# IMAGE-INTERNAL OPERATOR CALIBRATION FOR ALL-IN-ONE IMAGE RESTORATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

All-in-one image restoration requires adapting shared restoration knowledge to diverse degradations and image content. Representative prompt- and routing-based methods achieve this through feed-forward conditioning, without explicitly optimizing restoration operators on the current observation. We propose **CalibIR**, an *image-internal operator calibration* framework that uses each image to learn corrections to the operators used for its restoration. Starting from learned base operators, CalibIR performs a lightweight optimization step on an input-specific feature-prediction objective within the forward pass. To exploit image-wide information while accommodating regional differences, we develop a structured calibration rule that combines channel and spatial corrections. The channel correction aggregates information across the image, whereas the spatial correction is centered to capture departures from the image-wide adjustment. Both arise from the same optimization objective, enabling coordinated adaptation without maintaining an independent kernel for every channel–location pair. Trained end-to-end under restoration supervision, CalibIR combines knowledge learned across images with operator adaptation learned within each image. Experiments on standard all-in-one image restoration benchmarks show that CalibIR achieves leading average performance.

## 1 INTRODUCTION

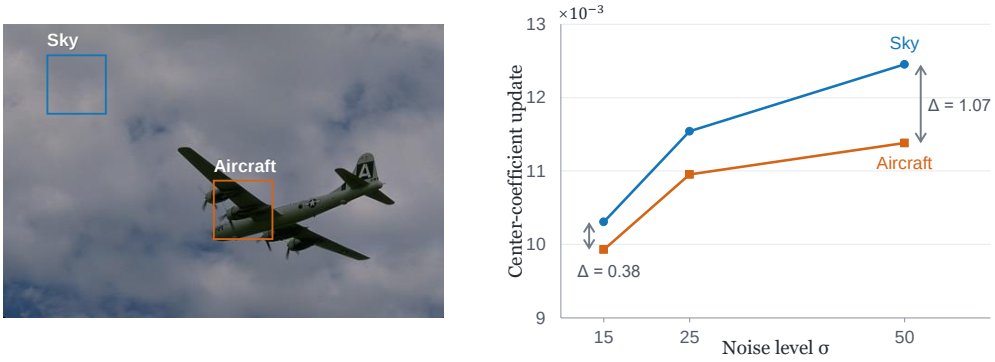
Convolutional networks have established effective learned models for image denoising (Zhang et al., 2017; 2018), with further advances in progressive reconstruction and simple restoration architectures (Zamir et al., 2021; Chen et al., 2022). Transformer-based methods extend this progress through windowed and efficient attention for image restoration (Liang et al., 2021; Wang et al., 2022; Zamir et al., 2022). Building on these developments, all-in-one image restoration aims to recover clean images from diverse degradations using a single model, without requiring the degradation type to be specified at inference (Li et al., 2022; Potlapalli et al., 2023; Cui et al., 2025). Sharing restoration knowledge across tasks reduces the need to maintain a separate model for each degradation. However, the appropriate restoration of an image depends on both its degradation and its content. Unified restoration therefore needs to combine knowledge learned across training images with adjustments suited to the current observation.

Existing approaches use learned degradation representations and input-dependent guidance to adapt a shared restoration backbone (Li et al., 2022; Potlapalli et al., 2023; Cui et al., 2025; Zamfir et al., 2025). Input-adaptive filtering also allows the processing operators to vary with the observation (Jia et al., 2016; Mildenhall et al., 2018). These approaches produce image-specific adjustments through feed-forward mappings learned during training.

Test-time training (TTT) layers offer a way to adapt through learning on the input itself (Sun et al., 2025; Han et al., 2026). In their original formulation for one-dimensional sequences, lightweight models update their parameters through an inner prediction task, encoding historical context in an adaptive state. For image restoration, this learning takes place within a complete observation with spatial structure. Update evidence is distributed across channels and spatial locations, combining information shared across the image with differences associated with local content. Even under the same degradation conditions, smooth regions and structured details can provide different update evidence. Spatially pooling this evidence produces a shared operator update that uses image-wide



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.




(a) Two regions of the same image

(b) Region-dependent updates

Figure 1: **Image-internal operator calibration under spatially uniform noise.** (a) A clean reference image with two fixed regions. (b) Center-coefficient updates relative to the learned base kernel in the first CalibIR block of the final decoder stage, from full-image inference under spatially uniform Gaussian noise at $\sigma = 15, 25, 50$. Updates include both channel and spatial corrections with their learned step sizes and are averaged over channels and each marked region. $\Delta$ denotes the between-region update gap.

information, but also merges position-dependent variation. Image-internal learning for restoration therefore calls for organizing corrections across channels and space to use image-wide information while retaining regional adjustments.

To this end, we propose **CalibIR**, an *image-internal operator calibration* framework for unified restoration. CalibIR starts from lightweight feature-space convolutional operators learned across training images and uses a prediction task on the current image to form channel and spatial corrections. Learned feature mappings construct the prediction task for the base operators, whose prediction residuals provide update evidence. A single structured gradient update organizes this evidence into two complementary corrections. Spatial aggregation produces channel-specific corrections shared across locations, allowing each channel to use information from the entire image. Channel aggregation produces position-specific corrections shared across channels, with spatial centering retaining regional deviations from the image-wide average update. Both components come from the same prediction objective and jointly calibrate the base operators, which then process the current features. The feature mappings, base operators, and update step sizes are learned jointly under restoration supervision. Calibration takes place within the forward pass using each image's own prediction task, without a clean target at inference. As shown in Figure 1, regional differences in the corrections change with the overall noise level, illustrating the combination of image-level adaptation and regional adjustment.

Our contributions are summarized as follows:

- We develop CalibIR, a unified restoration framework that combines operators learned across images with image-specific calibration through a lightweight optimization step within the forward pass.
- We introduce a structured calibration rule that derives channel and spatial corrections from the same feature-prediction objective to use image-wide information while retaining regional differences.
- We evaluate CalibIR through restoration experiments, component ablations, and operator analyses, and characterize how its calibration varies with the input.

## 2 Related Work

**All-in-one image restoration.** All-in-one restoration methods differ in how they represent degradations and condition a shared backbone. Early approaches explored unified adverse-weather removal through architectural search and Transformer-based modeling (Li et al., 2020; Valanarasu

et al., 2022). Degradation representations can be learned through contrastive objectives, multi-degradation modeling, or classification pre-training (Li et al., 2022; Zhang et al., 2023; Hu et al., 2025). Other approaches condition reconstruction through prompts, frequency modulation, and expert routing (Potlapalli et al., 2023; Cui et al., 2025; Zamfir et al., 2025), or use vision-language representations and human instructions as guidance (Luo et al., 2024; Conde et al., 2024). Recent work also improves restoration backbones. MIRAGE assigns channels to complementary convolution, attention, and MLP branches and applies cross-layer manifold regularization (Ren et al., 2026a). ExDA introduces nonlinear value transforms and a global spatial token to improve attention expressivity and degradation awareness (Ren et al., 2026b). Optimization-based restoration embeds iterative reconstruction into trainable networks, combining model-based updates with learned feature processing. DGUNet unfolds proximal gradient descent with learned gradient estimation (Mou et al., 2022), while VLU-Net combines deep unfolding with vision-language guidance for all-in-one restoration (Zeng et al., 2025).

**Input-adaptive filtering.** Dynamic Filter Networks generate filters from input images and support both image-specific and position-specific filtering (Jia et al., 2016). CondConv and Dynamic Convolution combine learned kernels with input-dependent weights (Yang et al., 2019; Chen et al., 2020). Pixel-Adaptive Convolution modulates spatial filtering using local features (Su et al., 2019), while Kernel Prediction Networks predict spatially varying kernels for burst alignment and denoising (Mildenhall et al., 2018). ADMS selects degradation-specific filters for all-in-one restoration (Park et al., 2023). These methods generate or select filters through learned feed-forward mappings. CalibIR derives operator corrections from an inner feature-prediction objective on the current image.

**Image-internal learning and test-time training.** Early image-internal methods exploit recurring patches and statistics within a single observation (Glasner et al., 2009; Zontak & Irani, 2011). Deep Image Prior fits an untrained generator to the observation using the network architecture as a prior (Ulyanov et al., 2018), while ZSSR trains on examples extracted from the test image to exploit internal recurrence (Shocher et al., 2018). Fast-weight models store input information in temporary parameters (Ba et al., 2016), and linear attention can be interpreted as fast-weight programming (Schlag et al., 2021). Test-time training adapts models through self-supervised objectives on test inputs (Sun et al., 2020). TTT layers represent a hidden state as a lightweight model updated through an inner objective within the forward pass (Sun et al., 2025; Han et al., 2026).

# 3 METHOD

CalibIR calibrates feature-space convolutional operators through a prediction task within each input image. Figure 2 shows the restoration network and expands its operator calibration module. The network uses a multi-scale backbone built from CalibIR (CR) blocks, following Restormer (Zamir et al., 2022). Each block contains operator calibration and a gated feed-forward network (FFN), each preceded by layer normalization (LN) and followed by a residual connection. During both training and inference, each calibration module computes one update from the current features. Configurations for CalibIR and CalibIR-s are provided in Appendix A.1.

## 3.1 IMAGE-INTERNAL OPERATOR CALIBRATION

A base operator learned across the training set provides the starting point for calibration. Following TTT layers and their visual instantiations (Sun et al., 2025; Han et al., 2026), we use prediction errors on features from the current input to compute a correction to this operator.

Given a layer-normalized feature map $X \in \mathbb{R}^{C\times H\times W}$ with $C$ channels and spatial size $H \times W$, learned mappings produce $Q$, $K$, and $V$ through a $1 \times 1$ convolution followed by a $3 \times 3$ depthwise convolution. $K$ and $V$ form the input and target of the prediction task, while $Q$ is processed by the calibrated operator. We $\ell_2$-normalize $Q$ and $K$ along channels at each spatial location.

Let $N = HW$ be the number of spatial locations, with $c$ indexing channels and $p$ denoting a spatial location. We use $o \in \Omega$ for a kernel offset, where $\Omega$ contains the nine offsets in a $3 \times 3$ neighborhood and $p + o$ denotes the neighboring location.

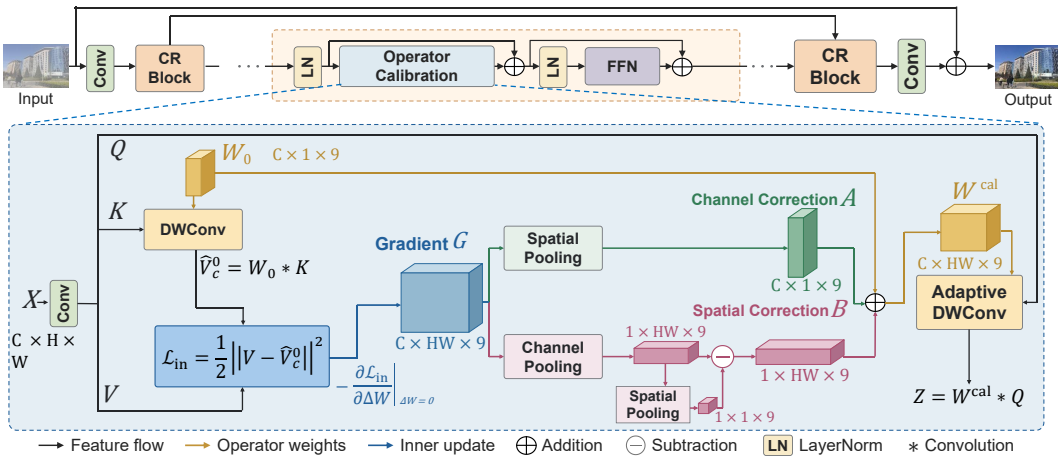



Figure 2: **Overview of CalibIR.** The upper part shows the restoration network and one block, with operator calibration expanded below. Cuboid dimensions denote channels, spatial locations, and kernel offsets, with $|\Omega| = 9$ for the $3 \times 3$ neighborhood. Pooling averages over the indicated dimension to produce the channel correction $A$ and spatial correction $B$, with spatial mean subtraction in the latter branch. The corrections are added to $W_0$ to form $W^{\mathrm{cal}}$.

The base operator $W_0$ contains one learnable depthwise kernel per channel, with coefficient $W_{0,c,o}$ at offset $o$ in channel $c$. Before calibration, it predicts $\widehat{V}^0$ from $K$ as

$$\widehat{V}_c^{\,0}(p) = \sum_{o \in \Omega} W_{0,c,o} K_c(p+o). \tag{1}$$

We introduce an operator correction $\Delta W$ to adapt $W_0$ to the current input. Here, $\Delta W_{c,p,o}$ denotes the correction to coefficient $W_{0,c,o}$ at output location $p$. The inner prediction objective is

$$\mathcal{L}_{\mathrm{in}}(\Delta W) = \frac{1}{2} \sum_{c,p} \left[ V_c(p) - \sum_{o \in \Omega} \big(W_{0,c,o} + \Delta W_{c,p,o}\big) K_c(p+o) \right]^2. \tag{2}$$

At $\Delta W = 0$, this objective gives the prediction loss $\frac{1}{2}\|V - \widehat{V}^{\,0}\|_F^2$ shown in Figure 2. Its negative gradient at this starting point provides the direction for operator calibration:

$$-\frac{\partial \mathcal{L}_{\mathrm{in}}}{\partial \Delta W_{c,p,o}}\bigg|_{\Delta W=0} = K_c(p+o)\big[V_c(p) - \widehat{V}_c^{\,0}(p)\big]. \tag{3}$$

The gradient retains contributions from individual channels and positions for each kernel offset. We aggregate these contributions into $\Delta W$, yielding the calibrated operator $W^{\mathrm{cal}}$ for the current input:

$$W^{\mathrm{cal}}_{c,p,o} = W_{0,c,o} + \Delta W_{c,p,o}. \tag{4}$$

We then use $W^{\mathrm{cal}}$ to process $Q$. Section 3.2 describes the form of $\Delta W$ and how it is computed.

### 3.2 Channel and Spatial Corrections

We denote the negative gradient in Equation (3) by $G_{c,p,o}$. We form the operator correction from two terms. The *channel correction* $A$ varies across channels and is shared across spatial positions, while the *spatial correction* $B$ varies across positions and is shared across channels.

$$\Delta W_{c,p,o} = A_{c,o} + B_{p,o}, \qquad \sum_p B_{p,o} = 0. \tag{5}$$

**Channel correction.** The upper branch in Figure 2 averages $G$ over all positions to obtain the channel correction $A$:

$$A_{c,o} = \frac{\eta_{\mathrm{ch}}}{N} \sum_p G_{c,p,o}. \tag{6}$$

The learned positive scalar step size $\eta_{\mathrm{ch}}$ controls the update magnitude. This correction allows each channel to use information from the entire image.

Table 1: Quantitative results PSNR (↑) and SSIM (↑) on the AIO3 setting. Best results are in **bold**.

| Method | Venue | Dehazing | Deraining | Denoising on BSD68 | | | Average | Params |
|---|---|---|---|---|---|---|---|---|
| | | SOTS | Rain100L | $\sigma = 15$ | $\sigma = 25$ | $\sigma = 50$ | | |
| Restormer (Zamir et al., 2022) | CVPR'22 | 30.39 / 0.968 | 36.19 / 0.971 | 33.91 / 0.929 | 31.18 / 0.883 | 28.02 / 0.786 | 31.94 / 0.907 | 26M |
| PromptIR (Potlapalli et al., 2023) | NeurIPS'23 | 30.58 / 0.974 | 36.37 / 0.972 | 33.98 / 0.933 | 31.31 / 0.888 | 28.06 / 0.799 | 32.06 / 0.913 | 36M |
| Art-PromptIR (Wu et al., 2024) | ACMMM'24 | 30.83 / 0.979 | 37.94 / 0.982 | 34.06 / 0.934 | 31.42 / 0.891 | 28.14 / 0.801 | 32.49 / 0.917 | 33M |
| AdaIR (Cui et al., 2025) | ICLR'25 | 31.06 / 0.980 | 38.64 / 0.983 | 34.12 / 0.935 | 31.45 / 0.892 | 28.19 / 0.802 | 32.69 / 0.918 | 29M |
| PromptIR-TUR (Wu et al., 2025) | AAAI'25 | 31.17 / 0.978 | 38.57 / 0.984 | 34.06 / 0.932 | 31.40 / 0.887 | 28.13 / 0.797 | 32.67 / 0.916 | 33M |
| VLU-Net (Zeng et al., 2025) | CVPR'25 | 30.71 / 0.980 | 38.93 / 0.984 | 34.13 / 0.935 | 31.48 / 0.892 | 28.23 / 0.804 | 32.70 / 0.919 | 35M |
| MoCE-IR (Zamfir et al., 2025) | CVPR'25 | 31.34 / 0.979 | 38.57 / 0.984 | 34.11 / 0.932 | 31.45 / 0.888 | 28.18 / 0.800 | 32.73 / 0.917 | 25M |
| ExDA (Ren et al., 2026b) | ICLR'26 | 31.58 / 0.982 | 39.13 / 0.985 | 34.22 / **0.937** | 31.56 / **0.895** | 28.32 / 0.808 | 32.96 / 0.921 | 22M |
| ClearAIR (Zhang et al., 2026) | AAAI'26 | 31.08 / 0.981 | 38.61 / 0.984 | 34.18 / 0.935 | 31.50 / 0.891 | 28.31 / 0.804 | 32.74 / 0.919 | 31M |
| **CalibIR-s(Ours)** | 2026 | **33.05 / 0.984** | 38.82 / 0.985 | 34.16 / 0.936 | 31.50 / 0.894 | 28.24 / 0.806 | 33.15 / 0.921 | 10M |
| **CalibIR(Ours)** | 2026 | 32.57 / **0.984** | 39.35 / **0.986** | **34.24 / 0.937** | **31.58 / 0.895** | **28.34 / 0.809** | **33.22 / 0.922** | 23M |

**Spatial correction.** The spatial correction $B$ retains variation across positions that spatial aggregation removes. The lower branch first averages $G$ over channels to obtain one correction per location shared by all channels. This average includes a spatially constant component that the channel correction $A$ can already represent. We therefore subtract the spatial mean and use a separate learned positive scalar step size $\eta_{\mathrm{sp}}$:

$$B_{p,o} = \eta_{\mathrm{sp}} \left[ \frac{1}{C} \sum_{c} G_{c,p,o} - \frac{1}{CN} \sum_{c,p'} G_{c,p',o} \right]. \tag{7}$$

Here, $p'$ ranges over all spatial locations in the mean term. Subtracting this mean leaves the deviation at each location and enforces the zero-mean constraint in Equation (5). Both corrections use the same prediction residual at $\Delta W = 0$ and are computed together in a single calibration step. Each module learns its own $\eta_{\mathrm{ch}}$ and $\eta_{\mathrm{sp}}$.

Adding $W_0$ and $A$ at every spatial position, and $B$ in every channel, gives $W^{\mathrm{cal}}$ in Equation (4). The adaptive depthwise convolution in Figure 2 uses this operator to process $Q$ and produce the feature map $Z$:

$$Z_c(p) = \sum_{o \in \Omega} W^{\mathrm{cal}}_{c,p,o} Q_c(p + o). \tag{8}$$

A $1 \times 1$ output projection maps $Z$ back to the block features before residual addition.

# 4 Experiments

## 4.1 Experimental Setup

We evaluate CalibIR on AIO3, AIO5, CDD11 (Guo et al., 2024), and WeatherBench (Guan et al., 2025), training one unified model for each setting. AIO3 combines Gaussian denoising, deraining, and dehazing, while AIO5 adds motion deblurring and low-light enhancement. CDD11 covers 11 single and composite degradation categories, and WeatherBench tests restoration under real-world haze, rain, and snow.

We train for 150 epochs with AdamW (Loshchilov & Hutter, 2019) and the pixel and Fourier reconstruction losses used in MoCE-IR (Zamfir et al., 2025). Appendix B provides the dataset protocols and training settings, with the loss definitions in Appendix B.3.

## 4.2 Main Results

We compare CalibIR and its compact variant, CalibIR-s, with recent all-in-one restoration methods under four settings. Additional visual comparisons are provided in Appendix F. Model configurations and computational costs are in Appendix A.

**AIO3.** Table 1 shows that CalibIR achieves the highest average PSNR and SSIM of 33.22 dB and 0.922. It reaches 39.35 dB on Rain100L and achieves the highest denoising PSNR at all three noise levels.

Table 2: Quantitative results PSNR (↑) and SSIM (↑) on the AIO5 setting. Best results are in **bold**.

| Method | Venue | Dehazing | Deraining | Denoising | Deblurring | Enhance | Average | Params |
|---|---|---|---|---|---|---|---|---|
| | | SOTS | Rain100L | $BSD68_{\sigma=25}$ | GoPro | LOL | | |
| PromptIR (Potlapalli et al., 2023) | NeurIPS'23 | 25.20 / 0.931 | 35.94 / 0.964 | 31.17 / 0.882 | 27.32 / 0.842 | 20.94 / 0.799 | 28.11 / 0.883 | 36M |
| Gridformer (Wang et al., 2024) | IJCV'24 | 26.79 / 0.951 | 36.61 / 0.971 | 31.45 / 0.885 | 29.22 / 0.884 | 22.59 / 0.831 | 29.33 / 0.904 | 34M |
| AdaIR (Cui et al., 2025) | ICLR'25 | 30.53 / 0.978 | 38.02 / 0.981 | 31.35 / 0.889 | 28.12 / 0.858 | 23.00 / 0.845 | 30.20 / 0.910 | 29M |
| VLU-Net (Zeng et al., 2025) | CVPR'25 | 30.84 / 0.980 | 38.54 / 0.982 | 31.43 / 0.891 | 27.46 / 0.840 | 22.29 / 0.833 | 30.11 / 0.905 | 35M |
| MoCE-IR (Zamfir et al., 2025) | CVPR'25 | 30.48 / 0.974 | 38.04 / 0.982 | 31.34 / 0.887 | 30.05 / 0.899 | 23.00 / 0.852 | 30.58 / 0.919 | 25M |
| Transweather-TUR (Wu et al., 2025) | AAAI'25 | 29.68 / 0.966 | 33.09 / 0.952 | 30.40 / 0.869 | 26.63 / 0.815 | 23.02 / 0.838 | 28.56 / 0.888 | 38M |
| ExDA (Ren et al., 2026b) | ICLR'26 | 31.14 / 0.981 | **39.23 / 0.985** | 31.52 / 0.893 | 29.06 / 0.878 | 23.19 / 0.859 | 30.83 / 0.919 | 22M |
| ClearAIR (Zhang et al., 2026) | AAAI'26 | 30.12 / 0.978 | 38.20 / 0.982 | **31.53** / 0.888 | 29.67 / 0.887 | 22.83 / 0.846 | 30.45 / 0.916 | 31M |
| **CalibIR-s(Ours)** | 2026 | 32.30 / 0.981 | 38.57 / 0.984 | 31.43 / 0.892 | 29.85 / 0.894 | 23.18 / 0.850 | 31.07 / 0.920 | 10M |
| **CalibIR(Ours)** | 2026 | **32.63 / 0.982** | 39.20 / **0.985** | 31.50 / **0.894** | **30.58 / 0.906** | **23.43 / 0.860** | **31.47 / 0.925** | 23M |

Table 3: Comparison on CDD11 setting. PSNR (↑) and SSIM (↑) are reported.

| Method | *CDD11-Single* | | | | | | | | *CDD11-Double* | | | | | | | | | | *CDD11-Triple* | | | | Avg. | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Low (L) | | Haze (H) | | Rain (R) | | Snow (S) | | L+H | | L+R | | L+S | | H+R | | H+S | | L+H+R | | L+H+S | | | |
| AirNet (Li et al., 2022) | 24.83 | .778 | 24.21 | .951 | 26.55 | .891 | 26.79 | .919 | 23.23 | .779 | 22.82 | .710 | 23.29 | .723 | 22.21 | .868 | 23.29 | .901 | 21.80 | .708 | 22.24 | .725 | 23.75 | .814 |
| PromptIR (Potlapalli et al., 2023) | 26.32 | .805 | 26.10 | .969 | 31.56 | .946 | 31.53 | .960 | 24.49 | .789 | 25.05 | .771 | 24.51 | .761 | 24.54 | .924 | 23.70 | .925 | 23.74 | .752 | 23.33 | .747 | 25.90 | .850 |
| WGWSNet (Zhu et al., 2023) | 24.39 | .774 | 27.90 | .982 | 33.15 | .964 | 34.43 | .973 | 24.27 | .800 | 25.06 | .772 | 24.60 | .765 | 27.23 | .955 | 27.65 | .960 | 23.90 | .772 | 23.97 | .771 | 26.96 | .863 |
| OneRestore (Guo et al., 2024) | 26.48 | .826 | 32.52 | .990 | 33.40 | .964 | 34.31 | .973 | 25.79 | .822 | 25.58 | .799 | 25.19 | .789 | 29.99 | .957 | 30.21 | .964 | 24.78 | .788 | 24.90 | .791 | 28.47 | .878 |
| MoCE-IR (Zamfir et al., 2025) | 27.26 | .824 | 32.66 | .990 | 34.31 | .970 | 35.91 | .980 | 26.24 | .817 | 26.25 | .800 | 26.04 | .793 | 29.93 | .964 | 30.19 | .970 | 25.41 | .789 | 25.39 | .790 | 29.05 | .881 |
| ExDA (Ren et al., 2026b) | 27.60 | .837 | 34.44 | .991 | 35.33 | .973 | 36.85 | .981 | 26.95 | .834 | 26.80 | .817 | 26.73 | .809 | 31.17 | .968 | 31.40 | .972 | 25.59 | .801 | 25.86 | .799 | 29.97 | .891 |
| **CalibIR-s(Ours)** | 27.61 | .837 | 34.83 | .992 | 35.53 | .975 | 37.26 | .982 | 27.00 | .835 | 26.74 | .816 | 26.74 | .810 | 31.77 | .971 | 32.24 | .975 | 26.18 | .811 | 26.36 | .811 | 30.21 | .892 |
| **CalibIR(Ours)** | **27.75** | **.840** | **36.56** | **.994** | **36.16** | **.978** | **38.02** | **.984** | **27.31** | **.838** | **26.87** | **.820** | **26.81** | **.814** | **33.12** | **.976** | **33.34** | **.979** | **26.38** | **.813** | **26.55** | **.814** | **30.81** | **.896** |

**AIO5.** With deblurring and low-light enhancement added to the joint training setting, CalibIR achieves 31.47 dB and 0.925 on average (Table 2), exceeding MoCE-IR (Zamfir et al., 2025) by 0.89 dB. It obtains the best PSNR on dehazing, deblurring, and low-light enhancement. In particular, it improves dehazing over ExDA by 1.49 dB and deblurring over MoCE-IR by 0.53 dB.

**CDD11.** CalibIR achieves the best PSNR and SSIM in all 11 degradation categories, with averages of 30.81 dB and 0.896 (Table 3). The gains extend to images affected by several degradations at once: CalibIR improves over ExDA by 1.95 and 1.94 dB for haze combined with rain and snow, respectively. For the two triple-degradation settings, low light with haze and rain, and low light with haze and snow, the gains are 0.79 and 0.69 dB.

Table 4: Comparisons for real-world WeatherBench. PSNR (↑) and SSIM (↑) are reported.

| Method | Venue | Dehaze | Derain | Desnow | Average | Params. |
|---|---|---|---|---|---|---|
| | | PSNR / SSIM | PSNR / SSIM | PSNR / SSIM | | |
| Restormer (Zamir et al., 2022) | CVPR'22 | 19.30 / 0.687 | 34.48 / 0.945 | 27.95 / 0.836 | 27.25 / 0.823 | 26M |
| AirNet (Li et al., 2022) | CVPR'22 | 20.94 / 0.705 | 33.59 / 0.942 | 22.05 / 0.780 | 25.53 / 0.809 | 9M |
| TransWeather (Valanarasu et al., 2022) | CVPR'22 | 19.79 / 0.680 | 29.34 / 0.903 | 24.96 / 0.796 | 24.70 / 0.793 | 22M |
| PromptIR (Potlapalli et al., 2023) | NeurIPS'23 | 21.11 / 0.713 | 34.53 / 0.944 | 27.93 / 0.836 | 27.86 / 0.831 | 36M |
| Histoformer (Sun et al., 2024) | ECCV'24 | 17.69 / 0.669 | 30.70 / 0.916 | 25.39 / 0.808 | 24.59 / 0.798 | 17M |
| AdaIR (Cui et al., 2025) | ICLR'25 | 23.08 / 0.731 | 34.87 / 0.946 | 28.44 / 0.837 | 28.80 / 0.838 | 29M |
| ExDA (Ren et al., 2026b) | ICLR'26 | 23.74 / 0.739 | 35.86 / 0.946 | 29.42 / 0.868 | 29.68 / 0.851 | 22M |
| **CalibIR-s(Ours)** | 2026 | 23.92 / 0.736 | 36.09 / 0.946 | 29.20 / 0.862 | 29.74 / 0.848 | 10M |
| **CalibIR(Ours)** | 2026 | **24.48 / 0.746** | **36.41 / 0.948** | **29.53 / 0.869** | **30.14 / 0.854** | 23M |

**WeatherBench.** On real-world weather degradations, CalibIR achieves the best results for all three tasks, reaching 30.14 dB and 0.854 on average (Table 4). It improves dehazing and deraining over ExDA by 0.74 and 0.55 dB, respectively. Figure 3 shows improved contrast around the staircase and railing, fewer residual snow particles on the brick wall, and reduced rain streaks with preserved car contours.

**Performance with a compact model.** CalibIR-s ranks second in average PSNR among the compared methods under all four settings, behind only CalibIR, while using 10.09M parameters. It exceeds ExDA by 0.19, 0.24, 0.24, and 0.06 dB on AIO3, AIO5, CDD11, and WeatherBench, respectively. On AIO3, it is only 0.07 dB below the full model in average PSNR and achieves the highest dehazing PSNR of 33.05 dB. On CDD11, it surpasses ExDA in 10 of the 11 degradation categories, including both triple-degradation settings. The compact model retains the same calibra-

tion module with reduced channel width and fewer refinement blocks, showing that the framework remains effective with a smaller backbone.

**Computational cost.** For $224 \times 224$ inputs, CalibIR-s requires 40G FLOPs, compared with 93G for CalibIR and 109G for ExDA. Relative to ExDA, CalibIR-s reduces parameter count by 55.0% and FLOPs by 63.3%. The full CalibIR model has a similar parameter count to ExDA (22.52M versus 22.42M) and uses 14.7% fewer FLOPs. Appendix A provides the model configurations and complete cost comparison.

**Discussion.** CalibIR's advantage over ExDA (Ren et al., 2026b) is larger under broader task mixtures and composite degradations: the average PSNR gain rises from 0.26 dB on AIO3 to 0.64 dB on AIO5 and 0.84 dB on CDD11, and remains 0.46 dB on real-world WeatherBench. From AIO3 to AIO5, the shared tasks retain similar performance: dehazing improves by 0.06 dB, while deraining and denoising at $\sigma = 25$ decrease by 0.15 and 0.08 dB, respectively (Tables 1 and 2). CalibIR thus accommodates deblurring and low-light enhancement while largely preserving its performance on the original three tasks.

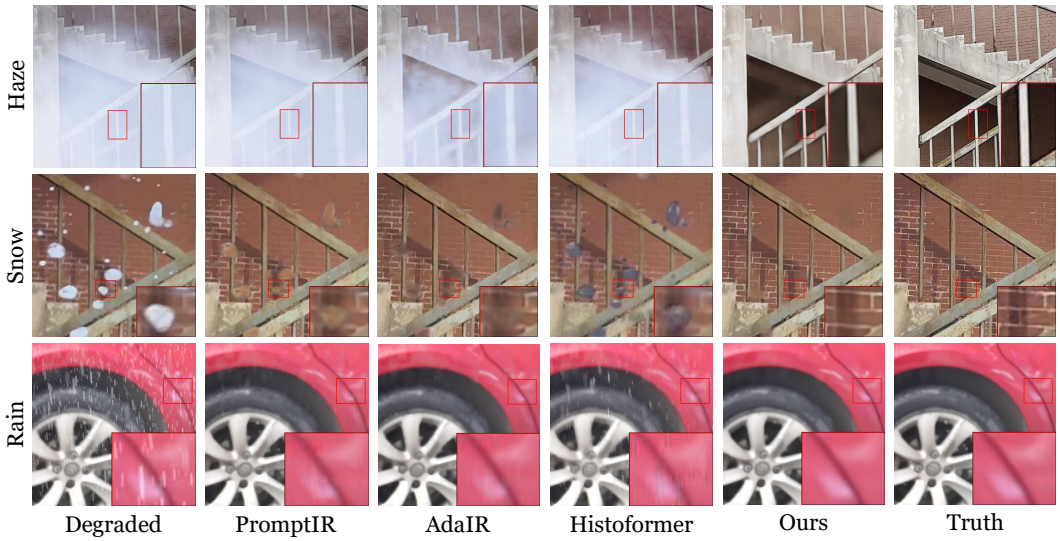



Figure 3: Visual comparisons on real-world WeatherBench. Rows show dehazing, desnowing, and deraining. The enlarged regions illustrate residual haze, snow artifacts, and rain streaks.

### 4.3 Ablation Study

We examine the contributions of the two corrections, the role of spatial centering, and how the corrections are generated. Table 5 reports the corresponding variants under the AIO3 setting.

**Operator components.** The static variant uses only the learned depthwise operator $W_0$, whose weights remain fixed across input images. Replacing Restormer's channel attention (MDTA) (Zamir et al., 2022) with the static depthwise operator $W_0$ raises PSNR from 31.85 to 32.54 dB. Adding the channel correction ($W_0 + A$) reaches 32.97 dB, while adding the spatial correction ($W_0 + B$) reaches 32.92 dB. Combining both corrections yields 33.22 dB, improving over the two individual variants by 0.25 and 0.30 dB. Both corrections therefore improve the same static operator, with a combined gain of 0.68 dB. The channel-only variant already adapts its kernels to each input image, but shares them across spatial positions within each channel. Its gap to the full model indicates that preserving regional variation provides benefits beyond image-level calibration. Conversely, the spatial-only variant applies the same correction across channels. Adding $A$ allows the update to vary across channels while retaining the position-dependent correction, supporting the complementary roles of the two branches.

**Spatial centering.** The uncentered variant retains both corrections and their learned step sizes, but uses the channel-averaged update evidence directly for $B$ without subtracting its spatial mean. It yields 33.08 dB and 0.921 SSIM, compared with 33.22 dB and 0.922 for the full model. Centering removes a spatially constant component from $B$ that can already be represented by $A$. The 0.14 dB improvement supports this separation, allowing $B$ to describe regional deviations alongside the image-wide channel correction.

Table 5: Ablation of CalibIR on AIO3. Best results are in **bold**.

| Variant | PSNR (dB) ↑ | SSIM ↑ |
|---|---|---|
| Attention (MDTA) | 31.85 | 0.907 |
| Static operator | 32.54 | 0.916 |
| Channel correction only | 32.97 | 0.919 |
| Spatial correction only | 32.92 | 0.918 |
| Direct prediction | 33.01 | 0.920 |
| Without spatial centering | 33.08 | 0.921 |
| **CalibIR (Full)** | **33.22** | **0.922** |

**Correction generation.** The direct prediction variant retains the form $W_0 + A + B$ and spatial centering, but uses two feed-forward heads to predict the corrections from $K$ and $V$. One head uses spatially pooled features for channel correction, while the other predicts a correction at each position. This variant preserves input-dependent channel and spatial adaptation while changing how the coefficients are obtained. It achieves 33.01 dB, improving over the static operator by 0.47 dB. CalibIR further improves PSNR by 0.21 dB and SSIM from 0.920 to 0.922. Input-dependent coefficients help, and calibration from the operator's prediction error provides further gains.

### 4.4 Analysis of Operator Calibration

We examine how operator calibration responds to image content and degradation conditions. All measurements use the AIO3 model and the first CalibIR block at the third encoder scale.

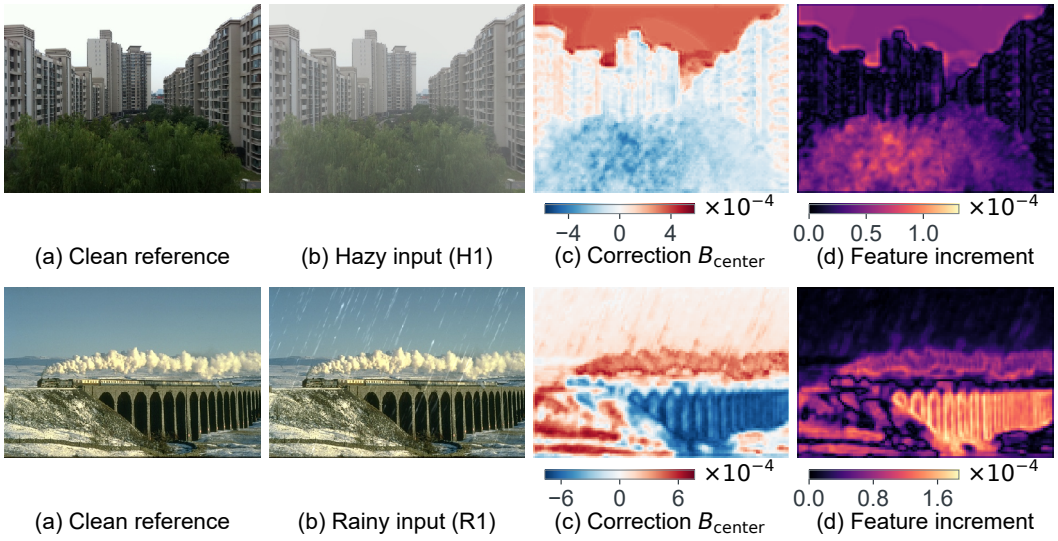



Figure 4: Spatial corrections and feature contributions for dehazing (top) and deraining (bottom). Columns show the clean reference, degraded input, center coefficient of $B$, and feature-increment magnitude $C^{-1}\sum_c \left|\sum_{o\in\Omega} B_{p,o} Q_c(p+o)\right|$ before the output projection. Corrections include learned step sizes. Each task uses its own color scales.

**Spatial corrections in feature processing.** Figure 4 shows the spatial correction $B$ and the feature increment it produces. In the hazy scene, the sky and trees receive different coefficient corrections from the building facades. In the rainy scene, the correction map distinguishes the bridge arches from the surrounding snow and sky. These regional patterns remain visible after the correction is applied to $Q$, combining all nine kernel offsets. The resulting feature increments are larger in the tree canopy than on the building facades, and in the bridge region than in the sky. Spatial calibration thus produces regional differences in both operator coefficients and feature increments.

We further keep three regions in one natural image fixed and vary the spatially uniform Gaussian noise level over $\sigma = 15, 25, 50$ (Appendix C). Relative to $\sigma = 15$, the penguin body and background receive increasing center-coefficient corrections, while the rock region changes much less. The same content regions therefore follow different correction trajectories as the input noise changes.

**Channel corrections across inputs.** We next examine how the spatially shared channel correction $A$ responds to different observations of the same scene. Figure 5 shows its center coefficients for two rainy inputs with the same clean reference. The overall distribution is similar, while individual channels undergo different adjustments. For example, the coefficient in channel 49 decreases from $0.0584$ to $0.0089$, whereas that in channel 191 increases from $-0.0067$ to $0.0249$. The sign reversal in channel 191 further shows that calibration adjusts both the magnitude and direction of coefficient updates. The difference map shows increases and decreases across channels. Together with Figure 4, these observations show how the same module adjusts channel-specific operators across inputs while preserving regional variation within each image.

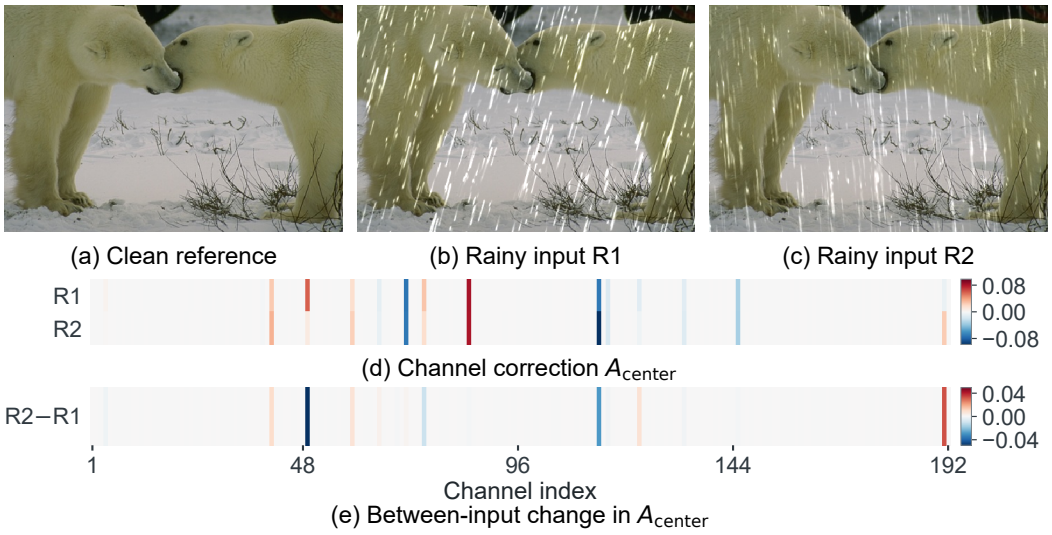



Figure 5: Channel corrections for two rain observations of the same scene. (a)–(c) Clean reference and Rain100L inputs R1 and R2. (d) Center coefficients of $A$, including learned step sizes. (e) R2 minus R1. Channels retain the same order in both maps, which use separate color scales.

## 5 Conclusion

We presented CalibIR, an image-internal operator calibration framework for all-in-one image restoration. CalibIR adapts learned base operators through a feature-prediction task on the current image. It derives channel and spatial corrections from the same prediction error, combining image-wide information with regional adjustments in a single forward pass. Experiments across four restoration settings demonstrate strong performance with both the full and compact models. Component ablations support the calibration design, and operator analyses show how the corrections respond to image content and degradation conditions.

## Reproducibility Statement

Section 3 specifies the operator calibration objective and the channel and spatial update equations. Appendix A.1 describes the model configurations and initialization, while Appendix B provides the dataset protocols and training settings, including the loss definitions in Appendix B.3. The comparison variants are described in Section 4.3, and the operator analyses are detailed in Section 4.4 and Appendix C. We will provide the model code as supplementary material.

## REFERENCES


Pablo Arbeláez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):898–916, 2011.

Jimmy Ba, Geoffrey E. Hinton, Volodymyr Mnih, Joel Z. Leibo, and Catalin Ionescu. Using fast weights to attend to the recent past. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 29, 2016.

Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *European Conference on Computer Vision (ECCV)*, pp. 17–33, 2022.

Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. Dynamic convolution: Attention over convolution kernels. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11030–11039, 2020.

Marcos V. Conde, Gregor Geigle, and Radu Timofte. InstructIR: High-quality image restoration following human instructions. In *European Conference on Computer Vision (ECCV)*, 2024.

Yuning Cui, Syed Waqas Zamir, Salman Khan, Alois Knoll, Mubarak Shah, and Fahad Khan. AdaIR: Adaptive all-in-one image restoration via frequency mining and modulation. In *International Conference on Learning Representations (ICLR)*, 2025.

Daniel Glasner, Shai Bagon, and Michal Irani. Super-resolution from a single image. In *International Conference on Computer Vision (ICCV)*, 2009.

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 *ACM International Conference on Multimedia (ACM MM)*, pp. 12607–12613, 2025.

Yu Guo, Yuan Gao, Yuxu Lu, Huilin Zhu, Ryan Wen Liu, and Shengfeng He. OneRestore: A universal restoration framework for composite degradation. In *European Conference on Computer Vision (ECCV)*, pp. 255–272, 2024.

Dongchen Han, Yining Li, Tianyu Li, Zixuan Cao, Ziming Wang, Jun Song, Yu Cheng, Bo Zheng, and Gao Huang. ViT$^3$: Unlocking test-time training in vision. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 51–61, 2026.

Jiakui Hu, Lujia Jin, Zhengjian Yao, and Yanye Lu. Universal image restoration pre-training via degradation classification. In *International Conference on Learning Representations (ICLR)*, 2025.

Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc Van Gool. Dynamic filter networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 29, 2016.

Boyi Li, Wenqi Ren, Dengpan Fu, Dacheng Tao, Dan Feng, Wenjun Zeng, and Zhangyang Wang. Benchmarking single-image dehazing and beyond. *IEEE Transactions on Image Processing*, 28 (1):492–505, 2019.

Boyun Li, Xiao Liu, Peng Hu, Zhongqin Wu, Jiancheng Lv, and Xi Peng. All-in-one image restoration for unknown corruption. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17452–17462, 2022.

Ruoteng Li, Robby T. Tan, and Loong-Fah Cheong. All in one bad weather removal using architectural search. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3175–3185, 2020.

Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. SwinIR: Image restoration using swin transformer. In *International Conference on Computer Vision Workshops (ICCVW)*, pp. 1833–1844, 2021.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

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.

Kede Ma, Zhengfang Duanmu, Qingbo Wu, Zhou Wang, Hongwei Yong, Hongliang Li, and Lei Zhang. Waterloo exploration database: New challenges for image quality assessment models. *IEEE Transactions on Image Processing*, 26(2):1004–1016, 2017. doi: 10.1109/TIP.2016.2631888.

David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *International Conference on Computer Vision (ICCV)*, volume 2, pp. 416–423, 2001.

Ben Mildenhall, Jonathan T. Barron, Jiawen Chen, Dillon Sharlet, Ren Ng, and Robert Carroll. Burst denoising with kernel prediction networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2502–2510, 2018.

Chong Mou, Qian Wang, and Jian Zhang. Deep generalized unfolding networks for image restoration. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17399–17410, 2022.

Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3883–3891, 2017.

Dongwon Park, Byung Hyun Lee, and Se Young Chun. All-in-one image restoration for unknown degradations using adaptive discriminative filters for specific degradations. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5815–5824, 2023.

Vaishnav Potlapalli, Syed Waqas Zamir, Salman H. Khan, and Fahad Shahbaz Khan. PromptIR: Prompting for all-in-one image restoration. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023.

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. In *International Conference on Learning Representations (ICLR)*, 2026a.

Bin Ren, Runyi Yang, Qi Ma, Xu Zheng, Mengyuan Liu, Danda Pani Paudel, Luc Van Gool, Rita Cucchiara, and Nicu Sebe. Rethinking expressivity and degradation-awareness in attention for all-in-one blind image restoration. In *International Conference on Learning Representations (ICLR)*, 2026b.

Imanol Schlag, Kazuki Irie, and Jürgen Schmidhuber. Linear transformers are secretly fast weight programmers. In *International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9355–9366. PMLR, 2021.

Assaf Shocher, Nadav Cohen, and Michal Irani. "zero-shot" super-resolution using deep internal learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3118–3126, 2018.

Hang Su, Varun Jampani, Deqing Sun, Orazio Gallo, Erik Learned-Miller, and Jan Kautz. Pixel-adaptive convolutional neural networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11166–11175, 2019.

Shangquan Sun, Wenqi Ren, Xinwei Gao, Rui Wang, and Xiaochun Cao. Restoring images in adverse weather conditions via histogram transformer. In *European Conference on Computer Vision (ECCV)*, pp. 111–129, 2024.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9229–9248, 2020.

Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, Tatsunori Hashimoto, and Carlos Guestrin. Learning to (Learn at test time): RNNs with expressive hidden states. In *International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 57503–57522. PMLR, 2025.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9446–9454, 2018.

Jeya Maria Jose Valanarasu, Rajeev Yasarla, and Vishal M. Patel. TransWeather: Transformer-based restoration of images degraded by adverse weather conditions. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2353–2363, 2022.

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(10): 4541–4563, 2024.

Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general U-shaped transformer for image restoration. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17683–17693, 2022.

Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. In *British Machine Vision Conference (BMVC)*, 2018.

Gang Wu, Junjun Jiang, Kui Jiang, and Xianming Liu. Harmony in diversity: Improving all-in-one image restoration via multi-task collaboration. In *ACM International Conference on Multimedia (ACM MM)*, pp. 6015–6023, 2024.

Gang Wu, Junjun Jiang, Yijun Wang, Kui Jiang, and Xianming Liu. Debiased all-in-one image restoration with task uncertainty regularization. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 39, pp. 8386–8394, 2025.

Brandon Yang, Gabriel Bender, Quoc V. Le, and Jiquan Ngiam. CondConv: Conditionally parameterized convolutions for efficient inference. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Wenhan Yang, Robby T. Tan, Jiashi Feng, Jiaying Liu, Zongming Guo, and Shuicheng Yan. Deep joint rain detection and removal from a single image. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1357–1366, 2017.

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 *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12753–12763, 2025.

Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14821–14831, 2021.

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 *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5728–5739, 2022.

Haijin Zeng, Xiangming Wang, Yongyong Chen, Jingyong Su, and Jie Liu. Vision-language gradient descent-driven all-in-one deep unfolding networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7524–7533, 2025.

Jinghao Zhang, Jie Huang, Mingde Yao, Zizheng Yang, Hu Yu, Man Zhou, and Feng Zhao. Ingredient-oriented multi-degradation learning for image restoration. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5825–5835, 2023.

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.

Kai Zhang, Wangmeng Zuo, and Lei Zhang. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *IEEE Transactions on Image Processing*, 2018. doi: 10.1109/TIP.2018.2839891.

Xu Zhang, Huan Zhang, Guoli Wang, Qian Zhang, and Lefei Zhang. ClearAIR: A human-visual-perception-inspired all-in-one image restoration. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2026.

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 *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21747–21758, 2023.

Maria Zontak and Michal Irani. Internal statistics of a single natural image. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

# A Model Configurations and Computational Cost

## A.1 Model Configurations

Table 6 details the two model sizes. CalibIR-s reduces the input embedding width from 48 to 32 and the number of refinement blocks from four to three. Both models use the same numbers of blocks at each encoder and decoder stage, the same FFN expansion factor, and the same operator calibration module.

Table 6: Configurations of CalibIR and CalibIR-s. Encoder lists include the bottleneck and run from full to $1/8$ resolution; decoder lists run from $1/4$ to full resolution. FLOPs are reported for $224 \times 224$ inputs.

| Configuration | CalibIR | CalibIR-s |
|---|---|---|
| Input embedding width | 48 | 32 |
| Encoder channels | $[48, 96, 192, 384]$ | $[32, 64, 128, 256]$ |
| Encoder blocks | $[4, 6, 6, 8]$ | $[4, 6, 6, 8]$ |
| Decoder channels | $[192, 96, 96]$ | $[128, 64, 64]$ |
| Decoder blocks | $[6, 6, 4]$ | $[6, 6, 4]$ |
| Refinement channels | 96 | 64 |
| Refinement blocks | 4 | 3 |
| FFN expansion factor | 2 | 2 |
| Calibration kernel size | $3 \times 3$ | $3 \times 3$ |
| Calibration steps per block | 1 | 1 |
| Learned step sizes per block | 2 | 2 |
| Parameters (M) | 22.52 | 10.09 |
| FLOPs (G) | 93 | 40 |

The backbone has four spatial scales, with the bottleneck operating at $1/8$ of the input resolution in each dimension. A $3 \times 3$ convolution embeds the RGB input. Downsampling uses a $3 \times 3$ convolution followed by pixel unshuffle, halving the spatial resolution and doubling the channel width. Upsampling uses a $3 \times 3$ convolution followed by pixel shuffle. At the $1/4$ and $1/2$ decoder scales, upsampled features are concatenated with the corresponding encoder features and reduced in width by a $1 \times 1$ projection. At full resolution, the concatenated features retain twice the input embedding width and pass through the final decoder and refinement blocks. A $3 \times 3$ output convolution predicts an RGB residual that is added to the input image.

Both models use layer normalization with learnable scale and bias, and omit biases in convolutional layers. Each calibration module initializes $W_0$ as an identity depthwise kernel. Its two positive step sizes are parameterized by softplus, with initial unconstrained values of $-5$ for $\eta_{\mathrm{ch}}$ and $-8$ for $\eta_{\mathrm{sp}}$. The channel and centered spatial corrections are computed together in one calibration step, as described in Section 3.2.

## A.2 Computational Cost

We compare parameter counts and FLOPs for $224 \times 224$ inputs, profiling all methods on the same machine equipped with an NVIDIA RTX 4090 GPU. Table 7 shows that CalibIR uses 22.52M parameters and 93G FLOPs, requiring 14.7% fewer FLOPs than ExDA at a similar parameter count. CalibIR-s uses 10.09M parameters and 40G FLOPs, the lowest parameter count and computational cost among the compared models.

Table 7: Parameter counts and FLOPs for $224 \times 224$ inputs. The lowest values are in **bold**.

| Method | Parameters (M) ↓ | FLOPs (G) ↓ |
|---|---|---|
| PromptIR (Potlapalli et al., 2023) | 35.59 | 132 |
| AdaIR (Cui et al., 2025) | 28.78 | 124 |
| MoCE-IR (Zamfir et al., 2025) | 25.35 | 81 |
| ExDA (Ren et al., 2026b) | 22.42 | 109 |
| CalibIR-s (Ours) | **10.09** | **40** |
| CalibIR (Ours) | 22.52 | 93 |

# B Experimental Details

## B.1 Datasets

**AIO3 and AIO5.** AIO3 jointly trains Gaussian denoising, deraining, and dehazing. For denoising, we train on BSD400 (Arbeláez et al., 2011) and WED (Ma et al., 2017) with Gaussian noise of standard deviation $\sigma \in \{15, 25, 50\}$ and test on BSD68 (Martin et al., 2001). Deraining uses Rain100L (Yang et al., 2017), and dehazing uses RESIDE (Li et al., 2019) with indoor SOTS for testing. AIO5 adds motion deblurring on GoPro (Nah et al., 2017) and low-light enhancement on LOL-v1 (Wei et al., 2018). Denoising is evaluated at all three noise levels in AIO3 and at $\sigma = 25$ in AIO5.

**CDD11 and WeatherBench.** CDD11 (Guo et al., 2024) comprises 11 categories of single and composite degradations involving low light, haze, rain, and snow. WeatherBench (Guan et al., 2025) provides paired real-world images affected by haze, rain, or snow. We train one unified model for each evaluation setting.

## B.2 Training Configuration

We train CalibIR end-to-end in PyTorch for 150 epochs on four NVIDIA L40S GPUs, using AdamW (Loshchilov & Hutter, 2019) with a base learning rate of $2 \times 10^{-4}$ and a batch size of 12 per GPU. Training uses $128 \times 128$ crops with random flips and rotations by multiples of $90^\circ$.

## B.3 Optimization Objectives

The overall training objective combines pixel and Fourier reconstruction losses:

$$\mathcal{L}_{\text{train}} = \mathcal{L}_{\text{pix}} + \lambda_{\text{fft}} \mathcal{L}_{\text{fft}}. \tag{9}$$

Given restored images $\widehat{I}$ and their clean targets $I_{\text{gt}}$, we use an $\ell_1$ loss to enforce pixel-wise similarity:

$$\mathcal{L}_{\text{pix}} = \frac{1}{M} \left\| \widehat{I} - I_{\text{gt}} \right\|_1, \tag{10}$$

where $M$ counts all image entries in the batch and $\| \cdot \|_1$ denotes the sum of absolute values.

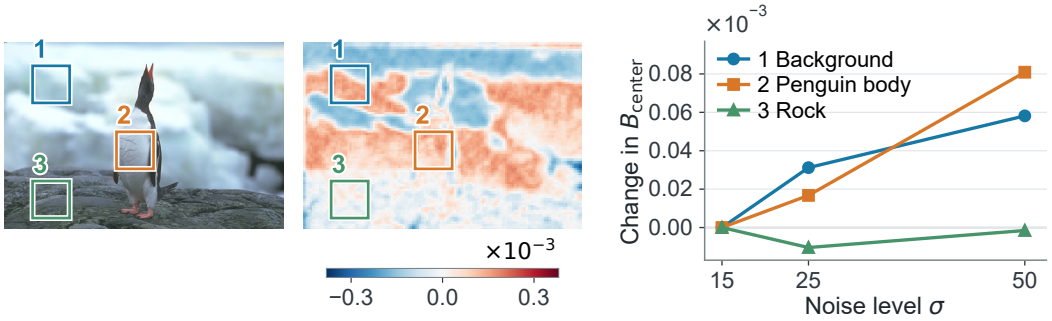
Following prior work (Zamfir et al., 2025; Ren et al., 2026b), we further adopt a Fourier loss to encourage frequency-domain consistency. Let $\mathcal{F}_{\text{re}}$ and $\mathcal{F}_{\text{im}}$ denote the real and imaginary components of the two-dimensional real-input Fourier transform. The loss is defined as

$$\begin{aligned} \mathcal{L}_{\text{fft}} = \frac{1}{2M_f} \Big[ & \left\| \mathcal{F}_{\text{re}}(\widehat{I}) - \mathcal{F}_{\text{re}}(I_{\text{gt}}) \right\|_1 \\ & + \left\| \mathcal{F}_{\text{im}}(\widehat{I}) - \mathcal{F}_{\text{im}}(I_{\text{gt}}) \right\|_1 \Big], \end{aligned} \tag{11}$$

where $M_f$ counts the retained complex Fourier coefficients in the batch, so the loss averages over both spectral components. The transform is computed with `rfft2` without forward normalization. We set $\lambda_{\text{fft}} = 0.1$ throughout our experiments.

## C Regional Responses to Degradation Conditions

We keep the image and observation regions fixed and vary the spatially uniform Gaussian noise level over $\sigma = 15, 25, 50$. As in Section 4.4, we use the AIO3 model and the first CalibIR block at the third encoder scale. Figure 6 plots changes in the mean center coefficient of $B$ relative to $\sigma = 15$. The penguin body and background both receive increasing corrections relative to $\sigma = 15$. The background changes more at $\sigma = 25$, whereas the body region shows the larger change from $\sigma = 15$ to $\sigma = 50$. The rock region varies much less across the three noise levels. The spatial difference map further locates these changes within the image. This example shows that changing the global noise condition adjusts the regional correction pattern, with different responses across locations in the same scene.




(a) Clean reference (b) Spatial change (c) Regional change

Figure 6: Regional changes under spatially uniform Gaussian noise. (a) The clean reference marks three fixed $64 \times 64$ regions. (b) The spatial difference in the center coefficient of $B$ between $\sigma = 50$ and $\sigma = 15$. (c) Region-averaged coefficient changes relative to $\sigma = 15$. The model processes the full noisy image at each noise level.

## D AI Use Statement

We used OpenAI's GPT-based Large Language Models the writing and improve the readability of the paper. The models were not used for developing the methodology, running experiments, or analyzing results. All scientific contributions remain entirely the work of the authors.

## E Ethics Statement

CalibIR studies general-purpose image restoration and may benefit photography and visual perception under adverse imaging conditions. Our experiments use established public benchmarks and involve no human-subject studies or new personal-data collection. Restored images may contain inaccurate details, so applications should account for restoration errors and respect the privacy of people depicted in the images.

# F Additional Visual Comparisons

## F.1 AIO3

Figure 7 presents visual comparisons on AIO3. CalibIR suppresses noise and rain streaks while preserving details in the building reflections and clothing, and restores contrast in the hazy street scene.




Figure 7: Visual comparisons on AIO3 for denoising, deraining, and dehazing, from top to bottom. Red boxes mark the regions shown in the enlarged insets.

## F.2 AIO5

Figure 8 illustrates restoration across the five tasks, including clearer vehicle boundaries in the blurred scene and recovered fabric structure in the low-light example.




Figure 8: Visual comparisons on AIO5. Rows show denoising, deraining, dehazing, deblurring, and low-light enhancement. Enlarged regions highlight local restoration details.

### F.3 CDD11

Figure 9 shows examples of double and triple degradations. CalibIR removes weather artifacts while recovering local contrast and detail, as seen in the roof contours and mountain textures.

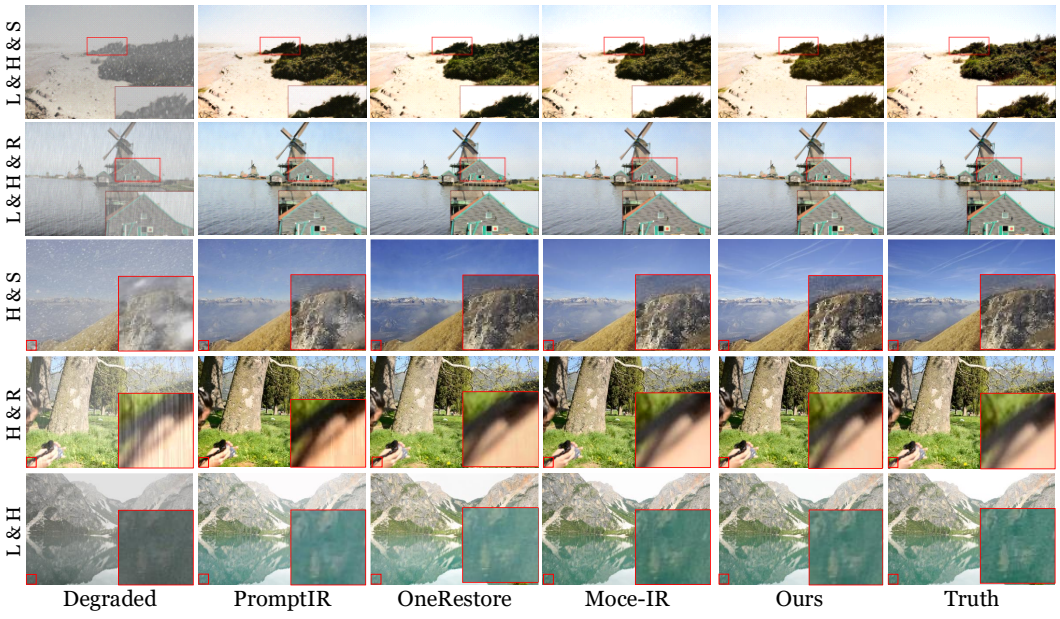



Figure 9: Visual comparisons on CDD11 under double and triple degradations. L, H, R, and S denote low light, haze, rain, and snow. The first two rows show triple degradations, followed by three double-degradation examples.