DEA-Net: Single Image Dehazing Based on Detail-Enhanced Convolution and Content-Guided Attention

Zixuan Chen, Zewei He, and Zhe-Ming Lu, *Senior Member, IEEE*

***Abstract*— Single image dehazing is a challenging ill-posed problem which estimates latent haze-free images from observed hazy images. Some existing deep learning based methods are devoted to improving the model performance via increasing the depth or width of convolution. The learning ability of Convolutional Neural Network (CNN) structure is still under-explored. In this paper, a Detail-Enhanced Attention Block (DEAB) consisting of Detail-Enhanced Convolution (DEConv) and Content-Guided Attention (CGA) is proposed to boost the feature learning for improving the dehazing performance. Specifically, the DEConv contains difference convolutions which can integrate prior information to complement the vanilla one and enhance the representation capacity. Then by using the re-parameterization technique, DEConv is equivalently converted into a vanilla convolution to reduce parameters and computational cost. By assigning the unique Spatial Importance Map (SIM) to every channel, CGA can attend more useful information encoded in features. In addition, a CGA-based mixup fusion scheme is presented to effectively fuse the features and aid the gradient flow. By combining above mentioned components, we propose our Detail-Enhanced Attention Network (DEA-Net) for recovering high-quality haze-free images. Extensive experimental results demonstrate the effectiveness of our DEA-Net, outperforming the state-of-the-art (SOTA) methods by boosting the PSNR index over 41 dB with only 3.653 M parameters. (The source code of our DEA-Net is available at https://github.com/cecret3350/DEA-Net.)**

***Index Terms*— Image dehazing, detail-enhanced convolution, content-guided attention, fusion scheme.**

I. Introduction

IMAGES captured under hazy scenes usually suffer from noticeable visual quality degradation in contrast or color distortion [1], leading to significant performance drop when inputting to some high-level vision tasks (e.g., object detection, semantic segmentation). Haze-free images are highly demanded or required among these tasks. Therefore, single image dehazing, which aims to recover the clean scene from the corresponding hazy image, has attracted significant attention among both the academic and industrial communities over the past decade. As a fundamental low-level image restoration task, image dehazing is usually considered as the pre-processing step of the subsequent high-level vision tasks. In this paper, we attempt to develop an effective algorithm to remove the haze and recover the details from the hazy input.

Recently, with the rapid development of deep learning, Convolution Neural Network (CNN) based dehazing methods achieve superior performance [2], [3], [4], [5], [6]. Earlier CNN-based methods [2], [7], [8] first estimate the transmission map and the atmospheric light separately, and then utilize the Atmospheric Scattering Model (ASM) [9] to derive the haze-free images. Typically, the transmission map is supervised by the ground truth, which is used for synthesizing the training dataset. However, inaccurate estimation of the transmission map or the atmospheric light would significantly influence the image restoration results. More recently, some methods [6], [10], [11] prefer to predict the latent haze-free images in an end-to-end manner since it tends to achieve promising results.

However, there still exists two main issues:

(1) *Less effectiveness of vanilla convolution.* Most of existing dehazing methods [5], [6], [12] adopt classical convolution layers for feature extraction without utilizing any priors. However, vanilla convolutions search the vast solution space without any constrains, which to some extent may limit the expressive ability (or modeling capacity). For example, the vanilla convolution tends to accentuate the low-frequency information and ignore the high-frequency information (refer to Appendix A). We argue that both low- and high-frequency information can facilitate haze removal. In addition, some transformer-based methods [13] expand the receptive field to the whole image for mining long-distance dependencies. They can enhance the expressive ability (or modeling capacity) at the cost of complex training strategy and tedious hyper-parameter tuning. Also, the prohibitive computational cost and vast GPU memory occupation cannot be ignored. In this regard, the ideal solution is to design a computation-friendly convolution layer that can simultaneously handle low- and high-frequency information.

(2) *Haze non-uniformity.* There are two kinds of non-uniformity in the dehazing problem: uneven haze distribution in image level and channel-wise haze difference in feature

Manuscript received 13 January 2023; revised 13 October 2023 and 17 December 2023; accepted 8 January 2024. Date of publication 22 January 2024; date of current version 25 January 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 52305590, in part by the National Key Research and Development Program of China under Grant 2020AAA0140004 and Grant 2022YFB3304204, in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LQ24F010004, in part by the China Postdoctoral Science Foundation under Grant 2022M712792, and in part by the Hangzhou Major Scientific and Technological Innovation Projects under Grant 2022AIZD0146. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Siwei Ma. *(Zixuan Chen and Zewei He contributed equally to this work.) (Corresponding author: Zhe-Ming Lu.)*

The authors are with the School of Aeronautics and Astronautics, Zhejiang University, Hangzhou 310027, China (e-mail: 22224039@zju.edu.cn; zeweihe@zju.edu.cn; zheminglu@zju.edu.cn).

Digital Object Identifier 10.1109/TIP.2024.3354108

level. To cope with the first one, Qin et al. [5] employed a pixel attention (i.e., spatial attention) to generate a Spatial Importance Map (SIM), which can adaptively indicate the importance levels of different pixel locations. Through this discriminative strategy, the FFA-Net model treats thin and thick haze regions unequally. Similarly, Ye et al. [11] tried to model the density of haze distribution via a density estimation module, which essentially is also a spatial attention. However, few researchers have paid attention to the non-uniformity in feature level, and it remains under-addressed. The haze information is encoded into the feature maps after applying convolution layers. Different channels in the feature space have different meanings depending on the role of the filters applied. For example, some filters highlight the hazy region, and some filters may weaken it reversely. Considering this, we argue that spatial importance maps should be channel-specific to handle feature level non-uniformity. A new attention mechanism that can treat each channel differently is needed.

To address above mentioned issues, we design a Detail-Enhanced Attention Block (DEAB), which consists of a Detail-Enhanced Convolution (DEConv) and a Content-Guided Attention (CGA) mechanism. The DEConv contains five convolution layers (four difference convolutions [14] and one vanilla convolution), which are deployed in parallel for feature extraction. Specifically, a Central Difference Convolution (CDC), an angular difference convolution (ADC), a Horizontal Difference Convolution (HDC), and a Vertical Difference Convolution (VDC) are adopted to integrate traditional local descriptors into the convolution layer, thus can enhance the representation and generalization capacity. In difference convolutions, the pixel differences in the image are firstly calculated, and then convolved with the convolution kernel to generate the output feature maps. The strategy of pixel pair's difference calculation can be designed to explicitly encode prior information into CNN. For instance, HDC and VDC explicitly encode the gradient prior into the convolution layers via learning beneficial gradient information.

Besides, we re-parameterize the learned kernel weights of the parallel convolutions to reduce the number of parameters and accelerate the training and testing process. The five parallel convolutions are simplified into one vanilla convolution layer with applying some constrains to the kernel weights and by using the linear property of convolution layers. Therefore, the proposed DEConv can extract rich features for improving dehazing performance while keeping the number of parameters and computational cost equal to the vanilla convolution. Fig. 1 shows the efficiency and effectiveness of our method.

Moreover, the Content-Guided Attention (CGA) is a two-step attention generator, which can produce the coarse spatial attention map firstly and then refine it to the fine version. Specifically, given certain input feature maps, we utilize the spatial attention mechanism presented in [15] and the channel attention presented in [16] to generate the initial SIMs (i.e., the coarse version). The novel idea of CGA is to utilize the input features in each channel to refine the corresponding initial SIM of this channel. Accordingly, every channel of initial SIM is refined with the guidance of corresponding channel of features to produce the final SIM. By using the content

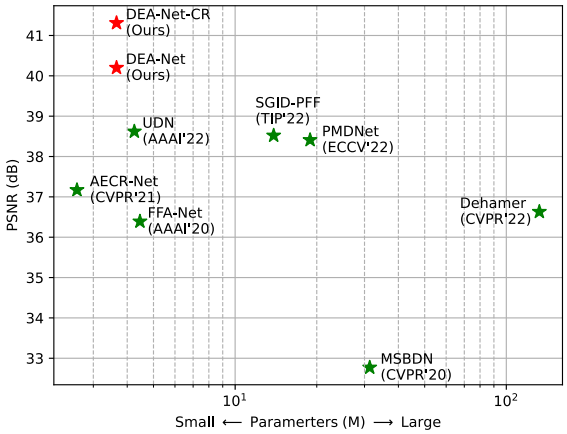

Fig. 1. PSNR vs. number of parameters for various algorithms. We compare our DEA-Net with some state-of-the-art methods (after 2020). The results are tested on the SOTS-indoor dataset. Note that AECR-Net adopts a sharing strategy to reduce the number of parameters.

of input features to guide the generation of SIMs, CGA can focus on the unique part of features in each channel. It is worth mentioning that CGA as a universal basic block can be plug into neural networks to improve the performance in various image restoration tasks.

Following [6], [10], [17], and [18], we also adopt a U-net-like framework to make the major time-consuming convolution computations in the low-resolution space. Among them, the fusion of shallow and deep features is widely used. Feature fusion can enhance the information flow from shallow layers to deep ones, which is effective for feature preserving and gradient back-propagation. The information encoded in the shallow features is tremendously different from the information encoded in the deep features, since the diverse receptive fields. One single pixel in the deep features are originated from a region of pixels in the shallow features. Simple addition or concatenation operation is unable to solve the receptive field mismatch problem. We further propose a CGA-based mixup scheme to adaptively fuse the low-level features in the encoder part with corresponding high-level features, by modulating the features via learned spatial weights.

The diagram of our proposed method is shown in Fig. 2. We term the proposed single image dehazing model as DEA-Net by introducing the **D**etail-**E**nhanced **A**ttention **B**lock (DEAB) with the **D**etail-**E**nhanced convolution and the content-guided **A**ttention.

To conclude, we have following main contributions:

- To the best of our knowledge, it is the first time that difference convolutions (DCs) are introduced to solve the image dehazing problem. Specifically, a Detail-Enhanced Convolution (DEConv), which contains parallel vanilla and difference convolutions, is designed. The DC encodes well-designed prior to accentuate the high-frequency information and the vanilla convolution has the complementary function. DEConv is able to combine both advantages. In addition, the re-parameterization technique is successfully applied to image dehazing domain for the

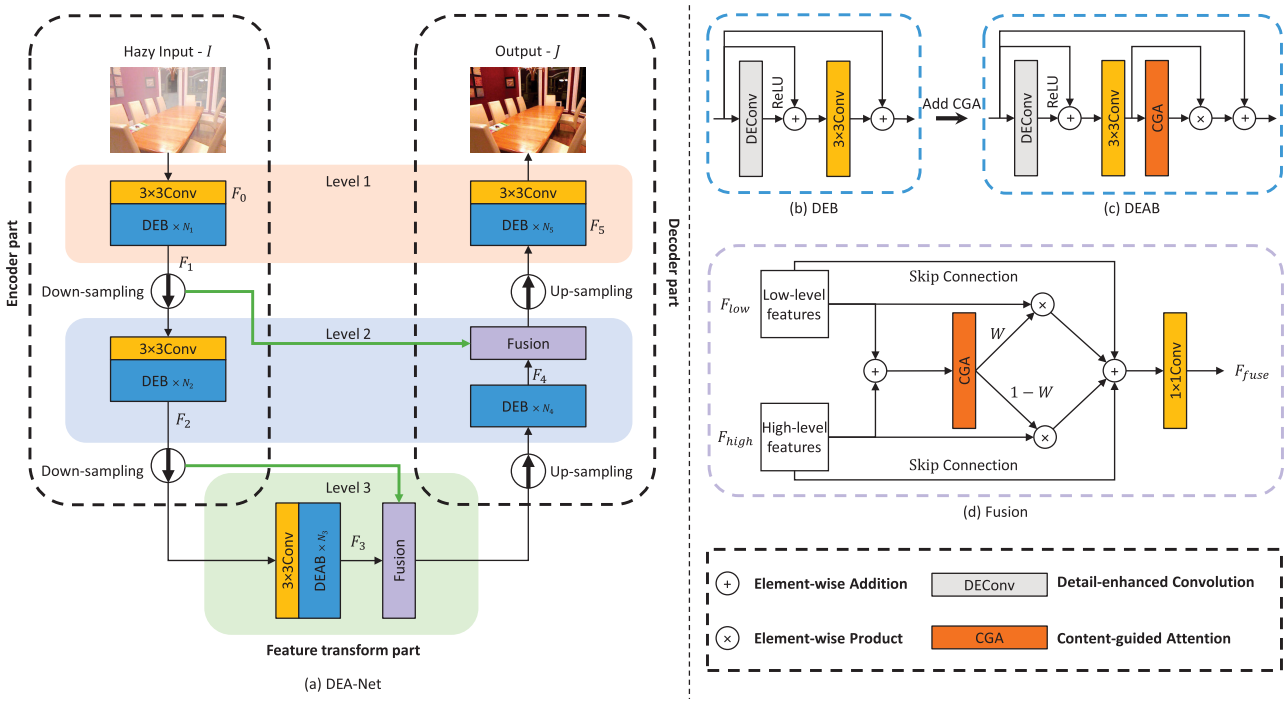

Fig. 2. The overall architecture of our detail-enhanced attention network (DEA-Net), which is a three-level encoder-decoder-like architecture. DEA-Net contains three parts: the encoder part, the feature transform part, and the decoder part. We deploy detail-enhanced attention blocks (DEABs) in the feature transform part, and deploy detail-enhanced blocks (DEBs) in the rest parts.

first time to facilitate the reductions of parameters and computational cost.

- We propose a novel attention mechanism called Content-Guided Attention (CGA) to generate the channel-specific SIMs in a coarse-to-fine manner. By using input features to guide the generation of SIMs, CGA assigns unique SIM to every channel, making the model attend significant regions of each channel. Thus, more useful information encoded in features can be emphasized to effectively improve the performance. Moreover, a CGA-based mixup fusion scheme is presented to effectively fuse the low-level features in the encoder part with corresponding high-level features.
- By combining DEConv and CGA, and using CGA-based mixup fusion scheme, we propose our Detail-Enhanced Attention Network (DEA-Net) for reconstructing high-quality haze-free images. DEA-Net shows superior performance over the state-of-the-art dehazing methods on multiple benchmark datasets, achieving more accurate results with faster inference speed.

The remainder of this paper is organized as follows. We first review a number of deep learning-based dehazing methods in Section II. Then, Section III describes the proposed DEA-Net model in detail, Section IV shows the experimental results, and Section V discusses the limitation and future work. Finally, Section VI concludes the whole paper.

II. Related Work

A. Single Image Dehazing

Existing methods of single image dehazing can be classified into prior-based methods and data-driven methods. Prior-based methods manually model the statistical discrepancy between the hazy and haze-free images as empirical priors. On the contrary, data-driven methods seek to learn the mapping function directly or indirectly based on large-scale datasets.

Prior-based methods are the pioneers of image dehazing. They usually rely on atmospheric scattering model (ASM) [9] and handcraft priors. The widely known priors include dark channel prior (DCP) [19], [20], non-local prior (NLP) [21], color attenuation prior (CAP) [22], etc. Specifically, He et al. [19], [20] propose DCP, leveraging a crucial observation that most local patches in haze-free outdoor images contain pixels with very low intensities in at least one color channel, facilitating transmission map estimation. CAP [22] starts from the HSV color model, and establishes a linear relationship between depth and the difference in brightness and saturation. Berman et al. [21] found that pixel clusters of haze-free images transform into haze-lines when haze presents. While prior-based methods have achieved promising dehazing results, their effectiveness is often limited to specific scenarios that align with their underlying assumptions.

In recent years, the field of image dehazing has witnessed a surge in the application of deep learning techniques as they demonstrated superior performance. For instance, DehazeNet [2], MSCNN [7] and MSCNN-HE [23] employ CNNs for transmission map estimation. Subsequently, AOD-Net [3] rewrites the ASM and simultaneously estimates atmospheric light and the transmission map. DCPDN [8] estimates the transmission map and atmospheric light by two different networks. However, the cumulative errors introduced by inaccurate estimations of transmission map and atmospheric light may cause the performance degradation.

PFDN [24] embeds physical constraints in feature space, thereby avoiding cumulative errors at image-level. PhysicsGAN [25] physically re-degrades reconstructed outputs, establishing consistency between hazy and the predicted images.

To take a step further, some methods entirely abandon the physical model. GFN [26] gates and fuses three enhanced images from original hazy inputs to generate haze-free images. GridDehazeNet [27] utilizes a three-stage attention-based grid network for haze-free image recovery. MSBDN [10] employs boosting strategy and back-projection technique to enhance the feature fusion. FFA-Net [5] introduces the feature attention mechanism (FAM) to dehazing network to deal with various types of information. AECR-Net [6] reuses the feature attention block (FAB) [5] and proposes a novel contrastive regularization, which leverages both positive and negative samples. UDN [18] analyzes two types of uncertainty in image dehazing to increase the dehazing performance. PMDNet [11] and Dehamer [13] adopt transformer to build long-range dependencies and perform dehazing with the guidance of haze density. However, as data-driven methods develop and dehazing performance improves, the complexity of dehazing networks also increases. Different from previous works, we rethink the limitations of vanilla convolution in image dehazing and design a novel convolution operator by combining well-designed priors. Additionally, we delve deeper into the unexploited non-uniformity of haze at the feature level. By harnessing the potential of CNNs, we achieve superior dehazing performance while maintaining low overheads.

B. Difference Convolution

The origin of difference convolutions can be traced back to the Local Binary Pattern (LBP) [28], which encodes the pixel differences in the local patch to a decimal number for texture classification. Since the success of CNNs in computer vision tasks, Xu et al. [29] proposed the Local Binary Convolution (LBC) which encodes the pixel differences by using non-linear activation functions and linear convolution layers. Recently, Yu et al. [14] proposed the Central Difference Convolution (CDC) to directly encode the pixel differences with completely learnable weights. Later, various forms of difference convolutions have been proposed, such as cross central difference convolution [30] and pixel difference convolution [31]. Considering the nature of the difference convolution for capturing gradient-level information, we firstly introduce it to single image dehazing for improving the performance.

III. Methodology

As shown in Fig. 2, our DEA-Net consists of three parts: encoder part, feature transform part, and decoder part. As the core of our DEA-Net, the feature transform part adopts stacked Detail-Enhanced Attention Blocks (DEABs) to learn haze-free features. There are three levels in the hierarchical structure, and we employ different blocks in different levels to extract corresponding features (level 1&2: DEB, level 3: DEAB). Given a hazy input image $I \in \mathbb{R}^{3\times H\times W}$, the goal of DEA-Net is to restore the corresponding haze-free image $J \in \mathbb{R}^{3\times H\times W}$.

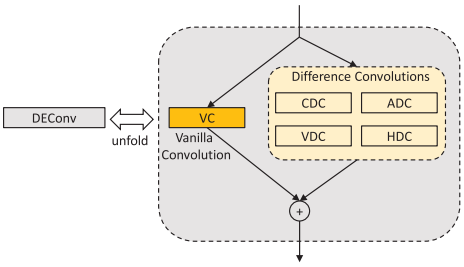

Fig. 3. Detail-enhanced convolution (DEConv). It contains five convolution layers deployed in parallel, i.e., a vanilla convolution (VC), a central difference convolution (CDC), an angular difference convolution (ADC), a horizontal difference convolution (HDC), and a vertical difference convolution (VDC).

A. Detail-Enhanced Convolution

In the single image dehazing domain, previous methods [5], [6], [12] usually utilize vanilla convolution (VC) layers for feature extraction and learning. Normal convolution layers search the vast solution space without any constrains (even start from random initialization), restricting the expressive ability or modeling capacity. From the viewpoint of image dehazing, the existence of haze will cause illumination and color change, which is low-frequency. Besides, natural scenes covered by haze may lose some high-frequency details. Then we notice that the low-frequency information is of great importance in correcting the illumination and color distribution and the high-frequency information (e.g., edges and contours) is also crucial in recovering missing fine details. However, vanilla convolutions tend to emphasize the low-frequency information and neglect the high-frequency information (refer to Appendix A). Previously, some researchers [8], [17], [32] adopted the edge prior in the dehazing model to help restore sharper contours. Inspired by their works [8], [32] and to take a further step, we design a Detail-Enhanced Convolution (DEConv) layer (see in Fig. 3) to take low- and high-frequency components into consideration by integrating well-designed priors into certain vanilla convolution layers (i.e., combine difference convolution layers with the complementary vanilla convolution layer in parallel).

Before elaborating the proposed DEConv in detail, we first recap the Difference Convolution (DC). Previous works [14], [30], [31], [33] usually describe the difference convolution as the convolution of pixel differences (the pixel differences are firstly calculated, and then convolved with the kernel weights to generate feature maps), which can enhance the representation and generalization capacity of vanilla convolution. Central Difference Convolution (CDC) and Angular Difference Convolution (ADC) are two kinds of typical DCs, and implemented by re-arranging learned kernel weights to save computational cost and memory consumption [31]. It proves to be effective for edge detection [31] and face anti-spoofing tasks [14], [30], [33]. **To the best of our knowledge, it is the first time that we introduce DC to solve the single image dehazing problem.**

In our implementation, we employ five convolution layers (four DCs [14] and one vanilla convolution), which are deployed in parallel for feature extraction. In DCs, the strategy of pixel pair's difference calculation can be designed

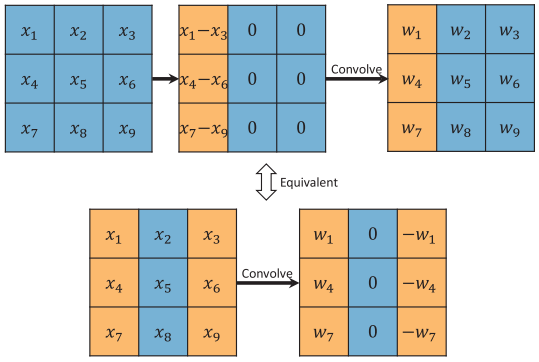

Fig. 4. The derivation of horizontal difference convolution (HDC).

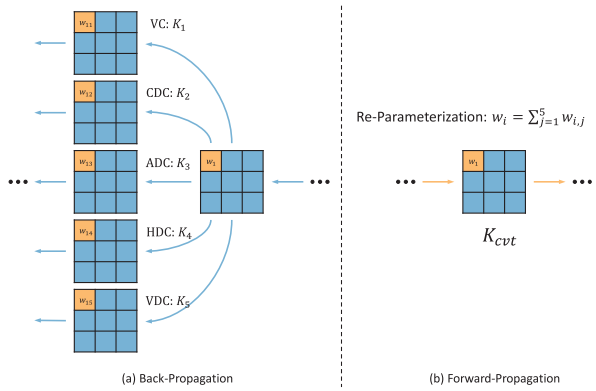

Fig. 5. The process of our re-parameterization technique.

to explicitly encode prior information into CNN. For our DEConv, besides the Central Difference Convolution (CDC) and the Angular Difference Convolution (ADC), we derive the Horizontal Difference Convolution (HDC) and the Vertical Difference Convolution (VDC) to integrate traditional local descriptors (like Sobel [34], Prewitt [35], or Scharr [36]) into the convolution layer. As shown in Fig. 4, taking HDC as an example, the horizontal gradient is firstly calculated by computing the differences of selected pixel pairs. After training, we re-arrange the learned kernel weights equivalently, and apply convolution directly to the untouched input features. Note that, the equivalent kernel has the similar format of traditional local descriptors (the sum of horizontal weights equals to zero). Horizontal kernels of Sobel [34], Prewitt [35], and Scharr [36] can be regarded as the special case of the equivalent kernel. VDC has the similar derivation by changing the horizontal gradient to corresponding vertical counterpart. Both HDC and VDC explicitly encode the gradient prior into the convolution layers to enhance the representation and generalization capacity via learning beneficial gradient information.

In our design, the vanilla convolution serves to obtain the intensity-level information while the difference convolutions are used to enhance gradient-level information. We simply add the learned features together to obtain the output of DEConv. We trust more sophisticated designs of the way for calculating the pixel difference can further benefit image restoration task, which is not the main direction of this paper.

However, deploying five parallel convolution layers for feature extraction will undesirably cause the increase of parameters and inference time. We seek to exploit the additivity of convolution layers for simplifying the convolutions deployed in parallel into a single standard convolution. We notice a useful property of the convolution: if several 2D kernels with the identical size operate on the same input with the same stride and padding to produce outputs, and their outputs are summed up to obtain the final output, we can add up these kernels on the corresponding positions to obtain an equivalent kernel which will produce the identical final output. Surprisingly, our DEConv exactly fits this situation. Given the input features F_{in}, DEConv can output F_{out} with identical computational cost and inference time to a vanilla convolution layer by utilizing re-parameterization technique. The formula is as follows (the biases are omitted for simplification):

$$\begin{aligned} F_{out} &= DEConv(F_{in}) = \sum_{i=1}^{5} F_{in} * K_i \\ &= F_{in} * (\sum_{i=1}^{5} K_i) = F_{in} * K_{cvt}, \end{aligned} \tag{1}$$

where $DEConv(\cdot)$ denotes the operation of our DEConv, $K_{i=1:5}$ represent the kernels of VC, CDC, ADC, HDC, and VDC, respectively, $*$ denotes the convolution operation, and K_{cvt} denotes the converted kernel, which combines the parallel convolutions together.

Fig. 5 visually shows the process of our re-parameterization technique. In the back-propagation phase, the kernel weights of five parallel convolutions are updated separately using the chain rule of gradient propagation. In the forward-propagation phase, the kernel weights of the parallel convolutions are fixed and the converted kernel weights are calculated by adding up them on the corresponding positions. Note that, the re-parameterization technique can accelerate the training and testing process simultaneously, since both of them contain the forward-propagation phase.

Compare with the vanilla convolution layer, the proposed DEConv can extract richer features while maintaining the parameter size, and introduces no extra computational cost and memory burdens in the inference stage. More discussions about DEConv can be found in Section IV-C.1.

B. Content-Guided Attention

Feature attention module (FAM) [5] consists of a channel attention and a spatial attention, which are sequentially placed to calculate the attentive weights in channel and spatial dimensions. The channel attention calculates a channel-wise vector, i.e., $W_c \in \mathbb{R}^{C\times1\times1}$, to re-calibrate the features. The spatial attention calculates a Spatial Importance Map (SIM), i.e., $W_s \in \mathbb{R}^{H\times W}$ to adaptively indicate the informative regions. The FAM treats different channels and pixels unequally, improving the dehazing performance.

We argue there are two main disadvantages: (1) no information exchange between these two attentive weights. W_c and W_s are sequentially calculated to enhance the features separately. (2) the spatial attention inside FAM can only address the

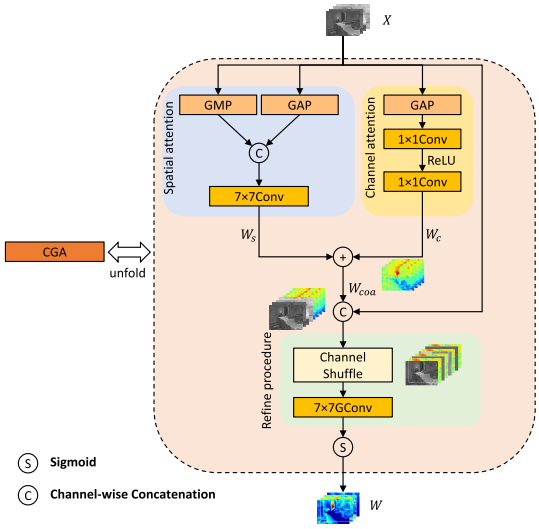

Fig. 6. The diagram of content-guided attention (CGA). CGA is a coarse–to-fine process: the coarse version of SIMs (i.e., $W_{coa} \in \mathbb{R}^{C\times H\times W}$) is generated firstly and then every channel is refined by the guidance of input features.

uneven haze distribution in the image level with one-channel W_s, and ignore the uneven distribution in the feature level. With the expanding of feature channels, the image-level haze distribution information is encoded into the feature maps. Different channels in the feature space have different meanings depending on the role of the filters applied. That means for every channel of features, the haze information is unevenly spread across the spatial dimensions. The channel-specific SIMs are desired in this situation.

To fully address the problems mentioned above, we propose a content-guided attention (CGA) mechanism to obtain the exclusive SIM for every single channel of input features in a coarse-to-fine manner, meanwhile fully mix channel attention weights and spatial attention weights to guarantee information interaction. The detailed procedures of CGA are illustrated in Fig. 6, let $X \in \mathbb{R}^{C\times H\times W}$ denote the proceeding input features, the goal of CGA is to generate channel-specific SIMs (i.e., $W \in \mathbb{R}^{C\times H\times W}$), which have the identical dimensions with X.

We first compute the corresponding W_c and W_s by following formulas [15], [16].

$$\begin{aligned} W_c &= \mathcal{C}_{1\times1}(max(0, \mathcal{C}_{1\times1}(X^c_{GAP}))), \\ W_s &= \mathcal{C}_{7\times7}([X^s_{GAP}, X^s_{GMP}]), \end{aligned} \tag{2}$$

where $max(0, x)$ denotes the ReLU activation function, $\mathcal{C}_{k\times k}(\cdot)$ denotes a convolution layer with $k \times k$ kernel size, $[\cdot]$ denotes the channel-wise concatenation operation. X^c_{GAP}, X^s_{GAP}, and X^s_{GMP} denote the features processed by global average pooling operation across the spatial dimensions, global average pooling operation across the channel dimension, and global max pooling operation across the channel dimension, respectively. To reduce the number of parameters and limit the model complexity, the first 1×1 convolution reduces the channel dimension from C to $\frac{C}{r}$ (r refers to the reduction ratio), and the second 1×1 convolution expands it back to C. In our implementation, we opt to reduce the channel dimension to a fixed value (i.e., 16) by setting r to $\frac{C}{16}$.

Then we fuse W_c and W_s together via a simple addition operation,[1] which follows broadcasting rules, to obtain the coarse SIMs $W_{coa} \in \mathbb{R}^{C\times H\times W}$.

$$W_{coa} = W_c + W_s. \tag{3}$$

Note that, W_{coa} and X are channel-wisely aligned, since the W_c is channel-based. In order to obtain the final refined SIMs W, every channel of W_{coa} is adjusted according to the corresponding input features. We utilize the content of input features as the guidance to generate the final channel-specific SIMs W. In particular, every channel of W_{coa} and X are re-arranged in an alternating manner via a channel shuffle operation [37]. Combined with the subsequent group convolution layer, the number of parameters can be greatly reduced.

$$W = \sigma(\mathcal{GC}_{7\times7}(CS([X, W_{coa}]))), \tag{4}$$

where σ denotes the sigmoid operation, $CS(\cdot)$ denotes the channel shuffle operation, $\mathcal{GC}_{k\times k}(\cdot)$ denotes a group convolution layer with the kernel size of $k \times k$, and in our implementation, the group number is set to C.

The CGA assigns an unique SIM to every channel, guiding the model to focus on significant regions of each channel. Therefore, more useful information encoded in features can be emphasized to effectively improve the dehazing performance. As shown in the right part of Fig. 2, we propose the main block of our DEA-Net, i.e., Detail-Enhanced Attention Block (DEAB), by combining DEConv with the CGA. By removing the CGA part, we obtain the Detail Enhanced Block (DEB).

C. CGA-Based Mixup Fusion Scheme

Following [6], [10], [17], [18], we adopt the encoder-decoder-like (or U-Net-like) architecture for our DEA-Net. We observe that fusing the features from the encoder part with that from the decoder part is an effective trick in dehazing and other low-level vision tasks [6], [10], [38], [39]. Low-level features (e.g., edges and contours), which have a non-negligible role for recovering haze-free images, gradually lose their impact after passing through many intermediate layers. Feature fusion can enhance the information flow from shallow layers to deep ones, which is beneficial for feature preserving and gradient back-propagation. The simplest way for fusion is element-wise addition, which is adopted in many previous approaches [10], [11], [17]. Later, Wu et al. [6] applied the adaptive mixup operation to adjust the fusion proportion via self-learned weights, which is more flexible than the addition.

However, there exists a receptive field mismatch problem in above mentioned fusion schemes. The information encoded in the shallow features is tremendously different from the information encoded in the deep features, since they have the totally different receptive fields. One single pixel in the deep features are originated from a region of pixels in the shallow features. The simple addition or concatenation operation or mixup operation fails to address the mismatch before fusion.

[1] We experimentally find the product operation can achieve similar results.

To mitigate this problem, we further propose a CGA-based mixup scheme to adaptively fuse the low-level features in the encoder part with corresponding high-level features, by modulating the features via learned spatial weights.

Fig. 2 (d) shows the details of proposed CGA-based mixup fusion scheme. The core part is that we opt to employ the CGA to calculate the spatial weights for feature modulation. The low-level features in the encoder part and corresponding high-level features are fed into the CGA to calculate the weights, and then combined by a weighted summation method. We also add the input features via skip connections to mitigate the gradient vanishing problem and ease the learning process. Finally, the fused features are projected by a 1×1 convolution layer to obtain the final features (i.e., F_{fuse}).

$$F_{fuse} = \mathcal{C}_{1\times1}(F_{low} \cdot W + F_{high} \cdot (\mathbf{1} - W) + F_{low} + F_{high}), \tag{5}$$

where $\mathbf{1}$ denotes the matrix whose elements are all 1. More discussions about the CGA-based mixup fusion scheme can be found in Section IV-C.3.

D. Overall Architecture

By combining DEConv, CGA, and the CGA-based mixup fusion scheme together, we propose our DEA-Net with DEAB and DEB as the basic blocks. As shown in Fig.2, our DEA-Net is a three-level encoder-decoder-like (or U-Net-like) architecture, which consists of three parts: encoder part, feature transform part, and decoder part. There are two down-sampling operations and two up-sampling operations in our DEA-Net. The down-sampling operation halves the spatial dimensions and doubles the number of channels. It is realized through a normal convolution layer by setting the value of stride to 2 and setting the number of output channels to 2 times of input channels. The up-sampling operation can be regarded as the inverse form of the down-sampling operation, which is realized through a deconvolution layer. The dimensional size of Level 1, Level 2, and Level 3 are $C \times H \times W$, $2C \times \frac{H}{2} \times \frac{W}{2}$, and $4C \times \frac{H}{4} \times \frac{W}{4}$, respectively. In our implementation, we set the value of C to 32. Previous methods [6], [18] transform the features only in the low-resolution space, resulting in information loss, which is non-trivial for the detail-sensitive task like dehazing. Differently, we deploy feature extraction blocks from Level 1 to Level 3. Specifically, we opt to employ different blocks in different levels (Level 1&2: DEB, Level 3: DEAB). For feature fusion, we fuse the features after the down-sampling operations and corresponding features before the up-sampling operations (highlighted with green arrow lines in Fig. 2). Finally, we simply employ a 3×3 convolution layer at the end to obtain the dehazing result J.

The DEA-Net is trained by minimizing the pixel-wise difference between the predicted haze-free image J and the corresponding ground truth GT. In our implementation, we choose L_1 loss function (i.e., mean absolute error) to drive the training.

$$\mathcal{L}_{L_1} = ||J - GT||_1, \tag{6}$$

IV. Experimental Results

A. Datasets and Metrics

1) Datasets: In our implementation, we train and test our DEA-Net on synthetic and real-captured datasets. REalistic Single Image DEhazing (RESIDE) [40] is a widely-used dataset, which contains five subsets: Indoor Training Set (ITS), Outdoor Training Set (OTS), Synthetic Objective Testing Set (SOTS), Real-world Task-driven Testing Set (RTTS), and Hybrid Subjective Testing Set (HSTS). We select ITS and OTS in the training phase and select SOTS in the testing phase. Note that, the SOTS is divided into two subsets (i.e., SOTS-indoor and SOTS-outdoor) for evaluating the models separately trained on ITS and OTS. ITS contains 1399 indoor clean images and for every clean image, 10 simulated hazy images are generated based on the physical scattering model. As for OTS, we pick around 296K images for the training process.[2] SOTS-indoor and SOTS-outdoor contain 500 indoor and 500 outdoor testing images, respectively. In addition, Haze4K dataset [41], which contains 3000 synthetic training images and 1000 synthetic testing images, is also employed to evaluate our DEA-Net. Besides, some real-captured hazy images are utilized to further verify the effectiveness on real scenes.

2) Evaluation Metrics: Peak Signal-to-Noise-Ratio (PSNR) and Structural Similarity Index Measurement (SSIM) [42], which are commonly used to measure the image quality among the computer vision community, are utilized for dehazing performance evaluation. For a fair comparison, we calculate the metrics based on the RGB color images without cropping pixels.

B. Implementation Details

We implement the proposed DEA-Net model on PyTorch deep learning platform with a single NVIDIA RTX3080Ti GPU. We deploy DEB in Level 1, DEB in Level 2, and DEAB in Level 3, respectively. The number of blocks deployed on different stages $[N_1, N_2, N_3, N_4, N_5]$ is set to $[4, 4, 8, 4, 4]$. The DEA-Net is optimized using the Adam [43] optimizer and β_1, β_2, ε are set to default values, i.e., 0.9, 0.999, $1e^{-8}$. Moreover, the initial learning rate and the batch size are set to $1e^{-4}$ and 16, respectively. The cosine annealing strategy [44] is adopted to adjust the learning rate from the initial value to $1e^{-6}$. To train the model, we randomly crop patches from the original images with size 256×256, and then two data augmentation techniques are adopted, i.e., 90° or 180° or 270° rotation and vertical or horizontal flipping. In the whole training phase, the model is trained for 1500K iterations, and it takes roughly 5 days to train our DEA-Net on ITS.

C. Ablation Study

To demonstrate the effectiveness of our proposed DEA-Net, we investigate on the designs and effects of (1) Detail-Enhanced Convolution (DEConv), (2) Content-Guided Attention (CGA), and (3) CGA-based mixup fusion scheme.

[2]Following [27], data cleaning is applied since the intersection of training and testing datasets.

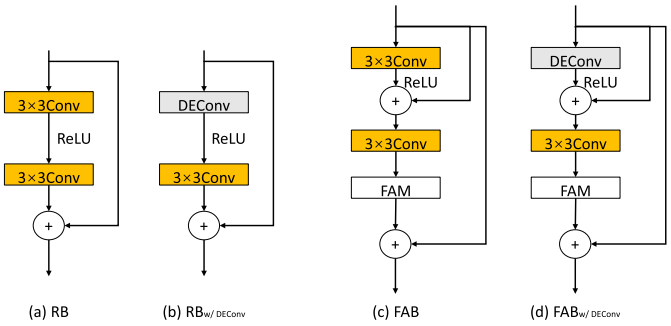

Fig. 7. The schematic diagrams of RB, RB$_{\text{w/ DEConv}}$, FAB, and FAB$_{\text{w/ DEConv}}$. The first vanilla convolution layer in RB/FAB is replaced with the proposed DEConv to generate the RB$_{\text{w/ DEConv}}$/FAB$_{\text{w/ DEConv}}$.

Fig. 8. Visualizations of intermediate features of FAB and FAB$_{\text{w/ DEConv}}$. Specifically, the feature maps before the ReLU are visualized, i.e., the outputs of vanilla convolution and the proposed DEConv.

TABLE I

ABLATION STUDY OF DECONV. ALL THE EXPERIMENTS ARE CONDUCTED ON SOTS-INDOOR [40] DATASET

Model		Base_RB	Base_FAB	Model_RB_D	Model_FAB_D
Setting	Level 1	–	–	–	–
	Level 2	–	–	–	–
	Level 3	RB	FAB	RB$_{\text{w/ DEConv}}$	FAB$_{\text{w/ DEConv}}$
	Attention	–	FAM	–	FAM
PSNR (dB)		30.74	33.07	31.01	33.67
SSIM		0.9729	0.9824	0.9739	0.9840
# Param. (K)		2105	2143	4467	4505

The contribution of each component is analyzed via ablation experiments.

1) DEConv: We first construct the baseline model by deploying classical Residual Block (RB) [45] in Level 3, and this model is denoted as **Base_RB**.

As a popular basic block used in the dehazing domain, we also employ the Feature Attention Block (FAB) from [5] in Level 3. The hyper-parameters are set to the default values as described in the original paper. We term this model as our second baseline, **Base_FAB**.

To extract more effective features, we revise the block by introducing DEConv into RB and FAB. As illustrated in Fig. 7, the first vanilla convolution layer is replaced with the proposed DEConv in both RB and FAB. The revised blocks are indicated as RB$_{\text{w/ DEConv}}$ and FAB$_{\text{w/ DEConv}}$, respectively. The corresponding models are denoted as **Model_RB_D** and **Model_FAB_D**.

For a fair comparison, all of the four blocks (i.e., RB, FAB, RB$_{\text{w/ DEConv}}$, and FAB$_{\text{w/ DEConv}}$) are cascaded for 6 times in Level 3, and the same fusion scheme is used (i.e., Mixup [5]). For convenience, we omit the blocks in Level 1 and Level 2, and train the models for only 500K iterations with initial learning rate is set to $2e^{-4}$ (These settings are with the ablation study). The experimental results are tested on the same testing dataset (i.e., SOTS-indoor [40] dataset). Although metrics are lower than the completely trained models reported in Table. VII, the trends and values are consistent and meaningful.

The performance of all these aforementioned models is summarized in Table I. Replacing the vanilla convolution layer with the parallel convolution layers (i.e., DEConv) brings 0.27 and 0.6 dB improvement in terms of PSNR on RB and FAB, respectively. We visualize the intermediate features of FAB and FAB$_{\text{w/ DEConv}}$ in Fig. 8. For both of FAB and FAB$_{\text{w/ DEConv}}$, we select the first block in level 3, which contains abundant textures and details. Specifically, the feature maps before the ReLU are visualized, i.e., the outputs of vanilla convolution and the proposed DEConv. In total, 3 input images are randomly selected from the testing datasets. We average the feature maps along the channel dimension in our implementation. The average values can roughly represent the situations of the whole feature maps. It is observed that the averaged feature map of the vanilla convolution from FAB is indistinct and fuzzy on image regions containing edges and textures. In comparison, the DEConv from FAB$_{\text{w/ DEConv}}$ introduces the difference convolutions to extract enhanced features with clearer boundary contours and more abundant high-frequency details.

By comparing **Model_FAB_D** with **Base_FAB**, the results indicate that DEConv can definitely improve the values of metrics (i.e., PSNR and SSIM) at the cost of around twice number of parameters (4505 K *vs.* 2143 K). That is very unfriendly and may cause failure under some memory-limited situations, prohibiting the usage of the DEConv on mobile or embedded devices.

In order to deal with the problem, we introduce the re-parameterization technique to equivalently transform the DEConv into a standard 3×3 convolution by adding up the learned kernel weights in the same positions during the testing phase. Table II shows the comparative results of the number of parameters (# Param.), the number of floating-point operations (# FLOPs) and inference time of **Model_FAB_D** before and after the re-parameterization operation. We can clearly see that the re-parameterization operation simplifies the parallel structure without triggering performance drop. In particular, after the simplification, **Model_FAB_D** still achieves 0.6 dB performance improvement when comparing with **Base_FAB** and no extra overhead is introduced.

In addition, we also explore the designs of parallel convolution layers from only a single vanilla convolution layer (i.e., FAB) to multiple streams, then to the complete DEConv (i.e., FAB$_{\text{w/ DEConv}}$). As shown in Table III, adding parallel streams with vanilla convolution layers causes the performance fluctuations between 33.07 dB and 32.92 dB (no better than FAB). The underlying reason behind this may be the training difficulty caused by redundant features extracted by the identical layers. On the contrary, adding parallel streams with difference

TABLE II

THE COMPARATIVE RESULTS OF THE NUMBER OF PARAMETERS (# PARAM.), THE NUMBER OF FLOATING-POINT OPERATIONS (# FLOPS), RUNTIME, AND PSNR VALUES OF **MODEL_FAB_D** BEFORE AND AFTER THE RE-PARAMETERIZATION (RE-PA.) OPERATION. THE PSNR VALUES ARE TESTED ON SOTS-INDOOR [40] DATASET. THE # FLOPS. AND INFERENCE TIME ARE MEASURED ON COLOR IMAGES WITH 256×256 RESOLUTION

	Model_FAB_D		Base_FAB
	w/o Re-Pa.	w/ Re-Pa.	–
# Param. (K)	4505	2143	2143
# FLOPs (G)	23.72	9.23	9.23
Runtime (ms)	4.53	1.76	1.76
PSNR (dB)	33.67	33.67	33.07

convolution (DC) layers can robustly boost the performance. The experimental results verify that by embedding traditional prior information, DC layers complement the vanilla convolution layer to produce more representative features. Based on the discussions above, we choose **Model_FAB_D** with basic block $\text{FAB}_{\text{w/ DEConv}}$ for the following study.

2) CGA: Further, we investigate the effectiveness of the proposed two-step coarse-to-fine attention mechanism (i.e., CGA). As mentioned in Section I, CGA generates channel-specific Spatial Importance Maps (SIMs) to indicate the important regions of individual channel. We compare the CGA with the other attention mechanisms such as Feature Attention Module (FAM) used in many dehazing methods [5], [6], [12] and the common Convolutional Block Attention Module (CBAM) [15]. Both FAM and CBAM contain sequential channel attention and spatial attention with slightly different implementations.

Model_FAB_D cascades $\text{FAB}_{\text{w/ DEConv}}$ blocks in Level 3 and inside the $\text{FAB}_{\text{w/ DEConv}}$ block, the FAM is adopted. Then, we replace FAM with CBAM and CGA to generate the corresponding models, i.e., **Model_DEAB** and **Model_CBAM**.

The spatial attention used in FAM or CBAM learns the SIM with only one single channel to indicate the important regions of the input features with relatively larger number of channels. Such approaches neglect the specificity of each channel of features and somehow restrict the powerful representation ability of CNNs. As shown in the left two and final columns of Table IV, **Model_DEAB** outperforms both **Model_FAB_D** and **Model_CBAM** by 1.5 dB and 0.49 dB in terms of PSNR. The results indicate that CGA can better re-calibrate the features via learning channel-specific SIMs to pay attention to channel-wise haze distribution difference.

we perform ablation study on the design of CGA. The results are listed in Table IV. As we can see, after removing the channel shuffle (CS) operation (the subsequent group convolution is replaced with a normal convolutional layer in this situation), the PSNR value decreases from 35.17 dB to 34.96 dB with the number of parameters increases from 4569 K to 14128 K. In our implementation, the channel attention (CA) and spatial attention (SA), which are used to generate the initial SIMs, are deployed in parallel. We also adopt the cascaded design and the results are listed in Table IV. We notice that parallel and cascaded designs achieve very similar performance in terms of PSNR (35.20 dB vs. 35.17 dB). Instead, if we omit the refine procedure (i.e., $\text{CGA}_{\text{w/o r}}$), the performance drops obviously.

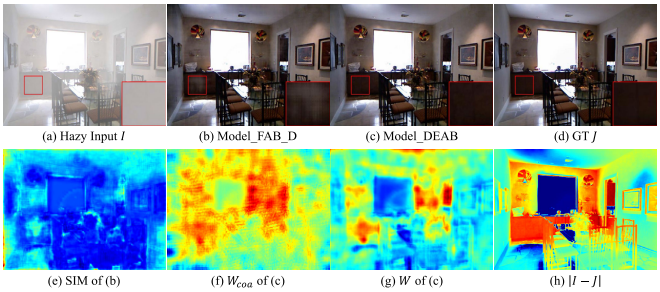

Fig. 9. Visual comparisons of FAM and our CGA. We show the learned SIMs and corresponding results.

Fig. 9 visually illustrates the SIMs learned by CGA and FAM, and the corresponding processing results. As we can see from Fig. 9 (e), one-channel SIM obtained by FAM can indicate the uneven haze distribution (to some extent). However, it is not accurate enough due to the mix of some contour patterns. Fig. 9 (f) shows the average map of the coarse SIMs W_{coa}, which is closer to the difference between ground-truth and the hazy input. By using the content of input features to guide the generation of SIMs, CGA can learn more accurate spatial weights. Fig. 9 (g) shows the average map of final SIMs. The channel-specific SIMs treat different channels of features with different spatial weights, which can better guide the model to focus on critical regions. Fig. 9 (b) and Fig. 9 (c) are the corresponding results. We observe that the cabinet region (highlighted by the red rectangle) recovered by **Model_FAB_D** has obvious haze residual.

3) CGA-based Mixup Fusion Scheme: We further perform ablation study to verify the effectiveness of our CGA-based mixup fusion scheme. We utilize **Model_DEAB** with the mixup fusion scheme from AECR-Net [6] as the baseline, and then evaluate another two schemes: element-wise addition [10], [11], [17] and the proposed CGA-based mixup. Their models are referred to **Model_DEAB_A** and **Model_DEAB_C**. The comparative results are shown in Table V. From these results, we see that "addition" achieves very similar performance with "mixup" (0.06 dB higher PSNR and 0.002 lower SSIM). "Addition" is a special case of mixup with the constant weights, and we experimentally find that the initial values have a significant impact on the performance of "mixup". Note that, our CGA-based mixup fusion scheme achieves the best performance in terms of PSNR and SSIM.

In addition, we deploy feature extraction blocks in Level 1 and Level 2 to further improve the performance. By deploying residual block (RB) in Level 1 and Level 2 (we refer this model to **Model_MS**), the performance improves by a large margin (2.52 dB in terms of PSNR). It means transforming features in high-resolution space even full-resolution space can repair the lost information, which is critical for image regression. As shown in Table V, our final DEA-Net-S achieves 39.16 dB in terms of PSNR and 0.9921 in terms of SSIM by deploying DEB in Level 1 and Level 2. The only difference between DEB and DEAB is the CGA part. The

TABLE III

THE EXPERIMENTAL RESULTS ON DESIGNS OF PARALLEL CONVOLUTION LAYERS. ✔×N MEANS THE SAME CONVOLUTION LAYER IS USED WITHIN N PARALLEL STREAMS. THE METRICS ARE TESTED ON SOTS-INDOOR [40] DATASET

Design	FAB	two-stream		four-stream		five-stream	$\text{FAB}_{\text{w/ DEConv}}$
Vanilla Conv.	✔	✔×2	✔	✔×4	✔	✔×5	✔
CDC			✔		✔		✔
HDC & VDC					✔		✔
ADC							✔
PSNR (dB)	33.07	32.92	33.23	33.07	33.43	32.98	33.67
SSIM	0.9824	0.9820	0.9826	0.9823	0.9833	0.9823	0.9840
# Param. (K)	2143	3028	3028	3620	3620	4505	4505

TABLE IV

ABLATION STUDY OF CGA. ALL THE EXPERIMENTS ARE CONDUCTED ON SOTS-INDOOR [40] DATASET

Model	Model_FAB_D	Model_CBAM	Model_DEAB			
Attention	FAM [5]	CBAM [15]	CGA_cas	$\text{CGA}_{\text{w/o r}}$	$\text{CGA}_{\text{w/o cs}}$	CGA
parallel CA&SA				✔	✔	✔
cascaded CA&SA	✔	✔	✔			
refine			✔		✔(remove CS)	✔
PSNR (dB)	33.67	34.68	35.20	34.50	34.96	35.17
SSIM	0.9840	0.9857	0.9869	0.9856	0.9858	0.9866
# Param. (K)	4505	4493	4569	4493	14128	4569

TABLE V

ABLATION STUDY OF CGA-BASED MIXUP FUSION SCHEME. WE COMPARE IT WITH ELEMENT-WISE ADDITION AND MIXUP [6]. ALL THE EXPERIMENTS ARE CONDUCTED ON SOTS-INDOOR [40] DATASET

Model		Model_DEAB_A	Model_DEAB	Model_DEAB_C	Model_MS	DEA-Net-S	DEA-Net-S*
Setting	Level 1	–	–	–	RB	DEB	DEAB
	Level 2	–	–	–	RB	DEB	DEAB
	Level 3	DEAB	DEAB	DEAB	DEAB	DEAB	DEAB
	Fusion scheme	Addition	Mixup	CGA-based Mixup	CGA-based Mixup	CGA-based Mixup	CGA-based Mixup
PSNR (dB)		35.23	35.17	35.40	37.92	39.16	38.78
SSIM		0.9864	0.9866	0.9875	0.9915	0.9921	0.9918

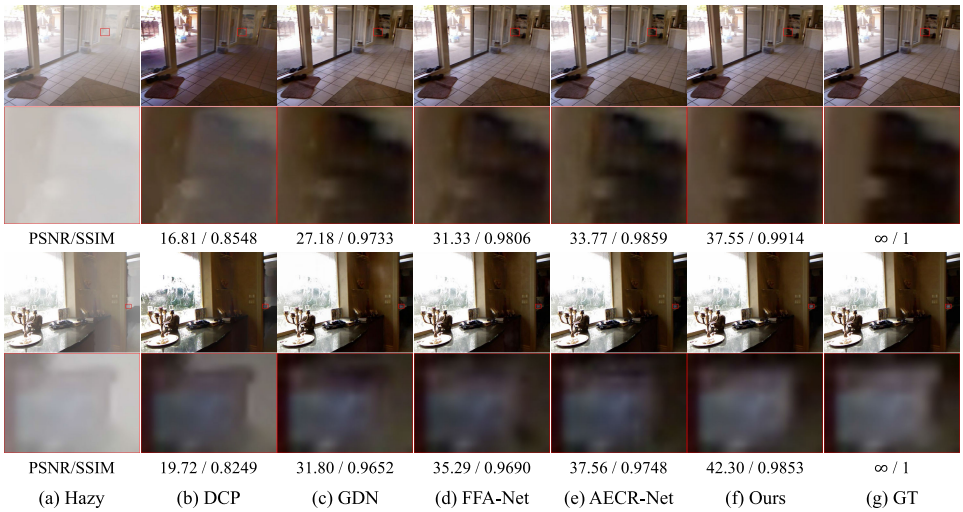

Fig. 10. Visual comparisons of various methods on synthetic SOTS-indoor [40] dataset. Please zoom in on screen for a better view.

suffix '-S' denotes the model is trained with the settings in ablation study, which is a simplified version. For **Model_MS** and DEA-Net-S, $[N_1, N_2, N_3, N_4, N_5]$ is set to $[3, 3, 6, 3, 3]$. Further deploying DEAB in level 1 and level 2 (denoted as

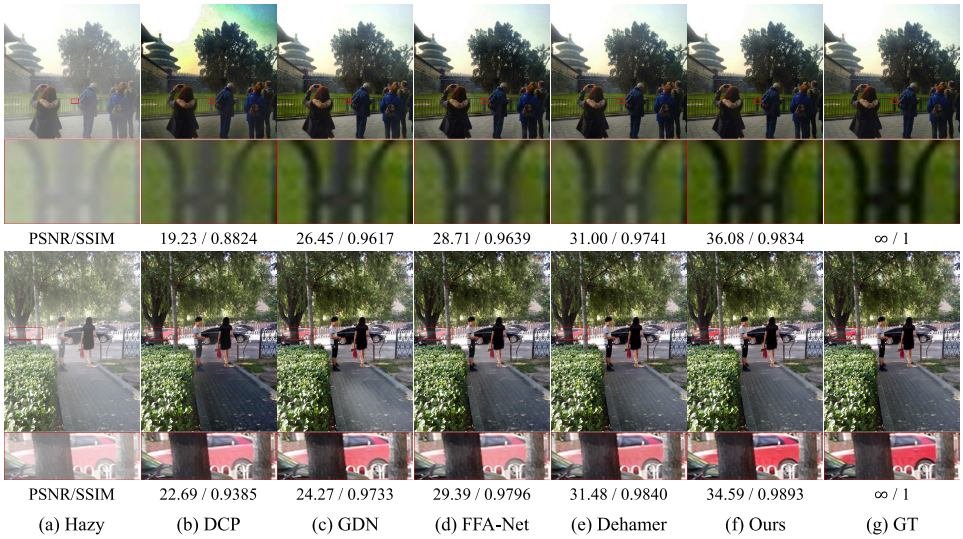

Fig. 11. Visual comparisons of various methods on synthetic SOTS-outdoor [40] dataset. Please zoom in on screen for a better view.

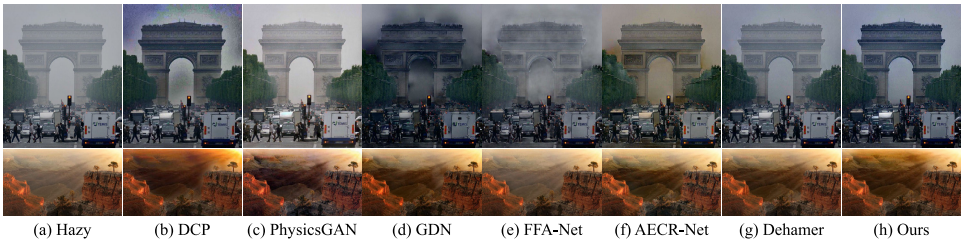

Fig. 12. Dehazing results of various methods on real-world hazy images. Please zoom in on screen for a better view.

TABLE VI

DETAILED COMPARISONS OF DEA-NET-S AND DEA-NET-S* ON PSNR, SSIM, # PARAM., # FLOPS. AND RUNTIME. THE PSNR AND SSIM VALUES ARE TESTED ON SOTS-INDOOR [40] DATASET. THE # FLOPS. AND INFERENCE TIME ARE MEASURED ON COLOR IMAGES WITH 256 × 256 RESOLUTION

Model		DEA-Net-S	DEA-Net-S*
Setting	Level 1	DEB	DEAB
	Level 2	DEB	DEAB
	Level 3	DEAB	DEAB
	Fusion scheme	CGA-based Mixup	CGA-based Mixup
PSNR (dB)		39.16	38.78
SSIM		0.9921	0.9918
# Param. (K)		5946	6013
# FLOPs (G)		68.36	70.28
Runtime (ms)		16.29	27.02

DEA-Net-S*) even negatively influence the performance in terms of PSNR and SSIM with more parameters and FLOPs (see in Table VI). We argue that the attribute transformations (such as illumination and color change) caused by the existence of haze relate more to the low-frequency component (i.e., low-resolution level 3), which is consistent with [46]. In our DEA-Net, CGA is designed to deal with haze non-uniformity in feature level. It is very straightforward that we only deploy CGA in level 3. Another advantage of this setting is that it can avoid complicated hyper-parameter tuning (e.g., the reduction ratio).

D. Comparisons With SOTA Methods

In this section, we compare our DEA-Net with 4 earlier dehazing approaches including DCP [20], DehazeNet [2], AOD-Net [3], GFN [26] and 9 recent state-of-the-art (SOTA) single image dehazing methods including PFDN [24], FFA-Net [5], MSBDN [10], DMT-Net [41], AECR-Net [6], SGID-PFF [17], UDN [18], PMDNet [11], Dehamer [13] on SOTS-indoor, SOTS-outdoor, and Haze4K datasets. We report three DEA-Net variants including DEA-Net-S with the settings in ablation study (i.e., the last but one model of Table. V), DEA-Net with normal settings, and DEA-Net-CR with normal settings and the contrastive regularization (CR) from AECR-Net [6]. DEA-Net-CR has identical setting of CR with AECR-Net [6]. Note that CR will not increase additional parameters and inference time, since it can be directly removed in the testing phase. For others, we adopt the official released codes or evaluation results of these methods for fair comparisons if they are publicly available, otherwise we retrain them using the same training datasets.

1) Quantitative Analysis: Table VII shows quantitative evaluation results (PSNR and SSIM indexes) of our DEA-Nets and other state-of-the-art methods on SOTS [40] and Haze4K [41]. As we can see, even our DEA-Net-S achieves the best performance with 39.16 dB in PSNR and 0.9921 in SSIM on SOTS-indoor than the alternatives. Further, our DEA-Net and DEA-Net-CR improve the performance by a large margin on both SOTS-indoor and SOTS-outdoor datasets. On the Haze4K

TABLE VII

QUANTITATIVE COMPARISONS OF VARIOUS DEHAZING METHODS ON SOTS-INDOOR, SOTS-OUTDOOR, AND HAZE4K. WE REPORT PSNR, SSIM, NUMBER OF PARAMETERS (# PARAM.), NUMBER OF FLOATING-POINT OPERATIONS (# FLOPS), AND RUNTIME TO PERFORM COMPREHENSIVE COMPARISONS. THE SIGN "-" DENOTES THE DIGIT IS UNAVAILABLE. **BOLD** AND UNDERLINED INDICATE THE BEST AND THE SECOND BEST RESULTS, RESPECTIVELY

Method	SOTS-indoor [40]		SOTS-outdoor [40]		Haze4K [41]		Overhead		
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	# Param. (M)	# FLOPs (G)	Runtime (ms)
(TPAMI'10) DCP [20]	16.61	0.8546	19.14	0.8605	14.01	0.76	-	-	-
(TIP'16) DehazeNet [2]	19.82	0.8209	27.75	0.9269	19.12	0.84	0.008	0.5409	0.9932
(ICCV'17) AOD-Net [3]	20.51	0.8162	24.14	0.9198	17.15	0.83	0.0018	0.1146	0.3159
(CVPR'18) GFN [26]	22.30	0.8800	21.55	0.8444	-	-	0.4990	14.94	-
(ECCV'20) PFDN [24]	32.68	0.9760	-	-	-	-	11.27	47.21	-
(AAAI'20) FFA-Net [5]	36.39	0.9886	33.57	0.9840	26.97	0.95	4.456	287.5	47.98
(CVPR'20) MSBDN [10]	32.77	0.9812	34.81	0.9857	22.99	0.85	31.35	41.54	9.826
(ACMMM'21) DMT-Net [41]	-	-	-	-	28.53	0.96	51.79	75.56	26.83
(CVPR'21) AECR-Net [6]	37.17	0.9901	-	-	-	-	**2.611**	52.20	-
(TIP'22) SGID-PFF [17]	38.52	0.9913	30.20	0.9754	-	-	13.87	152.8	20.92
(AAAI'22) UDN [18]	38.62	0.9909	34.92	0.9871	-	-	4.250	-	-
(ECCV'22) PMDNet [11]	38.41	0.9900	34.74	0.9850	33.49	0.98	18.90	-	-
(CVPR'22) Dehamer [13]	36.63	0.9881	35.18	0.9860	-	-	132.4	48.93	14.12
(Ours) DEA-Net-S w/o Re-Pa.	39.16	0.9921	-	-	-	-	5.946	68.36	16.29
(Ours) DEA-Net-CR w/o Re-Pa.	41.31	0.9945	36.59	0.9897	34.25	0.99	7.789	90.21	19.51
(Ours) DEA-Net-S	39.16	0.9921	-	-	-	-	2.844	**24.88**	**5.632**
(Ours) DEA-Net	40.20	0.9934	36.03	0.9891	33.19	**0.99**	3.653	32.23	7.093
(Ours) DEA-Net-CR	**41.31**	**0.9945**	**36.59**	**0.9897**	**34.25**	**0.99**	3.653	32.23	7.093

dataset, our DEA-Net and DEA-Net-CR achieve the best SSIM (0.9869 and 0.9885). We round the results to two decimals to keep consistent with [41]. Our DEA-Net-CR ranks first in all comparisons on SOTS and Haze4K datasets.

In addition, we adopt number of parameters (# Param.), number of floating-point operations (# FLOPs), and runtime as the major indicators of computational efficiency. The earlier dehazing methods contain very small parameters sizes at the cost of a big performance drop. Compared with recent SOTA methods, our DEA-Nets run fastest with acceptable # Param. and # FLOPs. Any one of our DEA-Net variants can rank second best in terms of # Param. and # FLOPs. This implies our DEA-Nets can reach a good trade-off between performance and model complexity. Note that # FLOPs and runtime are measured on color images with 256×256 resolution.

2) Qualitative Analysis: Fig. 10 shows the visual comparisons between our DEA-Net and previous SOTA methods on synthetic SOTS-indoor dataset. Our proposed DEA-Net can recover sharper and clearer contours or edges, and the results obtained by DEA-Net contains less haze residuals. Fig. 11 shows the visual comparisons on the synthetic SOTS-ourdoor dataset. We observe that in outdoor scenes, the results of our DEA-Net are closest to the ground truth than the other alternatives. We also test our DEA-Net on real-world hazy images, and compare the results with various SOTA methods. As shown in Fig. 12, the other methods either remain haze on the processed results or produce color deviations and artifacts. On the contrary, our DEA-Net can output more visually pleasing dehazing results. We also conduct comparisons with ultra-high-definition image dehazing (UHDD) model [47], which aims to remove haze for UHD (ultra-high-definition) or 4K resolution images. Please refer to Appendix B for more information.

V. LIMITATION AND FUTURE WORK

In the process of developing our DEA-Net, we notice there are still some difficulties that are urgent to be well addressed. We leave these meaningful and thought-provoking challenges for future work and hope to further contribute to dehazing community by addressing these issues.

- Domain shift: Collecting large-scale yet perfectly aligned hazy-clean pairs is incredibly difficult. Most existing methods trained on synthetic data perform poorly when applied to diverse real-world hazy scenarios. How to narrow the domain gap between synthetic training data and real-world testing images is still a challenging task, despite our DEA-Net model has achieved promising performance.
- Model efficiency: Existing dehazing models usually implement a highly non-linear mapping from hazy inputs to clean labels for promoting performance. In this case, the computational consumption is non-negligible, especially on resource-constrained and memory-limited devices. We plan to employ some model compression techniques (e.g., quantization, pruning) to reduce the model's computational complexity. How to further improve the model's efficiency is an interesting direction that is worth further study.

VI. CONCLUSION

In this paper, we propose a DEA-Net to deal with the challenging single image dehazing problem. Specifically, we design the detail-enhanced convolution (DEConv) by introducing the difference convolution to integrate local descriptors into normal convolution layer. Compare with vanilla convolution, DEConv has enhanced representation and generalization capacity. In addition, the DEConv can be equivalently converted into a vanilla convolution without triggering extra

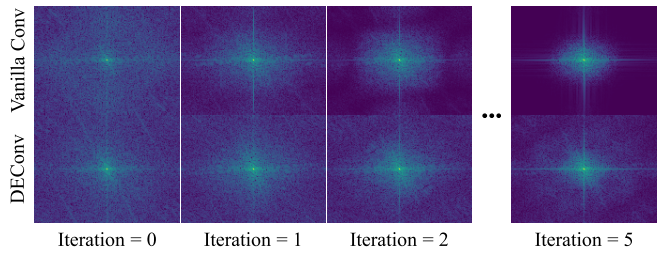

Fig. 13. The discrete Fourier transform (DFT) results of filtered features which are averaged firstly.

Fig. 14. Dehazing results of UHDD [47], Dehamer [13], and our methods on real-world hazy images. Please zoom in on screen for a better view.

parameters and computational cost. Then, we design a sophisticated attention mechanism termed content-guided attention (CGA), which assigns unique spatial importance map (SIM) to every channel. With CGA, more useful information encoded in features can be emphasized. Based on CGA, we further present a fusion scheme to effectively fuse low-level features in the encoder part with corresponding high-level features. Extensive experiments show that our DEA-Net achieves state-of-the-art results quantitatively and qualitatively.

APPENDIX

A. Analyses of Vanilla Convolution and DEConv

Recall the models in Table I, we extract the learned filters of FAB and $\text{FAB}_{\text{w/ DEConv}}$ from **Model_FAB** and **Model_FAB_D**, and iteratively impose the filters on features of a randomly selected image from the testing dataset. For FAB, the vanilla convolution's filters are extracted and for $\text{FAB}_{\text{w/ DEConv}}$, the DEConv's filters are extracted. For both of FAB and $\text{FAB}_{\text{w/ DEConv}}$, we select the first block in level 3.

We iteratively apply the filters and visualize the discrete Fourier transform (DFT) results of filtered features which are averaged firstly. As shown in Fig. 13, the spectra of vanilla convolution with different iterations are provided on the top row, and we notice that the high-frequency signals in the spectrum are reduced drastically. Instead, the spectra of DEConv can maintain the high-frequency information.

B. Comparison With UHDD Method

We also compare the qualitative results of UHDD [47], Dehamer [13], and our methods in Fig. 14. Despite the remarkable efficiency on high-resolution images, the dehazing results of UHDD model tend to lose some high-frequency information, making them become blur. Dehamer model can preserve the high-frequency information, but the residual haze still remains. Our method outperforms both competing models, yielding more visually pleasing results.

REFERENCES

[1] R. T. Tan, "Visibility in bad weather from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.
[2] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao, "DehazeNet: An end-to-end system for single image haze removal," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5187–5198, Nov. 2016.
[3] B. Li, X. Peng, Z. Wang, J. Xu, and D. Feng, "AOD-Net: All-in-one dehazing network," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4780–4788.
[4] P. Li, J. Tian, Y. Tang, G. Wang, and C. Wu, "Deep Retinex network for single image dehazing," *IEEE Trans. Image Process.*, vol. 30, pp. 1100–1115, 2021.
[5] X. Qin, Z. Wang, Y. Bai, X. Xie, and H. Jia, "FFA-Net: Feature fusion attention network for single image dehazing," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 11908–11915.
[6] H. Wu et al., "Contrastive learning for compact single image dehazing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10546–10555.
[7] W. Ren et al., "Single image dehazing via multi-scale convolutional neural networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 154–169.
[8] H. Zhang and V. M. Patel, "Densely connected pyramid dehazing network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3194–3203.
[9] S. G. Narasimhan and S. K. Nayar, "Contrast restoration of weather degraded images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 6, pp. 713–724, Jun. 2003.
[10] H. Dong et al., "Multi-scale boosted dehazing network with dense feature fusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2154–2164.
[11] T. Ye et al., "Perceiving and modeling density is all you need for image dehazing," 2021, *arXiv:2111.09733*.
[12] H. Wu, J. Liu, Y. Xie, Y. Qu, and L. Ma, "Knowledge transfer dehazing network for NonHomogeneous dehazing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 1975–1983.
[13] C. Guo, Q. Yan, S. Anwar, R. Cong, W. Ren, and C. Li, "Image dehazing transformer with transmission-aware 3D position embedding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 5802–5810.
[14] Z. Yu et al., "Searching central difference convolutional networks for face anti-spoofing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5294–5304.
[15] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.
[16] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2020.
[17] H. Bai, J. Pan, X. Xiang, and J. Tang, "Self-guided image dehazing using progressive feature fusion," *IEEE Trans. Image Process.*, vol. 31, pp. 1217–1229, 2022.
[18] M. Hong, J. Liu, C. Li, and Y. Qu, "Uncertainty-driven dehazing network," in *Proc. AAAI Conf. Artif. Intell.*, Jun. 2022, vol. 36, no. 1, pp. 906–913. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/19973
[19] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," in *Proc. CVPR*, 2009, pp. 1956–1963
[20] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2341–2353, Dec. 2011.
[21] D. Berman, T. Treibitz, and S. Avidan, "Non-local image dehazing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1674–1682.
[22] Q. Zhu, J. Mai, and L. Shao, "A fast single image haze removal algorithm using color attenuation prior," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3522–3533, Nov. 2015.
[23] W. Ren, J. Pan, H. Zhang, X. Cao, and M.-H. Yang, "Single image dehazing via multi-scale convolutional neural networks with holistic edges," *Int. J. Comput. Vis.*, vol. 128, no. 1, pp. 240–259, Jan. 2020.
[24] J. Dong and J. Pan, "Physics-based feature dehazing networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 188–204.

[25] J. Pan et al., "Physics-based generative adversarial models for image restoration and beyond," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 7, pp. 2449–2462, Jul. 2021.
[26] W. Ren et al., "Gated fusion network for single image dehazing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3253–3261.
[27] X. Liu, Y. Ma, Z. Shi, and J. Chen, "GridDehazeNet: Attention-based multi-scale network for image dehazing," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7313–7322.
[28] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.
[29] F. Juefei-Xu, V. N. Boddeti, and M. Savvides, "Local binary convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4284–4293.
[30] Z. Yu, Y. Qin, H. Zhao, X. Li, and G. Zhao, "Dual-cross central difference network for face anti-spoofing," in *Proc. 13th Int. Joint Conf. Artif. Intell.*, Z.-H. Zhou, Ed. Aug. 2021, pp. 1281–1287, doi: 10.24963/ijcai.2021/177.
[31] Z. Su et al., "Pixel difference networks for efficient edge detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 5097–5107.
[32] C. Wang et al., "EAA-Net: A novel edge assisted attention network for single image dehazing," *Knowl.-Based Syst.*, vol. 228, Sep. 2021, Art. no. 107279.
[33] Z. Yu, J. Wan, Y. Qin, X. Li, S. Z. Li, and G. Zhao, "NAS-FAS: Static-dynamic central difference network search for face anti-spoofing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 9, pp. 3005–3023, Sep. 2021.
[34] I. Sobel and G. Feldman, "A 3×3 isotropic gradient operator for image," Talk Stanford Artif. Intell. Lab., 1968, pp. 271–272.
[35] J. M. Prewitt, "Object enhancement and extraction," *Picture Process. Psychopictorics*, vol. 10, no. 1, pp. 15–19, 1970.
[36] H. Scharr, "Optimal operators in digital image processing," Ph.D. thesis, Interdiscipl. Center Sci. Comput., Heidelberg Univ., Heidelberg, Germany, 2000.
[37] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6848–6856.
[38] Z. He, G. Fu, Y. Cao, Y. Cao, J. Yang, and X. Li, "ESKN: Enhanced selective kernel network for single image super-resolution," *Signal Process.*, vol. 189, Dec. 2021, Art. no. 108274.
[39] Z. He et al., "Single image super-resolution based on progressive fusion of orientation-aware features," *Pattern Recognit.*, vol. 133, Jan. 2023, Art. no. 109038.
[40] B. Li et al., "Benchmarking single-image dehazing and beyond," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 492–505, Jan. 2019.
[41] Y. Liu et al., "From synthetic to real: Image dehazing collaborating with unlabeled real data," in *Proc. 29th ACM Int. Conf. Multimedia*, 2021, pp. 50–58.
[42] 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.
[43] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015, pp. 1–15.
[44] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, and M. Li, "Bag of tricks for image classification with convolutional neural networks," in *Proc. CVPR*, Jun. 2019, pp. 558–567.
[45] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[46] J. Liang, H. Zeng, and L. Zhang, "High-resolution photorealistic image translation in real-time: A Laplacian pyramid translation network," in *Proc. CVPR*, 2021, pp. 9392–9400.
[47] Z. Zheng et al., "Ultra-high-definition image dehazing via multi-guided bilateral learning," in *Proc. CVPR*, 2021, pp. 16185–16194.

Zixuan Chen received the B.E. degree from the School of Aeronautics and Astronautics, Zhejiang University, in 2022, where he is currently pursuing the master's degree. His research interests include image restoration and image enhancement.

Zewei He received the B.E. degree from the University of Science and Technology Beijing (USTB) in 2014 and the Ph.D. degree from Zhejiang University in 2019. After obtaining the Ph.D. degree, he was a Research Associate with Louisiana State University, Baton Rouge, LA, USA, from 2019 to 2020. He is currently a Postdoctoral Researcher with Zhejiang University. His research interests include infrared imaging, multi-sensor image fusion, and image restoration.

Zhe-Ming Lu (Senior Member, IEEE) was born in Zhejiang, China, in 1974. He received the B.S. and M.S. degrees in electrical engineering and the Ph.D. degree in measurement technology and instrumentation from the Harbin Institute of Technology (HIT), Harbin, China, in 1995, 1997, and 2001, respectively. In 1999, he became a Lecturer with HIT. Since 2003, he has been a Professor with the Department of Automatic Test and Control, HIT. He is currently a Full Professor with the School of Aeronautics and Astronautics, Zhejiang University. In the areas of multimedia signal processing and information hiding, he has published more than 300 papers, seven monographs in Chinese, one monograph in English, and three book chapters in English. His current research interests include multimedia signal processing, information security, and complex networks.