Continuous Expert Assembly: Instance-Conditioned Low-Rank Residuals for All-in-One Image Restoration

Haisen He[1,3] **Xiangyu Zou**[2,3] **SongLin Dong**[3]
Heng Li[3] **Yihong Gong**[4] **Zhiheng Ma**[3,5*]

[1]Southern University of Science and Technology
[2]Shenzhen University
[3]Faculty of Computility Microelectronics, Shenzhen University of Advanced Technology
[4]Xi'an Jiaotong University
[5]Primotion

Abstract

Real-world image degradation is often unknown, spatially non-uniform, and compositional, requiring all-in-one restoration models to handle diverse corruption patterns without test-time degradation labels. Existing approaches condition restoration features with prompts or descriptors, or allocate capacity through predefined experts. We propose **Continuous Expert Assembly** (CEA), an instance-conditioned low-rank parameterization framework with token-dependent residual assembly. A lightweight **Cross-Attention Hyper-Adapter** probes intermediate spatial features to generate low-rank routing bases and residual directions. The generated factors are shared across spatial tokens within each layer and target projection; token features induce dense signed coefficients and token-dependent residual responses. A linear-attention-style algebraic view exposes the coefficient matrix and efficient factorized computation. Experiments on AIO-3, AIO-5, and CDD-11 show moderate gains on standard AiO benchmarks and larger gains on compositional degradations, with favorable parameter, FLOP, and runtime efficiency. Additional evaluation on real-captured WeatherBench images supports the effectiveness of CEA beyond the synthetic benchmark settings.

1 Introduction

Real-world photographs rarely suffer from one form of degradation in isolation. Low light, haze, rain, motion blur, and sensor noise often co-occur in a single image, are spatially non-uniform, and are unknown at inference time. All-in-one (AiO) image restoration [28, 38, 19] therefore requires a single model to handle multiple degradation types and their compositions without test-time knowledge of the corruption. While Transformer-based restorers [30, 48, 53, 7] have substantially advanced single-degradation performance through long-range and multi-scale modeling, AiO introduces a distinct difficulty: one set of weights must serve heterogeneous, spatially non-uniform corruptions that often appear in composition.

Existing AiO restorers adapt a shared backbone through several forms of conditioning. Feature-space approaches use degradation representations, task priors, learned prompts, language instructions, or degradation descriptors [28, 31, 55, 38, 9, 19]. Other approaches route among expert branches or learned adapters, or generate input-conditioned parameters [43, 14, 27, 39, 17, 51, 42, 1, 2]. These mechanisms differ in the conditioning representation and the object being generated or combined. We

*Corresponding author. Email: mazhiheng@suat-sz.edu.cn.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

investigate a compact alternative that generates a low-rank residual component space from spatially probed features and uses it through dense signed assembly.

We argue that AiO restoration requires a more flexible form of parameter adaptation. Instead of selecting from a static set of experts or modulating the network with a single global condition, the model should dynamically assemble restoration parameters conditioned on the input image and apply them at the token level. This formulation is particularly suitable for compositional degradations, where different spatial tokens can receive different residual updates while sharing a compact and efficient input-conditioned parameter space.

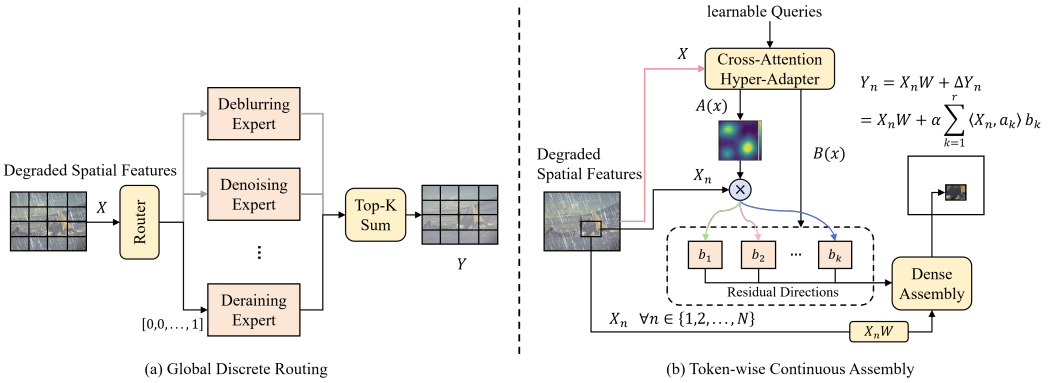


Figure 1: **Conceptual comparison of adaptation paradigms for All-in-One Image Restoration.** (a) Representative global or discrete expert adaptation. A compact image-level signal or sparse router selects or modulates a fixed set of parameters, which can become coarse when degradations vary spatially or appear in composition. (b) Continuous Expert Assembly (CEA). A lightweight hyper-adapter probes spatial features and generates instance-conditioned low-rank routing bases $A(x)$ and residual directions $B(x)$. Each spatial token uses dense signed dot-product affinities to assemble its own residual update ΔY_n, enabling instance-conditioned low-rank residual assembly without discrete expert selection.

To this end, we propose **Continuous Expert Assembly** (CEA), an instance-conditioned low-rank parameterization framework with token-dependent residual assembly. CEA synthesizes routing bases and residual directions from the current degraded image. These factors define a shared residual operator within each layer and target projection, while each spatial token induces its own signed coefficients and residual response. The generated component space supports continuous residual assembly under the current degradation instance.

CEA is implemented with a lightweight **Cross-Attention Hyper-Adapter**. Rather than pooling intermediate features into a single global descriptor, the adapter uses learnable queries to probe condensed spatial contexts and decode them into routing bases and residual directions. The assembled residual updates are injected into the attention projections of the restoration backbone, allowing spatial feature interactions to adapt to the degradation structure of each image.

The assembly rule admits a linear-attention-style algebraic view: spatial tokens serve as queries, and the generated routing bases and residual directions play roles analogous to keys and values. This identity exposes the token-to-component coefficient matrix and the two low-rank products used to evaluate the update. It explains how each token combines the generated residual directions through dense signed coefficients.

We evaluate CEA on AIO-3, AIO-5, and CDD-11, and additionally on real-captured adverse-weather images from WeatherBench [16]. The gains on standard AiO benchmarks are moderate but consistent and efficiency-preserving, while the largest improvements occur in spatially varying and compositional settings. WeatherBench reproduces the improvements at both model scales and across PSNR, SSIM, and LPIPS. Together, these results support the restoration quality and accuracy–efficiency trade-off of CEA.

Our contributions are summarized as follows:

- We propose **Continuous Expert Assembly**, a continuous expert-assembly mechanism that synthesizes an image-conditioned low-rank residual component space and applies it through token-dependent dense signed assembly.
- We design a lightweight **Cross-Attention Hyper-Adapter** that uses rank-oriented queries to probe intermediate spatial features and generate target-specific low-rank routing bases and residual directions.
- We provide a linear-attention-style algebraic view that exposes the token-to-component coefficients, signed residual combination, and efficient factorized computation.
- Experiments on AIO-3, AIO-5, CDD-11, and WeatherBench demonstrate consistent improvements with a favorable accuracy–efficiency trade-off, with the largest gains on spatially varying and compositional degradations.

2 Related Work

Restoration backbones. Image restoration has evolved from discriminative CNN restorers such as DnCNN [56], MPRNet [52], HINet [4], and NAFNet [5], to architectures with stronger global interaction, including IPT [3] and MAXIM [46]. Transformer-based restorers have further improved long-range dependency modeling and high-resolution restoration through efficient attention and hierarchical designs, as in SwinIR [30], Uformer [48], Restormer [53], and CAT [7]. These backbones provide strong restoration priors, but they are typically optimized for individual degradations. AiO restoration requires a single backbone to adapt its computation under unknown, spatially varying, and compositional degradations.

All-in-one restoration and conditioning. AiO restoration targets a single model that handles multiple degradation types under unknown corruption. AirNet learns degradation representations to guide a shared restorer [28]; TAPE introduces task-agnostic prior embeddings for cross-task restoration [31]; IDR formulates multi-degradation restoration through reusable degradation ingredients [55]; and TransWeather addresses unified adverse-weather restoration [24]. A related line conditions restoration with learned prompts, visual prompts, language instructions, or explicit degradation descriptors, including PromptIR [38], ProRes [34], PIP [29], InstructIR [9], and OneRestore [19]. PromptIR predicts image-level Softmax weights from globally pooled features to combine learned prompt components. The resulting spatial prompt is concatenated with restoration features and processed by a Transformer block with shared weights. CEA's Cross-Attention Hyper-Adapter probes spatial features to generate image-conditioned low-rank residuals for the Q/K projection parameters.

Conditional computation and expert allocation. Mixture-of-Experts allocates capacity by routing inputs to expert modules [43]. Scalable sparse MoE systems such as Switch Transformer and GShard activate only a subset of experts and typically introduce routing decisions, capacity handling, or load-balancing objectives [14, 27]. Vision MoE extends this idea to visual recognition [41]. In image restoration, AdaptIR uses heterogeneous expert branches for parameter-efficient adaptation [17], while MoCE-IR introduces complexity experts for all-in-one restoration [51]. Soft MoE replaces hard Top-K assignment with dense differentiable mixing over a fixed expert set [39]. CEA shares the goal of avoiding hard expert assignment, but differs in the object being mixed. Sparse and soft MoE methods mix outputs or representations produced by a sample-independent expert bank. In contrast, CEA first generates an input-conditioned low-rank residual subspace and then lets each spatial token assemble an additive update over this generated subspace. Thus, the rank-wise components are not fixed expert networks, and the routing weights are signed dot-product affinities rather than normalized probabilities over a static expert pool.

Low-rank and dynamic parameterization. Parameter-efficient adaptation methods, including Adapters [21], LoRA [22], AdaLoRA [57], and DoRA [32], show that low-rank updates can efficiently adapt large backbones. Hypernetworks and dynamic filters generate sample-conditioned parameters for adaptive computation [20, 10, 50, 6, 35]. LoRA-IR and ABAIR maintain learned banks of degradation- or task-specific low-rank adapters and use a degradation estimator to select or blend them [1, 42]. HAIR compresses the input into a global information vector and uses Softmax coefficients to generate full block parameters from a learned weight box [2]. CEA uses rank-oriented cross-attention queries to probe intermediate spatial features and directly generate target-specific low-rank factors at each equipped layer. Each token combines the generated residual directions through dense signed coefficients. The spatial-generator control and factorial study evaluate factor

generation and assembly in Appendix D.3 and Table 5. Quantitative comparisons with PromptIR and Res-HAIR are included in Tables 6 and 7.

Linear-attention-style computation. Efficient and linear attention methods reduce the quadratic cost of standard attention by exploiting associative matrix products or kernelized feature maps [44, 25, 8]. We observe that the CEA residual has the same algebraic form as dot-product attention without softmax, $QK^\top V$, where the keys and values are instance-conditioned routing bases and residual directions generated by the hyper-adapter. This view provides an efficient implementation of dense token-to-rank assembly.

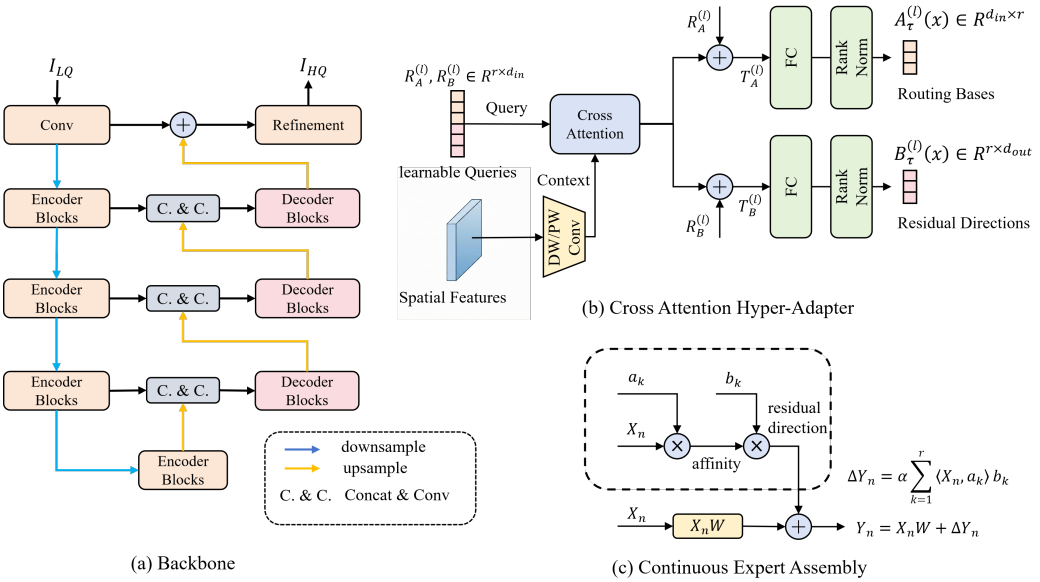


Figure 2: **Overall architecture of the proposed method.** (a) **Backbone.** We build upon a U-shaped Transformer backbone. To dynamically adapt to unknown degradations, our proposed modules are integrated into the decoder blocks to modulate the intermediate spatial features. (b) **Cross-Attention Hyper-Adapter.** Spatial features are condensed via depthwise-pointwise (DW/PW) convolutions to provide local context. A compact set of learnable queries $(R_A^{(l)}, R_B^{(l)})$ probes this context via cross-attention. The contextualized queries are decoded into instance-conditioned routing bases $A_\tau^{(l)}(x)$ and residual directions $B_\tau^{(l)}(x)$, followed by RankNorm for stable assembly. (c) **Continuous Expert Assembly.** For each spatial token X_n, CEA computes dense signed dot-product affinities $\langle X_n, a_k \rangle$ with the generated routing bases. These affinities weight the corresponding residual directions b_k to assemble a token-specific residual update ΔY_n, which is added to the base projection X_nW to produce the adapted feature Y_n.

3 Method

3.1 Problem Formulation & The Mixture-of-Experts Baseline

In All-in-One (AiO) image restoration, we are given a degraded observation $x \in \mathbb{R}^{H\times W\times 3}$ generated by an unknown degradation process $x = \mathcal{D}_t(y;\xi)$, where t denotes a latent degradation type or composition and ξ represents its severity and nuisance parameters. Our primary objective is to learn a unified model $F(\cdot)$ that reconstructs the clean image $\hat{y} = F(x) \approx y$ without prior knowledge of the explicit degradation parameters (t, ξ).

To address the severe heterogeneity of diverse degradations within a single model, a representative strategy is **expert allocation** via conditional computation, such as (sparse) Mixture-of-Experts (MoE) and MoE-style expert branches explored in recent restoration models [17, 51]. Formally, at decoder block l, we flatten the intermediate feature map of spatial size $H_l \times W_l$ into a token sequence $X^{(l)} \in \mathbb{R}^{N\times d_{\text{in}}}$ with $N = H_lW_l$ spatial tokens, and use $X_n^{(l)} \in \mathbb{R}^{d_{\text{in}}}$ to denote the n-th token (we

omit the batch index for clarity). For the n-th token $X_n^{(l)}$, a common *MoE-style expert augmentation* adds expert-dependent residual branches on top of a shared projection $W \in \mathbb{R}^{d_{\text{in}} \times d_{\text{out}}}$, using a sparse router over a *static* expert set $\{\mathcal{E}_i\}_{i=1}^{E}$:

$$Y_n^{(l)} = X_n^{(l)} W + \sum_{i=1}^{E} g_i(X_n^{(l)}) \mathcal{E}_i(X_n^{(l)}), \tag{1}$$

where $g_i(\cdot)$ is a discrete routing mechanism (e.g., Top-K Softmax) that assigns the token to specific experts.

Limitations in the restoration setting. While sparse MoE is a natural way to allocate conditional capacity, it introduces additional design considerations for AiO restoration. First, Top-K routing yields discontinuous expert assignments; in practice this can make optimization more sensitive and often requires capacity handling. Second, the expert parameters are typically shared across samples, so adaptation is limited to mixing a discrete set of functions even when degradations vary continuously in type and severity. These observations motivate a dense, continuous, and instance-conditioned alternative.

3.2 Continuous Expert Assembly: Generated Low-Rank Routing

We propose **Continuous Expert Assembly** (CEA), which relaxes sparse expert allocation into continuous low-rank residual assembly. Instead of selecting from a fixed pool of expert networks, CEA synthesizes rank-wise routing bases and residual directions from the current degraded image. Each spatial token then assembles an additive update over this generated subspace using dense signed inner products rather than sparse or probability-normalized routing weights.

We use rank-wise component for the pair $(a_{\tau,k}^{(l)}(x), b_{\tau,k}^{(l)}(x))$: the routing basis $a_{\tau,k}^{(l)}(x)$ determines a token's signed coefficient, and the residual direction $b_{\tau,k}^{(l)}(x)$ specifies its output-space contribution. For token $X_n^{(l)}$ and target projection τ, the residual is

$$\Delta Y_{\tau,n}^{(l)} = \alpha \sum_{k=1}^{r} \underbrace{\langle X_n^{(l)}, a_{\tau,k}^{(l)}(x) \rangle}_{\substack{\text{signed} \\ \text{affinity}}} \underbrace{b_{\tau,k}^{(l)}(x)}_{\substack{\text{residual} \\ \text{direction}}}, \tag{2}$$

where $a_{\tau,k}^{(l)}(x) \in \mathbb{R}^{d_{\text{in}}}$ is the k-th *routing basis* (key) for target τ, $b_{\tau,k}^{(l)}(x) \in \mathbb{R}^{d_{\text{out}}}$ is the corresponding instance-conditioned *residual direction*, and $\alpha = 1/r$ is a stabilizing scaling factor.

Crucially, when written for the full token sequence, Eq. (2) has the same *algebraic* form as dot-product attention without softmax, $QK^\top V$. Since the effective keys and values correspond to the generated rank-r factors, the update can be computed by two low-rank matrix multiplications, scaling linearly with the number of tokens N and the rank r. Let $A_\tau^{(l)}(x) \in \mathbb{R}^{d_{\text{in}} \times r}$ and $B_\tau^{(l)}(x) \in \mathbb{R}^{r \times d_{\text{out}}}$ be the matrices formed by columns $a_{\tau,k}^{(l)}(x)$ and rows $b_{\tau,k}^{(l)}(x)$, respectively. The residual assembly for the full token sequence is:

$$\begin{aligned} G_\tau^{(l)}(x) &\triangleq X^{(l)} A_\tau^{(l)}(x) \in \mathbb{R}^{N \times r}, \\ \Delta Y_\tau^{(l)} &= \alpha\, G_\tau^{(l)}(x) B_\tau^{(l)}(x) \\ &= \alpha\, X^{(l)} A_\tau^{(l)}(x) B_\tau^{(l)}(x) \\ &= \alpha\, \mathrm{Attn}_{\text{lin}}\Big(X^{(l)}, A_\tau^{(l)}(x)^\top, B_\tau^{(l)}(x)\Big). \end{aligned} \tag{3}$$

Here $\mathrm{Attn}_{\text{lin}}(Q, K, V) = QK^\top V$, with $Q = X^{(l)}$, $K = A_\tau^{(l)}(x)^\top$, and $V = B_\tau^{(l)}(x)$. This algebraic identity exposes $G_\tau^{(l)}(x)$ as the token-to-component coefficient matrix. Each row specifies how a token combines the residual directions in $B_\tau^{(l)}(x)$. The coefficients are dense, signed, and unnormalized, allowing each component to contribute in either direction.

For a fixed image, layer, and target projection, $A_\tau^{(l)}(x)$ and $B_\tau^{(l)}(x)$ are shared across all spatial tokens. Equivalently, they define one image-conditioned low-rank residual operator,

$$\Delta W_\tau^{(l)}(x) = \alpha A_\tau^{(l)}(x) B_\tau^{(l)}(x), \qquad \Delta Y_{\tau,n}^{(l)} = X_n^{(l)} \Delta W_\tau^{(l)}(x). \tag{4}$$

Token dependence arises through $G_{\tau,n}^{(l)}(x) = X_n^{(l)} A_\tau^{(l)}(x)$, which gives token-dependent coefficients and residual responses. The factorized evaluation $(XA)B$ costs $O(Nr(d_{\text{in}} + d_{\text{out}}))$, compared with $O(Nd_{\text{in}}d_{\text{out}})$ for applying the equivalent dense-shaped update.

3.3 Cross-Attention Hyper-Adapter

The low-rank factors in CEA should be conditioned on the current degradation while remaining efficient to generate. A simple option is to pool the feature map into a global descriptor and predict $A(x)$ and $B(x)$ with an MLP. However, such global conditioning can discard spatial evidence that is important for localized and compositional degradations. We therefore use spatial query probing rather than global factor prediction.

To this end, we introduce a lightweight **Cross-Attention Hyper-Adapter** (illustrated in Figure 2). Instead of relying on a single global descriptor that can compress spatially heterogeneous degradation cues, we use a compact set of learnable queries as spatial probes. The queries interact with condensed spatial features via cross-attention to aggregate local degradation context for generating routing bases and residual directions.

Given the intermediate spatial features $X^{(l)}$ at the l-th block, we first condense their spatial resolution via depthwise-pointwise convolutions to yield $\hat{X}^{(l)}$. This operation efficiently reduces the computational overhead of the subsequent attention mechanism while encoding local structural patterns. The learnable queries then interact with these condensed features via cross-attention to aggregate local degradation contexts:

$$T_\phi^{(l)}(x) = R_\phi^{(l)} + \text{CrossAttn}\Big(R_\phi^{(l)}, \hat{X}^{(l)}, \hat{X}^{(l)}\Big), \quad \phi \in \{A, B\}. \tag{5}$$

Finally, lightweight fully-connected (FC) heads decode these contextualized query representations into a set of *routing bases* and *residual directions* for continuous low-rank residual assembly. We share $T_A^{(l)}(x)$ and $T_B^{(l)}(x)$ across targets, while using *target-specific* heads to produce the corresponding routing bases and residual directions (i.e., matrices $A_\tau^{(l)}(x)$ and $B_\tau^{(l)}(x)$) for each modulated projection (e.g., Q and K):

$$A_\tau^{(l)}(x) = \Big(\text{FC}_{A,\tau}^{(l)}\big(T_A^{(l)}(x)\big)\Big)^\top, \qquad B_\tau^{(l)}(x) = \text{FC}_{B,\tau}^{(l)}\big(T_B^{(l)}(x)\big), \qquad \tau \in \{Q, K\}. \tag{6}$$

3.4 Stable Parameter Assembly and Injection

Resolving Scale Ambiguity (RankNorm). Dynamic factorization is inherently scale-ambiguous, as $AB = (cA)(B/c)$. To prevent norm drift and destabilization during training, we apply Rank-wise Factor Normalization (RankNorm) to the generated factors. For each target τ and rank index k,

$$\big(A_\tau^{(l)\prime}(x)\big)_{:,k} = \frac{\big(A_\tau^{(l)}(x)\big)_{:,k}}{\|\big(A_\tau^{(l)}(x)\big)_{:,k}\|_2 + \epsilon}, \qquad \big(B_\tau^{(l)\prime}(x)\big)_{k,:} = \frac{\big(B_\tau^{(l)}(x)\big)_{k,:}}{\|\big(B_\tau^{(l)}(x)\big)_{k,:}\|_2 + \epsilon}. \tag{7}$$

Crucially, this normalization is applied *per instance* to prevent cross-sample coupling, ensuring that each dynamically synthesized rank component remains well-conditioned. In implementation, the residual is computed as two low-rank products, $\alpha(X^{(l)} A_\tau^{(l)\prime}(x)) B_\tau^{(l)\prime}(x)$, rather than forming and applying a dense dynamic projection. This reduces the dominant computation from a full $Nd_{\text{in}}d_{\text{out}}$ dynamic projection to $Nd_{\text{in}}r + Nrd_{\text{out}}$, with $r \ll d_{\text{in}}, d_{\text{out}}$.

Injection Targets. The normalized factors are used to construct the residual $\Delta Y^{(l)}$. In our default architecture, we inject this dynamic residual into the Attention Query (Q) and Key (K) projections:

$$Q = X^{(l)} W_Q + \Delta Y_Q^{(l)}, \qquad K = X^{(l)} W_K + \Delta Y_K^{(l)}. \tag{8}$$

In particular, $\Delta Y_Q^{(l)}$ and $\Delta Y_K^{(l)}$ are assembled by Eq. (3) using their corresponding target-specific factors $(A_Q^{(l)}(x), B_Q^{(l)}(x))$ and $(A_K^{(l)}(x), B_K^{(l)}(x))$, respectively. These factors define image-conditioned residual operators shared across tokens, while each token's signed coefficients induce its own residual response. By modulating Q and K, CEA adjusts the backbone's attention patterns under the current degradation instance.

Training objective. We train the network with a reconstruction objective

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{rec}} + \lambda_f \mathcal{L}_{\text{fft}}.$$

Here $\mathcal{L}_{\text{rec}}$ is the pixel-domain ℓ_1 loss. Following recent frequency-aware restoration practice and the training recipe of MoCE-IR [51], we also use a Fourier-domain reconstruction loss $\mathcal{L}_{\text{fft}}$ as a shared regularizer. This term is not part of the proposed CEA module; it is used for all matched variants in the controlled analysis.

4 Experiments

4.1 Experiment Settings

Datasets and protocols. We evaluate CEA under the standard AIO-3 and AIO-5 protocols [38, 55] and the CDD-11 compositional protocol [19]. We additionally evaluate on the paired real-captured WeatherBench dataset [16], whose test set contains 600 image pairs, with 200 each for haze, rain, and snow. Results are reported in Section 4.4.

Implementation details. We adopt an asymmetric U-shaped Transformer backbone following prior all-in-one restorers. By default, CEA modules are inserted into decoder blocks and modulate the attention Q/K projections. We optimize all CEA models and controlled variants using Adam [26] with a reconstruction objective consisting of a spatial ℓ_1 loss and a Fourier-domain reconstruction regularizer, following recent all-in-one restoration practice [51] and frequency-aware restoration objectives [23]. This Fourier term is shared across matched variants and is not a label-based auxiliary loss. No degradation labels, prompts, or auxiliary classification/contrastive objectives are used.

4.2 All-in-One Restoration on Standard AIO Benchmarks

Table 1 summarizes the average performance on AIO-3 and AIO-5; full per-task comparisons are provided in Appendix C.1 and Appendix C.2.

AIO-3. In the lightweight regime, CEA-IR-S improves the average score from 32.57/0.916 to 32.89/0.918 compared with MoCE-IR-S while using fewer parameters. The gains are most visible on SOTS dehazing and Rain100L deraining, which involve spatially varying weather degradations. In the higher-capacity setting, CEA-IR improves the average score from 32.73/0.917 to 33.07/0.922. The denoising gains are smaller but consistent across noise levels, indicating that input-conditioned assembly does not sacrifice performance on more homogeneous degradations.

AIO-5. Compared with MoCE-IR-S, CEA-IR-S improves the average score from 30.08/0.913 to 30.80/0.919. The gain is mainly driven by deraining, deblurring, low-light enhancement, and a smaller improvement on denoising. On dehazing, CEA-IR-S obtains slightly lower PSNR than MoCE-IR-S but comparable SSIM, indicating that the average improvement does not rely on uniformly dominating every task. For heavy models, CEA-IR improves the average score from 30.58/0.919 to 31.13/0.922. The largest gains appear on dehazing and deraining, while the model remains competitive but slightly lower on GoPro deblurring compared with MoCE-IR.

4.3 Compositional Degradations on CDD-11

Compositional degradations require generalization beyond individual degradation operators to their interactions and spatial overlap. Table 2 reports group-level results on the CDD-11 protocol [19]. Single, Double, and Triple denote category-wise averages over the corresponding degradation groups, and Overall is the category-wise average over all 11 degradation categories following prior work.

Table 1: **Main all-in-one restoration results.** We report average PSNR/SSIM. Full per-task comparisons are provided in Appendix C.1 and Appendix C.2.

Benchmark	Scale	Method	Params.	Avg. PSNR	Avg. SSIM	Gain
AIO-3	Light	MoCE-IR-S	11M	32.57	.916	–
	Light	**CEA-IR-S**	**10M**	**32.89**	**.918**	**+0.32/+.002**
AIO-3	Heavy	MoCE-IR	25M	32.73	.917	–
	Heavy	**CEA-IR**	**22M**	**33.07**	**.922**	**+0.34/+.005**
AIO-5	Light	MoCE-IR-S	11M	30.08	.913	–
	Light	**CEA-IR-S**	**10M**	**30.80**	**.919**	**+0.72/+.006**
AIO-5	Heavy	MoCE-IR	25M	30.58	.919	–
	Heavy	**CEA-IR**	**22M**	**31.13**	**.922**	**+0.55/+.003**

Table 2: **CDD-11 compositional restoration.** We report group-level PSNR/SSIM on single, double, and triple degradation groups. Full per-category scores are provided in Appendix C.3.

Method	Single	Double	Triple	Overall
OneRestore	31.68/.938	27.35/.866	24.84/.790	28.47/.878
MoCE-IR-S	32.54/.941	27.73/.869	25.40/.790	29.05/.881
CEA-IR-S	**33.36/.943**	**28.51/.875**	**25.90/.798**	**29.79/.886**

CEA-IR-S achieves the best overall average of 29.79/0.886 using 10M parameters, improving over OneRestore and MoCE-IR-S by 1.32 dB and 0.74 dB, respectively. The gains are consistent across single, double, and triple degradation groups, with especially clear improvements on haze-related compositions in the full per-category table. These results match the motivation of CEA: compositional degradations often involve both global visibility changes and localized degradation patterns, making it useful to generate an input-conditioned residual subspace while allowing different spatial regions to assemble different updates. We therefore analyze the roles of factor generation and spatial assembly explicitly in Sec. 4.7.

4.4 Restoration on Real-Captured WeatherBench Images

Table 3 reports results on WeatherBench. At each model scale, CEA and the corresponding MoCE baseline use the same training and evaluation settings. CEA improves PSNR, SSIM, and LPIPS in all three weather categories, with average PSNR gains of 0.600 dB for CEA-IR-S and 0.430 dB for CEA-IR. The largest PSNR gains occur on snow.

Table 3: **Real-captured adverse-weather restoration on WeatherBench.** Each entry is PSNR / SSIM / LPIPS. The test set contains 200 pairs per weather category. Higher PSNR/SSIM and lower LPIPS are better.

Scale	Method	Haze	Rain	Snow	Average
Light	MoCE-IR-S	23.922/.748/.359	36.868/.959/.184	29.389/.847/.189	30.060/.851/.244
Light	CEA-IR-S	24.058/.759/.349	37.464/.962/.174	30.459/.882/.171	30.660/.867/.231
Heavy	MoCE-IR	24.388/.756/.347	37.232/.960/.177	29.938/.862/.177	30.519/.859/.234
Heavy	CEA-IR	24.652/.764/.342	37.517/.962/.172	30.679/.886/.167	30.949/.870/.227

4.5 Qualitative Evaluation

Representative qualitative comparisons are provided in Appendix F. Across individual and compositional degradations, CEA reduces localized residual artifacts such as rain streaks, haze boundaries, and low-light texture distortions, while preserving fine structures. These visual trends are consistent with the quantitative gains in Tables 1–2.

4.6 Efficiency

Table 4: **Computational demands.** FLOPs are computed on a 224×224 input. Runtime reports mean $\pm$ sample standard deviation (ms/image) across five complete-pass mean runtimes on an RTX 4090 (Appendix B.3). Runtime measurement is separated from FLOPs/memory profiling.

Method	Params.	Memory	FLOPs	Runtime (ms) $\downarrow$
AirNet	8.93M	4829M	238G	41.846 ± 0.098
PromptIR	35.59M	9830M	132G	41.179 ± 0.071
MoCE-IR	25.35M	5887M	80.59G	40.875 ± 0.055
MoCE-IR-S	11.47M	4228M	36.93G	38.177 ± 0.058
CEA-IR-S (Ours)	9.71M	**3517M**	**28.03G**	$\mathbf{23.338 \pm 0.064}$

Table 4 reports the computational demands. CEA-IR-S uses 9.71M parameters and 28.03 GFLOPs, with a mean runtime of 23.338 ms per image compared with 38.177 ms for MoCE-IR-S. Its low-rank residual is evaluated using two matrix multiplications. These results show a favorable accuracy–efficiency trade-off among the compared all-in-one restorers. Appendix B.3 describes the runtime measurement protocol.

4.7 Controlled Analysis

Disentangling factor source and routing rule. Table 5 tests whether the gain comes from the specific mechanism of CEA rather than from simply adding low-rank residual capacity. Static variants replace the input-conditioned hyper-adapter with per-block learnable factors A, B that are independent of the input image. Dynamic variants generate $A(x), B(x)$ with the CEA hyper-adapter. Top-2 variants replace CEA's dense assembly with sparse softmax routing over the top two rank components, whereas Dense uses CEA's softmax-free signed dot-product assembly in Eq. (3). All variants share the same lightweight backbone, Q/K injection, rank $r = 8$, and reconstruction losses.

Table 5: **Factorial ablation on CDD-11.** All variants use the same lightweight backbone, Q/K injection, rank $r = 8$, and reconstruction losses. "Static" learns input-independent low-rank factors, while "Dynamic" generates factors with the CEA hyper-adapter. "Top-2" applies sparse softmax routing over the top two rank components; "Dense" uses CEA's softmax-free signed dot-product assembly.

Factor source	Routing rule	Single	Double	Triple	Avg.
Static	Top-2 softmax	32.15/.939	27.50/.867	25.16/.787	28.77/.879
Static	Dense signed dot-product	32.35/.940	27.77/.869	25.40/.790	29.01/.881
Dynamic	Top-2 softmax	32.50/.941	27.94/.870	25.76/.792	29.20/.882
Dynamic	Dense signed dot-product, CEA	**33.36/.943**	**28.51/.875**	**25.90/.798**	**29.79/.886**

With Top-2 routing, dynamic factors improve the overall average from 28.77/0.879 to 29.20/0.882. With dense assembly, the same change improves the overall average from 29.01/0.881 to 29.79/0.886, showing the importance of input-conditioned factor generation. Static dense assembly remains close to the backbone-only result in Table 13, while dynamic dense assembly yields a higher score of 29.79/0.886. Dense signed assembly also improves over Top-2 routing under both factor sources, and the gain is larger with dynamic factors (+0.59 dB) than with static factors (+0.24 dB). This interaction supports the central design of CEA: rank-wise components should be generated from the current image and used through dense token-wise assembly. Additional controlled studies on spatial query probing, RankNorm, rank sensitivity, and injection targets are provided in Appendix D.

5 Limitations

This work evaluates CEA on standard individual and compositional restoration benchmarks, together with real-captured WeatherBench images. Evaluating CEA on additional restoration settings would further clarify its generalization behavior. The current implementation uses a U-shaped Transformer backbone and modulates attention projections; extending CEA to other backbones and dense prediction tasks is left for future work. As with other image enhancement methods, improved restoration

may raise privacy concerns or unintended downstream use, so responsible use should consider consent and appropriate data handling.

6 Conclusion

We proposed *Continuous Expert Assembly* (CEA), a parameter-efficient framework for all-in-one image restoration under unknown and compositional degradations. CEA generates an image-conditioned low-rank residual component space and applies it through continuous assembly. A lightweight Cross-Attention Hyper-Adapter probes intermediate spatial features to synthesize routing bases and residual directions shared within each layer and target projection, while dense signed coefficients produce token-dependent residual responses. The controlled studies support the combination of spatial-query factor generation and dense signed assembly. Experiments on AIO-3, AIO-5, and CDD-11 show moderate but consistent and efficiency-preserving gains on standard AiO benchmarks, with the largest improvements on spatially varying and compositional degradations. WeatherBench further demonstrates improvements at both model scales across all three real-captured weather subsets.

Acknowledgments and Disclosure of Funding

This work is supported by the National Natural Science Foundation of China under Grant (62576224 and 62606333); the Guangdong Provincial Key Laboratory of Computility Microelectronics (2024B1212010007); the Guangdong Basic and Applied Basic Research Foundation Project (2026A1515011733); the Characteristic Innovation Project of Ordinary Universities in Guangdong Province (2024KTSCX026); the Research and Development Fund Project of Xi'an Thermal Power Research Institute Co., Ltd. (TP-25-TYK10); the Research and Development Fund Project of China Huaneng Group Co., Ltd. (HNKJ26-H26).

The authors declare no competing interests.

References

[1] Yuang Ai, Huaibo Huang, and Ran He. Lora-ir: Taming low-rank experts for efficient all-in-one image restoration. *arXiv preprint arXiv:2410.15385*, 2024. doi: 10.48550/arXiv.2410.15385.

[2] Jin Cao, Yi Cao, Li Pang, Deyu Meng, and Xiangyong Cao. Hair: Hypernetworks-based all-in-one image restoration. *arXiv preprint arXiv:2408.08091*, 2024. doi: 10.48550/arXiv.2408.08091.

[3] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12294–12305, 2021. doi: 10.1109/CVPR46437.2021.01212.

[4] Liangyu Chen, Xin Lu, Jie Zhang, Xiaojie Chu, and Chengpeng Chen. Hinet: Half instance normalization network for image restoration. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 182–192, June 2021. doi: 10.1109/CVPRW53098.2021.00027.

[5] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *Computer Vision – ECCV 2022*, pages 17–33, 2022. doi: 10.1007/978-3-031-20071-7_2.

[6] Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. Dynamic convolution: Attention over convolution kernels. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11027–11036, 2020. doi: 10.1109/CVPR42600.2020.01104.

[7] Zheng Chen, Yulun Zhang, Jinjin Gu, Yongbing Zhang, Linghe Kong, and Xin Yuan. Cross aggregation transformer for image restoration. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 25478–25490, 2022.

[8] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamás Sarlós, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy J. Colwell, and Adrian Weller. Rethinking attention with performers. In *International Conference on Learning Representations (ICLR)*, 2021.

[9] Marcos V. Conde, Gregor Geigle, and Radu Timofte. Instructir: High-quality image restoration following human instructions. In *Computer Vision – ECCV 2024*, pages 1–21, 2024. doi: 10.1007/978-3-031-72764-1_1.

[10] Bert De Brabandere, Xu Jia, Tinne Tuytelaars, and Luc Van Gool. Dynamic filter networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 667–675, 2016. 10.48550/arXiv.1605.09673.

[11] Yu Dong, Yihao Liu, He Zhang, Shifeng Chen, and Yu Qiao. FD-GAN: Generative adversarial networks with fusion-discriminator for single image dehazing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020. doi: 10.1609/aaai.v34i07.6701.

[12] Huiyu Duan, Xiongkuo Min, Sijing Wu, Wei Shen, and Guangtao Zhai. Uniprocessor: A text-induced unified low-level image processor. In *Computer Vision – ECCV 2024*, pages 180–199, 2024. doi: 10.1007/978-3-031-72855-6_11.

[13] Qingnan Fan, Dongdong Chen, Lu Yuan, Gang Hua, Nenghai Yu, and Baoquan Chen. A general decoupled learning framework for parameterized image operators. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43:33–47, 2021. doi: 10.1109/TPAMI.2019.2925793.

[14] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23 (120):1–39, 2022.

[15] Xueyang Fu, Borong Liang, Yue Huang, Xinghao Ding, and John Paisley. Lightweight pyramid networks for image deraining. *IEEE Transactions on Neural Networks and Learning Systems*, 31(6):1794–1807, 2020. doi: 10.1109/TNNLS.2019.2926481.

[16] 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*, pages 12607–12613, 2025. doi: 10.1145/3746027.3758196.

[17] Hang Guo, Tao Dai, Yuanchao Bai, Bin Chen, Xudong Ren, Zexuan Zhu, and Shu-Tao Xia. Parameter efficient adaptation for image restoration with heterogeneous mixture-of-experts. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 13522–13547, 2024. doi: 10.52202/079017-0432.

[18] Hang Guo, Jinmin Li, Tao Dai, Zhihao Ouyang, Xudong Ren, and Shu-Tao Xia. Mambair: A simple baseline for image restoration with state-space model. In *Computer Vision – ECCV 2024*, 2024. doi: 10.1007/978-3-031-72649-1_13.

[19] Yu Guo, Yuan Gao, Yuxu Lu, Huilin Zhu, Ryan Wen Liu, and Shengfeng He. Onerestore: A universal restoration framework for composite degradation. In *Computer Vision – ECCV 2024*, pages 255–272, 2025. doi: 10.1007/978-3-031-72655-2_15.

[20] David Ha, Andrew Dai, and Quoc V. Le. Hypernetworks. In *International Conference on Learning Representations (ICLR)*, 2017.

[21] 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 *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97, pages 2790–2799, 2019.

[22] 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. In *International Conference on Learning Representations (ICLR)*, 2022. doi: 10.48550/arXiv.2106.09685.

[23] Liming Jiang, Bo Dai, Wayne Wu, and Chen Change Loy. Focal frequency loss for image reconstruction and synthesis. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13899–13909, October 2021. doi: 10.1109/ICCV48922.2021.01366.

[24] Jeya Maria Jose Valanarasu, Rajeev Yasarla, and Vishal 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)*, pages 2343–2353, 2022. doi: 10.1109/CVPR52688.2022.00239.

[25] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5156–5165, 2020.

[26] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. doi: 10.48550/arXiv.1412.6980.

[27] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations (ICLR)*, 2021.

[28] Boyun Li, Xiao Liu, Peng Hu, Zhongqin Wu, Jiancheng Lv, and Xi Peng. All-in-one image restoration for unknown corruption. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17431–17441, 2022. doi: 10.1109/CVPR52688.2022.01693.

[29] Zilong Li, Yiming Lei, Chenglong Ma, Junping Zhang, and Hongming Shan. Prompt-in-prompt learning for universal image restoration. *arXiv preprint arXiv:2312.05038*, 2023. doi: 10.48550/arXiv.2312.05038.

[30] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 1833–1844, 2021. doi: 10.1109/ICCVW54120.2021.00210.

[31] Lin Liu, Lingxi Xie, Xiaopeng Zhang, Shanxin Yuan, Xiangyu Chen, Wengang Zhou, Houqiang Li, and Qi Tian. TAPE: Task-agnostic prior embedding for image restoration. In *Computer Vision – ECCV 2022*, 2022. doi: 10.1007/978-3-031-19797-0_26.

[32] Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. DoRA: Weight-decomposed low-rank adaptation. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235, pages 32100–32121, 2024. doi: 10.48550/arXiv.2402.09353.

[33] 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 (ICLR)*, 2024.

[34] Jiaqi Ma, Tianheng Cheng, Guoli Wang, Qian Zhang, Xinggang Wang, and Lefei Zhang. Prores: Exploring degradation-aware visual prompt for universal image restoration. *arXiv preprint arXiv:2306.13653*, 2023. doi: 10.48550/ARXIV.2306.13653.

[35] Ben Mildenhall, Jonathan T. Barron, Jiawen Chen, Dillon Sharlet, Ren Ng, and Robert Carroll. Burst denoising with kernel prediction networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2502–2510, 2018. doi: 10.1109/CVPR.2018.00265.

[36] Chong Mou, Qian Wang, and Jian Zhang. Deep generalized unfolding networks for image restoration. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. doi: 10.1109/CVPR52688.2022.01688.

[37] Ozan Özdenizci and Robert Legenstein. Restoring vision in adverse weather conditions with patch-based denoising diffusion models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8):10346–10357, 2023. doi: 10.1109/TPAMI.2023.3238179.

[38] 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 (NeurIPS)*, pages 71275–71293, 2023. doi: 10.52202/075280-3121.

[39] Joan Puigcerver, Carlos Riquelme Ruiz, Basil Mustafa, and Neil Houlsby. From sparse to soft mixtures of experts. In *International Conference on Learning Representations*, pages 28435–28445, 2024.

[40] Bin Ren, Yawei Li, Xu Zheng, Yuqian Fu, Danda Pani Paudel, Hong Liu, Ming-Hsuan Yang, Luc Van Gool, and Nicu Sebe. Efficient degradation-agnostic image restoration via channel-wise functional decomposition and manifold regularization. *arXiv preprint arXiv:2505.18679*, 2026. doi: 10.48550/arXiv.2505.18679.

[41] Carlos Riquelme, Joan Puigcerver, Basil Mustafa, Maxim Neumann, Rodolphe Jenatton, André Susano Pinto, Daniel Keysers, and Neil Houlsby. Scaling vision with sparse mixture of experts. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 8583–8595, 2021. 10.48550/arXiv.2106.05974.

[42] David Serrano-Lozano, Luis Herranz, Shaolin Su, and Javier Vazquez-Corral. Adaptive blind all-in-one image restoration. *arXiv preprint arXiv:2411.18412*, 2025. doi: 10.48550/arXiv.2411.18412.

[43] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations (ICLR)*, 2017. 10.48550/arXiv.1701.06538.

[44] Zhuoran Shen, Mingyuan Zhang, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Efficient attention: Attention with linear complexities. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 3530–3538, 2021. doi: 10.1109/WACV48630.2021.00357.

[45] Chunwei Tian, Yong Xu, and Wangmeng Zuo. Image denoising using deep CNN with batch renormalization. *Neural Networks*, 121:461–473, 2020. doi: 10.1016/j.neunet.2019.08.022.

[46] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. MAXIM: Multi-axis MLP for image processing. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5759–5770, 2022. doi: 10.1109/CVPR52688.2022.00568.

[47] Tao Wang, Kaihao Zhang, Ziqian Shao, Wenhan Luo, Bjorn Stenger, Tong Lu, Tae-Kyun Kim, Wei Liu, and Hongdong Li. Gridformer: Residual dense transformer with grid structure for image restoration in adverse weather conditions. *International Journal of Computer Vision*, 132:4541–4563, 2024. doi: 10.1007/s11263-024-02056-0.

[48] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17662–17672, June 2022. doi: 10.1109/CVPR52688.2022.01716.

[49] Gang Wu, Junjun Jiang, Kui Jiang, and Xianming Liu. Harmony in diversity: Improving all-in-one image restoration via multi-task collaboration. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024. doi: 10.1145/3664647.3680762.

[50] Brandon Yang, Gabriel Bender, Quoc V. Le, and Jiquan Ngiam. Condconv: conditionally parameterized convolutions for efficient inference. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1307–1318, 2019.

[51] Eduard Zamfir, Zongwei Wu, Nancy Mehta, Yuedong Tan, Danda Pani Paudel, Yulun Zhang, and Radu Timofte. Complexity experts are task-discriminative learners for any image restoration. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12753–12763, 2025. doi: 10.1109/CVPR52734.2025.01190.

[52] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14816–14826, 2021. doi: 10.1109/CVPR46437.2021.01458.

[53] 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 *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5728–5739, 2022. doi: 10.1109/CVPR52688.2022.00564.

[54] Haijin Zeng, Xiangming Wang, Yongyong Chen, Jingyong Su, and Jie Liu. Vision-language gradient descent-driven all-in-one deep unfolding networks. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. doi: 10.1109/CVPR52734.2025.00705.

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

[56] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. doi: 10.1109/TIP.2017.2662206.

[57] Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adaptive budget allocation for parameter-efficient fine-tuning. In *International Conference on Learning Representations (ICLR)*, 2023. 10.48550/arXiv.2303.10512.

[58] Yurui Zhu, Tianyu Wang, Xueyang Fu, Xuanyu Yang, Xin Guo, Jifeng Dai, Yu Qiao, and Xiaowei Hu. Learning weather-general and weather-specific features for image restoration under multiple adverse weather conditions. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21747–21758, 2023. doi: 10.1109/CVPR52729.2023.02083.

A Additional Method Details

A.1 Cross-Attention Hyper-Adapter Details

Feature condensation. For each CEA-equipped decoder block, the hyper-adapter first condenses the intermediate feature map before cross-attention. The condensation module consists of a depthwise 3×3 convolution followed by a pointwise 1×1 convolution. The depthwise convolution uses stride s_c and padding 1, reducing the spatial resolution from $H_l \times W_l$ to $\lceil H_l/s_c \rceil \times \lceil W_l/s_c \rceil$, while the pointwise convolution preserves the channel dimension. Unless otherwise stated, we use $s_c = 2$. The resulting condensed tokens are used as keys and values in the Cross-Attention Hyper-Adapter, while the learnable queries remain compact and rank-oriented. The subsequent query probing and target-specific FC decoding follow Eqs. (5)–(6).

A.2 RankNorm and Sequential Low-Rank Assembly

The factorization $A_\tau^{(l)}(x)B_\tau^{(l)}(x)$ is scale-ambiguous because $AB = (cA)(B/c)$ for any nonzero scalar c. We therefore normalize each generated rank component before assembly. For each target τ and rank index k,

$$\big(A_\tau^{(l)\prime}(x)\big)_{:,k} = \frac{\big(A_\tau^{(l)}(x)\big)_{:,k}}{\|\big(A_\tau^{(l)}(x)\big)_{:,k}\|_2 + \epsilon}, \qquad \big(B_\tau^{(l)\prime}(x)\big)_{k,:} = \frac{\big(B_\tau^{(l)}(x)\big)_{k,:}}{\|\big(B_\tau^{(l)}(x)\big)_{k,:}\|_2 + \epsilon}. \tag{9}$$

This per-instance normalization stabilizes the token-to-rank affinities and avoids cross-sample coupling. The residual update is then implemented as

$$\Delta Y_\tau^{(l)} = \alpha \left(X^{(l)} A_\tau^{(l)\prime}(x) \right) B_\tau^{(l)\prime}(x), \tag{10}$$

which requires two low-rank matrix multiplications and avoids forming the equivalent dense $d_{\text{in}} \times d_{\text{out}}$ dynamic update.

B Detailed Experimental Settings

B.1 Datasets and Protocols

We follow standard all-in-one image restoration protocols. AIO-3 trains one model on mixed dehazing, deraining, and denoising data and evaluates it on SOTS, Rain100L, and BSD68 with Gaussian noise levels $\sigma \in \{15, 25, 50\}$. AIO-5 further includes deblurring and low-light enhancement and is reported using five benchmark entries: SOTS, Rain100L, BSD68 at $\sigma = 25$, GoPro, and LOLv1. For compositional restoration, we follow the CDD-11 protocol, which includes four single degradations, five double compositions, and two triple compositions. At test time, CEA receives only the degraded image. For WeatherBench, we use the official split with haze, rain, and snow; results are reported in Section 4.4.

B.2 Backbone and CEA Configuration

We use an asymmetric U-shaped Transformer encoder-decoder backbone following prior all-in-one restorers. The encoder contains three downsampling stages with $[4, 6, 6]$ Transformer blocks, followed by a latent stage with 8 blocks. The decoder contains $[2, 4, 4]$ Transformer blocks from low to high resolution, followed by 4 refinement blocks. We report two model scales: CEA-IR-S uses initial embedding dimension $C = 32$, while CEA-IR uses $C = 48$; in both cases the channel dimension doubles after each downsampling stage. The attention heads are $[1, 2, 4, 8]$ across the four resolution levels, the FFN expansion ratio is 2, and convolution/projection biases are disabled.

Unless otherwise stated, CEA modules are inserted into alternating decoder blocks, starting from the first block of each decoder stage. This gives 1, 2, and 2 CEA-equipped blocks in the three decoder stages, respectively. The default injection targets are the attention query and key projections (Q and K). The low-rank bottleneck is $r = 8$, and each hyper-adapter uses 4-head cross-attention with 16 learnable queries in total, with 8 queries for A and 8 queries for B. The CEA placement, injection targets, rank, and hyper-adapter configuration are kept the same across the two model scales.

B.3 Optimization and Training Objective

All CEA models and matched controlled variants are trained with the following reconstruction objective. We optimize

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{rec}} + \lambda_f \mathcal{L}_{\text{fft}}, \qquad \mathcal{L}_{\text{rec}} = \|\hat{y} - y\|_1, \tag{11}$$

with $\lambda_f = 0.10$. The Fourier-domain reconstruction regularizer is

$$\mathcal{L}_{\text{fft}} = \big\| \, |\mathcal{F}(\hat{y})| - |\mathcal{F}(y)| \, \big\|_1, \tag{12}$$

where $\mathcal{F}(\cdot)$ denotes FFT. Following recent restoration practice, this term is shared by CEA models and matched controlled variants in these experiments.

We train all CEA models and matched controlled variants using Adam with $(\beta_1, \beta_2) = (0.9, 0.999)$, a total batch size of 64, an initial learning rate of 5×10^{-4}, and cosine learning-rate decay. AIO models are trained for 120 epochs, while CDD-11 and WeatherBench models are trained for 200 epochs. We use random input crops of 128×128 for the first 80% of training epochs and 192×192 for the remaining 20%, with random horizontal and vertical flips throughout training.

Training hardware and wall-clock cost. All reported training runs were conducted on a single workstation with four NVIDIA RTX 4090 GPUs. For the lightweight CEA-IR-S model, CDD-11 training took approximately 15 hours for 200 epochs. AIO-3 and AIO-5 training took approximately 30–37.5 hours each, about 2–2.5$\times$ the CDD-11 wall-clock time with the same total batch size of 64 and two-stage crop schedule. The wall-clock time can vary with dataset I/O, implementation details, and model scale.

Inference measurement. Each method is evaluated in a fresh process, with FLOPs/memory profiling performed separately from runtime measurement. All methods use an NVIDIA RTX 4090, FP32, batch size 1, 224×224 inputs, and 100 warm-up iterations, followed by five complete passes over the same 804-image AIO-3 test set. CUDA events measure one model forward; file I/O, preprocessing, host-to-device transfer, metric computation, profiling, and memory queries are outside the timed region.

C Additional Quantitative Results

C.1 Full Results on AIO-3

Table 6 provides the full per-task AIO-3 comparison corresponding to the compact summary in the main paper.

Table 6: **Full AIO-3 benchmark.** Quantitative comparison on three degradations following the standard AiO protocol [38, 55]. We report PSNR (dB, ↑) / SSIM (↑) on SOTS (dehazing), Rain100L (deraining), and BSD68 Gaussian denoising with $\sigma \in \{15, 25, 50\}$. Methods are grouped by model capacity (*Light* vs. *Heavy*). Best results within each group are highlighted.

	Method	Params.	Dehazing		Deraining		Denoising						Average	
			SOTS		Rain100L		BSD68 $\sigma = 15$		BSD68 $\sigma = 25$		BSD68 $\sigma = 50$		PSNR	SSIM
Light	BRDNet [45]	–	23.23	.895	27.42	.895	32.26	.898	29.76	.836	26.34	.693	27.80	.843
	LPNet [15]	–	20.84	.828	24.88	.784	26.47	.778	24.77	.748	21.26	.552	23.64	.738
	FDGAN [11]	–	24.71	.929	29.89	.933	30.25	.910	28.81	.868	26.43	.776	28.02	.883
	DL [13]	2M	26.92	.931	32.62	.931	33.05	.914	30.41	.861	26.90	.740	29.98	.876
	AirNet [28]	9M	27.94	.962	34.90	.967	33.92	**.933**	31.26	.888	28.00	.797	31.20	.910
	MoCE-IR-S [51]	11M	30.94	.979	38.22	.983	34.08	**.933**	31.42	.888	28.16	.798	32.57	.916
	CEA-IR-S (ours)	10M	**31.90**	**.981**	**38.62**	**.984**	**34.16**	**.933**	**31.51**	**.889**	**28.27**	**.803**	**32.89**	**.918**
Heavy	MPRNet [52]	16M	25.28	.955	33.57	.954	33.54	.927	30.89	.880	27.56	.779	30.17	.899
	PromptIR [38]	36M	30.58	.974	36.37	.972	33.98	.933	31.31	.888	28.06	.799	32.06	.913
	Gridformer [47]	34M	30.37	.970	37.15	.972	33.93	.931	31.37	.887	28.11	.801	32.19	.912
	PromptIR+Art [49]	33M	30.83	.979	37.94	.982	34.06	.934	31.42	.891	28.14	.801	32.49	.917
	DA-CLIP [33]	125M	29.46	.963	36.28	.968	30.02	.821	24.86	.585	22.29	.476	28.58	.763
	UniProcessor [12]	1002M	31.66	.979	38.17	.982	34.08	.935	31.42	.891	28.17	.803	32.70	.918
	LoRA-IR [1]	–	30.68	.961	37.75	.979	34.06	.935	31.42	.891	28.18	.803	32.42	.914
	Res-HAIR [2]	29M	30.98	.979	38.59	.983	34.16	.935	31.51	.892	28.24	.803	32.70	.919
	VLU-Net [54]	35M	30.71	.980	**38.93**	.984	34.13	.935	31.48	.892	28.23	.804	32.70	.919
	MoCE-IR [51]	25M	31.34	.979	38.57	.984	34.11	.932	31.45	.888	28.18	.800	32.73	.917
	CEA-IR (ours)	22M	**32.33**	**.983**	38.92	**.985**	**34.22**	**.937**	**31.58**	**.895**	**28.32**	**.809**	**33.07**	**.922**

C.2 Full Results on AIO-5

Table 7 provides the full per-task AIO-5 comparison corresponding to the compact summary in the main paper.

C.3 Full Per-Category Results on CDD-11

Table 8 provides the full per-category CDD-11 comparison. The main paper reports group-level averages to save space.

D Additional Statistical and Ablation Studies

D.1 Paired Bootstrap Analysis on AIO-3 and AIO-5

We perform paired bootstrap analyses on the lightweight AIO-3 and AIO-5 comparisons to assess whether the improvements are consistent across matched test images. Table 9 reports the observed paired differences and bootstrap confidence intervals. For each benchmark entry, we compute the per-image PSNR/SSIM difference between CEA-IR-S and MoCE-IR-S, resample matched image pairs with replacement, and report the mean difference together with the 95% percentile confidence interval over 10,000 bootstrap resamples. This analysis quantifies consistency across matched test images for fixed trained models. For AIO-5, the overall statistic is computed as the unweighted mean

Table 7: **Full AIO-5 benchmark.** Comparison on five degradations under the same AiO training protocol [38, 55]. We evaluate on SOTS (dehazing), Rain100L (deraining), BSD68 ($\sigma = 25$), GoPro (deblurring), and LOLv1 (low-light). PSNR (dB, ↑) / SSIM (↑) are computed on full RGB images. Our method consistently achieves the best average across two model scales.

	Method	Params.	Dehazing		Deraining		Denoising		Deblurring		Low-Light		Average	
			SOTS		Rain100L		BSD68 σ=25		GoPro		LOLv1		PSNR	SSIM
Light	SwinIR [30]	1M	21.50	.891	30.78	.923	30.59	.868	24.52	.773	17.81	.723	25.04	.835
	DL [13]	2M	20.54	.826	21.96	.762	23.09	.745	19.86	.672	19.83	.712	21.05	.743
	TAPE [31]	1M	22.16	.861	29.67	.904	30.18	.855	24.47	.763	18.97	.621	25.09	.801
	AirNet [28]	9M	21.04	.884	32.98	.951	30.91	.882	24.35	.781	18.18	.735	25.49	.847
	MoCE-IR-S [51]	11M	**31.33**	.978	37.21	.978	31.25	.884	28.90	.877	21.68	.851	30.08	.913
	CEA-IR-S (ours)	10M	31.02	**.979**	**38.50**	**.984**	**31.45**	**.888**	**29.47**	**.886**	**23.58**	**.858**	**30.80**	**.919**
Heavy	NAFNet [5]	17M	25.23	.939	35.56	.967	31.02	.883	26.53	.808	20.49	.809	27.76	.881
	DGUNet [36]	17M	24.78	.940	36.62	.971	31.10	.883	27.25	.837	21.87	.823	28.32	.891
	Restormer [53]	26M	24.09	.927	34.81	.962	31.49	.884	27.22	.829	20.41	.806	27.60	.881
	MambaIR [18]	27M	25.81	.944	36.55	.971	31.41	.884	28.61	.875	22.49	.832	28.97	.901
	Transweather [24]	38M	21.32	.885	29.43	.905	29.00	.841	25.12	.757	21.21	.792	25.22	.836
	IDR [55]	15M	25.24	.943	35.63	.965	**31.60**	.887	27.87	.846	21.34	.826	28.34	.893
	Gridformer [47]	34M	26.79	.951	36.61	.971	31.45	.885	29.22	.884	22.59	.831	29.33	.904
	InstructIR-5D [9]	16M	27.10	.956	36.84	.973	31.40	.887	29.40	.886	23.00	.836	29.55	.907
	Res-HAIR [2]	29M	30.62	.978	38.11	.981	31.49	.891	28.52	.874	23.12	.847	30.37	.914
	VLU-Net [54]	35M	30.84	.980	38.54	.982	31.43	.891	27.46	.840	22.29	.833	30.11	.905
	MoCE-IR [51]	25M	30.48	.974	38.04	.982	31.34	.887	**30.05**	**.899**	23.00	.852	30.58	.919
	CEA-IR (ours)	22M	**31.96**	**.981**	**38.95**	**.985**	31.53	**.894**	29.90	.894	**23.32**	**.857**	**31.13**	**.922**

Table 8: **Full CDD-11 compositional degradations.** Results on the CDD-11 protocol introduced in OneRestore [19]. We report PSNR (dB, ↑) / SSIM (↑) on single, double, and triple degradation compositions (L: low-light, H: haze, R: rain, S: snow). Our method achieves the best overall average and yields consistent gains on challenging haze-related compositions.

(a) Single degradations

Method	Low (L)	Haze (H)	Rain (R)	Snow (S)
AirNet [28]	24.83/.778	24.21/.951	26.55/.891	26.79/.919
PromptIR [38]	26.32/.805	26.10/.969	31.56/.946	31.53/.960
WGWSNet [58]	24.39/.774	27.90/.982	33.15/.964	34.43/.973
WeatherDiff [37]	23.58/.763	21.99/.904	24.85/.885	24.80/.888
OneRestore [19]	26.48/.826	32.52/.990	33.40/.964	34.31/.973
MoCE-IR-S [51]	27.26/.824	32.66/.990	34.31/.970	35.91/.980
CEA-IR-S (ours)	**27.47/.827**	**34.75/.993**	**34.72/.971**	**36.49/.982**

(b) Double degradations

Method	L+H	L+R	L+S	H+R	H+S
AirNet [28]	23.23/.779	22.82/.710	23.29/.723	22.21/.868	23.29/.901
PromptIR [38]	24.49/.789	25.05/.771	24.51/.761	24.54/.924	23.70/.925
WGWSNet [58]	24.27/.800	25.06/.772	24.60/.765	27.23/.955	27.65/.960
WeatherDiff [37]	21.83/.756	22.69/.730	22.12/.707	21.25/.868	21.99/.868
OneRestore [19]	25.79/.822	25.58/.799	25.19/.789	29.99/.957	30.21/.964
MoCE-IR-S [51]	26.24/.817	26.25/.800	26.04/.793	29.93/.964	30.19/.970
CEA-IR-S (ours)	**26.67/.824**	**26.65/.806**	**26.58/.802**	**31.15/.968**	**31.49/.974**

(c) Triple degradations and overall average

Method	Params.	L+H+R	L+H+S	Avg.
AirNet [28]	9M	21.80/.708	22.24/.725	23.75/.814
PromptIR [38]	36M	23.74/.752	23.33/.747	25.90/.850
WGWSNet [58]	26M	23.90/.772	23.97/.771	26.96/.863
WeatherDiff [37]	83M	21.23/.716	21.04/.698	22.49/.799
OneRestore [19]	6M	24.78/.788	24.90/.791	28.47/.878
MoCE-IR-S [51]	11M	25.41/.789	25.39/.790	29.05/.881
CEA-IR-S (ours)	**10M**	**25.87/.797**	**25.93/.799**	**29.79/.886**

over five benchmark entries: SOTS, Rain100L, BSD68 at $\sigma = 25$, GoPro, and LOLv1. Because Gaussian noise is generated online for denoising and the full benchmark tables report rounded aggregate scores, the bootstrap mean computed from raw matched per-image differences may not exactly equal the arithmetic difference obtained from the displayed rounded table entries.

Table 9: **Paired bootstrap analysis on AIO-3 and AIO-5.** Differences are computed as CEA-IR-S minus MoCE-IR-S on matched test images. Confidence intervals are estimated from 10,000 paired bootstrap resamples.

Benchmark	Metric	Observed difference	95% CI
AIO-3	PSNR	+0.3046	[+0.2565, +0.3582]
AIO-3	SSIM	+0.0019	[+0.0017, +0.0022]
AIO-5	PSNR	+0.6892	[+0.3591, +1.0192]
AIO-5	SSIM	+0.0054	[+0.0033, +0.0073]

All confidence intervals are strictly positive. For both benchmarks, all 10,000 bootstrap resamples yield positive mean differences. On AIO-5, CEA-IR-S improves most benchmark entries, with the only negative PSNR difference occurring on SOTS dehazing; the overall paired improvement remains positive.

D.2 Paired Bootstrap Analysis on CDD-11

To further assess the robustness of the CDD-11 improvement, we perform a paired bootstrap analysis between CEA-IR-S and MoCE-IR-S. The analysis uses matched test images: for each image, we first compute the PSNR/SSIM difference between the two methods, and then resample matched image pairs with replacement. We run 10,000 bootstrap resamples, and Table 10 reports the mean difference together with the 95% percentile confidence interval.

Table 10: **Paired bootstrap analysis on CDD-11.** Differences are computed as CEA-IR-S minus MoCE-IR-S on matched test images. Confidence intervals are estimated from 10,000 paired bootstrap resamples.

Metric	MoCE-IR-S	CEA-IR-S	Difference with 95% CI
PSNR	29.0490	29.7924	+0.7434 [+0.6692, +0.8205]
SSIM	0.8806	0.8857	+0.0051 [+0.0047, +0.0057]

Both confidence intervals are strictly positive, and all 10,000 bootstrap resamples yield positive mean differences. These results support the consistency of the CDD-11 improvement under paired test-image resampling.

D.3 Global Conditioning vs. Spatial Query Probing

To evaluate whether spatial probing is necessary for factor generation, Table 11 compares CEA with a Global-Dynamic variant that predicts $A(x)$ and $B(x)$ from a global average-pooled feature using an MLP. Both variants use the same backbone, Q/K injection, rank $r = 8$, reconstruction losses, and dense softmax-free dot-product assembly; only the factor generator differs.

Table 11: **Global conditioning vs. spatial query probing on CDD-11.** All variants use the same reconstruction objective and are trained without degradation labels.

Generator	Single	Double	Triple	Avg.
GAP + MLP	32.53/.941	27.87/.871	25.53/.792	29.14/.882
Cross-attention queries (CEA)	**33.36/.943**	**28.51/.875**	**25.90/.798**	**29.79/.886**

Spatial query probing improves the overall average from 29.14/0.882 to 29.79/0.886. The gain is observed across all degradation groups, supporting the use of cross-attention queries to collect spatial degradation evidence before generating the low-rank factors.

D.4 RankNorm and Rank Sensitivity

We further analyze two design choices in the low-rank assembly: the use of RankNorm and the rank r. All variants are evaluated on CDD-11 using the lightweight CEA-IR-S setting with the same backbone, training objective, and $Q{+}K$ injection.

The left part of Table 12 shows that removing RankNorm reduces the overall average from 29.79/0.886 to 29.59/0.885. This indicates that RankNorm improves the conditioning of dynamically generated factors, although the model remains effective without it. The right part of Table 12 studies the rank r. Increasing the rank from 4 to 8 improves the average PSNR from 29.40 to 29.79, while further increasing the rank to 16 slightly decreases performance to 29.70. We therefore use $r = 8$ as the default setting, which provides the best accuracy with negligible parameter overhead.

Table 12: **RankNorm and rank sensitivity on CDD-11.** (a) RankNorm improves the conditioning of dynamically generated low-rank factors. (b) Performance saturates at $r = 8$, indicating that a compact residual subspace is sufficient.

(a) Effect of RankNorm

Variant	PSNR ↑	SSIM ↑
w/o RankNorm	29.59	.885
w/ RankNorm	**29.79**	**.886**

(b) Effect of rank r

r	Params.	PSNR ↑	SSIM ↑
4	9.708M	29.40	.884
8	9.712M	**29.79**	**.886**
16	9.718M	29.70	.885

D.5 Ablation on Injection Targets

Table 13: **Ablation on injection targets (CDD-11).** We report PSNR/SSIM averaged over CDD11-Single, CDD11-Double, CDD11-Triple, and the overall average.

Injection targets	Single Avg.	Double Avg.	Triple Avg.	Overall Avg.
None (backbone only)	32.35/.941	27.74/.870	25.42/.792	29.00/.882
Q only	32.88/.942	28.12/.873	25.62/.794	29.39/.884
K only	32.81/.942	28.00/.872	25.39/.792	29.28/.883
V only	32.73/.942	27.99/.871	25.51/.792	29.26/.882
$Q{+}K$ **(default)**	**33.36/.943**	**28.51/.875**	**25.90/.798**	**29.79/.886**
$Q{+}K{+}V$	32.94/.942	28.18/.873	25.68/.795	29.46/.884
W_1 only (FFN-in)	32.73/.942	27.93/.871	25.50/.792	29.23/.883

Table 13 studies where to inject the CEA residual. Modulating Q and K gives the best overall result, improving over the backbone-only variant by 0.79 dB in PSNR. All single-target variants improve PSNR over the backbone, but their gains are smaller than $Q{+}K$ injection. This suggests that CEA is most effective when it reshapes attention matching rather than only changing value content or feed-forward features. Adding V on top of $Q{+}K$ reduces the overall average from 29.79/0.886 to 29.46/0.884. We therefore use $Q{+}K$ injection as the default accuracy–efficiency trade-off.

E Additional Out-of-Distribution Evaluation

E.1 Zero-Shot Underwater Enhancement on UIEB

Table 14 reports out-of-domain generalization on UIEB under the zero-shot protocol used in recent degradation-agnostic restoration work. Models are trained under the all-in-one restoration setting and directly evaluated on the paired UIEB subset without underwater-specific training or finetuning. This experiment evaluates transfer from all-in-one restoration to underwater images.

CEA-IR-S obtains 17.31/0.774 on UIEB without underwater-specific training or finetuning. Compared with zero-shot all-in-one baselines, it improves over MoCE-IR by 1.40 dB and is slightly above MIRAGE-S [40]. These results indicate transfer to underwater images outside the training degradation set.

Table 14: **Zero-shot underwater enhancement on UIEB.** PSNR/SSIM are evaluated on the paired UIEB subset. All methods are evaluated under a zero-shot restoration setting without underwater-specific training or finetuning.

Method	PSNR	SSIM
SwinIR	15.31	.740
NAFNet	15.42	.744
Restormer	15.46	.745
AirNet	15.46	.745
IDR	15.58	.762
PromptIR	15.48	.748
MoCE-IR	15.91	.765
MIRAGE-S	17.29	.773
CEA-IR-S	**17.31**	**.774**

F Additional Qualitative Results

F.1 Qualitative Comparison on AIO-3

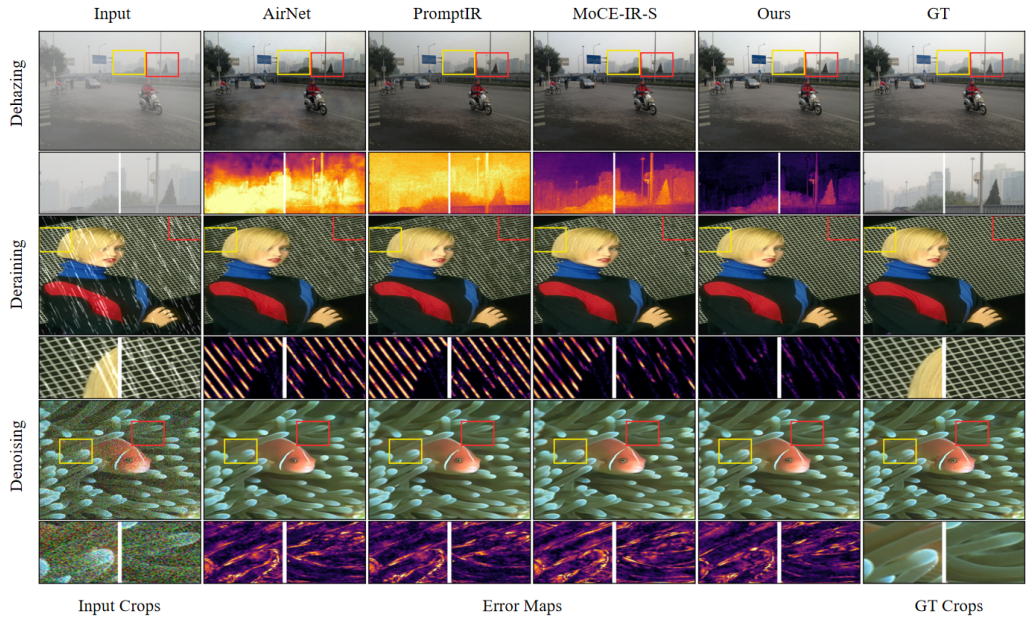


Figure 3: **Qualitative comparison on the AIO-3 benchmark.** We compare CEA-IR-S against AirNet, PromptIR, and MoCE-IR-S across dehazing, deraining, and denoising. The bottom rows show zoom-in crops and absolute error maps, where darker regions indicate lower reconstruction error. CEA reduces residual degradation artifacts and preserves fine structures across the three restoration tasks.

Figure 3 provides representative AIO-3 examples. CEA-IR-S produces fewer residual artifacts in localized degraded regions, such as rain streaks and haze boundaries, while preserving fine structures under denoising.

F.2 Qualitative Comparison on AIO-5

Figure 4 shows representative AIO-5 examples. Across the five tasks, CEA reduces residual artifacts in local regions and better preserves structural details, while maintaining natural brightness and color in the low-light case.

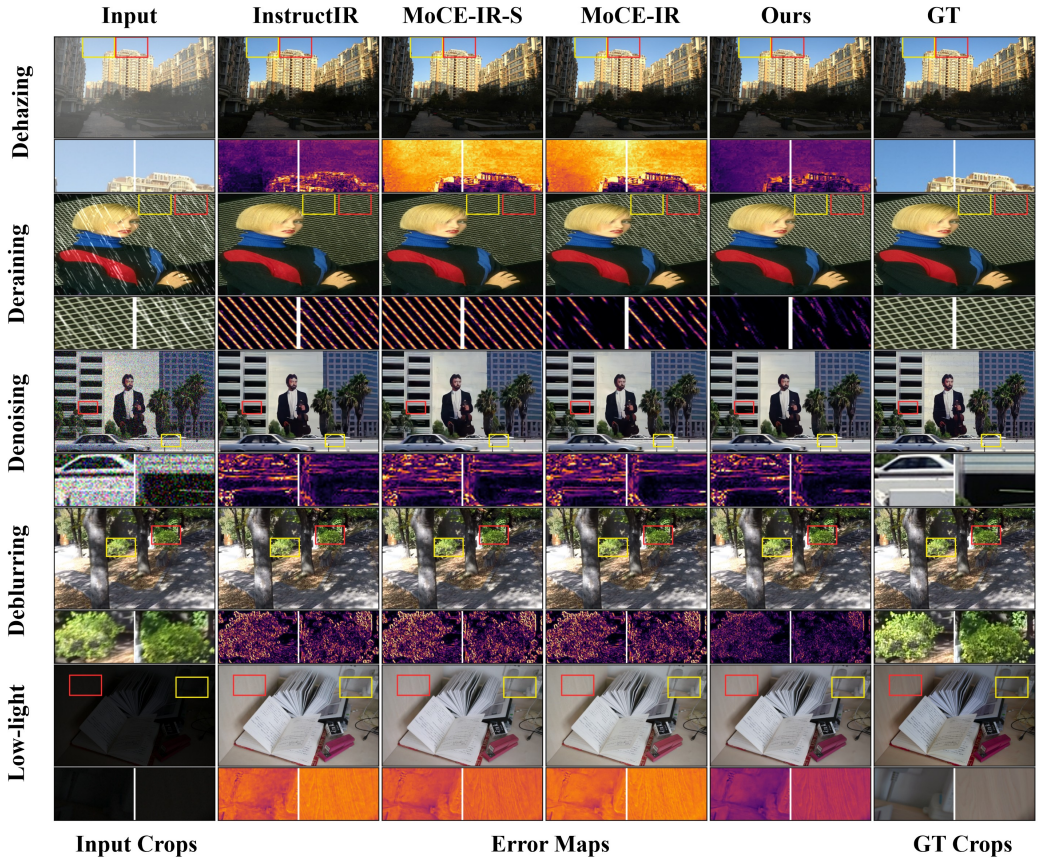


Figure 4: **Additional qualitative comparison on the AIO-5 benchmark.** We compare our method with representative baselines across dehazing, deraining, denoising, deblurring, and low-light enhancement.

F.3 Qualitative Comparison on CDD-11

Figures 5–7 provide additional CDD-11 comparisons. The advantage of CEA is most visible when multiple degradation operators interact, especially in haze-related compositions where both global visibility and local detail preservation are required.

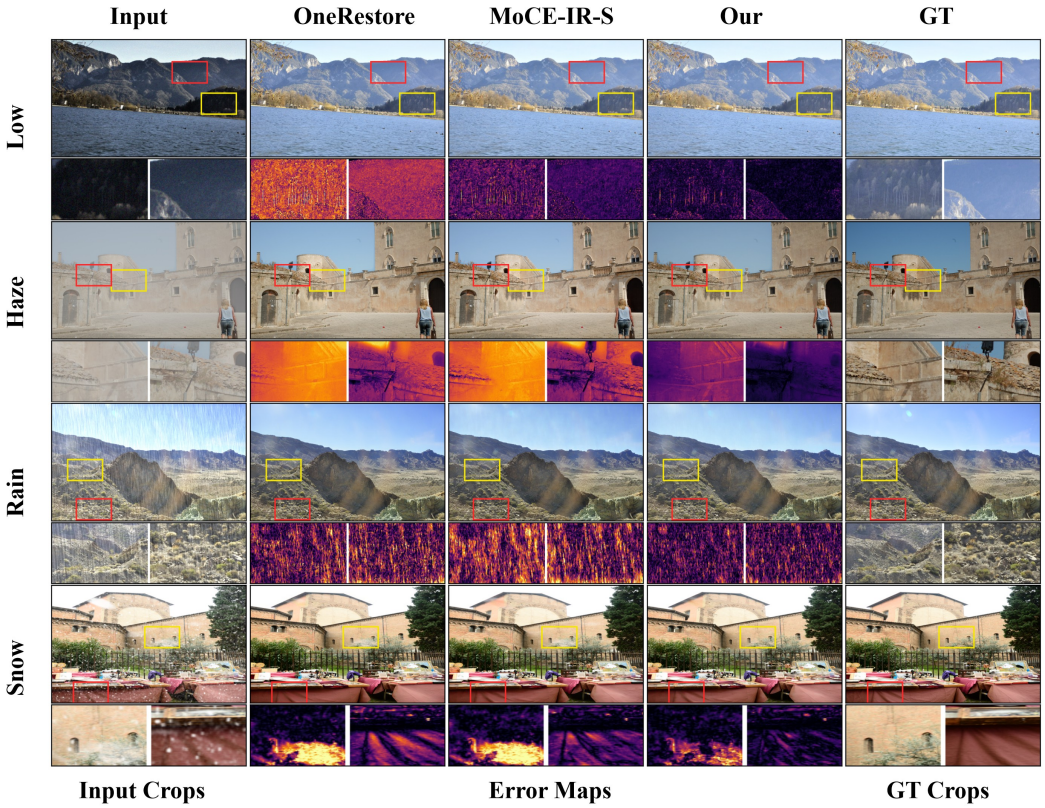


Figure 5: **Qualitative comparison on CDD-11 single degradations.** We compare OneRestore, MoCE-IR-S, and CEA-IR-S on low-light, haze, rain, and snow.

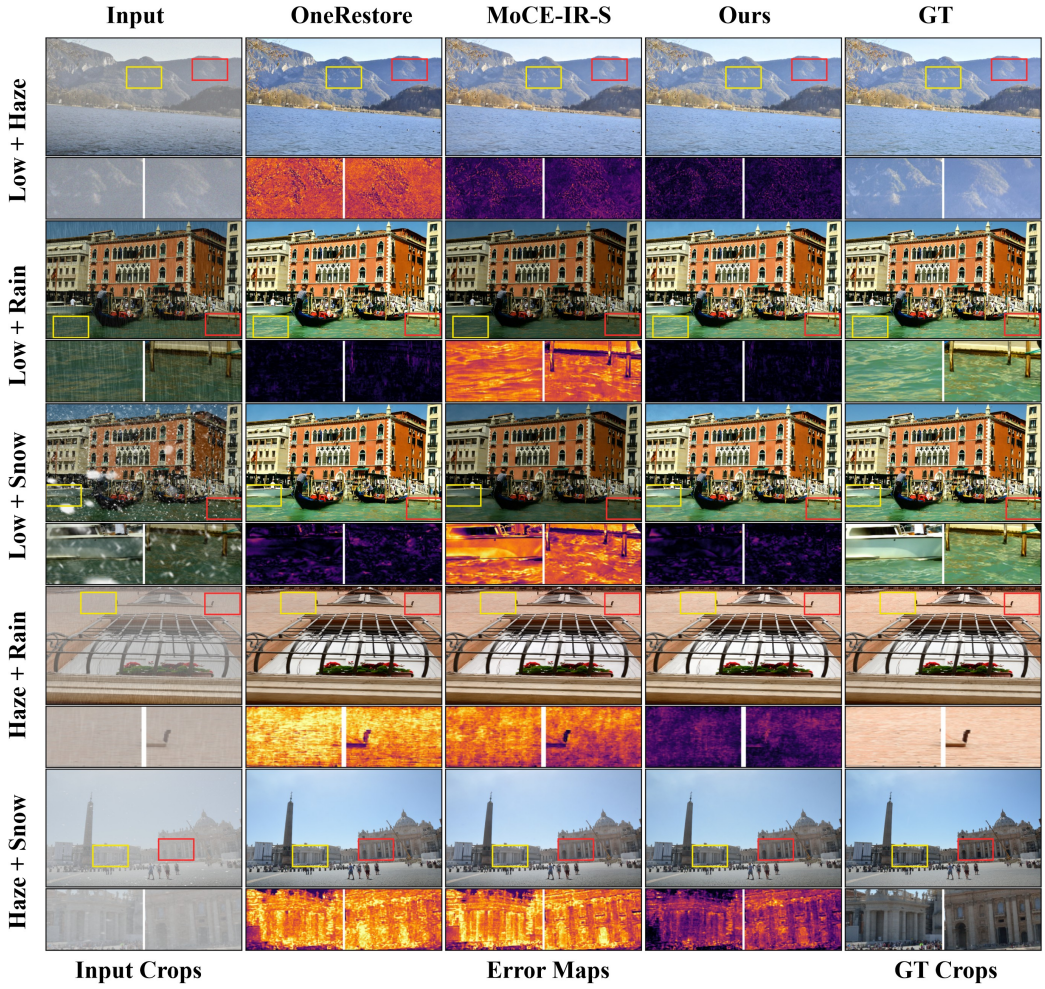


Figure 6: **Qualitative comparison on CDD-11 double degradations.** Representative examples include low-light+haze, low-light+rain, low-light+snow, haze+rain, and haze+snow.

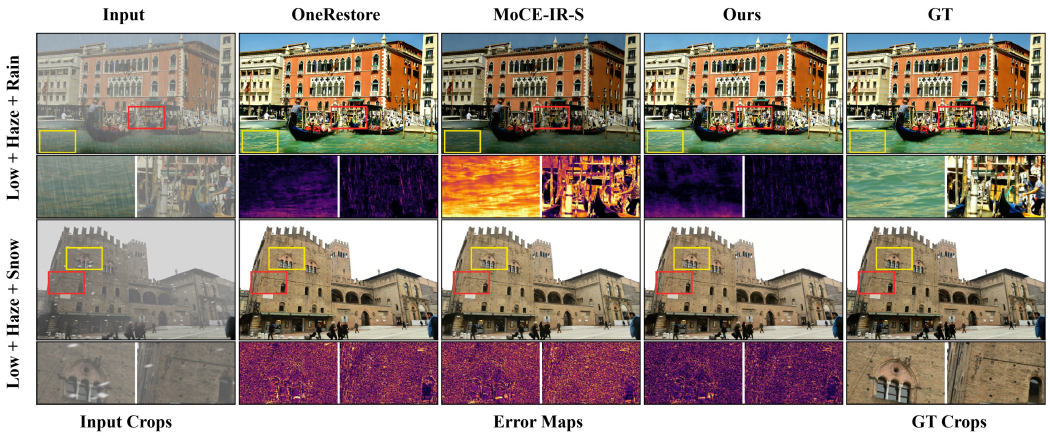
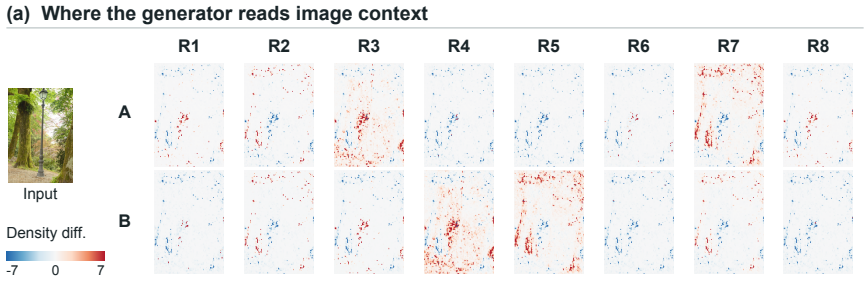


Figure 7: **Qualitative comparison on CDD-11 triple degradations.** We show representative examples with low-light+haze+rain and low-light+haze+snow.

F.4 Continuous assembly on CDD-11

At each token, the signed coefficients in Eq. (3) combine the generated residual directions. At site 1, the queries emphasize different regions while the update magnitudes largely overlap (Figure 8). R3's mean affinity share rises from 12.89% under rain to 14.41% under low light with rain. At site 4, mean shares stay close to 12.5%.



(c) Component weighting across degradation conditions

Degradation	Injection layer 1								Injection layer 4							
	R1	R2	R3	R4	R5	R6	R7	R8	R1	R2	R3	R4	R5	R6	R7	R8
Haze	12.4	12.5	13.2	12.3	12.4	12.4	12.4	12.4	12.4	12.4	12.6	12.6	12.4	12.4	12.6	12.5
Rain	12.2	12.8	12.9	12.3	12.4	12.2	12.9	12.2	12.4	12.4	12.7	12.4	12.4	12.5	12.6	12.5
Snow	12.4	12.6	13.0	12.3	12.4	12.3	12.6	12.4	12.4	12.5	12.7	12.5	12.4	12.4	12.6	12.6
Low light	12.4	12.3	13.8	12.2	12.2	12.4	12.3	12.4	12.4	12.4	12.6	12.4	12.5	12.5	12.6	12.5
Haze + rain	12.2	12.7	13.5	12.2	12.2	12.2	12.7	12.2	12.6	12.5	12.5	12.8	12.4	12.4	12.5	12.4
Haze + snow	12.3	12.6	13.2	12.3	12.4	12.3	12.5	12.4	12.5	12.5	12.5	12.6	12.4	12.4	12.5	12.5
Low + haze	12.4	12.4	13.5	12.3	12.3	12.3	12.3	12.4	12.5	12.4	12.7	12.7	12.4	12.3	12.6	12.5
Low + rain	12.2	12.2	14.4	12.2	12.4	12.2	12.2	12.2	12.5	12.4	12.6	12.5	12.5	12.5	12.6	12.5
Low + snow	12.2	12.2	14.4	12.2	12.3	12.2	12.3	12.2	12.5	12.5	12.6	12.5	12.5	12.5	12.5	12.5
Low + haze + rain	12.3	12.4	13.7	12.3	12.3	12.3	12.4	12.3	12.5	12.4	12.6	12.8	12.3	12.3	12.5	12.4
Low + haze + snow	12.8	12.1	13.8	12.1	12.1	12.1	12.1	12.8	12.5	12.5	12.6	12.7	12.4	12.4	12.5	12.5

Cell values: mean affinity share (%)

Color: deviation from 12.5%

-2 pp 0 +2 pp

Figure 8: **Component responses in CEA-IR-S on CDD-11.** Panels (a,b) show a rain-degraded image at decoder site 1, selected after inspecting five sites. (a) Where A and B queries read image features. Red and blue mark attention above and below the local rank mean within each branch. (b) Rows show coefficient share $|G_{n,k}|/\sum_j |G_{n,j}|$ relative to $1/8$, pre-summation update magnitude, and its deviation from the local rank mean. (c) Rows are degradations and columns are components. Cells give mean shares (%) over tokens and 20 matched scenes at sites 1 and 4. Rank indices are local to each site.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction describe Continuous Expert Assembly, its instance-conditioned low-rank parameterization with token-dependent residual assembly, the spatial query-probing hyper-adapter, and the evaluation scope including WeatherBench. The quantitative and qualitative claims are supported by Sections 3–4 and Appendices B–F, while scope limitations are stated in Section 5.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 5 explicitly discusses limitations, including evaluation on standard restoration benchmarks, the current focus on Transformer Q/K modulation, the need to test other dense-prediction tasks/backbones.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not present formal theoretical results, theorems, or lemmas requiring proof. It provides algebraic formulations of the CEA update and its linear-attention-style computation in Section 3 and Appendix A.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper discloses the architecture, CEA insertion strategy, loss function, optimization setup, training schedule, data protocols, and evaluation metrics in Sections 3–4 and Appendices A–B. These details are sufficient to reproduce the main experimental comparisons to the extent supported by the manuscript.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The current submission does not include an anonymized code release, exact run commands, or a packaged reproduction artifact. Public datasets, evaluation protocols, architecture details, and training settings are described in the paper and appendix. We plan to release the code.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Sections 4.1 and Appendices B.1–B.3 specify the datasets and protocols, blind evaluation setting, backbone and CEA configuration, optimizer, losses, batch size, learning rate, training epochs, augmentations, and runtime measurement procedure.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: Appendix D.1 reports paired bootstrap analyses with 95% confidence intervals for the lightweight AIO-3 and AIO-5 comparisons between CEA-IR-S and MoCE-IR-S. Appendix D.2 further reports paired bootstrap results on CDD-11 with 10,000 resamples over matched test images. Table 4 reports runtime as mean $\pm$ sample standard deviation across five complete-pass mean runtimes. The paired-bootstrap intervals describe variability across matched test images for fixed trained models, and the runtime error bars describe variability across repeated test-set passes.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix B.3 reports the training hardware and wall-clock cost. All training runs used four NVIDIA RTX 4090 GPUs; CDD-11 training of CEA-IR-S took approximately 15 hours, while AIO-3 and AIO-5 took approximately 30–37.5 hours each under the same batch and crop setting. Section 4.6 and Table 4 further report inference memory, FLOPs, and runtime means with sample standard deviations on a single NVIDIA RTX 4090 GPU.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research uses established image-restoration benchmarks and does not involve human subjects, sensitive personal data collection, or deceptive deployment.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Section 5 discusses the scope of the evaluation and notes that, as with other image enhancement methods, improved restoration may raise privacy concerns or be used in unintended downstream applications, motivating attention to consent and appropriate data handling.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release a high-risk asset such as a pretrained language model, image generator, or scraped sensitive dataset. The work uses existing image-restoration benchmarks, and broader dual-use concerns are discussed in Section 5.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [No]

    Justification: The paper credits the datasets, protocols, baselines, and codebases used in the experiments through citations. However, license or terms-of-use information is not consistently available from the official sources for all legacy restoration benchmarks and baseline implementations, so we do not claim complete license coverage in the paper. We use these assets only for academic research and evaluation, do not redistribute the datasets or baseline code, and will update the asset-license information where official terms can be verified.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper proposes a new method and model architecture, but it does not introduce or release a new dataset, benchmark, or standalone asset requiring separate documentation in this paper.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing experiments or research with human subjects; all evaluations use existing image-restoration benchmarks.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or human-subject research, so IRB approval or equivalent human-subjects review is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not used as an important, original, or non-standard component of the core method. Any use of LLMs solely for writing, editing, or formatting would not affect the methodology, experiments, scientific rigor, or originality of the research.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.