb{R}^d$, an independent controller $g_\ell : \mathbb{R}^d \to \mathbb{R}^{C_\ell} \times \mathbb{R}^{C_\ell}$ predicts a scale offset and a shift for each channel:

$$(\gamma_\ell, \beta_\ell) = g_\ell(\phi), \tag{3}$$

$$\widetilde{z}_\ell = (1+\gamma_\ell) \odot z_\ell + \beta_\ell. \tag{4}$$

Here $\gamma_\ell, \beta_\ell \in \mathbb{R}^{C_\ell}$ are the scale offset and additive shift, and $\odot$ denotes element-wise multiplication. Both vectors are broadcast over spatial positions, preserving the output shape. One controller produces both vectors. This channel-wise affine transformation is related to FiLM (Perez et al., 2018) and follows the conditional operator of TCN (Hou & Liu, 2026). The condition code is shared, but each block has its own controller parameters, even when blocks have the same number of channels. The same condition can therefore produce different feature adjustments at different depths.

The modulated output $\widetilde{z}_\ell$ replaces z_ℓ, and subsequent operations use these features through the backbone's existing connections. Modulation follows the block's internal residual addition, when present, rather than acting only on its residual branch. This placement applies to convolutional and Transformer blocks that output spatial feature maps, in both sequential and U-shaped networks. Downsampling, upsampling, and skip connections between blocks retain their original structure.

We initialize each controller to output zero for every condition, so $\gamma_\ell = \beta_\ell = 0$ and $\widetilde{z}_\ell = z_\ell$ at initialization. The wrapped network thus initially preserves the function of a backbone with the same weights, while learning condition-dependent adaptations during training.

3.4 BLIND CONDITION PREDICTION

For blind restoration, an image-based predictor s_ψ produces condition logits $s(x) = s_\psi(x) \in \mathbb{R}^K$. These replace the supplied label as the source of conditioning, without changing the modulation modules. We convert the logits to weights according to the experiment's label setting:

$$q(x) = \begin{cases} \mathrm{softmax}(s(x)/\tau), & \text{single-label}, \\ \mathrm{sigmoid}(s(x)/\tau), & \text{multi-label}. \end{cases} \tag{5}$$

Here $\tau > 0$ is a temperature parameter. Softmax weights sum to one; independent sigmoid weights need not. Without thresholding or an argmax in the restoration path, we compute

$$\bar{\phi}(x) = \sum_{c=0}^{K-1} q_c(x)\, \Phi(\epsilon_c), \qquad \hat{y} = R_\theta(x, \bar{\phi}(x)). \tag{6}$$

This uses Equation 2 with $\mathbf{a} = q(x)$ and allows the restoration loss to train the predictor through the condition code. The resulting code conditions a single restoration pass, rather than one pass per category. When an image is processed in tiles, each tile uses the same predicted image-level condition. At inference, only the degraded image is required. Our experiments use a CLIP-based predictor (Radford et al., 2021), but the modulation mechanism does not require that encoder architecture.

3.5 TRAINING OBJECTIVES

Let $x_p, y_p \in \mathbb{R}^{3\times H_p \times W_p}$ denote aligned degraded and target training patches, and let ϕ denote the supplied condition code. The reconstruction objective is the mean absolute error per RGB element,

$$\mathcal{L}_{\mathrm{rest}}(x_p, y_p; \phi) = \frac{1}{3H_pW_p} \left\| R_\theta(x_p, \phi) - y_p \right\|_1 . \tag{7}$$

Oracle training uses $\phi = \phi(\mathbf{a})$ for one-hot or multi-hot ground-truth labels. The plain baseline uses the same reconstruction objective without conditioning. Let v be the view of the degraded image supplied to the predictor for the same training example; it may cover the complete image while the restorer receives a patch. The blind model jointly minimizes

$$\mathcal{L}_{\mathrm{joint}} = \mathcal{L}_{\mathrm{rest}}(x_p, y_p; \bar{\phi}(v)) + \lambda_{\mathrm{cls}} \mathcal{L}_{\mathrm{cls}}(s(v), \mathbf{a}), \tag{8}$$

Here λ_{cls} balances reconstruction and condition supervision. We use cross-entropy on raw logits for single-label experiments and mean binary cross-entropy for all examples in multi-label experiments, including one-hot examples. Both losses are averaged over the mini-batch; their expanded definitions are in Appendix A.1. The backbone, controllers, and predictor are trained jointly, while Fourier frequencies remain fixed. All degradations use the same L1 reconstruction objective.

4 EXPERIMENTS

4.1 IMPLEMENTATION DETAILS

Models and training. We evaluate NAFNet, Restormer, and PromptIR. Controllers use a 64-dimensional fixed Fourier code with $\sigma = 1$, hidden width 128, SiLU, and zero-initialized output layers. The blind model uses CLIP ViT-B/32 and a linear head, jointly trained with temperature $\tau = 1$ and classification weight 0.01. Our backbone–NC experiments use seed 2026, batch size 16, 128×128 crops, L1 reconstruction, and at most 100k updates. NAFNet uses Adam with initial learning rate 10^{-3}; Restormer and PromptIR use AdamW at 2×10^{-4}, 5k warmup steps, and gradient clipping at norm 1. Each Transformer plain–NC pair has matched shared-weight initialization and optimization. Architecture-specific settings and validation schedules are in Appendix A.2. TransWeather is trained on the same mixture with its official Smooth L1 and perceptual objective; details are in Appendix A.2.

Data. The shared training mixture contains WeatherBench rain, haze, and snow (Guan et al., 2025), LOL-v2 real and synthetic low light (Yang et al., 2021), RainCityScapes rain–fog (Hu et al., 2019), and clean identity pairs: 95,414 indexed pairs in total. The condition order is [clean, rain, haze, snow,

Table 1: Single-degradation restoration on WeatherBench and LOL-v2: full-image Y-channel PSNR (dB) / SSIM, both higher is better. Rows are grouped by backbone; gray shading identifies NC variants, and bold marks the best value within each backbone group. The Input column distinguishes blind inference from oracle access to the true degradation label. All rows use the same 800 images (200 per task) and evaluation protocol. Average excludes clean and mixtures. Optimization recipes are described in Section 4.1.

Model	Input	Rain	Haze	Snow	Low light	Average
AirNet	Image	29.58 / 0.895	16.75 / 0.645	23.18 / 0.799	16.03 / 0.801	21.39 / 0.785
TransWeather	Image	31.06 / 0.930	20.26 / 0.691	26.68 / 0.811	23.53 / 0.893	25.38 / 0.831
NAFNet-32	Image	27.54 / 0.807	17.14 / 0.569	22.14 / 0.745	18.75 / 0.646	21.39 / 0.692
NC-NAFNet	Image + label	32.90 / 0.932	20.64 / 0.694	26.23 / 0.819	**23.74** / 0.898	25.88 / 0.836
CLIP+NC-NAFNet	Image	**35.04 / 0.946**	**23.62 / 0.729**	**28.07 / 0.836**	23.14 / **0.903**	**27.46 / 0.853**
Restormer	Image	33.89 / 0.940	22.56 / 0.722	28.32 / 0.838	24.55 / 0.914	27.33 / 0.854
NC-Restormer	Image + label	34.86 / 0.949	22.44 / 0.728	28.62 / 0.842	**25.27** / 0.917	27.80 / 0.859
CLIP+NC-Restormer	Image	**35.97 / 0.951**	**24.01 / 0.742**	**29.04 / 0.847**	23.57 / **0.920**	**28.15 / 0.865**
PromptIR	Image	33.04 / 0.932	21.36 / 0.712	28.09 / 0.833	24.40 / 0.909	26.72 / 0.847
NC-PromptIR	Image + label	35.24 / 0.950	22.35 / 0.728	28.66 / 0.841	**24.75 / 0.918**	27.75 / 0.859
CLIP+NC-PromptIR	Image	**36.11 / 0.955**	**24.00 / 0.741**	**29.09 / 0.849**	23.28 / 0.912	**28.12 / 0.864**

low light]; rain–fog uses $[0, 1, 1, 0, 0]$. The same trained models are evaluated on 800 single-weather images (200 per task). A separate CDD-11 experiment (Guo et al., 2024) evaluates four single conditions and seven mixtures, with 200 images per condition. All evaluated weather combinations are represented in the corresponding training mixture.

Evaluation. Full-image evaluation uses 512-pixel tiles with 32-pixel overlap and 8-bit output quantization. WeatherBench and LOL-v2 report Y-channel PSNR/SSIM after a four-pixel border crop; CDD-11 reports RGB metrics without border cropping. SSIM uses an 11×11 Gaussian window. Task averages exclude clean identity, with equal weight per displayed condition.

4.2 Single-degradation restoration

We first evaluate whether one restorer can handle individual weather degradations without selecting a separate task-specific network. Table 1 reports rain, haze, snow, and low light; all unified models are trained on the weather mixture described in Section 4.1.

On NAFNet, oracle conditioning raises the four-task average from 21.39 to 25.88 dB (+4.49 dB), and the blind CLIP+NC variant reaches 27.46 dB and 0.853 SSIM (+6.07 dB). Figure 2 shows example NAFNet restorations.

Transformer backbones. Plain Restormer and PromptIR (Potlapalli et al., 2023) obtain 27.33 and 26.72 dB, respectively. With oracle conditioning, NC-Restormer reaches 27.80 dB (+0.47 dB) and NC-PromptIR reaches 27.75 dB (+1.03 dB); each plain–NC pair shares the same optimization recipe and matched initialization. NC-PromptIR improves all four tasks, while NC-Restormer improves rain, snow, and low light but lowers haze by 0.13 dB. Because NC-PromptIR retains PromptIR's original prompts, these gains indicate that outside-block modulation complements prompt-based conditioning. The blind CLIP+NC variants reach 28.15 and 28.12 dB, improving their plain backbones by 0.82 and 1.40 dB.

Blind versus oracle conditioning. For all three backbones, the blind variant exceeds the oracle variant on rain, haze, and snow but has lower low-light PSNR. The two variants are trained separately, and the predictor supplies soft weights rather than one-hot labels, so this gap does not indicate that predicted labels are more accurate than ground truth (Appendix E). The predictor identifies the degradation correctly on 95.5–100% of images for every backbone and task; low light, the only task where blind trails oracle, is among the least accurately predicted (95.5–96.5%).

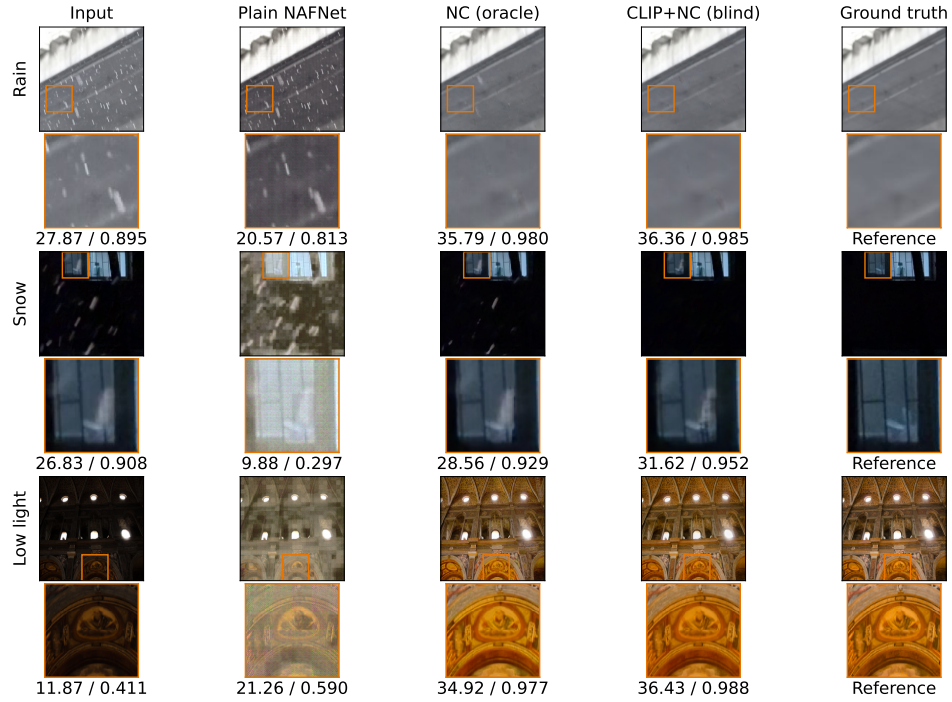

Figure 2: NAFNet restoration on rain, snow, and low light. Each full image is followed by an enlarged region, marked in orange and identical across methods. Values are full-image PSNR-Y/SSIM-Y. The complete four-condition comparison and the example-selection rule are in Appendix G.

Other all-in-one baselines. TransWeather (Valanarasu et al., 2022), trained on the same mixture with its own objective, obtains 25.38 dB, below both NC-NAFNet variants, although it exceeds oracle NC-NAFNet on snow. AirNet (Li et al., 2022) obtains 21.39 dB; its training produced non-finite gradients near 26.7k steps, so we report its best saved checkpoint at 25k steps (Appendix A.2).

Comparison with task specialists. Appendix C compares the unified model with four task-specific NAFNets on the same 800 images. Blind NC-NAFNet exceeds the corresponding specialist on rain, haze, and snow and trails the low-light specialist by only 0.02 dB, so one conditioned model matches per-task specialists across all four tasks.

4.3 Mixed-degradation restoration

We next evaluate weather combinations on CDD-11 (Guo et al., 2024), which provides five two-degradation and two three-degradation conditions. Each unified model restores all seven combinations with one set of weights; known multi-hot labels supply the oracle condition, and the CLIP-based predictor supplies the blind condition. All evaluated combinations occur in training, so these experiments measure restoration of seen mixtures rather than generalization to unseen combinations.

Table 2 reports the seven compound conditions, with 200 full images each. Restormer and PromptIR are trained as matched plain and NC pairs on all CDD-11 conditions, with checkpoints selected on a held-out training split. Oracle NC improves every mixture for both backbones, raising average PSNR from 25.78 to 26.62 dB for Restormer (+0.84 dB) and from 25.78 to 26.61 dB for PromptIR (+0.83 dB). Blind CLIP+NC reaches 25.95 dB for Restormer and 26.08 dB for PromptIR, 0.17 and 0.30 dB above the plain models. For both backbones, the blind variant exceeds oracle NC on haze+snow but falls below the plain model on the three two-degradation low-light mixtures, so on CDD-11 blind conditioning recovers only part of the oracle gain. For NAFNet, we compare with a specialist-averaging baseline that averages the outputs of the relevant single-weather specialists. Oracle and blind NC-NAFNet reach 24.89 and 24.37 dB, exceeding this baseline by 7.26 and 6.74 dB.

Table 2: Mixed-degradation restoration on CDD-11, grouped by backbone. Quantized full-frame RGB PSNR (dB; higher is better) uses 200 images per column. R, H, S, and L denote rain, haze, snow, and low light; Average covers only the seven mixtures. Oracle methods receive the true condition; blind methods receive only the image. Bold marks the best value within each backbone group. †: checkpoints selected on the evaluation images (Appendix A.2); Restormer and PromptIR rows select checkpoints on a source-disjoint held-out training split.

	Two degradations					Three degradations		
Model	R+H	H+S	L+H	L+R	L+S	L+H+R	L+H+S	Average
NAFNet specialist averaging†	18.56	19.30	17.08	18.44	18.07	16.14	15.81	17.63
NC-NAFNet (oracle)†	**27.23**	**26.34**	**24.38**	**24.91**	**24.32**	**23.62**	**23.45**	**24.89**
CLIP+NC-NAFNet (blind)†	26.66	25.84	23.89	24.12	23.93	23.24	22.93	24.37
Restormer (plain unified)	28.42	28.01	25.22	25.40	24.99	24.28	24.17	25.78
NC-Restormer (oracle)	**29.66**	29.42	**25.89**	**25.97**	**25.54**	**25.00**	**24.89**	**26.62**
CLIP+NC-Restormer (blind)	29.21	**29.55**	24.84	24.89	24.68	24.20	24.27	25.95
PromptIR (plain unified)	28.59	28.21	25.04	25.41	24.97	24.15	24.06	25.78
NC-PromptIR (oracle)	**29.91**	29.31	**25.82**	**25.99**	**25.52**	**24.94**	**24.78**	**26.61**
CLIP+NC-PromptIR (blind)	29.56	**29.68**	24.87	25.06	24.88	24.31	24.23	26.08

Table 3: Restorer parameter counts with and without NC. Counts exclude the optional CLIP predictor.

Backbone	Plain	Conditioned	Increase
NAFNet	29.160M	32.002M	9.75%
Restormer	26.127M	28.425M	8.80%
PromptIR	35.592M	38.221M	7.38%

Because the specialists never see mixtures during training, this comparison reflects unified training as well as conditioning. The full specialist matrix is in Appendix B; each specialist remains best on its own single-weather task.

4.4 Ablation studies

We separate the contributions of additional parameters, degradation-specific encoding, and the condition supplied at inference.

Capacity and implementation cost. Table 3 reports the parameters added by the block-wise controllers, 7–10% of each backbone. Because the controllers depend only on the condition, they are evaluated once per image regardless of resolution, and each block adds only a channel-wise affine operation. A plain NAFNet-34 has more parameters than NC-NAFNet-32 (32.90M versus 32.00M) but reaches only 21.03 dB on the single-weather average, versus 25.88 dB (Appendix F), so added capacity alone does not explain NC's gain.

Condition encoding on CDD-11. We retrain constant-condition and learned-embedding variants alongside the fixed-RFF reference, with matched backbone and adapter initialization. Constant control supplies the same code to every image; learned embeddings replace the fixed category codes with trainable vectors while retaining the same sum for mixtures. Appendix D gives initialization, training, and selection details.

Table 4 shows that degradation-specific Fourier conditioning improves overall PSNR by 0.76 dB over constant conditioning, with gains of 0.97 dB on single weather and 0.65 dB on mixtures. The constant-condition model still reaches 25.92 dB because its backbone observes the degraded image and learns joint restoration; degradation-specific codes add a consistent gain on top of this. Fixed RFF also exceeds learned embeddings by 1.81 dB overall. The learned variant stopped improving early (Appendix D), so fixed codes are the more reliable choice under our recipe.

Table 4: Condition-encoding ablations on CDD-11. PSNR is measured in RGB on all 2,200 degraded test images with 8-bit quantization, no border crop, and 512-pixel tiles with 32-pixel overlap. Single and mixed denote unweighted averages over four single-weather and seven mixed-weather conditions; overall averages all 11 conditions, excluding clear identity. SSIM is the overall average. Higher is better.

	PSNR (dB) ↑			SSIM ↑
Condition encoding	Single	Mixed	Overall	Overall
Constant condition	28.84	24.24	25.92	0.832
Learned embeddings	27.80	23.19	24.87	0.811
Fixed RFF (full NC)	**29.81**	**24.89**	**26.68**	**0.844**

Does the restorer use its condition? On a fixed, separately trained blind NAFNet checkpoint, replacing predicted soft conditions with uniform or cyclically incorrect conditions decreases the four-task average by 9.36 and 14.03 dB, respectively (Appendix E), confirming that the restorer relies on its conditioning path rather than ignoring it.

5 Discussion and limitations

Reliability of predicted conditions. The CLIP-based predictor is reliable on single weather (95.5–100% exact-match accuracy) but less so on CDD-11 mixtures (57.5–97.5% across backbones). Incorrect weights can lead to inappropriate corrections and reduce restoration quality. CLIP is optional: alternative degradation classifiers and uncertainty-aware predictions could be evaluated without changing NC's modulation interface.

Real-world adverse weather. Although our evaluation includes real-weather imagery, available paired datasets cover limited conditions, and tested mixtures occur during training. Robustness to unseen combinations and spatially varying severity remains to be tested. Broader evaluation requires suitable real-world data, for which obtaining aligned degraded and clean images is challenging.

Untested design alternatives. We apply modulation after complete blocks so that NC can be added without changing a backbone's internal operations; this is what allows one interface to wrap NAFNet, Restormer, and PromptIR. We have not compared this placement with modulation inside blocks, nor independent block-wise controllers with a single shared controller, under matched training. Inside-block modulation could condition features more directly but requires backbone-specific changes; quantifying this trade-off is left to future work.

Gated modulation. NC gains vary across backbones and tasks. A learnable gate on each MLP-to-block connection could scale its affine correction relative to the unmodulated block output, adjusting conditioning strength at different depths. This extension would retain the plug-and-play interface, but its effect on restoration quality and stability has not been evaluated.

6 Conclusion

We present NCRestoration, a reusable plug-and-play modulation interface for both CNN and Transformer restoration backbones. Independent outside-block controllers use shared Fourier-based degradation codes to handle single and mixed degradations without changing the internal backbone blocks. The image-based predictor is optional; our contribution lies in the restoration interface rather than any particular classifier, and the condition source can be replaced without changing the restorer. Across three backbones, NCRestorationconsistently improves restoration performance over matched plain models for both single and mixed weather degradations, and ablations confirm the benefit of degradation-specific conditioning, with gains varying across backbones and tasks. A natural direction for future work is to evaluate the framework under previously unseen combinations of weather degradations and to further assess its ability to generalize beyond the conditions observed during training.

Reproducibility Statement

Architecture, optimization, preprocessing, checkpoint selection, and metric definitions are described in Section 4.1 and Appendix A. Anonymous code is provided in the supplementary material, including the conditioning modules, backbone wrappers, training and evaluation code, dataset configurations, and a launch script for every reported experiment. Public datasets remain subject to their original licenses, and the implementation preserves upstream license notices, including PromptIR's academic non-commercial license.

Ethics Statement

This study uses existing image datasets and does not involve human-subject intervention or collection of personally identifying information by the authors.

AI Use Statement

In this work, we used generative AI tools to implement methods, clean and reformat datasets, and help interpret results. Specifically, AI coding assistants implemented the model, training, and evaluation code, built the dataset manifests and train/test audits, and assisted in interpreting experimental results. We did not use generative AI tools to develop the research idea, hypotheses, or method, which were conceived by the authors, and generating synthetic data sets, formulating mathematical claims, and writing proofs are not applicable to this work. Additionally, we used generative AI tools to edit the paper for readability, revise parts of the text, identify related literature, and format references. We have reviewed all AI-assisted work, and an author checked the AI-implemented code. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

References

Yuang Ai, Huaibo Huang, Xiaoqiang Zhou, Jiexiang Wang, and Ran He. Multimodal prompt perceiver: Empower adaptiveness, generalizability and fidelity for all-in-one image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25432–25444, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Ai_Multimodal_Prompt_Perceiver_Empower_Adaptiveness_Generalizability_and_Fidelity_for_All-in-One_CVPR_2024_paper.html.

Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *European Conference on Computer Vision*, 2022. doi: 10.48550/arXiv.2204.04676. URL https://arxiv.org/abs/2204.04676.

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, 2024. doi: 10.1007/978-3-031-72764-1_1.

Yuning Cui, Syed Waqas Zamir, Salman Khan, Alois Knoll, Mubarak Shah, and Fahad Khan. AdaIR: Adaptive all-in-one image restoration via frequency mining and modulation. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/fb44a668c2d4bc984e9d6ca261262cbb-Abstract-Conference.html.

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. *arXiv preprint arXiv:2509.11642*, 2025. doi: 10.48550/arXiv.2509.11642. URL https://arxiv.org/abs/2509.11642.

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, 2024. doi: 10.1007/978-3-031-72655-2_15.

Jingwen He, Chao Dong, and Yu Qiao. Modulating image restoration with continual levels via adaptive feature modification layers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11048–11056, 2019. doi: 10.1109/CVPR.2019.01131.

Jingwen He, Chao Dong, and Yu Qiao. Interactive multi-dimension modulation with dynamic controllable residual learning for image restoration. In *European Conference on Computer Vision*, pp. 53–68, 2020. doi: 10.1007/978-3-030-58565-5_4.

Zhichao Hou and Xiaorui Liu. Towards one-for-all robustness across a continuum of threat levels. *arXiv preprint arXiv:2609.02440*, 2026. doi: 10.48550/arXiv.2609.02440. URL https://arxiv.org/abs/2609.02440.

Jiakui Hu, Lujia Jin, Zhengjian Yao, and Yanye Lu. Universal image restoration pre-training via degradation classification. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2501.15510.

Xiaowei Hu, Chi-Wing Fu, Lei Zhu, and Pheng-Ann Heng. Depth-attentional features for single-image rain removal. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8022–8031, 2019. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Hu_Depth-Attentional_Features_for_Single-Image_Rain_Removal_CVPR_2019_paper.html.

Boyun Li, Xiao Liu, Peng Hu, Zhongqin Wu, Jiancheng Lv, and Xi Peng. All-in-one image restoration for unknown corruption. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17452–17462, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Li_All-in-One_Image_Restoration_for_Unknown_Corruption_CVPR_2022_paper.html.

Ziwei Luo, Fredrik K. Gustafsson, Zheng Zhao, Jens Sjölund, and Thomas B. Schön. Controlling vision-language models for multi-task image restoration. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/467e109270678b3e2b3c2e7fa6b3d575-Abstract-Conference.html.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. doi: 10.1609/aaai.v32i1.11671. URL https://aaai.org/papers/11671-film-visual-reasoning-with-a-general-conditioning-layer/.

Vaishnav Potlapalli, Syed Waqas Zamir, Salman H. Khan, and Fahad Shahbaz Khan. PromptIR: Prompting for all-in-one blind image restoration. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/e187897ed7780a579a0d76fd4a35d107-Abstract-Conference.html.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Yang Song and Stefano Ermon. NCSN: Noise conditional score networks. GitHub repository, 2019a. URL https://github.com/ermongroup/ncsn.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, volume 32, 2019b. doi: 10.48550/arXiv.1907.05600. URL https://arxiv.org/abs/1907.05600.

Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/55053683268957697aa39fba6f231c68-Abstract.html.

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 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28165–28175, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Tian_Degradation-Aware_Feature_Perturbation_for_All-in-One_Image_Restoration_CVPR_2025_paper.html.

Jeya Maria Jose Valanarasu, Rajeev Yasarla, and Vishal M. Patel. Transweather: Transformer-based restoration of images degraded by adverse weather conditions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2353–2363, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Valanarasu_TransWeather_Transformer-Based_Restoration_of_Images_Degraded_by_Adverse_Weather_Conditions_CVPR_2022_paper.html.

Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 606–615, 2018. doi: 10.1109/CVPR.2018.00070.

Wenhan Yang, Wenjing Wang, Haofeng Huang, Shiqi Wang, and Jiaying Liu. Sparse gradient regularized deep retinex network for robust low-light image enhancement. *IEEE Transactions on Image Processing*, 30:2072–2086, 2021. doi: 10.1109/TIP.2021.3050850.

Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5728–5739, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Zamir_Restormer_Efficient_Transformer_for_High-Resolution_Image_Restoration_CVPR_2022_paper.html.

Jinghao Zhang, Jie Huang, Mingde Yao, Zizheng Yang, Hu Yu, Man Zhou, and Feng Zhao. Ingredient-oriented multi-degradation learning for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5825–5835, 2023. doi: 10.1109/CVPR52729.2023.00564.

Ruoxi Zhu, Zhengzhong Tu, Jiaming Liu, Alan C. Bovik, and Yibo Fan. MWFormer: Multi-weather image restoration using degradation-aware transformers. *IEEE Transactions on Image Processing*, 33:6790–6805, 2024. doi: 10.1109/TIP.2024.3501855. URL https://doi.org/10.1109/TIP.2024.3501855.

A Additional implementation details

Restoration backbones. Our primary weather-restoration NAFNet uses width 32, encoder depths $(2, 2, 4, 8)$, 12 middle blocks, and decoder depths $(2, 2, 2, 2)$, for 36 independently modulated NAFBlocks. The Restormer configuration uses encoder/latent depths $(4, 6, 6, 8)$, decoder depths $(6, 6, 4)$, and four refinement blocks, for 44 independently modulated Transformer blocks. One adapter follows each complete NAFBlock or Transformer block, after its two internal residual additions. Downsampling, upsampling, and input/output convolutions are not modulated. All backbones retain the backbone-level image residual connection, so $\hat{y} = x + F_\theta(x, \phi)$, with padding and output cropping handled by the backbone or evaluation wrapper as needed. Network outputs are not clamped inside the reconstruction training loss.

PromptIR adaptation. The NC-PromptIR implementation retains the official image-derived prompts and adds independent adapters after all 47 complete Transformer blocks, including three prompt-fusion blocks. It contains 38.221M parameters versus 35.592M for plain PromptIR. The matched plain and NC runs are reported in Tables 1 and 2.

Controller implementation. We use Fourier dimension $d = 64$ and frequency scale $\sigma = 1$, with frequencies registered as a fixed model buffer. Each g_ℓ is a two-layer MLP with hidden width 128 and a SiLU activation:

$$u_\ell = W_\ell^{(2)} \, \mathrm{SiLU}\left(W_\ell^{(1)}\phi + b_\ell^{(1)}\right) + b_\ell^{(2)}, \tag{9}$$

where $W_\ell^{(1)} \in \mathbb{R}^{128\times d}$ and $W_\ell^{(2)} \in \mathbb{R}^{2C_\ell\times 128}$. We initialize $W_\ell^{(2)} = b_\ell^{(2)} = 0$ to obtain identity modulation. The first C_ℓ outputs specify γ_ℓ and the remaining outputs specify β_ℓ. All wrappers use these parameters directly, without additional normalization or bounding. Each adapter adds $128(d+1) + 2C_\ell(128+1)$ trainable parameters; the fixed Fourier frequencies add none. Table 3 reports the resulting model sizes. A width-34 plain NAFNet provides a conservative capacity control: it is slightly larger than the width-32 conditioned model but contains no degradation-dependent computation.

At initialization, the zero output weights also make the restoration gradient through the condition code zero. This gradient can become nonzero as the controllers learn; classification supervision trains the predictor from the first update.

Blind predictor implementation. We implement $s_\psi(x) = A(E(P(x)))$ using a pretrained CLIP ViT-B/32 image encoder E and a trainable linear head A. The preprocessing operator P resizes its input to 224×224 using antialiased bicubic interpolation with `align_corners=False`, then applies the pretrained encoder's channel-wise CLIP normalization. We use temperature $\tau = 1$ and classification weight $\lambda_{\text{cls}} = 0.01$. The backbone, controller MLPs, classification head, and image encoder are optimized jointly, with separate optimizer groups for the restoration network, head, and encoder and experiment-specific learning-rate schedules. No CLIP text encoder or score-matching objective is used. Discrete label reporting uses argmax in single-label mode and a 0.5 threshold per coordinate in multi-label mode; these decisions do not replace soft conditioning during normal restoration inference.

Predictor views. During full-view weather training, the restoration branch receives an aligned random 128×128 patch, while the classifier receives a resized view of the complete degraded source image taken before patch cropping and augmentation. All datasets in this paper consist of stored paired images. Full-image evaluation computes one global condition and reuses it for all restoration tiles when tiling is enabled.

A.1 Classification objective

Here λ_{cls} balances reconstruction and condition supervision. Classification uses raw logits: cross-entropy in single-label experiments, where c denotes the target category, and mean binary cross-entropy in multi-label experiments. The latter applies to all samples in those experiments, including images with only one active degradation:

$$\mathcal{L}_{\text{cls}}(s, \mathbf{a}) = \begin{cases} -\log \operatorname{softmax}(s)_c, & \text{single-label,} \\ -\dfrac{1}{K}\displaystyle\sum_{c=0}^{K-1} \left[a_c \log \operatorname{sigmoid}(s_c) + (1-a_c)\log(1-\operatorname{sigmoid}(s_c))\right], & \text{multi-label.} \end{cases} \tag{10}$$

Both reconstruction and classification losses are averaged over the mini-batch. The backbone, modulation controllers, and image-based predictor are optimized jointly; the Fourier frequencies remain fixed. The same reconstruction objective is used across degradations, without a degradation-specific loss or task-specific output head.

A.2 Training and evaluation details

Datasets. The weather experiment uses the aligned rain, haze, and snow subsets of WeatherBench (Guan et al., 2025) and the real and synthetic subsets of LOL-v2 (Yang et al., 2021), with clean targets also sampled as identity pairs. The training mixture also includes 9,432 RainCityScapes (Hu et al., 2019) rain–fog pairs, which are used for training but not evaluated. Conditions follow the order [clean, rain, haze/fog, snow, low light], so a rain–fog sample uses $[0, 1, 1, 0, 0]$; the combined manifest contains 95,414 indexed pairs. CDD-11 (Guo et al., 2024) covers four individual degradations and seven two- or three-degradation combinations.

Optimization. Unless otherwise stated, runs train for at most 100,000 steps on NVIDIA GPUs with batch size 16, 128×128 aligned crops, Adam, an initial learning rate of 10^{-3}, cosine decay to 10^{-6}, and an L1 reconstruction loss. A weighted sampler balances the condition vectors (six for

the weather mixture, twelve for CDD-11). Weather runs validate every 5,000 steps and stop after six validations without at least 0.01 dB macro-PSNR improvement; tables report the best validation checkpoint. Plain and NC Restormer and PromptIR instead use AdamW at 2×10^{-4} with zero weight decay, 5,000 warmup steps, cosine decay, and gradient clipping at norm 1, with an effective batch size of 16 from four microbatches of four. Each plain–NC pair shares matched initialization.

Blind runs. The blind NAFNet run uses separate initial learning rates of 10^{-3} for the restorer, 10^{-4} for the classification head, and 10^{-5} for the CLIP encoder, with gradient clipping at norm 1. It validates every 2,000 steps on two batches of eight images per source, with patience ten and a 0.02 dB improvement threshold. The blind Restormer and PromptIR runs use AdamW with peak learning rates of 2×10^{-4} for the restorer, 10^{-4} for the classification head, and 10^{-5} for the CLIP encoder, with 5k warmup steps and norm-1 clipping. They validate every 5k steps on 16 batches of four images per source, with patience six and a 0.01 dB threshold; both selected checkpoints are at 100k steps.

TransWeather baseline. We train the unchanged official TransWeather architecture (Valanarasu et al., 2022) from scratch on the same 95,414-pair mixture for 100k steps, retaining its input normalization to $[-1, 1]$, targets in $[0, 1]$, and Smooth L1 plus 0.04 times the frozen ImageNet-VGG16 perceptual loss. Adam uses a peak learning rate of 2×10^{-4}, 5k warmup steps, cosine decay to 10^{-6}, zero weight decay, and norm-1 clipping. The effective batch of 16 consists of four microbatches, with 128×128 crops and seed 2026. Validation uses 64 images per source every 5k steps; the selected checkpoint is step 100k. This setup matches our data and training budget rather than the original training recipe.

AirNet baseline. We use the unchanged official AirNet architecture (Li et al., 2022), initialized from scratch on the same training mixture, with two independently augmented crops of each sampled source image. The 100k-step budget allocates 10k steps to encoder-only contrastive cross-entropy training, followed by joint L1 restoration and 0.1-weighted contrastive loss. Adam uses an encoder learning rate of 10^{-3} for 6k steps and 10^{-4} for the next 4k; the joint-stage rate decays from 10^{-4} to 10^{-6} with cosine scheduling. Training uses seed 2026, 128×128 crops, effective batch 16 with microbatches of four, a 4,096-entry contrastive queue, FP32, and norm-1 clipping. Validation covers 64 images per source every 5k steps. Training produced non-finite gradients near step 26.7k, so Table 1 reports the best saved checkpoint at 25k steps (10k encoder-only and 15k joint steps).

Metric. Patch-based validation reports RGB PSNR averaged per validation source; LOL-v2 real and synthetic scores are averaged into one low-light score. For Table 1 and the NAFNet CDD-11 experiments, checkpoints are selected by periodic validation on the benchmark evaluation images, using the same procedure for every method; the selection macro includes clean, unlike the restoration-only reporting average. Validation scores are used only for checkpoint selection; all reported results are full-image. For WeatherBench and LOL-v2, predictions are quantized to 8-bit RGB, a four-pixel border is removed, and PSNR/SSIM are computed on the MATLAB-style luminance channel with an 11×11 Gaussian SSIM window. CDD-11 reports quantized full-frame RGB metrics without border cropping. All full-image evaluations use 512-pixel tiles with 32-pixel overlap and average overlapping predictions.

B Complete CDD-11 specialist matrix

Table 5 extends Table 2 with the four single-weather conditions, all specialist cross-task scores, and the 11-condition macro.

All four specialists outperform both unified models on their own single-weather tasks. The snow and low-light rows use the final EMA checkpoints from 100,000-step retraining with a learning rate of 3×10^{-4}, gradient clipping at 0.01, and EMA decay 0.999; their backbone, training data, and L1 loss are unchanged. The rain and haze specialists use their original validation-selected checkpoints. Selecting the corresponding specialist for a single degradation and averaging the relevant specialist outputs for a mixture yields 23.00 dB over all 11 conditions and 17.63 dB over the seven mixtures. Oracle NC-NAFNet reaches 26.68 and 24.89 dB, respectively, improving these macros by 3.68 and 7.26 dB. Blind CLIP+NC-NAFNet reaches 25.85 and 24.37 dB, improving them by 2.85 and 6.74 dB

Table 5: Cross-degradation and mixed-weather restoration on CDD-11. Entries report quantized full-frame RGB PSNR (dB; higher is better) on the same 200 images per column. Gray diagonal cells mark each specialist's corresponding single-weather task. The specialist selection/averaging baseline selects the corresponding specialist for a single degradation and averages the outputs of the relevant specialists for a mixture. Gray rows denote one unified model evaluated across all conditions. Macro is the unweighted mean over all 11 columns.

Model	Rain	Haze	Snow	Low	R+H	H+S	L+H	L+R	L+S	L+H+R	L+H+S	Macro
Rain specialist	**35.69**	15.99	24.06	11.78	15.80	14.91	14.80	12.37	13.17	15.03	14.58	17.11
Haze specialist	22.98	**30.55**	22.87	11.41	20.40	22.65	12.74	13.63	12.59	14.07	13.69	17.96
Snow specialist	23.72	15.88	**36.55**	11.97	15.11	15.65	14.90	13.61	12.06	15.23	14.70	17.22
Low-light specialist	13.10	10.87	13.58	**26.76**	10.34	10.41	11.85	18.81	17.78	11.32	11.03	14.17
Specialist selection/averaging	**35.69**	**30.55**	**36.55**	**26.76**	18.56	19.30	17.08	18.44	18.07	16.14	15.81	23.00
NC-NAFNet (oracle)	32.58	28.76	32.33	25.57	**27.23**	**26.34**	**24.38**	**24.91**	**24.32**	**23.62**	**23.45**	**26.68**
CLIP+NC-NAFNet (blind)	30.90	27.65	30.28	24.97	26.66	25.84	23.89	24.12	23.93	23.24	22.93	25.85

Table 6: Cross-degradation generalization on single-weather restoration. Entries report full-image quantized Y-channel PSNR (dB; higher is better). Gray diagonal cells mark the matched specialist–test setting, and gray rows denote one unified model evaluated across all four conditions. Macro is the unweighted mean over rain, haze, snow, and low light.

Training model	Rain	Haze	Snow	Low light	Macro
Rain specialist	33.48	14.06	23.07	11.45	20.52
Haze specialist	23.27	19.06	20.95	13.26	19.14
Snow specialist	27.63	13.56	27.66	11.42	20.07
Low-light specialist	13.13	10.54	14.80	23.16	15.41
NC-NAFNet (oracle)	32.90	20.64	26.23	**23.74**	25.88
CLIP+NC-NAFNet (blind)	**35.04**	**23.62**	**28.07**	23.14	**27.46**

while accepting only the degraded image. One conditioned model therefore covers all 11 conditions, while specialists remain stronger on their own single-weather tasks.

C Single-weather specialist comparison

We train four independent plain NAFNet models, each seeing only rain, haze, snow, or low-light data, and evaluate every checkpoint on all four single-weather conditions. Table 6 reports full-image results on the same 800 single-weather images used by the corresponding columns of Table 1.

The specialist matrix shows strong task dependence rather than perfect diagonal dominance. The rain, snow, and low-light specialists achieve their strongest scores on their matched tasks, whereas the haze specialist scores higher on rain than on haze. Their single-weather macros range from 15.41 to 20.52 dB. One oracle NC-NAFNet raises the four-task macro to 25.88 dB, and the blind model reaches 27.46 dB. The blind model exceeds the corresponding specialist on rain, haze, and snow and trails the low-light specialist by only 0.02 dB. One conditioned model thus avoids most of the off-diagonal degradation of task-specific restorers. Several specialist runs, particularly rain and snow, became unstable after their best checkpoint, so we report validation-selected checkpoints. Because specialists and the unified model see different training data, this matrix measures task coverage rather than isolating the NC operator.

D Condition-encoding implementation

We retrain two encoding variants from scratch to distinguish degradation-specific control from the presence of additional modulation parameters. The constant-condition variant feeds the scalar zero to the Fourier encoder for every image, yielding the same 64-dimensional code $[\mathbf{0}_{32}, \mathbf{1}_{32}]$ for both single and mixed weather; mixture cardinality is also withheld. Its independent block-wise MLPs remain

Table 7: Same-checkpoint condition intervention on the 800 full-resolution single-weather images of Table 1. Task columns and Restore-4 are Y-channel PSNR (dB), and SSIM is the four-task macro. Every row uses the same 100k-step joint weights; only the condition input changes. Hard and ground-truth one-hot controls are distribution shifts because training uses soft CLIP distributions, so they are not oracle upper bounds.

Condition input	Rain	Haze	Snow	Low light	Restore-4	SSIM
Predicted soft	**34.34**	**22.94**	**27.35**	**23.97**	**27.15**	**0.851**
Predicted hard	31.70	16.22	24.32	9.27	20.38	0.761
Ground-truth one-hot	31.70	16.22	24.32	8.67	20.23	0.758
Uniform	22.98	13.27	22.06	12.84	17.79	0.697
Cyclically wrong	21.97	12.53	6.19	11.80	13.12	0.577

trainable but produce image-independent modulation parameters. The learned-embedding variant replaces the fixed class vectors with a trainable table $E \in \mathbb{R}^{5\times 64}$ for clear, rain, haze, snow, and low light. Entries are initialized independently from $\mathcal{N}(0, 0.5)$ using a separate random generator with seed zero. Active rows are summed without normalization, preserving the multi-hot composition rule. The table adds 320 trainable parameters and is optimized jointly with the backbone and modulation MLPs using only the same L1 restoration loss; no predictor or classification objective is used.

Both variants follow the CDD-11 oracle NC-NAFNet recipe: NAFNet-32, seed 2026, batch size 16, 128×128 training crops, Adam with zero weight decay, and a cosine learning rate from 10^{-3} to 10^{-6} over at most 100,000 steps. Backbone and adapter initialization are matched. Evaluation occurs every 2,000 steps on two batches of eight images per source, with early stopping after ten checks without a PSNR improvement exceeding 0.02 dB. Neither EMA nor gradient clipping is used. The constant-condition run completes 100,000 steps and selects its 86,000-step checkpoint; learned embeddings stop at 46,000 steps and select the 26,000-step checkpoint. The full-NC reference completes 100,000 steps and selects the 98,000-step checkpoint. All rows use the validation protocol described in Appendix A.2.

E Fixed-checkpoint condition interventions

To test whether the blind restorer uses its predicted condition, we evaluate one fixed 100k-step blind NAFNet checkpoint with five controller inputs. This model is trained separately from the Table 1 model, on the same weather mixture without the rain–fog pairs, so its scores differ from Table 1. The inputs are: the predicted soft distribution $q(x)$; a hard one-hot vector at $\arg\max q(x)$; the ground-truth one-hot class; a uniform distribution; and a deliberately incorrect one-hot class obtained by cyclically shifting the ground-truth class. All variants use the same weights and the same 800 full-resolution single-weather images as Table 1. Because the model is trained with soft predicted distributions, hard and ground-truth one-hot inputs are inference-time distribution shifts rather than oracle upper bounds.

Table 7 evaluates the same trained joint model while changing only the condition delivered to its layer-wise adapters. The predicted soft condition exceeds the uniform condition by 9.36 dB and the cyclically wrong condition by 14.03 dB on Restore-4. A wrong condition is especially destructive for snow, reducing PSNR from 27.35 to 6.19 dB. The restoration network therefore relies on its conditioning pathway.

The soft vector outperforms its hard argmax because the joint objective trains the adapters on the full probability vector, whereas one-hot vectors lie outside that training distribution. Together with the large uniform and wrong-condition drops, this suggests that $q(x)$ acts as a continuous image-dependent routing signal that may encode ambiguity, severity, or domain information beyond its displayed class label.

F Full-image capacity control

The NAFNet-34 control uses the same weather mixture and full-image evaluation protocol as Table 1. It has 32.90M parameters, compared with 32.002M for NC-NAFNet-32. Its Y-channel PSNR/SSIM

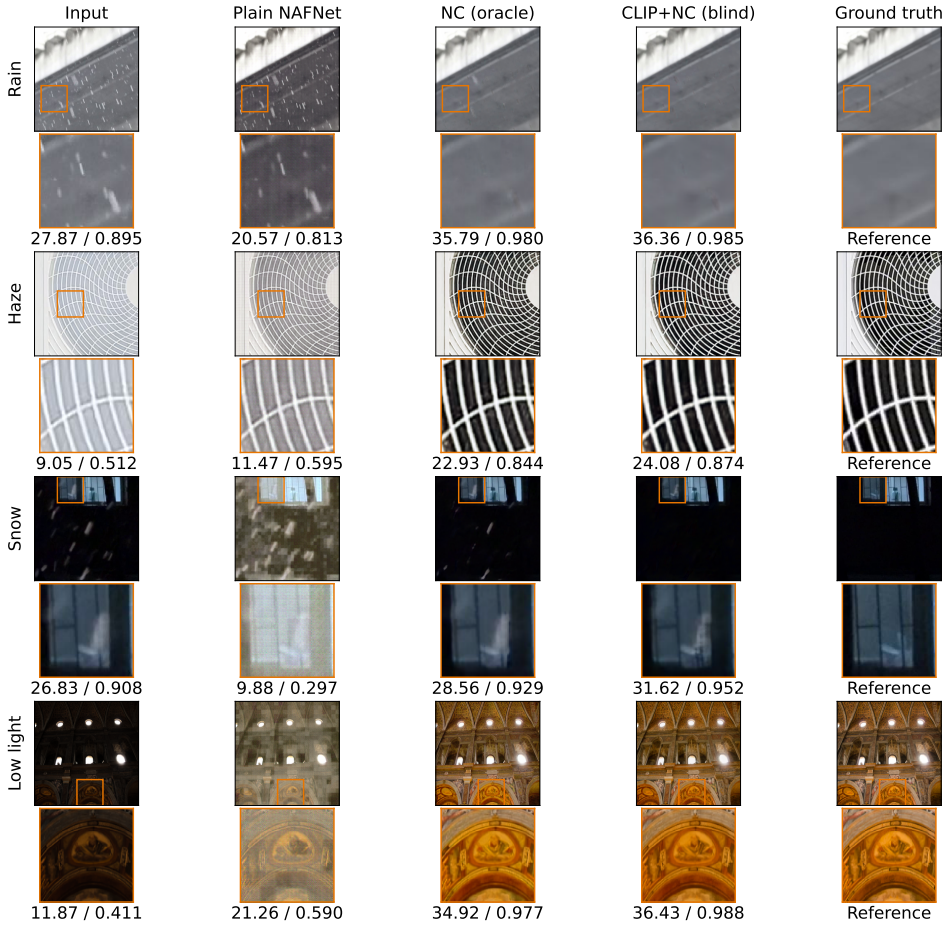

Figure 3: Complete NAFNet qualitative comparison with matched detail regions. NC uses oracle labels; CLIP+NC predicts its condition from the image. Scores are full-image PSNR-Y/SSIM-Y. The same rendering scale and region are used across methods.

is 27.21/0.801 on rain, 16.91/0.580 on haze, 22.10/0.759 on snow, and 17.90/0.618 on low light, averaging 21.03/0.690.

G Additional qualitative comparisons

Figure 3 shows all four single-weather examples: WeatherBench rain, haze, and snow, and LOL-v2 low light. Figure 2 shows the rain, snow, and low-light rows. For each condition, we consider the eight test images with the largest PSNR-Y gain of both NC variants over plain NAFNet and choose one with clearly visible differences; these examples illustrate favorable cases, and Table 1 reports average performance. All methods use the same saved input and target. Each orange box covers a quarter of the image width and height, placed where the error of plain NAFNet most exceeds the larger error of the two NC variants; the enlarged regions use identical coordinates across methods. No restoration output is enhanced. Reported PSNR-Y/SSIM-Y values are quantized full-image metrics with a four-pixel border crop, not metrics on the enlarged regions.