IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 5 March 2024, accepted 16 April 2024, date of publication 22 April 2024, date of current version 29 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3392016

RESEARCH ARTICLE

Multiple Adverse Weather Removal Using Masked-Based Pre-Training and Dual-Pooling Adaptive Convolution

SHUGO YAMASHITA AND MASAAKI IKEHARA, (Senior Member, IEEE)
Faculty of Science and Technology, Department of Electrical and Information Engineering, Keio University, Yokohama, Kanagawa 223-8522, Japan
Corresponding author: Shugo Yamashita (yamashita@tkhm.elec.keio.ac.jp)

ABSTRACT Removing artifacts caused by multiple adverse weather, including rain, fog, and snow, is crucial for image processing in outdoor environments. Conventional high-performing methods face challenges, such as requiring pre-specification of weather types and slow processing times. In this study, we propose a novel convolutional neural network-based hierarchical encoder-decoder model that addresses these issues effectively. Our model utilizes knowledge of feature representations obtained from masked-based pre-training on a large-scale dataset. To remove diverse degradations efficiently, we employ a proposed dual-pooling adaptive convolution, which improves representational capability of weight generating network by using average pooling, max pooling, and filter-wise global response normalization. Experiments conducted on both synthetic and real image datasets show that our model achieves promising results. The performance on real images is also improved by a novel learning strategy, in which a model trained on the synthetic image dataset is fine-tuned to the real image dataset. The proposed method is notably cost-effective in terms of computational complexity and inference speed. Moreover, ablation studies show the effectiveness of various components in our method.

INDEX TERMS Convolutional neural network, dehazing, deraining, desnowing, large-scale pre-training, raindrop removal, weight generating network.

I. INTRODUCTION

Removing degradations due to adverse weather conditions in images is important for the practicality of image processing systems in outdoor environments. Image processing algorithms, such as object detection, object tracking, and segmentation, are used in automated vehicle driving, security systems, and the like. Adverse weather conditions, such as rain, fog, and snow, cause a loss of information in images and degrade the performance of these algorithms. Thus, removing adverse weather degradations can improve the effectiveness and reliability of image processing across numerous fields. Deep learning based solutions have been explored extensively for raindrop removal [1], [2], [3], [4], rain and fog removal [5], [6], [7], [8], [9], and snow removal [10], [11], [12].

Recently, various methods [13], [14], [15], [16], [17] employ a unified set of network parameters to remove degradations caused by multiple weather conditions. Multiple adverse weather removal initially proposed by [13] has been subsequently developed through transformer [14] and knowledge distillation [15]. Most recent methods [16], [17] achieve notable performance improvements, but face challenges in real-world applications. WeatherDiffusion [16], a denoising diffusion model, is computationally expensive and requires a long inference time. WGWS [17] learns

The associate editor coordinating the review of this manuscript and approving it for publication was Chuan Li.

general and specific weather features through a two-step learning process. However, its inference is limited in that a type of weather has to be specified beforehand.

In this study, we propose a novel convolution-based encoder-decoder model that removes different weather-related artifacts rapidly without specifying weather types. To design a powerful encoder, we use large-scale pre-training. Masked-based self-supervised learning [18], [19], [20] enables extracting valuable features from images. Pre-training with these methods on a large dataset has recently demonstrated remarkable success in image classification, object detection, and semantic segmentation. We utilize fully convolutional masked autoencoder (FCMAE) [20] for pre-training to tackle adverse weather removal problems.

To deal with multiple adverse weather degradations, we employed adaptive convolution for processing features extracted by the encoder. We develop WeightNet [21], which generates convolution weights by a network, to dual-pooling adaptive convolution (DPAC). This module aggregates spatial information through average pooling and max pooling, enhancing the weight generation process. In addition, filter-wise global response normalization (FGRN) encourages capturing diverse features across filters.

We experimentally use the synthetic image dataset All-Weather [13] and the real image dataset WeatherStream [22]. Our model obtains superior performance across multiple weather conditions. To boost the performance on WeatherStream, we introduce a novel learning strategy in which models trained on All-Weather are fine-tuned to WeatherStream. The proposed method has a relatively low computational cost and fast inference speed. Furthermore, ablation studies are conducted to validate the effectiveness of the components of the proposed method.

The contributions of this paper are summarized as:

- A novel model for removing multiple weather corruptions is proposed. Masked-based pre-training and adaptive convolution make our model effective and efficient.
- We propose dual-pooling adaptive convolution (DPAC). It combines average pooling and max pooling to improve the expressive power of the weight generating network and employs filter-wise global response normalization (FGRN) to extract various features.
- The proposed method achieves state-of-the-art performance on both synthetic and real image datasets. The performance on real images is also improved by a novel learning strategy that fine-tunes a model trained on the synthetic image dataset to the real image dataset.

II. RELATED WORKS

A. ADVERSE WEATHER REMOVAL

There has been extensive research on employing deep learning to remove degradation caused by adverse weather conditions in single images. Conventional weather removal models, such as raindrop removal [1], [2], [3], [4], rain and fog removal [5], [6], [7], [8], [9], and desnowing [10], [11], [12] concentrated on specific situations. Recent works removed a variety of weather degradations using a unified model [13], [14], [15], [16], [17].

1) SINGLE ADVERSE WEATHER REMOVAL

To remove raindrops, a dual residual network [1], a CNN with a dual attention mechanism [2], an attention GAN [3], and an image deraining transformer [4] were proposed. To remove rain and fog, a multi-stage recurrent network [5] with the squeeze-and-excitation block [23], a spatial attentive mechanism [6], a heavy rain GAN [7], a progressive coupled network [8], and a multi-stage progressive image restoration network [9] were used. Image-to-image translation models based on GAN [24], like pix2pix [25] and CycleGAN [26], were demonstrated their ability to capture intrinsic structures of image backgrounds when applied to these tasks. To remove snow, a joint size and transparency-aware algorithm [10], a two-stage network [11], and a deep dense multi-scale network [12] were proposed. Concurrently, it was found that several deraining methods [5], [6] also yielded favorable results on this task.

2) MULTIPLE ADVERSE WEATHER REMOVAL

Li et al. [13] proposed an All-in-One network with multiple task-specific encoders and a common decoder, which is the first method to remove all weather degradation. Valanarasu et al. [14] used a network based on Vision Transformer with a learnable weather query. Chen et al. [15] proposed a two-stage knowledge learning with teacher networks for each weather type and a single student network and regularization by contrast learning. Özdenizci and Legenstein [16] removed weather-related artifacts using a patch-based denoising diffusion model, but it took a long inference time. Zhu et al. [17] designed a two-stage learning strategy, which optimizes the network to learn weather-general features and then weather-specific features. Some multiple weather removal models [13], [17] require specification of a weather type in advance. For applying these models in the real world, a type of weather has to be specified by either a person or a system. This study utilizes large-scale pre-training and adaptive convolution to design a model that achieves short inference times without pre-specifying a weather type.

B. MASKED-BASED PRE-TRAINING

Masked autoencoder (MAE) [18] is a self-supervised method that masks a portion of the input image and learns to reconstruct the original image from the masked image. This learning process acquires the ability of the model to extract meaningful information from images. While the original MAE used a vanilla vision transformer (ViT) [27], Liu et al. [19] proposed an efficient masked image modeling method for hierarchical ViTs. ConvNeXt V2 [20] enables masked-based self-supervised learning in hierarchical CNNs with fully convolutional masked autoencoder (FCMAE). This approach uses a sparse convolution [28], [29], [30]

during pre-training. These works [18], [19], [20] showed that encoder with self-supervised pre-training on the large dataset ImageNet-1K [31] is effective for downstream tasks such as image recognition, object detection, and semantic segmentation. However, its effectiveness in the field of image restoration has not been verified. In this work, we utilize masked-based pre-training for removing weather degradation, which is one of the image restoration tasks.

C. ADAPTIVE CONVOLUTION

Various methods for adaptively generating convolution weights based on input features have been explored. CondConv [32] and Dynamic Convolution [33] realized dynamic weights by preparing multiple expert convolution weights and predicting the coefficients that combine them. WeightNet [21] proposed directly generating convolution weights employing a weight generation network. DDF [34] generated decoupled spatial and channel dynamic weights. Our proposed dual-pooling adaptive convolution extends WeightNet [21] to enable more effective processing.

III. PROPOSED METHOD

As shown in Fig. 1, the proposed method is a convolution-based hierarchical encoder-decoder model. Initially, an RGB image $I \in \mathbb{R}^{3\times H\times W}$ degraded by adverse weather conditions is forwarded to an encoder that is pre-trained by a masked-based method. We then use a middle network that performs adaptive convolutional processing to handle various degradations efficiently. The features extracted from both the encoder and the middle network are fed into a decoder, generating a clear image $\hat{I} \in \mathbb{R}^{3\times H\times W}$. The following describes these components in detail.

A. MASKED-BASED PRE-TRAINED ENCODER

To effectively extract features from a weather degraded image I, we use a convolution-based encoder pre-trained by fully convolutional masked autoencoder (FCMAE) in ConvNeXt V2 [20]. The encoder captures both low-level and high-level features by sequentially processing in four resolution stages. This approach helps extract features related to local degradation, such as raindrops, rain streaks, and snowflakes, as well as global degradation, such as fog and widespread loss of illumination. Each resolution stage has multiple blocks. As shown in Fig. 1 (b), the encoder block initially uses a 7×7 depth-wise convolution, following the success of ConvNeXt [35]. Pre-training is performed on the large-scale dataset ImageNet-1K [31], using the FCMAE strategy, which is a masked-based learning method. FCMAE acquires the ability to extract useful features from images by learning to recover masked areas. The proposed method uses the pre-trained weights as initial values to learn weather degradation removal to take advantage of representation learning knowledge obtained from the masked-based pre-training.

B. DUAL-POOLING ADAPTIVE CONVOLUTION MIDDLE NETWORK

Our proposed dual-pooling adaptive convolution (DPAC) is used to handle degradation caused by multiple types of adverse weather efficiently. DPAC is an extension of WeightNet [21], which performs convolution using weights generated from input features by a network. DPAC is intended to replace ordinary depth-wise convolution, consisting of a weight generating network and a depth-wise convolution with kernel size k, as shown in Fig. 2. The main differences between WeightNet and DPAC are the way of aggregating spatial information and the presence of a normalization layer. Both methods first aggregate the spatial information of the input features $x \in \mathbb{R}^{c\times h\times w}$. WeightNet uses only average pooling, while DPAC combines average pooling and max pooling. This improves the expressive capability of the weight generating network. The features extracted by the two types of pooling are processed by a convolution, then added and an activation function GELU [36] is applied. This operation can be summarized as:

$$z_{\mathrm{w}} = \mathrm{GELU}(\mathrm{Conv}(\mathrm{Avg\ Pooling}(x)) + \mathrm{Conv}(\mathrm{Max\ Pooling}(x))). \tag{1}$$

Then, a depth-wise convolution are performed with output $\hat{z}_{\mathrm{w}} \in \mathbb{R}^{k^2c\times 1\times 1}$:

$$\hat{z}_{\mathrm{w}} = \mathrm{DW\text{-}Conv}(z_{\mathrm{w}}). \tag{2}$$

After that, we use filter-wise global response normalization (FGRN) to encourage each filter of convolution weights to extract various features. FGRN applies GRN [20], a normalization method to increase channel contrast and selectivity, to each filter in a convolution. FGRN calibrates a convolution weight using normalized scores computed based on information aggregated in an L2 norm for each filter. We reshape $\hat{z}_{\mathrm{w}} \in \mathbb{R}^{k^2c\times 1\times 1}$ to the convolution weights $w \in \mathbb{R}^{c\times k\times k}$, and let its i-th filter be $w_i \in \mathbb{R}^{1\times k\times k}$. FGRN can be formulated as:

$$\hat{w}_i = \gamma \frac{\|w_i\|}{\sum_{j=1}^{c} \|w_j\|} w_i + \beta, \tag{3}$$

where γ and β are trainable variables. Unlike the original GRN implementation, skip connections are not used. The above process of generating convolution weights can be summarized as:

$$\hat{w} = \mathrm{FGRN}(\hat{z}_{\mathrm{w}}). \tag{4}$$

Finally, a depth-wise convolution is performed using the generated weights $\hat{w} \in \mathbb{R}^{c\times k\times k}$:

$$y = \mathrm{DW\text{-}Conv}(x; \hat{w}). \tag{5}$$

Our weather removal model uses a DPAC middle network to perform adaptive convolution on the features extracted by the encoder. The DPAC middle network consists of multiple DPAC middle blocks (Fig. 1 (c)), which employ 3×3 DPAC for the first depth-wise convolution of the ConvNeXt [20]

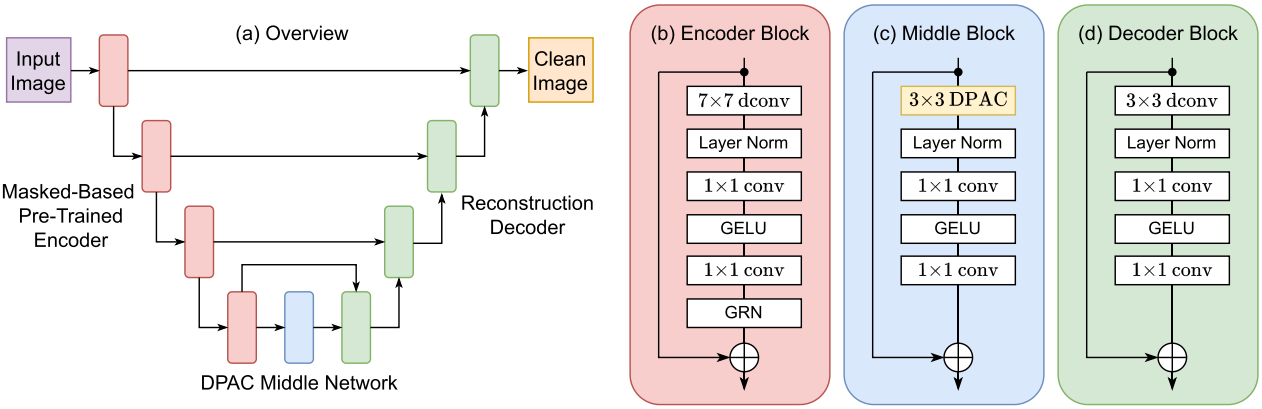


FIGURE 1. (a) The overall illustration of our method. (b), (c), (d) The detailed structure of encoder, middle, and decoder blocks, respectively.

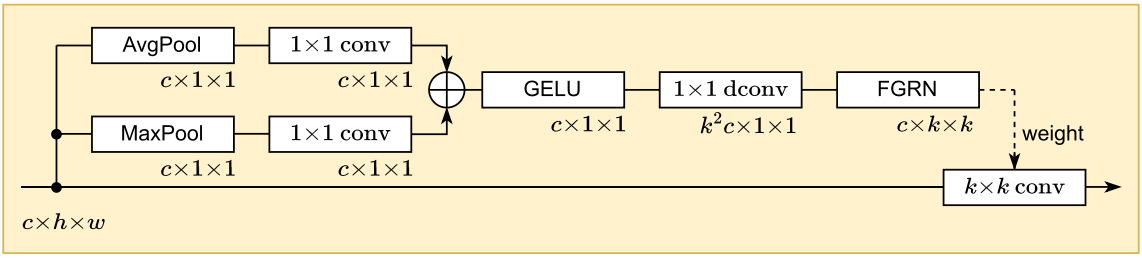


FIGURE 2. The architecture of dual-pooling adaptive convolution (DPAC).

block. In the original ConvNeXt, the number of channels is increased by a factor of 4 in the first point-wise convolution. However, in our middle block, the number of channels is not increased, thus reducing the computational complexity and suppressing overfitting of the model.

C. RECONSTRUCTION DECODER

The features extracted by the encoder and the middle network are fed into a reconstruction decoder to obtain a clear image equal in size to the input image. The decoder uses deconvolutions, decoder blocks, and skip connections with the encoder at the corresponding resolutions. The decoder block (Fig. 1 (d)) is based on the structure of ConvNeXt [35], but a 3×3 depth-wise convolution is employed instead of a 7×7 depth-wise convolution, following experimental results. Finally, a tanh activation function is applied.

D. LOSS FUNCTION

The loss function of the proposed method is computed based on the model output $\hat{I}$ and the ground truth G. Following TransWeather [14], we use a smooth L1-loss and a perceptual loss [37]. The smooth L1-loss is formulated as:

$$\mathcal{L}_{\text{smooth}L_1} = \begin{cases} 0.5(\hat{I} - G)^2 & \text{if } |\hat{I} - G| < 1, \\ |\hat{I} - G| - 0.5 & \text{otherwise.} \end{cases} \tag{6}$$

In the perceptual loss, $\hat{I}$ and G are fed into a VGG16 [38] network pre-trained on ImageNet-1K [31], and the mean squared error (MSE) of the features extracted in its 3rd, 8th, and 15th layers is calculated:

$$\mathcal{L}_{\text{perceptual}} = \sum_{i \in \{3,8,15\}} \mathcal{L}_{\text{MSE}}(\text{VGG}(\hat{I}), \text{VGG}(G)). \tag{7}$$

In addition, we employ a structural similarity (SSIM) loss [39]. SSIM [40] is a metric that evaluates the similarity between two images based on their luminance, contrast, and structural attributes. The SSIM Loss is designed to maximize the SSIM value towards its maximum value of 1. It is computed as follows:

$$\mathcal{L}_{\text{SSIM}} = 1 - \text{SSIM}(\hat{I}, G). \tag{8}$$

The total loss is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{smooth}L_1} + \lambda_{\text{p}}\mathcal{L}_{\text{perceptual}} + \lambda_{\text{s}}\mathcal{L}_{\text{SSIM}}, \tag{9}$$

where λ_{p} and λ_{s} are coefficients to adjust the ratio of the losses.

IV. EXPERIMENTS

A. SET UP

1) DATASETS

We use a synthetic image dataset and a real-world image dataset. These datasets comprise pairs of images, each pair consisting of a degraded image due to adverse weather and its corresponding clear image. The synthetic image dataset All-Weather [13], used in previous studies [13], [14], [16],

TABLE 1. Quantitative comparison on the synthetic image dataset all-weather [13]. Best and second-best values are indicated with bold text and underlined text respectively.

Type	Method	RainDrop [3]		Outdoor-Rain [7]		Snow100K-L [11]		Average	
		PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
RainDrop	pix2pix [25]	28.02	0.8547	-	-	-	-	-	-
	DuRN [1]	31.24	0.9259	-	-	-	-	-	-
	RaindropAttn [2]	31.44	0.9263	-	-	-	-	-	-
	AttentiveGAN [3]	31.59	0.9170	-	-	-	-	-	-
	IDT [4]	31.87	0.9313	-	-	-	-	-	-
Rain & Fog	CycleGAN [26]	-	-	17.62	0.6560	-	-	-	-
	pix2pix [25]	-	-	19.09	0.7100	-	-	-	-
	HRGAN [7]	-	-	21.56	0.8550	-	-		
	PCNet [8]	-	-	26.19	0.9015	-	-	-	-
	MPRNet [9]	-	-	28.03	0.9192	-	-	-	-
Snow	SPANet [6]	-	-	-	-	23.70	0.7930	-	-
	JSTASR [10]	-	-	-	-	25.32	0.8076	-	-
	RESCAN [5]	-	-	-	-	26.08	0.8108	-	-
	DesnowNet [11]	-	-	-	-	27.17	0.8983	-	-
	DDMSNet [12]	-	-	-	-	28.85	0.8772	-	-
Multi	TW [14]	30.86	0.9227	28.98	0.8998	29.40	0.8870	29.75	0.9032
	TSK [15]	31.95	0.9316	25.67	0.8747	28.60	0.8776	28.74	0.8946
	WD [16]	30.42	0.9311	29.48	**0.9278**	29.60	0.8916	29.83	0.9168
	WGWS [17]	**33.43**	**0.9476**	26.65	0.9173	<u>30.66</u>	<u>0.9114</u>	30.25	<u>0.9254</u>
	Ours-S	31.86	0.9367	<u>30.23</u>	0.9203	30.34	0.9045	<u>30.81</u>	0.9205
	Ours-L	<u>32.46</u>	<u>0.9418</u>	**30.62**	<u>0.9272</u>	**31.04**	**0.9127**	**31.37**	**0.9272**

[17], comprises three subsets: RainDrop [3], containing images of raindrops; Outdoor-Rain [7], including images of rain and fog; and Snow100K [11], consisting of snow images. The training set is composed of 18, 069 image pairs, with the test set including 58 pairs from RainDrop [3], 750 pairs including rain and fog from Test1 [7], and 16, 801 pairs from Snow100K-L [11]. The real image dataset WeatherStream [22] contains rain, fog, and snow images collected from videos taken in the real world. The training data contains 163, 800 image pairs, with the test data including 3, 000 pairs of rain images, 4, 500 pairs of fog images, and 3, 960 pairs of snow images.

2) IMPLEMENTATION DETAILS

The proposed method is implemented with two model sizes using the PyTorch framework [41]. The smaller model is denoted as Ours-S and the larger model as Ours-L. We use both Ours-S and Ours-L for comparison with the conventional method, and only Ours-S for ablation studies. The number of blocks and channels at each stage of the encoder are {2, 2, 6, 2}, {64, 128, 256, 512} for Ours-S, and {3, 3, 9, 3}, {96, 192, 384, 768} for Ours-L. In the decoder's architecture, the number of blocks at each stage is {1, 1, 1, 1} for Ours-S, and {2, 2, 2, 2} for Ours-L. All other conditions are the same between Ours-S and Ours-L. The number of blocks in the middle network is set to 3. Additionally, hyperparameters of the loss function are set as: $\lambda_p = 0.04$, $\lambda_s = 0.15$.

3) TRAINING SPECIFICATIONS

For the initial encoder values, the pre-trained weights published from the official implementation of ConvNeXt V2 [20] are used. The pre-training on ImageNet-1K [31] is carried out through 1,600 epochs by FCMAE.

The proposed weather removal model is trained using the Adam optimizer [42] with a batch size of 32. During training, all input images are randomly cropped to 256×256. In accordance with previous studies [14], [17], the model is trained for 200 epochs on All-Weather [13] dataset. Considering that WeatherStream [22] dataset is approximately ten times larger than All-Weather dataset, the model is trained for 20 epochs on this larger dataset. To enhance performance on the real image dataset WeatherStream, we adopt a novel training strategy that fine-tunes a model trained on the synthetic image dataset All-Weather to WeatherStream. This learning is conducted in 5 epochs. A learning rate is initially set low and increased incrementally to prevent pre-training parameters from being updated too much in the early stages of training. Specifically, the learning rate is increased linearly from 1×10^{-5} to 2×10^{-4} for the first 10% epochs, and then keeps constant at 2×10^{-4} until 50% epochs. The learning rate is halved after 50% epochs and 75% epochs.

4) EVALUATION METRICS

We use peak signal-to-noise ratio (PSNR) [43] and structural similarity (SSIM) [40] to evaluate model performance. Following previous studies [14], [15], [16], PSNR and SSIM are assessed based on the luminance channel Y in the YCbCr color space. We also measure the number of Multiply-Accumulates (MACs), which is a metric of computational complexity, and inference time for an input image with an image of 256×256 resolution, using a single NVIDIA GeForce GTX 1080 Ti.

5) COMPARISON METHODS

On the commonly used All-Weather [13] dataset, we compare the proposed method with conventional removal methods

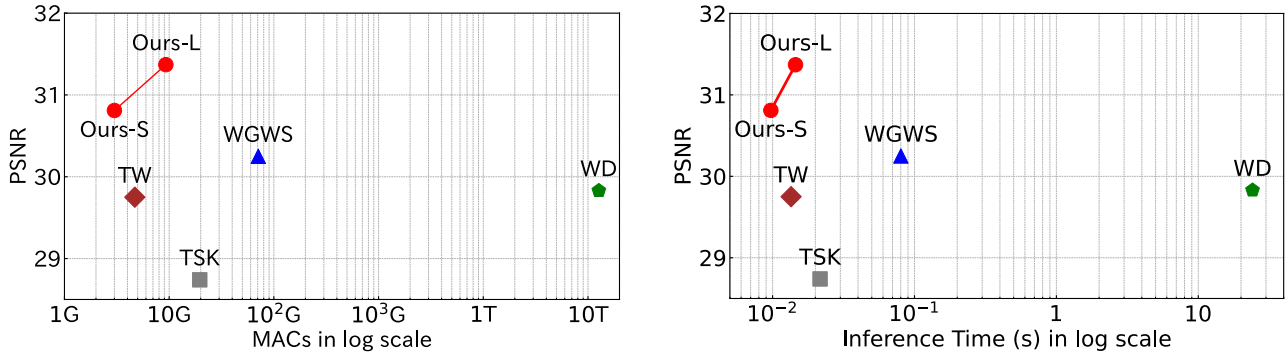


FIGURE 3. PSNR vs. computational complexity MACs (left), and PSNR vs. inference time (right) on All-Weather [13] dataset.

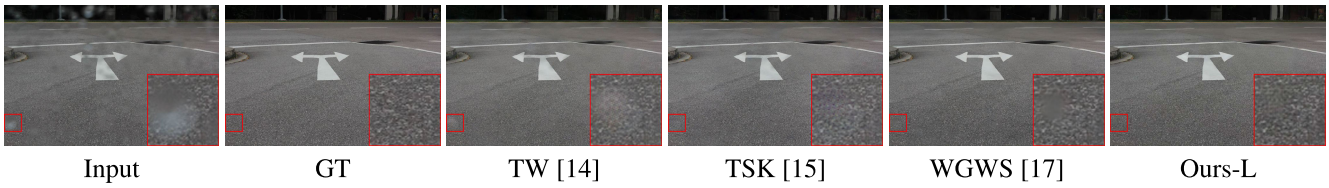


FIGURE 4. Visualization comparison on the synthetic raindrop image of RainDrop [3] dataset. Red boxes correspond to the zoomed-in patches.

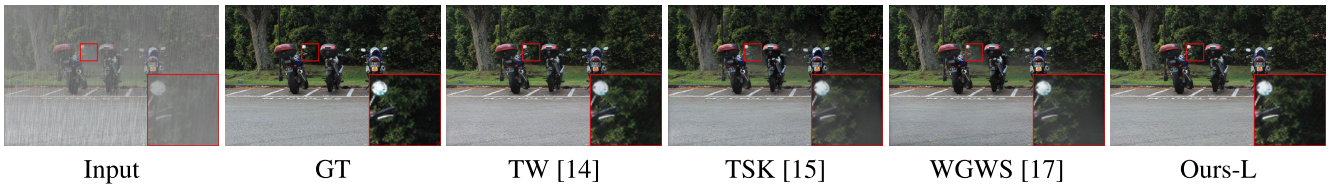


FIGURE 5. Visualization comparison on the synthetic rain and fog image of Test1 [7] dataset. Red boxes correspond to the zoomed-in patches.

for raindrop [1], [2], [3], [4], [25], combined rain with fog [7], [8], [9], [25], [26], snow [5], [6], [10], [11], [12], and multiple weather [14], [15], [16], [17]. For the single weather removal methods, we use the evaluation values reported by WeatherDiffusion (WD) [16]. Regarding multiple weather removal, we choose methods with publicly available code, reporting the evaluation values of models trained by ourselves or publicly provided pre-trained models. In experiments conducted on the recently released WeatherStream [22] dataset, we compare the multiple weather models, excluding WD [16] with long training and inference times. Our method and comparison methods are trained under similar conditions.

B. QUANTITATIVE COMPARISON

The quantitative results in terms of PSNR and SSIM on the synthetic image dataset All-Weather [13] are reported in Tab. 1. Both Ours-S and Ours-L outperform single weather removal models. Compared to previous multiple weather removal models, Ours-S achieves competitive performance, and Ours-L achieves the first or second-best performance in each weather condition. Averaged over all weather conditions, Ours-L surpasses the current state-of-the-art model WGWS [17] by 1.12 dB in PSNR and by +0.0018 in SSIM. Fig. 3 presents the comparison of average PSNR and computational costs between multiple weather removal models. Ours-S outperforms conventional methods with the least amount of computational complexity MACs and the shortest inference time. In addition, Ours-L achieves the best performance with superior computational efficiency.

Tab. 2 depicts the quantitative evaluations on the real image dataset WeatherStream [22]. For each method, PSNR and SSIM averaged over all weather conditions are higher when models initially trained on All-Weather [13] undergo further fine-tuning on WeatherStream, compared to training solely on WeatherStream. Thus, the ability to remove adverse weather degradations, obtained by training on All-Weather, contributes to the improved performance on WeatherStream. Among the different methods, Ours-L achieves the best results regarding PSNR and SSIM across all weather conditions. Ours-L outperforms WGWS [17] by 0.35 dB in PSNR and TSK [15] by 0.0164 in SSIM, compared to the best value of the conventional methods. Following Ours-L, Ours-S ranks second in quantitative performance.

FIGURE 6. Visualization comparison on the synthetic snow image of Snow100K [11] dataset. Red boxes correspond to the zoomed-in patches.

TABLE 2. Quantitative comparison on the real image dataset WeatherStream [22]. Best and second-best values are indicated with bold text and underlined text respectively. In "dataset" column, "WS" denotes models trained on WeatherStream, and "AW → WS" denotes models initially trained on all-weather [13] undergo further trained on WeatherStream.

Method	Dataset	Rain		Fog		Snow		Average	
		PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
TW [14]	WS	24.86	0.7249	24.13	0.7170	23.87	0.7530	24.29	0.7316
	AW → WS	24.85	0.7257	24.32	0.7138	23.92	0.7561	24.36	0.7319
TSK [15]	WS	24.94	0.7249	24.08	0.7113	23.72	0.7559	24.25	0.7307
	AW → WS	25.07	0.7316	24.34	0.7252	24.01	0.7584	24.47	0.7384
WGWS [17]	WS	25.14	0.7226	24.37	0.7123	23.69	0.7508	24.40	0.7286
	AW → WS	25.10	0.7241	<u>24.58</u>	0.7213	23.76	0.7591	24.48	0.7348
Ours-S	WS	24.95	0.7354	24.53	0.7309	23.87	0.7602	24.45	0.7422
	AW → WS	<u>25.36</u>	<u>0.7460</u>	24.34	<u>0.7384</u>	<u>24.07</u>	<u>0.7698</u>	<u>24.59</u>	<u>0.7514</u>
Ours-L	WS	25.10	0.7377	24.50	0.7351	23.96	0.7619	24.52	0.7449
	AW → WS	**25.50**	**0.7499**	**24.85**	**0.7386**	**24.15**	**0.7758**	**24.83**	**0.7548**

FIGURE 7. Visualization comparison on the real-world rain and fog image of WeatherStream [22] dataset. Red boxes correspond to the zoomed-in patches.

FIGURE 8. Visualization comparison on the real-world fog image of WeatherStream [22] dataset. Red boxes correspond to the zoomed-in patches.

C. QUALITATIVE COMPARISON

We conduct the qualitative comparison of weather removal methods, excluding WD [16], which has a long inference time. Fig. 4, 5, 6 visualize the results on synthetic raindrops, rain with fog, and snow. Ours-L removes raindrops more cleanly than TW [14], TSK [15], and WGWS [17]. Additionally, our method works very well in removing fog and snow particles, while other methods fail to remove completely.

Fig. 7, 8, 9 show the restoration results of images taken under real-world weather conditions of rain, fog, and snow. Our method can remove rain and snow particles that previous methods cannot. Regarding fog, our method can generate the clean image, whereas the other methods have noisy results.

D. ABLATION STUDY AND ANALYSIS

In this section, we conduct ablation studies to analyze the effects of each component in the proposed method. Our

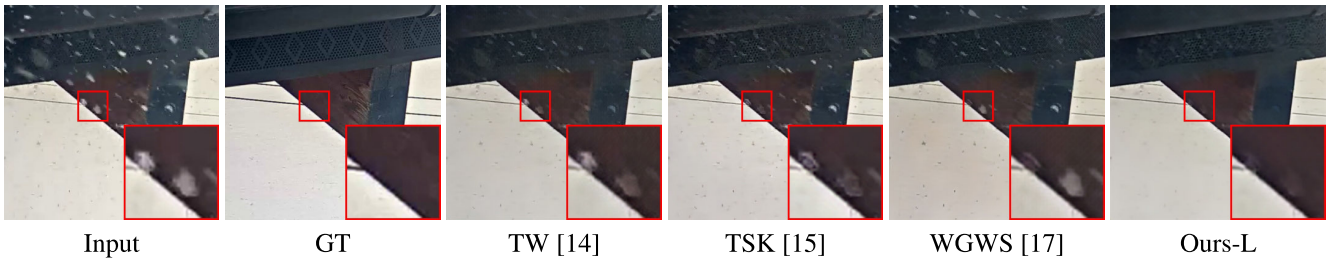


FIGURE 9. Visualization comparison on the real-world snow image of WeatherStream [22] dataset. Red boxes correspond to the zoomed-in patches.

TABLE 3. Ablation study about masked-based pre-training.

Masked-based pre-training	PSNR ↑	SSIM ↑
	30.13	0.9151
✓	**30.81**	**0.9205**

TABLE 4. Comparison of the proposed dual-pooling adaptive convolution (DPAC), standard convolution, and previous adaptive convolutions.

Architecture	PSNR ↑	SSIM ↑
standard conv	30.46	0.9199
CondConv [32]	30.50	0.9198
WeightNet [21]	30.60	0.9201
DDF [34]	30.59	0.9202
DPAC (Ours)	**30.81**	**0.9205**

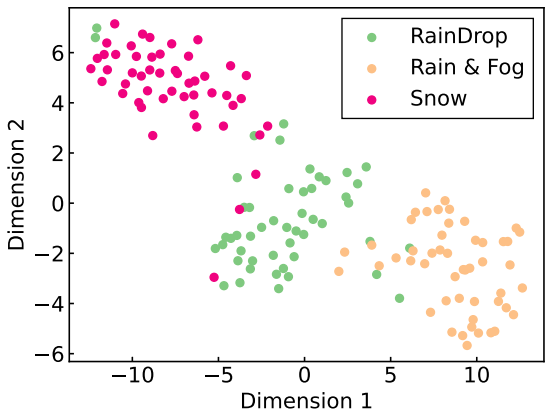


FIGURE 10. Visualization of dual-pooling adaptive convolution (DPAC) weights in the well-trained Ours-S by t-SNE [44]. The weights are visualized by different colors when the model processes three types of weather images from All-Weather [13] dataset. Each point represents the weights of one image.

smaller model, referred to as Ours-S, and All-Weather [13] dataset are used for evaluation. We report PSNR and SSIM averaged over all weather conditions.

1) MASKED-BASED PRE-TRAINING

Tab. 3 shows that PSNR and SSIM are improved by employing masked-based pre-training. This indicates that knowledge of feature representations acquired through large-scale pre-training with FCMAE [20] is effective for adverse weather removal.

TABLE 5. Ablation study about dual-pooling adaptive convolution (DPAC).

Architecture	PSNR ↑	SSIM ↑
Full	**30.81**	**0.9205**
- Avg Pooling	30.60	0.9200
- Max Pooling	30.69	0.9204
- FGRN	30.65	**0.9205**

TABLE 6. Comparison of depth-wise convolution kernel sizes in reconstruction decoder.

kernel size	PSNR ↑	SSIM ↑
3	**30.81**	**0.9205**
5	30.67	0.9204
7	30.58	0.9198

TABLE 7. Ablation study about loss function.

Architecture	PSNR ↑	SSIM ↑
Full	**30.81**	**0.9205**
- Smooth L1 Loss	30.59	**0.9205**
- Perceptual Loss	30.41	0.9202
- SSIM Loss	30.71	0.9122

2) DUAL-POOLING ADAPTIVE CONVOLUTION

We compare the use of dual-pooling adaptive convolution (DPAC) with a standard convolution and some existing adaptive convolutions, such as CondConv [32], WeightNet [21], and DDF [34]. The results for replacing DPAC in the middle network with these convolutions are shown in Tab. 4. Using DPAC instead of the standard convolution yields higher PSNR and SSIM. This suggests that processing convolution with adaptive weights corresponding to the input features helps remove multiple adverse weather degradations. In addition, DPAC outperforms all other adaptive convolutions.

Tab. 5 presents an ablation study about DPAC. DPAC aggregates input features using average and max pooling, but removing either pooling degrades performance. Removing filter-wise global response normalization (FGRN) also degrades performance. These results suggest that each component of DPAC functions effectively.

Fig. 10 depicts visualization of the DPAC weights. We randomly select 50 images each of raindrops, combined rain with fog, and snow from test set of All-Weather [13] dataset, and input them into the trained model. The DPAC weights in the middle network's initial block are projected into a

two-dimensional space using t-SNE [44]. The visualization demonstrates that DPAC assigns varying weights to distinct images. For images under identical adverse weather, DPAC generates similar weights. This suggests that DPAC adapts its processing based on degradation characteristics of input images.

3) RECONSTRUCTION DECODER

Tab. 6 presents experimental results for the structure of the reconstruction decoder block. In the depth-wise convolution, a kernel size of 3 yields the highest performance compared to 5 and 7. Hence, our method sets the kernel size of the depth-wise convolution to 3.

4) LOSS FUNCTION

As shown in Tab. 7, removing Smooth L1 Loss, Perceptual Loss [37], or SSIM Loss [39] causes performance degradation, indicating that any of the loss functions are working effectively.

V. CONCLUSION

We present an efficient adverse weather removal model. It exploits knowledge of feature representations acquired through mask-based pre-training. It also efficiently handles multiple weather degradations by our proposed dual-pooling adaptive convolution (DPAC). Experiments on synthetic and real-world image datasets demonstrate that our method achieves encouraging performance compared to other state-of-the-art methods. The proposed method, characterized by its rapid inference without specifying weather types, shows significant potential for application in real-world, real-time image processing scenarios.

REFERENCES

[1] X. Liu, M. Suganuma, Z. Sun, and T. Okatani, "Dual residual networks leveraging the potential of paired operations for image restoration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7000–7009.

[2] Y. Quan, S. Deng, Y. Chen, and H. Ji, "Deep learning for seeing through window with raindrops," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2463–2471.

[3] R. Qian, R. T. Tan, W. Yang, J. Su, and J. Liu, "Attentive generative adversarial network for raindrop removal from a single image," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2482–2491.

[4] J. Xiao, X. Fu, A. Liu, F. Wu, and Z.-J. Zha, "Image de-raining transformer," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–18, 2022.

[5] X. Li, J. Wu, Z. Lin, H. Liu, and H. Zha, "Recurrent squeeze-and-excitation context aggregation net for single image deraining," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 254–269.

[6] T. Wang, X. Yang, K. Xu, S. Chen, Q. Zhang, and R. W. H. Lau, "Spatial attentive single-image deraining with a high quality real rain dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12262–12271.

[7] R. Li, L.-F. Cheong, and R. T. Tan, "Heavy rain image restoration: Integrating physics model and conditional adversarial learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1633–1642.

[8] K. Jiang, Z. Wang, P. Yi, C. Chen, Z. Wang, X. Wang, J. Jiang, and C.-W. Lin, "Rain-free and residue hand-in-hand: A progressive coupled network for real-time image deraining," *IEEE Trans. Image Process.*, vol. 30, pp. 7404–7418, 2021.

[9] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang, and L. Shao, "Multi-stage progressive image restoration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14816–14826.

[10] W.-T. Chen, H.-Y. Fang, J.-J. Ding, C.-C. Tsai, and S.-Y. Kuo, "JSTASR: Joint size and transparency-aware snow removal algorithm based on modified partial convolution and veiling effect removal," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 754–770.

[11] Y.-F. Liu, D.-W. Jaw, S.-C. Huang, and J.-N. Hwang, "DesnowNet: Context-aware deep network for snow removal," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 3064–3073, Jun. 2018.

[12] K. Zhang, R. Li, Y. Yu, W. Luo, and C. Li, "Deep dense multi-scale network for snow removal using semantic and depth priors," *IEEE Trans. Image Process.*, vol. 30, pp. 7419–7431, 2021.

[13] R. Li, R. T. Tan, and L.-F. Cheong, "All in one bad weather removal using architectural search," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3172–3182.

[14] J. M. Jose Valanarasu, R. Yasarla, and V. M. Patel, "TransWeather: Transformer-based restoration of images degraded by adverse weather conditions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 2343–2353.

[15] W.-T. Chen, Z.-K. Huang, C.-C. Tsai, H.-H. Yang, J.-J. Ding, and S.-Y. Kuo, "Learning multiple adverse weather removal via two-stage knowledge learning and multi-contrastive regularization: Toward a unified model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 17632–17641.

[16] O. Özdenizci and R. Legenstein, "Restoring vision in adverse weather conditions with patch-based denoising diffusion models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 8, pp. 10346–10357, Aug. 2023.

[17] Y. Zhu, T. Wang, X. Fu, X. Yang, X. Guo, J. Dai, Y. Qiao, and X. Hu, "Learning weather-general and weather-specific features for image restoration under multiple adverse weather conditions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 21747–21758.

[18] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 15979–15988.

[19] J. Liu, X. Huang, J. Zheng, Y. Liu, and H. Li, "MixMAE: Mixed and masked autoencoder for efficient pretraining of hierarchical vision transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 6252–6261.

[20] S. Woo, S. Debnath, R. Hu, X. Chen, Z. Liu, I. S. Kweon, and S. Xie, "ConvNeXt v2: Co-designing and scaling ConvNets with masked autoencoders," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 16133–16142.

[21] N. Ma, X. Zhang, J. Huang, and J. Sun, "WeightNet: Revisiting the design space of weight networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 776–792.

[22] H. Zhang, Y. Ba, E. Yang, V. Mehra, B. Gella, A. Suzuki, A. Pfahnl, C. C. Chandrappa, A. Wong, and A. Kadambi, "WeatherStream: Light transport automation of single image deweathering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 13499–13509.

[23] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[24] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, vol. 2, 2014, pp. 2672–2680.

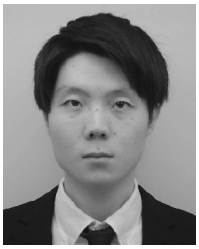
[25] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5967–5976.

[26] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2242–2251.

[27] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent.*, 2021, pp. 1–21.

[28] B. Graham and L. van der Maaten, "Submanifold sparse convolutional networks," 2017, *arXiv:1706.01307*.

[29] B. Graham, M. Engelcke, and L. van der Maaten, "3D semantic segmentation with submanifold sparse convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9224–9232.
[30] C. Choy, J. Gwak, and S. Savarese, "4D spatio-temporal ConvNets: Minkowski convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3070–3079.
[31] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[32] B. Yang, G. Bender, Q. V. Le, and J. Ngiam, "CondConv: Conditionally parameterized convolutions for efficient inference," in *Proc. Conf. Workshop Neural Inf. Process. Syst.*, 2019, pp. 1305–1316.
[33] Y. Chen, X. Dai, M. Liu, D. Chen, L. Yuan, and Z. Liu, "Dynamic convolution: Attention over convolution kernels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11027–11036.
[34] J. Zhou, V. Jampani, Z. Pi, Q. Liu, and M.-H. Yang, "Decoupled dynamic filter networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 6643–6652.
[35] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 11966–11976.
[36] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," 2016, *arXiv:1606.08415*.
[37] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 694–711.
[38] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent.* San Diego, CA, USA: Computational and Biological Learning Society, 2015, pp. 1–14.
[39] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 47–57, Mar. 2017.
[40] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[41] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 8024–8035.
[42] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–15.
[43] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment," *Electron. Lett.*, vol. 44, no. 13, pp. 800–801, 2008.
[44] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.



SHUGO YAMASHITA received the B.E. degree in electronics and electrical engineering from Keio University, Yokohama, Japan, in 2024, where he is currently pursuing the M.E. degree, under the supervision of Prof. Masaaki Ikehara. His research interests include machine learning, image processing, and computer vision.

MASAAKI IKEHARA (Senior Member, IEEE) received the B.E., M.E., and Ph.D. degrees in electrical engineering from Keio University, in 1984, 1986, and 1989, respectively. He is currently a Full Professor with the Department of Electronics and Electrical Engineering, Keio University. His research interests include multi-rate signal processing, wavelet image coding, and filter design problems.