Restore Anything with Masks: Leveraging Mask Image Modeling for Blind All-in-One Image Restoration

Chu-Jie Qin[1,2*], Rui-Qi Wu[1,2], Zikun Liu[3], Xin Lin[5], Chun-Le Guo[1,2], Hyun Hee Park[4], and Chongyi Li[1,2†]

[1] VCIP, CS, Nankai University
[2] NKIARI, Shenzhen Futian
{chujie.qin,wuruiqi}@mail.nankai.edu.cn
{guochunle, lichongyi}@nankai.edu.cn
[3] Samsung Research, China, Beijing (SRC-B)
[4] The Department of Camera Innovation Group, Samsung Electronics
{zikun.liu,inextg.park}@samsung.com
[5] Sichuan University
linxin@stu.scu.edu.cn

Abstract. All-in-one image restoration aims to handle multiple degradation types using one model. This paper proposes a simple pipeline for all-in-one blind image restoration to **R**estore **A**nything with **M**asks (**RAM**). We focus on the image content by utilizing Mask Image Modeling to extract intrinsic image information rather than distinguishing degradation types like other methods. Our pipeline consists of two stages: masked image pre-training and fine-tuning with mask attribute conductance. We design a straightforward masking pre-training approach specifically tailored for all-in-one image restoration. This approach enhances networks to prioritize the extraction of image content priors from various degradations, resulting in a more balanced performance across different restoration tasks and achieving stronger overall results. To bridge the gap of input integrity while preserving learned image priors as much as possible, we selectively fine-tuned a small portion of the layers. Specifically, the importance of each layer is ranked by the proposed Mask Attribute Conductance (**MAC**), and the layers with higher contributions are selected for finetuning. Extensive experiments demonstrate that our method achieves state-of-the-art performance. Our code and model will be released at https://github.com/Dragonisss/RAM.

Keywords: Image Restoration · All-in-One · Mask Image Modeling

1 Introduction

Image restoration involves the restoration of low-quality images affected by various degradation, typically arising from adverse environmental conditions (*e.g.*,

*A part of this work is done during Chu-Jie Qin's internship at Samsung.
†Chongyi Li is the corresponding author.

rain, haze, low-light), hardware-related issues (*e.g.*, noise and blur), and post-processing artifacts (*e.g.*, JPEG compression). Image restoration serves not only to enhance the visual appeal of images but also contributes to practical application scenarios such as autonomous driving and surveillance.

Modern techniques in this field mainly focus on learning fixed patterns formed during the degradation process, *i.e.*, degradation priors. Some works [29, 30, 60] utilize task-specific priors to solve a certain degradation problem, while another research line [3, 28, 39, 48, 55] tries to design a general network architecture that can effectively learn each degradation pattern. Nevertheless, the above methods only enable the network to learn a single degradation, resulting in an imbalanced situation when dealing with multiple types of degradation.

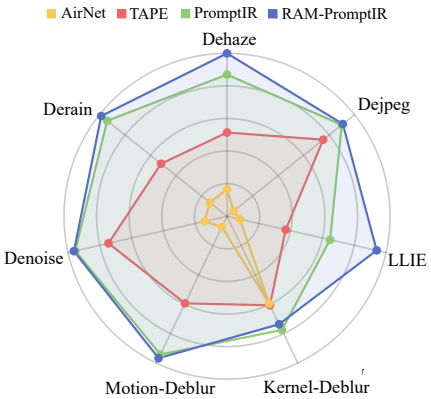


Fig. 1: Our RAM achieves more balanced and more powerful performance than the state-of-the-art methods (AirNet [21], TAPE [31], PromptIR [42]) for all-in-one blind image restoration.

To tackle the problem stated above, all-in-one methods have emerged, aiming to handle multiple degradations using one model. Most of these approaches tend to utilize explicit priors (*e.g.*, AirNet [21]) or introduce an extra module (*e.g.*, PromptIR [42]) to discern image degradation patterns, thereby assisting the model in performing the restoration. However, these methods place their emphasis on distinguishing degradation types in images rather than the image content, leading to lower scalability and fuzzy decision boundaries when more degradation types are involved. We argue that the essence of image restoration is to extract intrinsic image information from corrupted images rather than eliminate degradation patterns, *i.e.*, learning image prior rather than degradation prior. It is worth noting that TAPE [31] similarly suggests that understanding normal image nature aids restoration by introducing a natural image prior. Nevertheless, TAPE utilizes the model output as the optimization target, which causes the model to amplify its own errors and learn the image prior with bias.

In this paper, we focus on tackling **how to extract intrinsic image information from diverse corrupted images.** Some attempts [2, 7] by Mask Image Modeling (MIM) in low-level vision have caught our attention. As a pre-training strategy, MIM has been widely validated for its effectiveness in high-level tasks, thanks to its generic representation of images. Simultaneously, the model also learns the distribution of natural images, which encompasses the intrinsic information we aim to extract from the images. Built on MIM, we propose a simple pipeline for all-in-one blind image restoration that **R**estores **A**nything with **M**asks (**RAM**), which includes two stages: the mask pre-training stage and the

fine-tuning stage with Mask Attribute Conductance (MAC). In the pre-training stage, we randomly mask corrupted images at the pixel-wise level and force the network to predict the clear one corresponding to the masked pixels, extracting inherent image information from corrupted images. In the fine-tuning stage, we focus on overcoming the input integrity gap caused by changing masked input during pre-training into the whole image during inference while preserving learned prior as much as possible.

Specifically, we first evaluated the importance of each network layer in addressing this gap by the proposed MAC. Following that, we chose the top $k\%$ most critical layers for fine-tuning while keeping the rest of the network layers frozen. We demonstrate that after a brief fine-tuning period (even if only 10% layers are tuned), the model can achieve a highly satisfactory performance level, surpassing models trained using traditional pair-wise training. Additionally, our pipeline can be plug-and-play used in any network without introducing additional computational overhead.

The contributions of this work are as follows:

- We discuss the challenge of adopting MIM in low-level vision and propose a MIM-based pre-training strategy tailored to all-in-one blind image restoration, which allows the restoration networks to effectively learn inherent image information while guaranteeing reconstruction results.
- We proposed Mask Attribute Conductance to evaluate the importance of each layer in addressing the input integrity gap so that a very small portion (*e.g.* 10%) of critical layers are tuned to bridge this gap while preserving the image prior learned by MIM.
- Our proposed RAM provides a fresh perspective to achieve more balanced and powerful all-in-one blind image restoration, which focuses on extracting inherent image information from corrupted images. Our pipeline can be applied to any image restoration network without introducing additional computational overhead.

2 Related Work

2.1 Image Restoration for Multi Degradations

While neural networks have demonstrated impressive performance in single degradation image restoration [9, 13, 14, 17, 22, 23, 29, 30, 49, 60], recent works have shifted their focus towards addressing the more challenging domain of multi-degradation image restoration. A group of methods [3, 28, 39, 48, 55] aims at designing a general architecture that can effectively learn each degradation pattern. SwinIR [28] employs a window attention mechanism to convert global attention into a localized approach, effectively reducing computational overhead. In addition, the U-shaped transformer-based methods [48, 55] are employed to extract multi-scale features and reduce computational overhead. However, these methods have to train individually on each restoration task. Several methods [1, 24] leverage multiple input and output heads to empower the network to restore

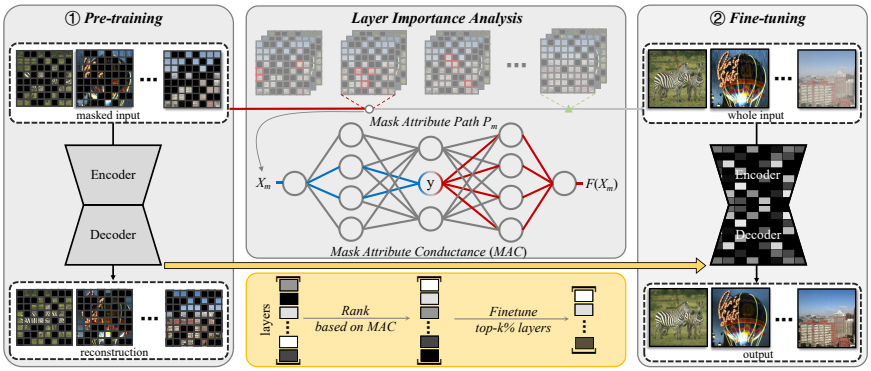


Fig. 2: The illumination of our overall pipeline. 1) Pre-training the model with mask image pre-training method tailored to low-level vision. We randomly mask degraded images at the pixel level with a 50% masking ratio and reconstruct the clean images. 2) The Fine-tuning stage is followed to overcome the input integrity gap caused by changing masked input during pre-training into the whole image during inference. We analyze the importance of each network layer for resolving the input integrity gap according to the proposed MAC and rank them in descending order. The top $k\%$ of network layers are selected for fine-tuning on the complete image.

various types of degraded images. Nonetheless, this kind of approach may lead to the diminished scalability of the model. Recently, several subsequent methods [4, 21, 36, 41, 42, 56, 61] have been proposed to employ a unified network to address multiple restoration issues. Most of these methods put emphasis on learning how to distinguish different types of degradations and restore corrupted images. Typically, AirNet [21] first proposed an all-in-one image restoration task. The method initially pretrains a degradation classifier based on contrastive learning and subsequently utilizes it to assist in all-in-one image restoration. PromptIR [42] has introduced a learnable prompt-based module. Instead of constraining the degradation category, it enables the model to autonomously learn features that are advantageous to its performance by using an adaptive prompt. Our RAM takes a fresh perspective that focuses on extracting common content information from corrupted images, without any extra design to distinguish degradations, which helps us achieve balance and powerful performance when more degradation types are taken into consideration.

2.2 Mask Image Modeling

Inspired by Mask Language Modeling [18,43], Mask Image Modeling (MIM) [15, 51] is introduced as a pretraining approach to learn general representations in high-level vision. MAE [15] effectively utilizes MIM for predicting hidden tokens, demonstrating strong performance and generalization across various downstream tasks. SimMIM [51] proposed a general masked image modeling method based on Swin-ViT [34]. Painter [47] unifies multiple tasks under image-to-image trans-

lation and leverages MIM pretraining. In recent years, there have been efforts to incorporate MIM into the realm of low-level vision to enhance model generalization. Among them, [2] and [7] are the most closely aligned with our focus. [2] employs the MIM model to enhance the model's generalization for denoising tasks but has not explored its potential in multi-task scenarios. [7] utilizes MIM for pre-training the model encoder to introduce generative prior and subsequently employs the decoder for restoration. However, it does not fully harness the potential of MIM. Our proposed RAM utilizes MIM to unify the optimization objective for various image restoration tasks into reconstructing intrinsic image information. This allows the network to learn restoration functions more balanced and effectively. Moreover, to preserve the image priors learned by MIM, we designed a fine-tuning strategy based on MAC analysis (in Sec. 3.3). This enables us to achieve comparable performance by fine-tuning only a small portion (*e.g.* 10%) of layers, fully tapping into the potential of MIM.

2.3 Gradient-based Attribution

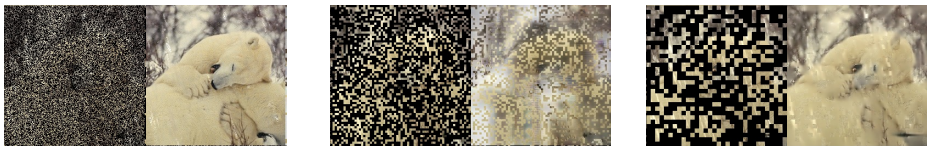
Gradient-based attribution methods [6,12,44–46,50] are often used to clarify how hidden units (or inputs) impact the output of networks. One commonly used approach is Integrated Gradients (IG) [45,46], which accumulates gradients along a linear path from the baseline input to the target input in the pixel/feature space. After that, IntInf [19] and layer conductance [6] alter IG to attribute neuron importance along the same path. In our work, we expect to find the key layers that can effectively overcome the distribution shift between training data and inference data. We propose Mask Attribute Conductance (MAC) based on the layer conductance and accumulated MAC of each layer along the Mask Attribute Path (MAP). MAC can represent the layer's importance along the MAP. In this way, we can fine-tune the top $k\%$ critical layers of the pre-trained network, preserving to a great extent the image priors learned during pretraining.

3 Methodology

In this section, we start with discussing the challenges of using MIM in low-level vision tasks (Sec. 3.1). Following that, we present our pipeline for all-in-one blind image restoration, which contains two parts: pre-training with MIM (Sec. 3.2) and fine-tuning with Mask Attribute Conductance (MAC) Analysis (Sec. 3.3).

3.1 Rethinking MIM in Low-Level Vision

MIM is a process that randomly masks certain parts of an image and extracts features from the remaining visible parts to reconstruct the entire image. It allows models to acquire a generic representation of images and thus achieve good pre-training, which is verified in many high-level tasks [15, 51]. Moreover, the models also learn the distribution of natural images during the image reconstruction, *i.e.* MIM pre-training. This incidental acquisition of prior knowledge



patch size=1 patch size=4 patch size=8

Fig. 3: Mask Image Modeling reconstruction with different patch sizes. We pre-trained with different patch sizes and visualized the mask inputs (left), and the corresponding MIM reconstructions (right).

is instrumental in tasks like image restoration. Despite these advantages, applying MIM in pretraining a model for low-level vision tasks is still under-explored, primarily due to the challenges that must be addressed in the process.

Firstly, the main purpose of vanilla MIM is not high-quality reconstruction but good feature extraction for high-level tasks. Therefore, it masks a wider range of images to gather semantic information but not pixel-level content, reflected in token-level masking and a high mask ratio. CSFormer [7] directly adopts this strategy on low-level vision pre-training. However, some studies verify that semantic information is not as important for image restoration as it is in pattern recognition tasks [33, 37]. Moreover, high-degree masking leads to producing detail-deficient results, as shown in Fig. 3, which is harmful to low-level tasks.

Secondly, the training objective of MIM is to reconstruct the masked input images, so it can only produce results with the same domain as the input image. However, we hope the model gains the ability to bridge low-quality domain to high-quality domain, *i.e.* recover clean content from degraded input. Therefore, it is necessary to introduce paired data when pre-training image restoration models by MIM (see the experiment in Sec. 4.3 for details). Chen *et al.* [2] demonstrate that pair-wise MIM training enhances the generalization performance over different types of noisy images. In this paper, we take a step forward to explore the effectiveness of MIM on multiple degradations with larger variance.

3.2 Pretraining with MIM

Based on the above analysis, we design a MIM pre-training paradigm tailored for low-level vision.

Masking. During the pre-training stage, we randomly mask the pixels of degraded images (mask images in a 1×1 patch size) with a 50% mask ratio. We found that fine-grained masked patches and balanced mask ratio are beneficial to image restoration, which can be demonstrated in Sec. 4.3.

Besides, since our MIM pre-training has a similar target to subsequent low-level tasks, we do not need to change the decoder like MAE [15] does but just fine-tune it.

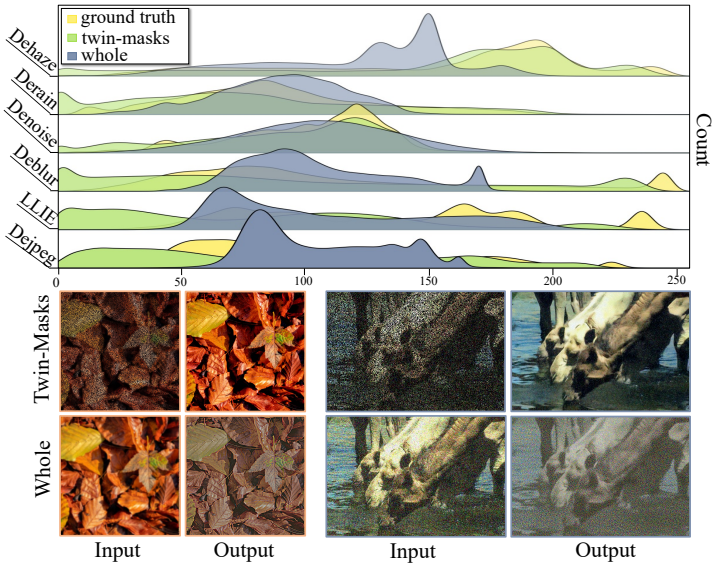


Fig. 4: The effect of MIM reconstruction with different input integrity on kernel deblurring (orange border) and denoising (blue border). We also visualize the color distributions of reconstructions in various tasks above. It shows that the distribution of the reconstruction results obtained using the twin-masks method as input is closer to the real images (ground truth) compared to the results obtained using the whole input.

Reconstruction target. Following the Bert [18] and MAE [15], we choose L1 loss to supervise the masked part. The training objective can be written as:

$$\arg\min_{\theta} \mathbb{E}[||\tilde{\mathcal{M}}(I - f(\mathcal{M}(I_d), \theta))||], \tag{1}$$

where $\{I, I_d\}$ represents a pair of clean image and degraded image, $f(\cdot, \theta)$ denotes a network with parameters θ, $\mathcal{M}(\cdot)$ is a random binary masking operation and $\tilde{\mathcal{M}}(\cdot) = 1 - \mathcal{M}(\cdot)$.

3.3 Finetuning with Mask Attribute Conductance Analysis

Observation. During pre-training, the network learns rich content priors. However, the incompleteness of the masked input prevents the direct use of the pre-trained model for inference, as it would result in a distribution shift in the outputs. As shown in Fig. 4, We start by feeding the entire image into a pre-trained model, leading to a color-distorted result. Next, we use a pair of complementary masks, referred to as twin-masks, to individually mask the image. Subsequently, we input both of these complementarily masked images into the network. By combining the pixel values predicted by each image, we generate a higher-quality image. This observation indicates that the hindrance to using

mask pre-trained model directly for inference lies in input incompleteness rather than the model's inability to learn the restoration function.

Building upon this insight, we explore the possibility of minimizing the influence of disparities in data input formats via model fine-tuning. To maintain the learned priors, it is essential to retain pre-trained parameters as extensively as possible while employing the fewest but most effective layers for fine-tuning. To tackle this, we introduce the concept of mask attribution conductance, which quantifies the importance of each layer concerning the fine-tuning objective. We then identify the top-k% most critical layers for fine-tuning.

Preliminary. Before giving the definition of Mask Attribute Conductance (MAC), we briefly recall the definition of integrate gradient [45] (IG) and neuron conductance [6] (Cond). Considering a linear path $\gamma(\alpha) = x' + \alpha(x - x')$ from base input x' to target input x, we can attribute output change $F(x) - F(x')$ to i-th dimension of input/feature x_i (*e.g.* a pixel) by calculating its integrate gradient, which formally as below:

$$\mathrm{IG}_i(x) := (x_i - x_i') \cdot \int_0^1 \frac{\partial F(x' + \alpha(x - x'))}{\partial x_i} \, d\alpha. \tag{2}$$

We can also attribute output change to a specific neuron y by improving IG, which involves calculating the conductance. The conductance [6] of the hidden neuron y along the $\gamma(\alpha)$ is:

$$\begin{aligned} \mathrm{Cond}^y(x) &:= \sum_i (x_i - x_i') \cdot \int_0^1 \frac{\partial F(x' + \alpha(x - x'))}{\partial y} \cdot \frac{\partial y}{\partial x_i} \, d\alpha \\ &= \sum_i \int_0^1 \frac{\partial F(\gamma(\alpha))}{\partial y} \cdot \frac{\partial y}{\partial \alpha} \, d\alpha, \end{aligned} \tag{3}$$

Note that $(x_i - x_i') = \frac{\partial (x' + \alpha(x_i - x_i'))}{\partial \alpha}$. Certainly, we can broaden Eq. (3) to compute conductance when integrating along any given path $\alpha : [s, t] \to P$:

$$\mathrm{GeneralCond}^y(x) := \sum_i \int_P \frac{\partial F(X_i(\alpha))}{\partial y} \cdot \frac{\partial y}{\partial \alpha} \, d\alpha, \tag{4}$$

where $X : R \to R^n$ is the function of the path from x' to x, which satisfies $X(s) = x'$, $X(t) = x$. $[s, t]$ represent the domain of the path function X."

Finetuning with MAC. To find effective layers to finetune, we propose **M**ask **A**ttribute **C**onductance (**MAC**) to evaluate how effective each layer is in overcoming the gap of input integrity. Considering such a nonlinear path $\alpha : [0, 1] \to P_m$ from zero input x' to whole input x, which path function X^m satisfies:

$$X_i^m(\alpha; \alpha_i) = \begin{cases} x_i', & \alpha < \alpha_i \\ x_i, & else \end{cases}, \tag{5}$$

where i refers to the index of pixels, $\alpha_i \in (0, 1]$ is a set of parameters that indicate when each pixel gets masked. We define this path as a Mask Attribute Path (MAP). Apparently, $X^m(0) = x'$ and $X^m(1) = x$.

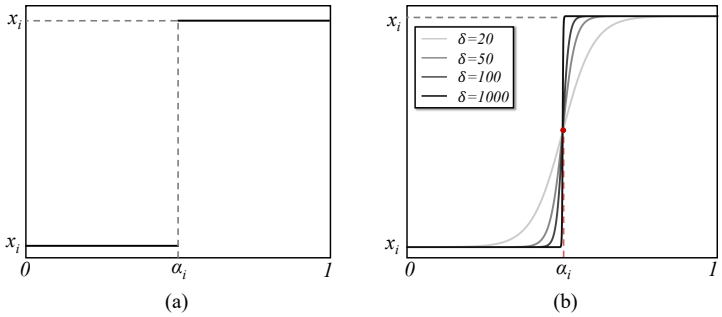


Fig. 5: Illumination of (a) X_i^m in Eq. (5) and (b) $\tilde{X}_i^m$ in Eq. (6).

However, X^m is not differentiable, making it an invalid attribute path function. To solve this problem, we use a group of sigmoid-like functions $\tilde{X}^m$ to approximate X^m:

$$\tilde{X}_i^m(\alpha;\alpha_i) = \frac{(x_i' - x_i)}{1 + e^{-\delta(x_i' - \alpha_i)}}. \tag{6}$$

We can see that $\tilde{X}^m$ is very close to X^m when δ is sufficiently large (as depicted in Fig. 5). And for each $\tilde{X}_i^m$, it will change sharply from x_i' to x_i when α is in the neighborhood of α_i.

Here, we can give a definition of **MAC** as below:

$$\begin{aligned} \mathrm{MAC}^y(x) &:= \sum_i \int_{P_m} \frac{\partial F(X_i(\alpha))}{\partial y} \cdot \frac{\partial y}{\partial \alpha}\, d\alpha \\ &\approx \sum_i \int_0^1 \frac{\partial F(\tilde{X}_i^m(\alpha;\alpha_i))}{\partial y} \cdot \frac{\partial y}{\partial \alpha}\, d\alpha. \end{aligned} \tag{7}$$

In fact, a partial path is also available to attribute from a masked input x_m with any mask ratio r to whole input x:

$$\mathrm{MAC}_r^y(x) \approx \sum_i \int_{1-r}^1 \frac{\partial F(\tilde{X}_i^m(\alpha;\alpha_i))}{\partial y} \cdot \frac{\partial y}{\partial \alpha}\, d\alpha. \tag{8}$$

In practice, we use N-steps discretization to approximate the integral form of Eq. (8), which follows [44]:

$$\begin{aligned} \mathrm{MAC}_r^y(x) \approx \sum_i \sum_{j=1}^N & \frac{\partial F(\tilde{X}_i^m(\frac{jr}{N};\alpha_i))}{\partial y} \\ & \cdot (F_y(\tilde{X}_i^m(\frac{(j+1)r}{N})) - F_y(\tilde{X}_i^m(\frac{jr}{N}))). \end{aligned} \tag{9}$$

We compute the MAC of each layer of pre-trained networks, rank them in descending order based on their MAC values, and pick top-k% layers for fine-tuning. The networks are initialized by pre-trained weight and only top-k% layers

Table 1: Quantitative comparison on seven challenging image restoration tasks, including dehazing, deraining, denoising, motion deblurring, low-light image enhancement (LLIE), kernel deblurring, and JPEG artifact removal. **boldface** and underlined indicate the best and second-best results, respectively.

Method	SOTS [20] PSNR↑/SSIM↑	Rain13k-Test [32] PSNR↑/SSIM↑	BSD68 [38] PSNR↑/SSIM↑	GoPro [40] PSNR↑/SSIM↑	LOL [5] PSNR↑/SSIM↑	LSDIR-Blur [26] PSNR↑/SSIM↑	LSDIR-Jpeg [26] PSNR↑/SSIM↑	Average PSNR↑/SSIM↑
Restormer [55]	22.89/0.9172	27.05/0.8469	**30.95**/**0.8657**	27.46/0.8497	23.65/0.8458	19.60/0.3658	**30.46**/**0.9141**	26.01/0.8007
MPRNet [39]	25.23/0.9463	25.36/0.8068	29.83/0.8317	25.90/0.7949	22.29/0.8170	25.68/0.8281	28.96/0.8865	26.18/0.8445
NAFNet [3]	25.74/0.9445	24.65/0.7877	30.37/0.8540	25.53/0.7909	21.50/0.8104	29.08/0.9130	29.09/0.8955	26.57/0.8566
DL [8]	21.16/0.9042	19.56/0.6508	16.15/0.5861	17.63/0.5862	19.26/0.7777	17.98/0.6121	19.55/0.6965	18.75/0.6877
TAPE [31]	25.14/0.9319	23.66/0.7818	30.11/0.8354	25.97/0.7962	18.95/0.7632	24.26/0.7654	29.28/0.8965	25.34/0.8243
AirNet [21]	21.66/0.8366	20.21/0.6402	27.99/0.7250	23.36/0.7503	16.65/0.6708	23.84/0.7358	24.36/0.8020	22.58/0.7372
SwinIR [28]	27.29/0.9622	25.32/0.8258	30.65/0.8540	26.61/0.8125	18.66/0.8048	27.82/0.8839	30.13/0.9071	26.64/0.8643
RAM-SwinIR	28.47/0.9689	26.31/0.8486	30.83/0.8611	26.89/0.8200	21.62/0.8291	26.66/0.8514	30.22/0.9096	27.28/0.8698
PromptIR [42]	28.70/0.9659	27.46/0.8585	30.84/0.8625	27.71/0.8565	21.19/0.8356	**31.01**/**0.9385**	30.30/0.9117	28.17/0.8899
RAM-PromptIR	**29.64**/**0.9695**	**28.47**/**0.8751**	30.86/0.8624	**28.02**/**0.8592**	**24.46**/**0.8581**	29.57/0.9179	30.33/0.9119	**28.76**/**0.8935**

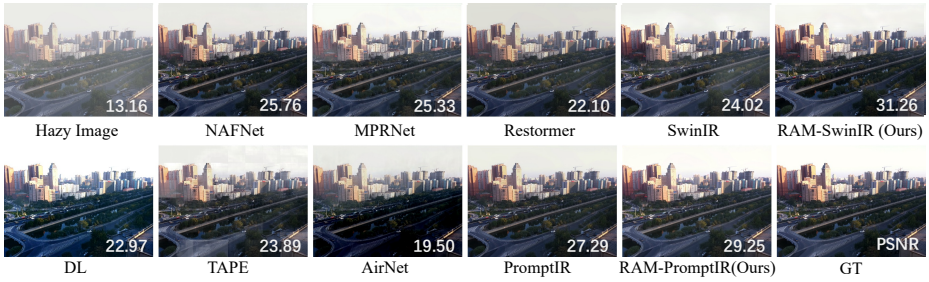


Fig. 6: Dehaze visual comparison on SOTS dataset. Zoom in for details.

will be fine-tuned. More implementation details can be found in the supplementary material.

4 Experiment

4.1 Experiments Settings

Datasets and Metrics. We combine datasets from various restoration tasks to form the training set, following [59]. For high-cost tasks that degradations are difficult to synthesize, we leverage existing paired datasets, including RESIDE [20] for dehazing, Rain13k [10, 25, 27, 35, 53] for deraining, GoPro [40] for motion deblurring, and LOL-v2 [54] for low-light image enhancement (LLIE). For low-cost tasks that degradations are easy to synthesize (*e.g.* noise, kernel blur, and JPEG artifact), we generate corrupted images on the LSDIR dataset [26] during the training process, which involves generating Gaussian noise with random variation $\sigma \in (0, 50]$, creating gaussian blurred images with a blur kernel of size $k = 15$ and random $\sigma \in [0.1, 3.1]$, and introducing JPEG artifacts with a random quality parameter $q \in [20, 90]$.

For evaluation, we use SOTS-outdoor [20] for dehazing, Rain13k-Test (the combination of Rain100L [52], Rain100H [52], Test100 [58], Test1200 [57] and

Table 2: Quantitative Gaussian denoising results at different noise levels on BSD68 and Urban100 datasets in terms of PSNR.

Method	BSD68 [38]				Urban100 [16]			
	$\sigma = 15$	$\sigma = 25$	$\sigma = 50$	Average	$\sigma = 15$	$\sigma = 25$	$\sigma = 50$	Average
NAFNet [3]	33.22	30.59	27.30	30.37	32.67	30.21	26.97	29.92
MPRNet [39]	32.73	30.11	26.65	29.83	32.06	29.46	25.77	29.10
Restormer [55]	33.79	31.17	27.90	30.95	33.83	31.40	27.99	31.07
DL [8]	16.04	16.20	16.19	16.15	19.17	19.11	18.47	18.92
TAPE [31]	33.10	30.37	26.86	30.11	32.59	29.93	26.19	29.57
AirNet [21]	31.63	28.83	23.52	27.99	29.79	26.90	21.35	26.01
SwinIR [28]	33.53	30.89	27.54	30.65	33.50	30.99	27.37	30.62
RAM-SwinIR	33.65	31.06	27.77	30.82	33.82	31.43	27.94	31.07
performance gains	(↑0.12)	(↑0.17)	(↑0.23)	(↑0.17)	(↑0.32)	(↑0.44)	(↑0.57)	(↑0.45)
PromptIR [42]	33.67	31.06	27.80	30.84	33.56	31.08	27.64	30.76
RAM-PromptIR	33.70	31.08	27.79	30.86	33.70	31.30	27.92	30.97
performance gains	(↑0.03)	(↑0.02)	(↓0.01)	(↑0.02)	(↑0.14)	(↑0.22)	(↑0.28)	(↑0.21)

Test2800 [11]) for deraining, GoPro for motion deblurring, LOL [5] for low-light enhancement, BSD68 [38] for denoising, LSDIR-val for kernel deblurring and jpeg artifact removal. Furthermore, We conducted evaluations including denoising tests with variances of 15, 25, and 50, deblurring tests at $k = 15$ and $\sigma = 2.0$, and JPEG artifact removal tests at $q = 50$.

Implementation Details. We apply our proposed **RAM** to SwinIR [28] and PromptIR [42]. The input size for RAM-SwinIR is 64, while for RAM-PromptIR it is 128. During the pre-training phase, we use the Adam optimizer to train RAM-SwinIR and RAM-PromptIR for 300 epochs, with the learning rate decaying from 1e-4 to 6e-5 following a cosine schedule. In the fine-tuning phase, we use the Adam optimizer to fine-tune the network layers obtained from the MAC analysis of RAM-SwinIR and RAM-PromptIR for 40 epochs, with the learning rate decaying from 2e-4 to 1e-7 following a cosine schedule. The batch sizes for RAM-SwinIR and RAM-PromptIR during the pre-training and fine-tuning phases are (12,4) and (4,4), respectively.

4.2 Comparisons

To validate the gain capability and effectiveness of our RAM, we apply the proposed RAM to SwinIR (a general image restoration method) and PromptIR (an all-in-one image restoration method). Four general architecture-based image restoration methods [3, 28, 39, 55] and four all-in-one methods [8, 21, 31, 42] are considered for comparison. We ensure that the number of supervised pixels employed by all other methods equals that used during the pre-training stage.

As illustrated in Tab. 1, our approach achieves the best or comparable performance on each task. On the average score across seven different tasks, our method with PromptIR [42] achieves 0.59dB performance gains compared to the second-best algorithm. Besides, the SwinIR equipped with RAM also yields 2.40% improvement on PSNR. Specifically, our RAM has significant benefits for dehazing and low-light enhancement. Tab. 2 shows the Quantitative denois-

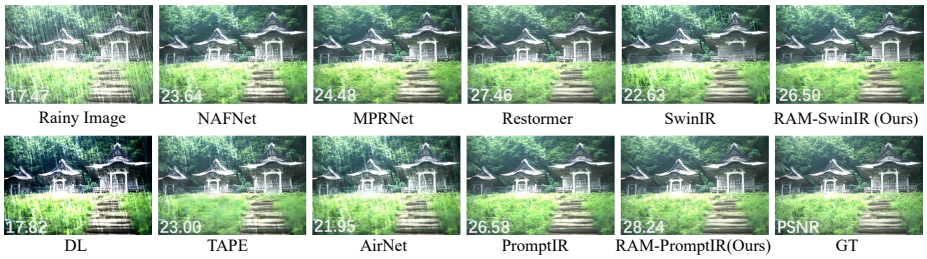


Fig. 7: Derain visual comparsion on Rain13k-Test dataset. Zoom in for details.

Fig. 8: Motion deblur visual comparison on GoPro dataset. Zoom in for details.

ing result at different noise levels. Both RAM-SwinIR and RAM-PromptIR get higher performance than the origin versions.

Fig. 6-Fig. 10 show the qualitative results of various methods on different datasets. In Fig. 6, our method achieves better dehazing effects (right region) and exposure correction (sky). In the deraining task (Fig. 7, our method better removes rain streaks and restores textures in the occluded regions. In terms of denoising (Fig. 9) and deblurring (Fig. 8), we achieve clearer results with fewer artifacts. We also demonstrate better color correction (the purple blanket on the left) and exposure correction in low-light image enhancement tasks (Fig. 10). For simplicity, the qualitative effects of kernel deblurring and JPEG artifact removal will be presented in the supplementary material.

4.3 Ablation Study

In this section, we conduct an ablative study on the masking ratio, mask patch size, pre-training strategy, fine-tuning strategy, and fine-tuning ratio to demonstrate the effectiveness of our MIM pre-training and fine-tuning strategy.

Table 3: Ablative results on masking ratios.

Masking ratio	20%	40%	50%	60%	80%
PSNR↑	27.28	27.21	**27.28**	27.26	27.08
SSIM↑	0.8663	0.8683	**0.8698**	0.8694	0.8642

Table 4: Ablative results of different pre-training strategies.

RAM-SwinIR	PSNR↑	SSIM↑
pre-trained w/ gt	26.62	0.8580
pre-trained w/ paired data	27.28	0.8698

Table 5: Ablative results of different fine-tuning strategies.

RAM-SwinIR	PSNR↑	SSIM↑
random	26.86	0.8535
IG [45]	26.92	0.8554
MAC (Ours)	**27.28**	**0.8698**

Patch size & masking ratio are two essential hyper-parameters that determine the continuity and area of the masking of an image. In high-level tasks, MAE [15] masks 75% of an image with 16×16 patch size. However, it can corrupt the local details of images, which is not suitable for image restoration.

Table 6: Ablative results in terms of the PSNR on fine-tuning ratios. We compared the performance in restoring images with unseen noises (Out-of-Distribution Denoising) and known degraded images (In-Distribution). In this case, the settings of In-Distribution are the same as Tab. 1.

Method	Out-of-Distribution Denoising				In-Distribution
	Possion	Pepper	Speckle	Average	Average
SwinIR [28]	12.83	10.00	20.86	14.56	26.64
RAM-SwinIR$_{10\%}$	**13.67**	**19.23**	**21.07**	**17.99**	27.28
RAM-SwinIR$_{20\%}$	13.27	19.09	20.68	17.68	27.35
RAM-SwinIR$_{50\%}$	12.75	16.51	20.36	16.54	27.38
RAM-SwinIR$_{100\%}$	12.47	15.31	20.01	15.93	**27.54**

We first find the best choice of patch size by pre-training SwinIR [28] on 1×1, 4×4, and 8×8, as shown in Fig. 3. Since the attention layers of SwinIR treat an 8×8 patch as a token, the 4×4 pre-training produces heavy artifacts. Besides, the results generated by 8×8 pre-training are highly missing details, *e.g.* the texture of the polar bear's paws. In contrast, the model pre-trained with 1×1 patch size, which is also our final choice, achieves a satisfactory reconstruction and removes most of the rain streaks.

Then, we adjust the masking ratio from 20% to 80%. As we can see in Tab. 3, the model pre-trained with 50% achieves the highest performance. Moreover, the performance is significantly dropped from 27.28dB to 27.08dB in terms of PSNR when we continue to increase the masking ratio, which also demonstrates our opinion that a high masking ratio is harmful to image restoration.

Pre-trained with paired data. Tab. 4 compares the results of using paired data for mask image pretraining (our pretraining strategy) with those using only ground truth for mask image pretraining. It shows that pre-trained with paired data is necessary for our RAM. Pretraining the model on high-quality images does not effectively enable learning for image restoration tasks. It still requires paired data to guide the model in the learning process.

Fine-tuning strategy. To verify the effectiveness of our fine-tuning strategy, we fine-tune 10% of the network layers selected through MAC analysis, IG [45], and uniform sampling, respectively, and the results are shown in Tab. 5. Compared

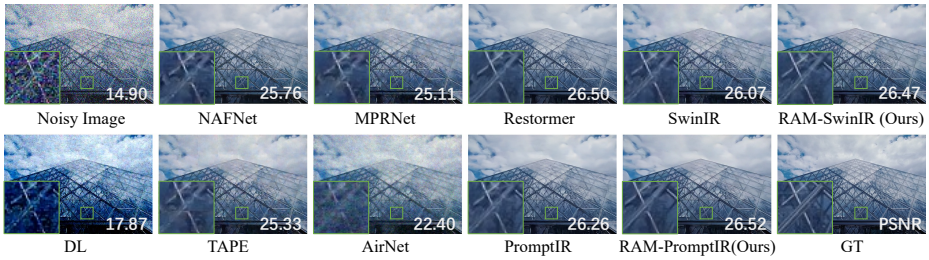


Fig. 9: Denoising visual comparison on CBSD68 dataset. Zoom in for details.

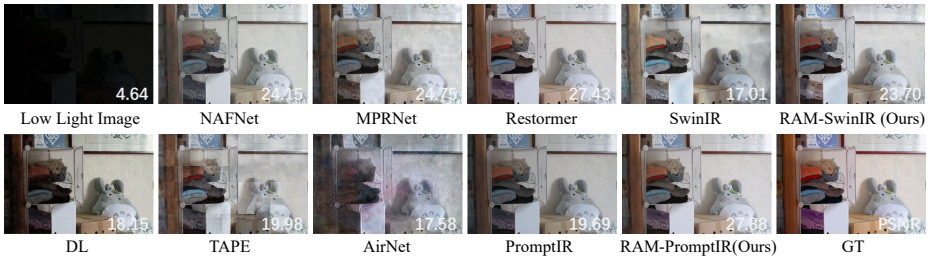


Fig. 10: LLIE visual comparison on LOL dataset. Zoom in for details.

to IG, we have improved by 0.36 dB in PSNR and 1.6% in SSIM, which indicates that our selection strategy is superior to IG.

Fine-tune ratio. We conduct the ablation experiment to compare the network's performances with different fine-tune ratios in Tab. 6. We found that using our finetune strategy, a pre-trained network could achieve comparable performance by fine-tuning only a few layers (*e.g.* 10%). At the same time, we need to fine-tune almost all network parameters to get the best performance on given tasks.

Performance *vs* Generalization capability. We found a trade-off between in-distribution performance and out-of-distribution generalization in Tab. 6. We found that the more layers fine-tuned, the less generalization capability to tackle the out-of-distribution tasks. With our fine-tuning method, the model can have stronger generalization while maintaining comparable performance.

5 Conclusion

This paper presents RAM, a pipeline for extracting intrinsic image information from corrupted images using Mask Image Modeling (MIM) pre-training. We design a MIM pre-training strategy tailored for image restoration and a fine-tuning algorithm to handle the transition from masked to complete images. By analyzing layer importance with MAC, we achieve high performance with minimal parameter tuning. Extensive experiments demonstrate that our RAM can bring boosts to various architectures and achieve state-of-the-art performance, moving towards a unified solution for all-in-one image restoration.

References

1. Chen, H., Wang, Y., Guo, T., Xu, C., Deng, Y., Liu, Z., Ma, S., Xu, C., Xu, C., Gao, W.: Pre-trained image processing transformer. In: CVPR. pp. 12299–12310 (2021)
2. Chen, H., Gu, J., Liu, Y., Magid, S.A., Dong, C., Wang, Q., Pfister, H., Zhu, L.: Masked image training for generalizable deep image denoising. In: CVPR. pp. 1692–1703 (2023)
3. Chen, L., Chu, X., Zhang, X., Sun, J.: Simple baselines for image restoration. In: ECCV. pp. 17–33. Springer (2022)
4. Chen, W.T., Huang, Z.K., Tsai, C.C., Yang, H.H., Ding, J.J., Kuo, S.Y.: Learning multiple adverse weather removal via two-stage knowledge learning and multi-contrastive regularization: Toward a unified model. In: CVPR. pp. 17653–17662 (2022)
5. Chen Wei, Wenjing Wang, W.Y.J.L.: Deep retinex decomposition for low-light enhancement. In: BMVC. British Machine VLOLision Association (2018)
6. Dhamdhere, K., Sundararajan, M., Yan, Q.: How important is a neuron. In: ICLR (2019)
7. Duan, H., Shen, W., Min, X., Tu, D., Teng, L., Wang, J., Zhai, G.: Masked autoencoders as image processors. arXiv preprint arXiv:2303.17316 (2023)
8. Fan, Q., Chen, D., Yuan, L., Hua, G., Yu, N., Chen, B.: A general decoupled learning framework for parameterized image operators. PAMI **43**(1), 33–47 (2019)
9. Fang, Y., Zhang, H., Wong, H.S., Zeng, T.: A robust non-blind deblurring method using deep denoiser prior. In: CVPRW. pp. 735–744 (June 2022)
10. Fu, X., Huang, J., Ding, X., Liao, Y., Paisley, J.: Clearing the skies: A deep network architecture for single-image rain removal. TIP **26**(6), 2944–2956 (2017)
11. Fu, X., Huang, J., Zeng, D., Huang, Y., Ding, X., Paisley, J.: Removing rain from single images via a deep detail network. In: CVPR. pp. 3855–3863 (2017)
12. Gu, J., Dong, C.: Interpreting super-resolution networks with local attribution maps. In: CVPR. pp. 9199–9208 (2021)
13. Guo, C.L., Yan, Q., Anwar, S., Cong, R., Ren, W., Li, C.: Image dehazing transformer with transmission-aware 3d position embedding. In: CVPR. pp. 5812–5820 (2022)
14. Guo, C., Li, C., Guo, J., Loy, C.C., Hou, J., Kwong, S., Cong, R.: Zero-reference deep curve estimation for low-light image enhancement. In: CVPR. pp. 1780–1789 (2020)
15. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick., R.: Masked autoencoders are scalable vision learners. In: CVPR. pp. 16000–16009 (2022)
16. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: CVPR. pp. 5197–5206 (2015)
17. Jin, X., Han, L.H., Li, Z., Guo, C.L., Chai, Z., Li, C.: Dnf: Decouple and feedback network for seeing in the dark. In: CVPR. pp. 18135–18144 (2023)
18. Kenton, J.D.M.W.C., Toutanova, L.K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: NAACL. pp. 4171–4186 (2019)
19. Leino, K., Sen, S., Datta, A., Fredrikson, M., Li, L.: Influence-directed explanations for deep convolutional networks. In: ITC. pp. 1–8. IEEE (2018)
20. Li, B., Ren, W., Fu, D., Tao, D., Feng, D., Zeng, W., Wang, Z.: Benchmarking single-image dehazing and beyond. TIP **28**(1), 492–505 (2018)
21. Li, B., Liu, X., Hu, P., Wu, Z., Lv, J., Peng, X.: All-in-one image restoration for unknown corruption. In: CVPR. pp. 17452–17462 (2022)

22. Li, C., Guo, C.L., Liang, Z., Zhou, S., Feng, R., Loy, C.C., et al.: Embedding fourier for ultra-high-definition low-light image enhancement. In: ICLR (2022)
23. Li, D., Zhang, Y., Cheung, K.C., Wang, X., Qin, H., Li, H.: Learning degradation representations for image deblurring. In: ECCV. pp. 736–753. Springer (2022)
24. Li, R., Tan, R.T., Cheong, L.F.: All in one bad weather removal using architectural search. In: CVPR. pp. 3175–3185 (2020)
25. Li, X., Wu, J., Lin, Z., Liu, H., Zha, H.: Recurrent squeeze-and-excitation context aggregation net for single image deraining. In: ECCV. pp. 254–269 (2018)
26. Li, Y., Zhang, K., Liang, J., Cao, J., Liu, C., Gong, R., Zhang, Y., Tang, H., Liu, Y., Demandolx, D., et al.: Lsdir: A large scale dataset for image restoration. In: CVPR. pp. 1775–1787 (2023)
27. Li, Y., Tan, R.T., Guo, X., Lu, J., Brown, M.S.: Rain streak removal using layer priors. In: CVPR
28. Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: Swinir: Image restoration using swin transformer. In: CVPR. pp. 1833–1844 (2021)
29. Lin, X., Ren, C., Liu, X., Huang, J., Lei, Y.: Unsupervised image denoising in real-world scenarios via self-collaboration parallel generative adversarial branches. In: ICCV. pp. 12642–12652 (2023)
30. Lin, X., Yue, J., Ren, C., Guo, C.L., Li, C.: Unlocking low-light-rainy image restoration by pairwise degradation feature vector guidance. arXiv preprint arXiv:2305.03997 (2023)
31. Liu, L., Xie, L., Zhang, X., Yuan, S., Chen, X., Zhou, W., Li, H., Tian, Q.: Tape: Task-agnostic prior embedding for image restoration. In: ECCV. pp. 447–464. Springer (2022)
32. Liu, Y., He, J., Gu, J., Kong, X., Qiao, Y., Dong, C.: Degae: A new pretraining paradigm for low-level vision. In: CVPR. pp. 23292–23303 (2023)
33. Liu, Y., Liu, A., Gu, J., Zhang, Z., Wu, W., Qiao, Y., Dong, C.: Discovering distinctive "semantics" in super-resolution networks. arXiv preprint arXiv:2108.00406 (2021)
34. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: CVPR. pp. 10012–10022 (2021)
35. Luo, Y., Xu, Y., Ji, H.: Removing rain from a single image via discriminative sparse coding. In: ICCV. pp. 3397–3405 (2015)
36. Luo, Y., Zhao, R., Wei, X., Chen, J., Lu, Y., Xie, S., Wang, T., Xiong, R., Lu, M., Zhang, S.: Mowe: mixture of weather experts for multiple adverse weather removal. arXiv preprint arXiv:2303.13739 (2023)
37. Magid, S.A., Lin, Z., Wei, D., Zhang, Y., Gu, J., Pfister, H.: Texture-based error analysis for image super-resolution. In: CVPR. pp. 2118–2127 (2022)
38. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: ICCV. vol. 2, pp. 416–423. IEEE (2001)
39. Mehri, A., Ardakani, P.B., Sappa, A.D.: Mprnet: Multi-path residual network for lightweight image super resolution. In: CVPR. pp. 2704–2713 (2021)
40. Nah, S., Hyun Kim, T., Mu Lee, K.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: CVPR
41. Park, D., Lee, B.H., Chun, S.Y.: All-in-one image restoration for unknown degradations using adaptive discriminative filters for specific degradations. In: CVPR. pp. 5815–5824 (2023)
42. Potlapalli, V., Zamir, S.W., Khan, S.H., Shahbaz Khan, F.: Promptir: Prompting for all-in-one image restoration. NeurIPS **36** (2024)

43. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al.: Improving language understanding by generative pre-training (2018)
44. Shrikumar, A., Su, J., Kundaje, A.: Computationally efficient measures of internal neuron importance. arXiv preprint arXiv:1807.09946 (2018)
45. Sundararajan, M., Taly, A., Yan, Q.: Axiomatic attribution for deep networks. In: ICML. pp. 3319–3328. PMLR (2017)
46. Sundararajan, M., Taly, A., Yan, Q.: Gradients of counterfactuals. ICLR (2017)
47. Wang, X., Wang, W., Cao, Y., Shen, C., Huang, T.: Images speak in images: A generalist painter for in-context visual learning. In: CVPR. pp. 6830–6839 (2023)
48. Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., Li, H.: Uformer: A general u-shaped transformer for image restoration. In: CVPR. pp. 17683–17693 (2022)
49. Wu, R.Q., Duan, Z.P., Guo, C.L., Chai, Z., Li, C.: Ridcp: Revitalizing real image dehazing via high-quality codebook priors. In: CVPR. pp. 22282–22291 (2023)
50. Xie, L., Wang, X., Dong, C., Qi, Z., Shan, Y.: Finding discriminative filters for specific degradations in blind super-resolution. NeurIPS **34**, 51–61 (2021)
51. Xie, Z., Zhang, Z., Cao, Y., Lin, Y., Bao, J., Yao, Z., Dai, Q., Hu, H.: Simmim: a simple framework for masked image modeling. In: CVPR. pp. 9653–9663 (2022)
52. Yang, W., Tan, R.T., Feng, J., Liu, J., Guo, Z., Yan, S.: Deep joint rain detection and removal from a single image. In: CVPR. pp. 1357–1366 (2017)
53. Yang, W., Tan, R.T., Wang, S., Fang, Y., Liu, J.: Single image deraining: From model-based to data-driven and beyond. PAMI **43**(11), 4059–4077 (2020)
54. Yang, W., Wang, W., Huang, H., Wang, S., Liu, J.: Sparse gradient regularized deep retinex network for robust low-light image enhancement. TIP **30**, 2072–2086 (2021)
55. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H.: Restormer: Efficient transformer for high-resolution image restoration. In: CVPR. pp. 5728–5739 (2022)
56. Zhang, C., Zhu, Y., Yan, Q., Sun, J., Zhang, Y.: All-in-one multi-degradation image restoration network via hierarchical degradation representation. In: ACMMM. pp. 2285–2293 (2023)
57. Zhang, H., Patel, V.M.: Density-aware single image de-raining using a multi-stream dense network. In: CVPR. pp. 695–704 (2018)
58. Zhang, H., Sindagi, V., Patel, V.M.: Image de-raining using a conditional generative adversarial network. TCSVT **30**(11), 3943–3956 (2019)
59. Zhang, J., Huang, J., Yao, M., Yang, Z., Yu, H., Zhou, M., Zhao, F.: Ingredient-oriented multi-degradation learning for image restoration. In: CVPR. pp. 5825–5835 (2023)
60. Zheng, N., Zhou, M., Dong, Y., Rui, X., Huang, J., Li, C., Zhao, F.: Empowering low-light image enhancer through customized learnable priors. In: ICCV. pp. 12559–12569 (2023)
61. Zhu, Y., Wang, T., Fu, X., Yang, X., Guo, X., Dai, J., Qiao, Y., Hu, X.: Learning weather-general and weather-specific features for image restoration under multiple adverse weather conditions. In: CVPR. pp. 21747–21758 (2023)