RefineDNet: A Weakly Supervised Refinement Framework for Single Image Dehazing

Shiyu Zhao, Lin Zhang, *Senior Member, IEEE*, Ying Shen, *Member, IEEE*, and Yicong Zhou, *Senior Member, IEEE*

***Abstract*— Haze-free images are the prerequisites of many vision systems and algorithms, and thus single image dehazing is of paramount importance in computer vision. In this field, prior-based methods have achieved initial success. However, they often introduce annoying artifacts to outputs because their priors can hardly fit all situations. By contrast, learning-based methods can generate more natural results. Nonetheless, due to the lack of paired foggy and clear outdoor images of the same scenes as training samples, their haze removal abilities are limited. In this work, we attempt to merge the merits of prior-based and learning-based approaches by dividing the dehazing task into two sub-tasks, i.e., visibility restoration and realness improvement. Specifically, we propose a two-stage weakly supervised dehazing framework, RefineDNet. In the first stage, RefineDNet adopts the dark channel prior to restore visibility. Then, in the second stage, it refines preliminary dehazing results of the first stage to improve realness via adversarial learning with unpaired foggy and clear images. To get more qualified results, we also propose an effective perceptual fusion strategy to blend different dehazing outputs. Extensive experiments corroborate that RefineDNet with the perceptual fusion has an outstanding haze removal capability and can also produce visually pleasing results. Even implemented with basic backbone networks, RefineDNet can outperform supervised dehazing approaches as well as other state-of-the-art methods on indoor and outdoor datasets. To make our results reproducible, relevant code and data are available at https://github.com/xiaofeng94/RefineDNet-for-dehazing.**

***Index Terms*— Single image dehazing, weak supervision, image fusion, unpaired dehazing dataset.**

I. Introduction

UNDER haze conditions, the visibility of images is seriously degraded due to the scattering of atmospheric aerosol particles, making it difficult to further perceive and understand for many computer vision applications, such as object detection, recognition, and ADAS (Advanced Driver Assistance System). Therefore, haze removal, especially single image dehazing, is highly valuable and has been extensively studied in the past decade [1]–[7].

Existing dehazing methods can be roughly classified into two categories, i.e., the prior-based and the learning-based. Methods of the first class rely on a widely accepted physical model for atmospheric scattering, Koschmieder's law [9]. In our case, this law can be defined as,

$$I(\mathbf{x}) = J(\mathbf{x})\, t(\mathbf{x}) + A\,(1 - t(\mathbf{x}))\,. \tag{1}$$

Here, $\mathbf{x}$ refers to the position of a pixel. $I(\mathbf{x})$ and $J(\mathbf{x})$ are the apparent luminance (the foggy image) and the intrinsic luminance (the clear scene), respectively. A is the global skylight representing ambient light in the atmosphere. $t(\mathbf{x})$ is the transmission of the intrinsic luminance in the atmosphere and it can be further modeled as,

$$t(\mathbf{x}) = e^{-\beta d(\mathbf{x})} \tag{2}$$

where β is the extinction coefficient, and $d(\mathbf{x})$ is the scene depth of $\mathbf{x}$. Since there are more than two unknown variables in Koschmieder's law, we cannot pinpoint them using the input hazy image only. Thus, researchers of prior-based methods have proposed various priors as extra constraints to find a proper solution for $J(\mathbf{x})$. Those priors usually aim to restore the contrast of objects against the ambient light. Since the visibility is decided by the contrast, prior-based methods can generate dehazing results with high visibility. Although those priors perform well in specific cases, they are unable to fit all circumstances and thus overly enhance the contrast, producing unwanted artifacts, e.g., halos and color blockings.

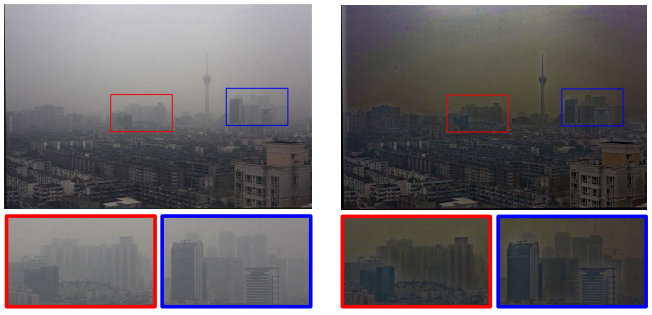
Unlike prior-based dehazing methods, learning-based approaches learn to estimate A and $t(\mathbf{x})$, or to recover $J(\mathbf{x})$ directly from the input hazy image via supervised learning. Since they adopt convolutional neural networks (CNNs) that are inborn to generate images with few artifacts according to [10], those methods are able to produce dehazing results with satisfactory realness. However, their training processes require a large number of clear and hazy image pairs from the same scenes, which are hardly possible to collect in bulk under real-world conditions. Therefore, they often make a trade-off and synthesize hazy images by applying Koschmieder's law on indoor scenes where the essential depth information is available. Since there are certain gaps between indoor synthetic and real-world outdoor images, learning-based methods are likely to overfit the synthetic data, and their ability to remove real haze is limited.

Manuscript received June 24, 2020; revised November 29, 2020 and January 29, 2021; accepted February 12, 2021. Date of publication March 2, 2021; date of current version March 9, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61973235, Grant 61936014, and Grant 61972285; in part by the Natural Science Foundation of Shanghai under Grant 19ZR1461300; and in part by the Shanghai Science and Technology Innovation Plan under Grant 20510760400. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xiaolin Hu. *(Corresponding author: Lin Zhang.)*

Shiyu Zhao, Lin Zhang, and Ying Shen are with the School of Software Engineering, Tongji University, Shanghai 201804, China (e-mail: 1731558@tongji.edu.cn; cslinzhang@tongji.edu.cn; yingshen@tongji.edu.cn).

Yicong Zhou is with the Department of Computer and Information Science, University of Macau, Zhuhai, Macau (e-mail: yicongzhou@um.edu.mo).

Digital Object Identifier 10.1109/TIP.2021.3060873



(a) AODNet [5] (b) DCP [8]

Fig. 1. Drawbacks of learning-based and prior-based methods. (a) is generated by the learning-based method AODNet [5]. (b) is generated by the prior-based method DCP [8]. The result of AODNet is visually better but includes more haze, whereas DCP removes more haze at the cost of introducing artifacts. The red and the blue boxes highlight their differences.

Interestingly, due to the characteristics of the two categories, prior-based methods are relatively better for restoring visibility whereas learning-based methods are preferable for improving the result's realness. Fig. 1 provides dehazing results of (a) the learning-based AODNet [5] and (b) the prior-based DCP [8] to illustrate this phenomenon. As we can see, DCP's result has less haze but more artifacts, whereas the result of AODNet is in high realness but with more haze. In Appendix A, we provide some theoretical explanations for the preference of prior-based and learning-based methods.

To further improve the dehazing results, it is a natural idea to exploit the advantages of both categories, but surprisingly such a simple idea has seldom been explored in the literature. In this work, based on the above-mentioned findings, we propose a two-stage weakly supervised dehazing framework, RefineDNet (Refinement Dehazing Network), to merge the merits of the two categories.

Specifically, in the first stage, RefineDNet restores the visibility of the input hazy image by producing preliminary results with DCP. We embed DCP dehazing in our framework to enable end-to-end training and evaluations. In the second stage, RefineDNet improves the realness of the preliminary dehazed image and the quality of the transmission map by refining them using two refiner networks. During training, we update the refiner networks via adversarial learning with a discriminator on unpaired images. This weak supervision with unpaired data is beneficial to dehazing because it is possible to collect a large amount of unpaired images from the real world to train our model. In this way, RefineDNet suits to process real-world foggy images better than supervised methods that are trained on simulated images and may overfit those data.

Beside the refined dehazed image, RefineDNet reconstructs another dehazed image using the hazy input and refined transmission. Since the refined and the reconstructed dehazed images are generated in different ways, they are unlikely to perform the same in all regions. It is highly possible that either of them may outperform the other in some regions. Thus, fusing better regions in either of them can boost performance. To this end, we propose a perceptual fusion strategy to fuse the refined and the reconstructed dehazed images. In this strategy, greater weights are assigned to regions that are closer to natural images. To obtain such weights, we exploit powerful features in the field of image quality assessment (IQA).

With the two-stage dehazing strategy, RefineDNet divides the dehazing task into two less intractable subtasks, namely, visibility restoration and realness improvement, and leverages priors and learning to handle the two subtasks, respectively. Since priors and learning are used in separate stages of RefineDNet, they are unlike to affect each other. Thus, RefineDNet merits the advantages of both prior-based and learning-based methods. Besides, RefineDNet only needs to remove artifacts in the refinement stage, and thus, its learning encounters less ambiguity of dehazing. As a result, it is blessed with stable weak supervision and circumvents the issue of the lack of data that supervised methods suffer from. To support our claims, we show that even implemented with basic backbone networks, RefineDNet is able to outperform state-of-the-art supervised methods on both indoor and outdoor datasets. Moreover, since there is no off-the-shelf outdoor training set for RefineDNet, we built an unpaired outdoor training dataset, RESIDE-unpaired, using images from RESIDE [11].

The main contributions of this work are summarized as:

- We propose a two-stage weakly supervised framework RefineDNet which first adopts prior-based DCP to restore visibility and then employs GANs to improve realness. It is demonstrated that RefineDNet integrates the advantages of both prior-based and learning-based dehazing methods and generates visually pleasing results with high visibility. Moreover, due to the two-stage dehazing strategy, RefineDNet boasts effective weak supervision with unpaired foggy and clear images, which avoids the issue of the lack of paired data for supervised methods.
- We propose a novel perceptual fusion strategy to blend different dehazing results. Our experimental results demonstrate that this strategy is effective with performance gain in various datasets.
- We also construct a necessary unpaired dataset with 6,480 outdoor images to facilitate the relevant studies of weakly supervised dehazing approaches.

The remainder of this paper is organized as follows. Section II reviews the related work. Section III describes the proposed RefineDNet with the perceptual fusion in detail. Section IV presents experimental results and ablation studies. Finally, Section V concludes this paper.

II. Related Work

This work is related to the prior-based and learning-based dehazing methods and generative adversarial networks (GANs). Since GANs have been widely explored in recent years, we mainly review their applications on dehazing.

A. Prior-Based Dehazing Methods

In the literature, various priors or assumptions have been explored. Fattal [1] decomposed $J(\mathbf{x})$ in Koschmieder's law [9] into surface reflectance coefficients and a shading factor, and he solved all unknown variables by assuming

that the shading factor and transmission are independent. Tan [12] constructed a Markov Random Field (MRF) with the energy function based on their observation that clear images have higher contrast, and the atmosphere scattering term of Koschmieder's law (the second term) changes smoothly across small regions. Following Tan's observation [12], Tarel and Hautiere [13] defined the atmospheric veil and provided its closed-form solution. Later, He *et al.* [8] proposed the aforementioned dark channel prior (DCP) to estimate transmission maps. Salazar-Colores *et al.* [14] combined DCP with mathematical morphology operations, e.g., erosion and dilation, to compute transmission maps efficiently. Meng *et al.* [3] generalized DCP to the boundary constraint and adopted this constraint together with a weighted contextual regularization to get optimized transmission maps. More recently, Liu *et al.* [15] proposed the non-local total variation regularization (NLTV) to refine preliminary transmission maps obtained by the boundary constraint.

Besides, distributions of different parts of Koschmieder's law were studied. Nishino *et al.* [16] analyzed the distributions of the scene albedo and image depth and then applied a Factorial MRF [17] to estimate them jointly. Fattal [18] found that pixels in small patches of natural images typically exhibit one-dimensional distributions called color-lines in RGB color space. Specifically, the color-lines of hazy images own the exclusive offsets. Berman and Avidan [19] pointed out that haze-free images can be well approximated by a few hundreds of distinct colors, and pixels can be grouped into clusters according to their colors. In haze conditions, pixels of each cluster become a haze-line in RGB space. Thus, dehazing is equal to identifying those haze-lines. More recently, based on the observation that pixels of image patches are clustered in an ellipsoid region instead of color-lines, Bui and Kim [20] proposed the color ellipsoid prior to maximize the contrast of dehazed pixels.

B. Learning-Based Dehazing Methods

With the popularity of CNNs, learning-based methods have emerged in this field. Cai *et al.* [4] proposed an end-to-end CNN called DehazeNet to estimate the transmission map from a hazy image. Ren *et al.* [21] exploited multi-scale information to predict transmission by using a coarse-scale net and a fine-scale one. Differently, Li *et al.* [5] combined the two unknown variables, i.e., the transmission and the ambient light, into one by reformulating Koschmieder's law. Then, they constructed the AODNet to estimate this variable. In [22], Zhang *et al.* adopted AODNet's formulation and proposed a fast and accurate multi-scale dehazing network called FAMED-Net to estimate the same variable. Later, Ren *et al.* [23] proposed the gate fusion network (GFN) to conflate three intermediate results generated by white balance, contrast enhancement, and gamma correction as dehazing results. Santra *et al.* [24] constructed a patch quality comparator (PQC) with CNNs to attain the best dehazing patches. More recently, based on the finding that the atmospheric illumination has a greater impact on the illumination channel of the YCrCb color space than the chrominance channels, Want *et al.* proposed AIPNet [25] which adopts multi-scale CNNs to restore the Y channel of a hazy image. Liu *et al.* [26] solved the dehazing problem in an iterative manner. For each iteration, the input was optimized via a variational model and then put into a CNN to generate the output as the input for the next iteration. Liu *et al.* [27] constructed a grid network with several residual dense blocks [28] and a channel-wise attention mechanism to remove haze. All those approaches rely on supervision with paired images, whereas our method is weakly supervised with unpaired data.

C. GANs in Dehazing

GAN originated in [29], where a generator and a discriminator are involved to play a maximin game during training in an adversarial way. Many studies [30]–[32] have proved that GANs are superior in fields of image generation and restoration. For dehazing, GAN was first introduced in [33] where dehazed images are generated by the network according to Koschmieder's law and judged by a discriminator. Later, Zhang *et al.* [6] proposed more complex structures to generate unknown variables of Koschmieder's law and adopted one discriminator to jointly judge the transmission map and the dehazed output. Li *et al.* [34] employed a conditional GAN to directly generate dehazing results without any physical model. Following Li *et al.*'s work [34], Qu *et al.* [7] proposed enhancing blocks, multi-scale generators, and multi-scale discriminators to further enhance the results. Although GANs are involved, all those dehazing methods still require paired training data. As the pioneer of leveraging unpaired data, DisentGAN [35] employed three generators to produce dehazed images, transmission maps, and ambient light from the hazy input, and then it resorted to a multi-scale discriminator to conduct adversarial training. Our method focuses on training with unpaired images as well, but it solves the dehazing problem by restoring visibility and improving realness separately.

III. Proposed Framework

In this section, the proposed RefineDNet will be presented in detail. We first introduce its overall architecture and then review how to get the preliminary dehazing results of DCP, which are essential to RefineDNet. After that, the perceptual fusion is detailed. Finally, the loss function is described.

A. Overall Framework

We divide the dehazing task into two sub-tasks, i.e., visibility restoration and realness improvement, and propose the weakly supervised framework RefineDNet. Our motivation is twofold. First, we have found that prior-based methods are more likely to remove haze at the expense of introducing artifacts whereas learning-based methods are good at producing visually pleasing results but with more haze. Thus, it should be promising to combine the merits of both kinds of methods. Second, supervised learning-based methods require pairs of clear and hazy images which are difficult to obtain in the real-world conditions, whereas the weak supervision with unpaired data can tackle this issue appropriately.

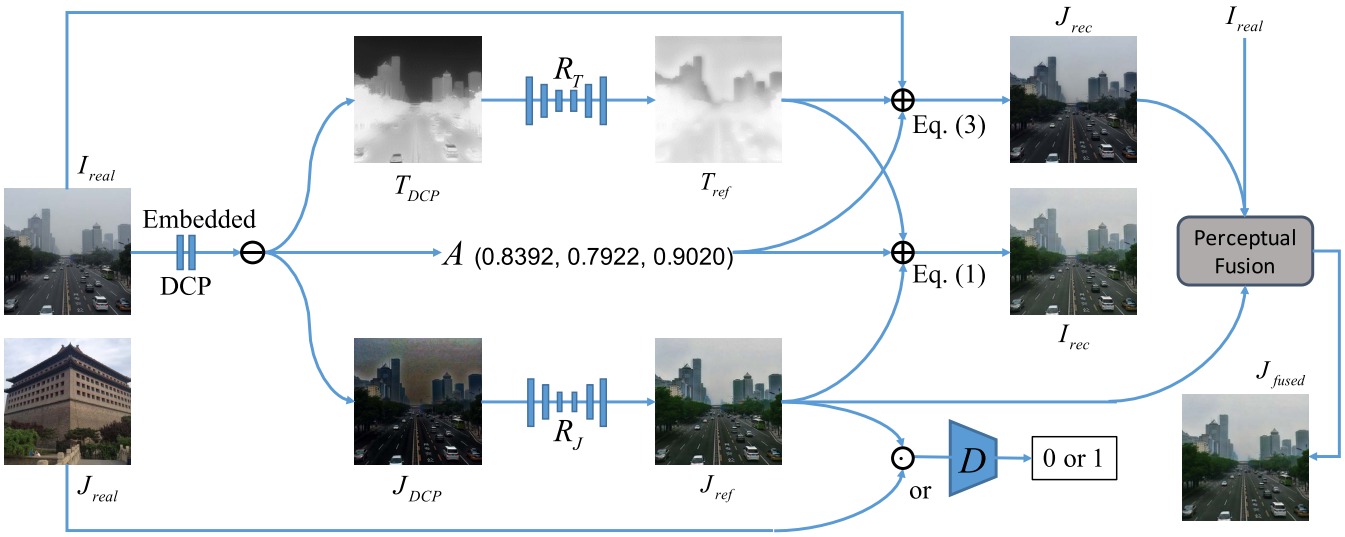


Fig. 2. Overview of RefineDNet. R_T and R_J represent the two refiner networks. D refers to the discriminator. I_{real} and J_{real} are the unpaired input images. T_{ref} and J_{ref} are the refined results of T_{DCP} and J_{DCP}, respectively. A is the ambient light, and the numbers in the brackets are the values for the R, G, B channels of A, respectively. J_{rec} is the reconstructed dehazed image via Eq. (3). I_{rec} is the reconstructed hazy image via Eq. (1). The outline of the perceptual fusion is presented in Fig. 3.

1) Two-Stage Framework: As shown in Fig. 2, RefineDNet includes two stages. In the first stage, it adopts DCP to generate the ambient light A, the preliminary dehazed image J_{DCP}, and transmission map T_{DCP}. In the second stage, T_{DCP} is refined by the refiner network R_T as T_{ref}, and J_{DCP} is refined by another refiner network R_J as J_{ref}. Note that the DCP stage is embedded in our framework, and thus, I_{real} is the only input for RefineDNet in the inference. Also, Fig. 2 indicates that T_{ref} values of sky regions are larger than their true values. However, the magnified T_{ref} values at sky regions do not affect the dehazing results, which is discussed in Appendix B in detail.

2) Weakly Supervised Learning: During training, to ensure that T_{ref} is appropriately refined, we reconstruct the hazy input as I_{rec} using T_{ref}, J_{ref}, and A according to Koschmieder's law (i.e., Eq. (1)). Then, the refiner R_T is updated by minimizing the distance between I_{real} and I_{rec}. For the reason why we can updated R_T in this way, please refer to Appendix C. Besides, there is an additional discriminator notated by D, which receives either J_{ref} or the clear sample J_{real} to enable adversarial learning. Since there is no requirement that J_{real} must be taken from the same scene of the hazy input I_{real}, the whole framework is weakly supervised. In RefineDNet, D plays a paramount role in the weak supervision. Without D, we are not able to conduct adversarial learning, and as a result, R_J will not be appropriately updated.

3) Dehazing Result Fusion: In RefineDNet, although J_{ref} is a dehazed image, it doesn't fit any physical model. To obtain a more qualified result, we reconstruct another clear output J_{rec} by reformulating Koschmieder's law as,

$$J_{rec}(\mathbf{x}) = \frac{I_{real}(\mathbf{x}) - A}{T_{ref}(\mathbf{x})} + A. \tag{3}$$

Then, we adopt powerful features for IQA to compute weights to fuse J_{ref} and J_{rec} as the final dehazed output J_{fused}. IQA metrics based on those features can generate effective judgments that are close to human perceptions, and thus we call our fusion strategy the perceptual fusion. This strategy is elaborated in Section III-C.

4) Network Structures: To justify the effectiveness of RefineDNet's motif rather than that of backbone networks, we adopt basic backbone networks provided by CycleGAN [31] to implement R_T, R_J, and D without incorporating any multi-scale or other customized structures popular in modern state-of-the-art dehazing pipelines [6], [7], [23], [35]. Specifically, R_T is a U-Net [36] that includes 8 downsampling and 8 upsampling convolution layers. R_J is a ResNet [37] with 9 residual blocks. D is a CNN with 5 convolution layers.

B. Preliminary DCP Results

DCP [8] is embedded in RefineDNet to enable the end-to-end training and inference. In this section, we briefly introduce how we gain the preliminary dehazing results, *i.e.*, T_{DCP}, J_{DCP}, and A, for RefineDNet with the imbedded DCP.

1) Dark Channel Extraction: For an input RGB image I, we calculate the channel-wise minimum value image denoted as I^{min}. Then, we apply the max pooling with the kernel size of 5×5 on the additive inverse of I^{min} and then get the additive inverse of the pooling result as the dark channel image I^{dark}. The extraction of the dark channel can be formulated as,

$$I^{dark}(\mathbf{x}) = -maxpool(-\min_{c\in R,G,B}(I^c(\mathbf{x}))) \tag{4}$$

where I^c refers to one of the R, G, B channels of I.

2) Transmission Estimation: We get the dark channels at both sides of Koschmieder's law as,

$$I^{dark}(\mathbf{x}) = J^{dark}(\mathbf{x})\, t(\mathbf{x}) + A\,(1 - t(\mathbf{x})) \tag{5}$$

where $I^{dark}(\mathbf{x})$ and $J^{dark}(\mathbf{x})$ are the dark channels of images I and J at pixel $\mathbf{x}$, respectively. According to DCP's assumption that pixels in most of the non-sky patches of natural images have the intensity values close to zero at least in one color channel, $J^{dark}(\mathbf{x}) \to 0$. Then

$$t(\mathbf{x}) = 1 - \frac{I^{dark}(\mathbf{x})}{A}. \tag{6}$$

If A is known, T_{DCP} can be obtained accordingly. In addition, we employ a guided filter to make T_{DCP} smooth. The guided filter is also imbedded in our framework and implemented using one average pooling with the kernel size of 19×19 and the stride of 1.

3) Ambient Light Estimation and the Dehazed Image: As for A, since the large pixel values (e.g., the pixel values of the sky region) in an image are very close to ambient light, the top 0.1% brightest pixels in $I^{dark}(\mathbf{x})$ are picked, and their values in color channels of $I(\mathbf{x})$ are averaged as A. With acquired A and T_{DCP}, J_{DCP} can be attained by reversing Koschmieder's law like Eq. (3).

C. Perceptual Fusion

Since J_{ref} and J_{rec} are produced in their own ways, it is highly possible that either of them is better than the other in some regions. In this sense, if better regions from either of J_{rec} and J_{ref} are assigned with larger weights, we can obtain a better result by fusing J_{rec} and J_{ref}.

Since both J_{ref} and J_{rec} are dehazed images with good visibility, their fusion with arbitrary normalized weights should not impair the visibility. Thus, we fuse them based on the image realness. Since I_{real} is a natural image with high realness, the similarity map of I_{real} and J_{ref} (or I_{real} and J_{rec}) is an informative indicator for the realness of J_{ref} (or J_{rec}). In this sense, we should assign a larger weight to either of $J_{rec}(\mathbf{x})$ and $J_{ref}(\mathbf{x})$, which has a larger corresponding value in the similarity map. To get appropriate similarity maps, we adopt two features, i.e., gradient modulus (GM) and chrominance information of the LMN color space (ChromMN), which are widely adopted in the field of IQA.

1) Feature Extraction: According to IQA studies [38]–[41], GM is computed in the Y channel (luminance channel) of the YIQ color space, and ChromMN refers to the M and N channels of the LMN color space [42], [43]. Therefore, to obtain GM, we first calculate the Y channel of YIQ using its definition as,

$$Y = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B. \tag{7}$$

Then, the GM of an image is computed as $G(\mathbf{x}) = \sqrt{G_x^2(\mathbf{x}) + G_y^2(\mathbf{x})}$, where $\mathbf{x}$ is a pixel of that image, and $G_x(\mathbf{x})$ and $G_y(\mathbf{x})$ are its partial derivatives at $\mathbf{x}$ in the Y channel. For ChromMN, we compute the M and N channels of the LMN color space as follows,

$$\begin{aligned} M &= 0.30 \cdot R + 0.04 \cdot G - 0.35 \cdot B \\ N &= 0.34 \cdot R - 0.60 \cdot G + 0.17 \cdot B. \end{aligned} \tag{8}$$

2) Similarity Calculation: We calculate the similarity with GM and ChromMN to evaluate the realness of the dehazing result. Given the GM values of two images notated as $G_1(\mathbf{x})$ and $G_2(\mathbf{x})$, the similarity $S^G(\mathbf{x})$ at pixel $\mathbf{x}$ is defined as,

$$S^G(\mathbf{x}) = \frac{2G_1(\mathbf{x}) \cdot G_2(\mathbf{x}) + C_1}{G_1^2(\mathbf{x}) + G_2^2(\mathbf{x}) + C_1} \tag{9}$$

where C_1 is set to 160 as suggested in [39].

As for the ChromMN, supposing that $M_1(\mathbf{x})$ and $N_1(\mathbf{x})$ are computed from the first image, and $M_2(\mathbf{x})$ and $N_2(\mathbf{x})$ are derived from the second, the similarity $S^C(\mathbf{x})$ at pixel $\mathbf{x}$ is calculated as,

$$S^C(\mathbf{x}) = \frac{2M_1(\mathbf{x}) \cdot M_2(\mathbf{x}) + C_2}{M_1^2(\mathbf{x}) + M_2^2(\mathbf{x}) + C_2} \cdot \frac{2N_1(\mathbf{x}) \cdot N_2(\mathbf{x}) + C_2}{N_1^2(\mathbf{x}) + N_2^2(\mathbf{x}) + C_2} \tag{10}$$

where C_2 is set to 130 as suggested in [40].

We consider both $S^G(\mathbf{x})$ and $S^C(\mathbf{x})$ and define the overall similarity map $S^{GC}(\mathbf{x})$ as,

$$S^{GC}(\mathbf{x}) = S^G(\mathbf{x}) \cdot [S^C(\mathbf{x})]^{\alpha} \tag{11}$$

where α is a parameter used to adjust the relative importance between the GM and ChromMN. Following the previous IQA study [40], we set $\alpha = 0.4$ in our experiments.

3) Fusion Weights: In this step, we convert the similarity into fusion weights. Supposing that $S_{ref}^{GC}(\mathbf{x})$ is the similarity value of $I_{real}(\mathbf{x})$ and $J_{ref}(\mathbf{x})$ at pixel $\mathbf{x}$, and $S_{rec}^{GC}(\mathbf{x})$ is the similarity value of $I_{real}(\mathbf{x})$ and $J_{rec}(\mathbf{x})$ at pixel $\mathbf{x}$, the weights of $J_{ref}(\mathbf{x})$ and $J_{rec}(\mathbf{x})$ at pixel $\mathbf{x}$ are defined as the softmax of $S_{ref}^{GC}(\mathbf{x})$ and $S_{rec}^{GC}(\mathbf{x})$. We notate the weights as $W_{ref}(\mathbf{x})$ and $W_{rec}(\mathbf{x})$, respectively. Hence,

$$\begin{bmatrix} W_{ref}(\mathbf{x}) \\ W_{rec}(\mathbf{x}) \end{bmatrix} = softmax\left(\begin{bmatrix} S_{ref}^{GC}(\mathbf{x}) \\ S_{rec}^{GC}(\mathbf{x}) \end{bmatrix}\right). \tag{12}$$

Note that $W_{ref}(\mathbf{x}) + W_{rec}(\mathbf{x}) = 1$.

In the end, we fuse J_{ref} and J_{rec} with their weights, and the final result J_{fused} is defined as,

$$J_{fused} = J_{ref} \odot W_{ref} + J_{rec} \odot W_{rec} \tag{13}$$

where $\odot$ refers to pixelwise product. Fig. 3 illustrates the outline of our perceptual fusion. Its efficacy is discussed in ablation studies in Section IV-C.

4) Adaptation to Fusing Multiple Results: The perceptual fusion can be easily adapted to fusing more than two dehazing results. Suppose that $J_1, J_2, \ldots,$ and J_n are n dehazing results to be fused. For J_i ($i \in 1, 2, \ldots, n$), we calculate the similarity map S_i^{GC} of J_i and the foggy input I according to Eq. (11). Then, for a pixel $\mathbf{x}$, the fusion weight for $J_i(\mathbf{x})$ is $W_i(\mathbf{x})$ of the softmax defined as,

$$\begin{bmatrix} W_1(\mathbf{x}) \\ W_2(\mathbf{x}) \\ \vdots \\ W_n(\mathbf{x}) \end{bmatrix} = softmax\left(\begin{bmatrix} S_1^{GC}(\mathbf{x}) \\ S_2^{GC}(\mathbf{x}) \\ \vdots \\ S_n^{GC}(\mathbf{x}) \end{bmatrix}\right). \tag{14}$$

Finally, the fused image $\hat{J}_{fused}$ is derived as,

$$\hat{J}_{fused} = \sum_{i}^{n} W_i \odot J_i \tag{15}$$

where $\odot$ refers to the pixel-wise product.

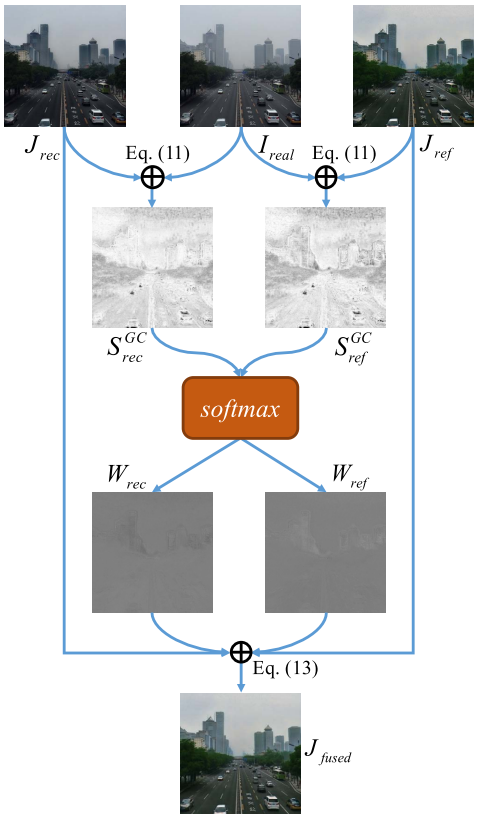


Fig. 3. Outline of the perceptual fusion. I_{real} is the hazy image. J_{rec} and J_{ref} are the reconstructed and the refined dehazing results, respectively. S^{GC}_{rec} and S^{GC}_{ref} are their similarity maps, and W_{rec} and W_{ref} are their weights in the fusion. J_{fused} is the final fused result.

D. Loss Function

The loss function of RefineDNet includes 3 terms, i.e., the GAN loss L_G, the reconstruction loss L_{rec}, and the identity loss L_{idt}. Their definitions are as follows, and we demonstrate their effectiveness in Section IV-C.

GAN loss is originally used to update the generator and the discriminator in an adversarial way [29]. In our case, L_G is used to supervise R_J and D. It is defined as,

$$L_G(R_J, D) = \mathbb{E}_{J_{real}\sim\mathcal{J}_{real}}[\log D(J_{real})] + \mathbb{E}_{J_{DCP}\sim\mathcal{J}_{DCP}}[\log(1 - D(R_J(J_{DCP})))] \quad (16)$$

where $\mathcal{J}_{real}$ is the set of all possible J_{real}, and $\mathcal{J}_{DCP}$ refers to the set of all possible J_{DCP}.

Reconstruction loss is adopted to regularize the reconstructed hazy image. As mentioned in Section III-A, we define L_{rec} as the distance between I_{real} and I_{rec}, namely,

$$L_{rec} = \|I_{real} - I_{rec}\| \quad (17)$$

where I_{real} is the hazy input, I_{rec} is obtained via Eq. (1), and $\|\cdot\|$ denotes the distance metric.

Identity loss is applied to depress the artifacts which may be introduced by the refiner R_J. Generally, this term encourages R_J to output something similar to its input when the input is a real-world clear image. In this way, R_J is less likely to cheat the discriminator by adding extra textures. We define L_{idt} as,

$$L_{idt} = \|J_{real} - R_J(J_{real})\| \quad (18)$$

where $\|\cdot\|$ is a distance metric the same as the one appearing in Eq. (17). $\|\cdot\|$ can be the L_1-norm or the L_2-norm. In our experiments, we trained RefineDNet with both L_1 and L_2 and found that the achieved models exhibited nearly the same performance. This indicates that there is no need to deliberately choose the metric form, L_1 or L_2, to train RefineDNet. For more details, please refer to Section IV-C. By default, we report the results of RefineDNet trained with L_1.

Overall loss function. Combining all loss terms, the whole objective is formulated as,

$$R^*_T, R^*_J = \arg\min_{R_T, R_J}\max_{D} \lambda L_G + L_{rec} + L_{idt} \quad (19)$$

where λ is a hyperparameter indicating the weight of L_G. the default value of λ is set to 0.02.

IV. Experiments and Discussions

Our experiments are aimed at answering the following questions: 1) Is the proposed framework RefineDNet effective? 2) Does each part of RefineDNet indeed contribute to its performance? To this end, we implemented RefineDNet with basic backbone networks to diminish the performance gain brought by advanced network architectures and compared it with several state-of-the-art methods on various datasets. Then, we conducted ablation studies concerning the two-stage strategy, loss terms, the refinement with various priors, the perceptual fusion, and the weight of L_G (λ in Eq. (19)).

A. Experimental Protocols

In this subsection, we present training and test datasets, evaluation metrics, and implementation details of RefineDNet in the experiments.

1) Indoor Training Set: We trained our framework and other competitive learning-based models on the training set of RESIDE-standard [11] called ITS (Indoor Training Set). ITS contains 13,990 clear and synthetic hazy image pairs from indoor scenes, which are generated with the images and depth maps from NYU Depth v2 [44]. Note that we didn't exploit the paired information in ITS and randomly shuffled the images during the training of RefineDNet.

2) Indoor Evaluation: We evaluated different dehazing approaches on both RESIDE-standard's test set SOTS (Synthetic Objective Testing Set) and the cross-domain Middlebury part of D-HAZY [45]. SOTS has 500 indoor pairs generated in the same manner as ITS's. The Middlebury part of D-HAZY contains 23 indoor pairs generated from images and high-quality depth maps of the Middlebury dataset [46]. Following previous studies, we adopted PSNR and SSIM [47] as the evaluation metrics on both SOTS and D-HAZY.

3) Outdoor Training Set: To verify the effectiveness of RefineDNet on outdoor scenes, a large number of unpaired clear and foggy outdoor images are required. RESIDE [11] provides 8,970 clear images and 9,129 foggy ones. However, many of those images are of low quality or out of the

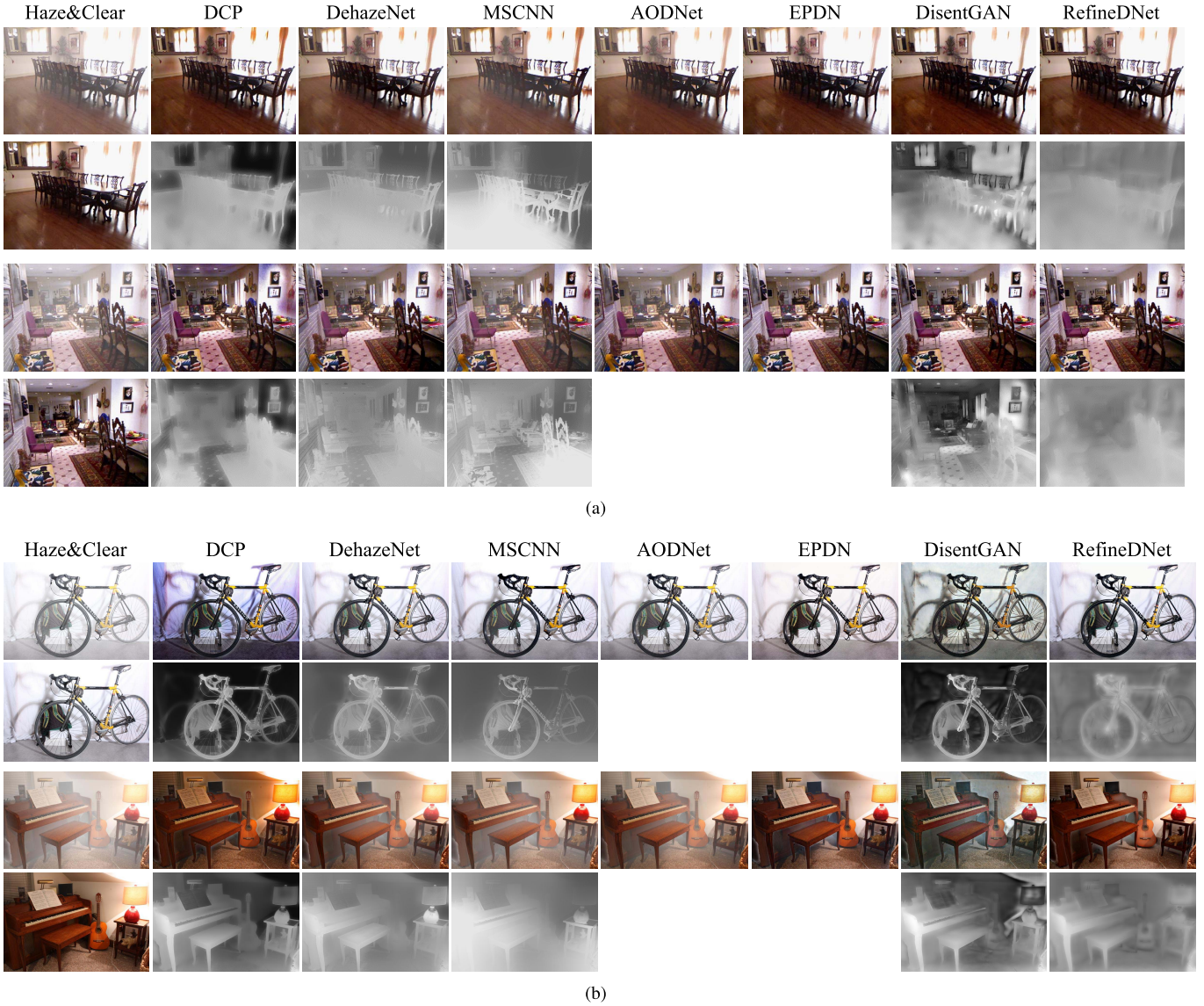


Fig. 4. Qualitative comparisons on (a) SOST and (b) D-HAZY. For each test sample, the leftmost two images are the hazy and the clear images, respectively. As for other images, the first row presents dehazed images, and the second row shows transmission maps. Note that AODNet and EPDN have no transmission map, and thus the associated slots in the figure remain blank.

dehazing scope, e.g., in very low resolutions or strong sunlight. Therefore, for clear images, we manually selected high-quality cloudy ones. For foggy images, we filtered out low-quality ones with obvious artifacts or blur. Eventually, 3,577 clear images and 2,903 foggy images were chosen as the training set notated by RESIDE-unpaired.

4) Outdoor Evaluation: We adopted the recently released real-world outdoor benchmark dataset BeDDE [48] and the recommended metrics VSI [40], VI [41], and RI [41]. This dataset contains 208 clear and foggy image pairs of high quality. Those images were collected from 23 provincial capital cities of China under different weather conditions.

5) Implementation Details: RefineDNet was trained on the deep learning platform PyTorch with the acceleration of an Nvidia Titan X GPU. We employed the Adam optimizer [49] with a learning rate of 0.0002. The default value for λ in Eq. (19) is set to 0.02. The default value for α in Eq. (11) is set to 0.4. Similar to the training of GANs, the two refiner networks of RefineDNet (R_T and R_J) and the discriminator D are alternately updated. That is, with the parameters of D fixed, R_T and R_J are trained for one iteration. Then, R_T and R_J are fixed, and D is trained for another iteration. To make our results reproducible, relevant code and datasets have been released online.[1]

B. Comparisons With State-of-the-Art Methods

We compare RefineDNet[2] with several state-of-the-art methods on indoor and outdoor datasets quantitatively and

[1]Released code: https://github.com/xiaofeng94/RefineDNet-for-dehazing

[2]RefineDNet models trained with L_1 and L_2 actually have similar performance. Unless otherwise specified, the reported RefineDNet's results were obtained with L_1.

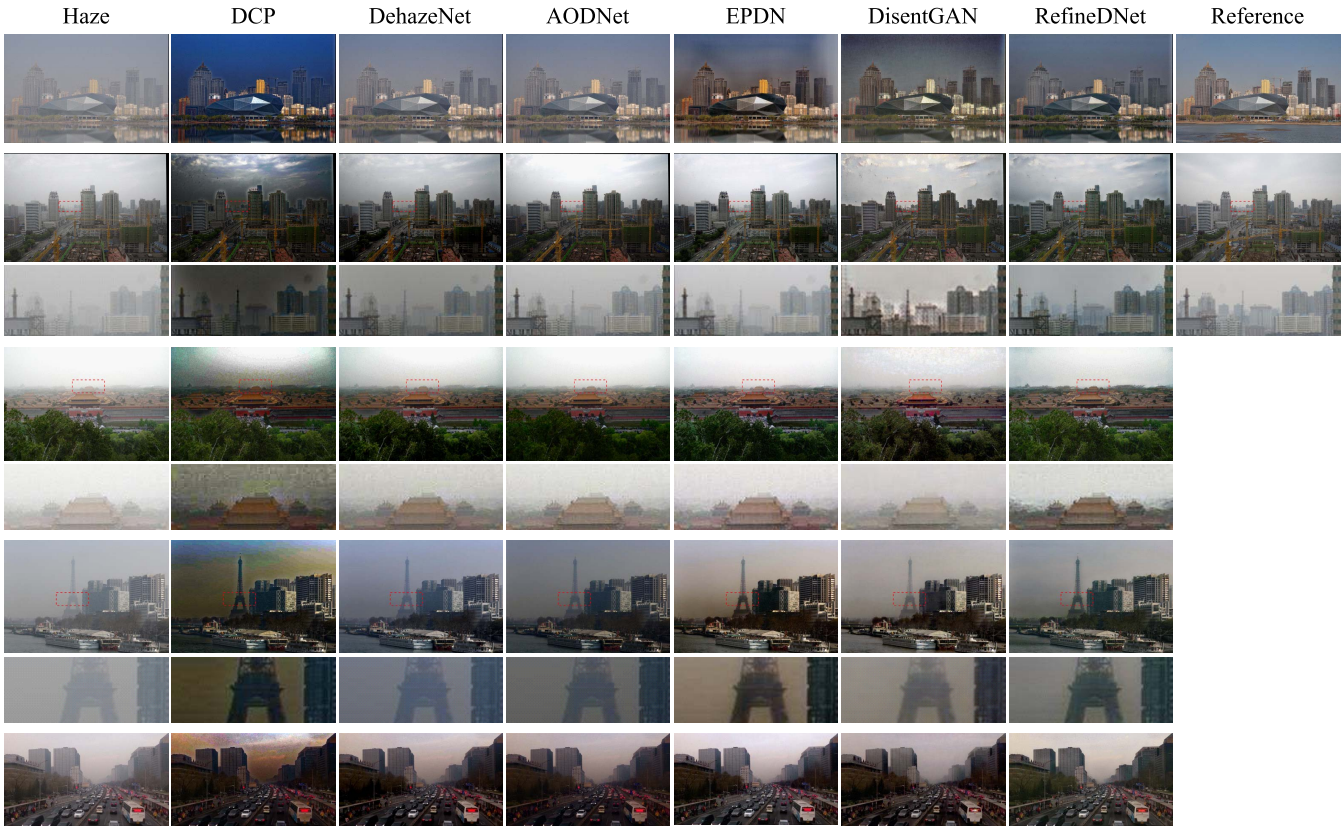


Fig. 5. Qualitative comparisons on BeDDE and hazy images from the Internet. Note that there is no reference for the Internet samples.

TABLE I
QUANTITATIVE EVALUATIONS OF VARIOUS DEHAZING METHODS ON INDOOR DATASETS, I.E., SOST AND D-HAZY

Method	Type	SOST		D-HAZY	
		PSNR	SSIM	PSNR	SSIM
FVR [13]	Prior	15.72	0.7483	14.35	0.8051
DCP [8]	Prior	16.62	0.8179	15.09	**0.8303**
BCCR [3]	Prior	16.88	0.7913	15.02	0.8207
CAP [50]	Prior	19.05	0.8364	13.68	0.7269
NLD [19]	Prior	17.29	0.7489	13.46	0.7561
DehazeNet [4]	Supervised	21.14	0.8472	13.76	0.8087
MSCNN [21]	Supervised	17.57	0.8102	13.57	0.7985
AODNet [5]	Supervised	19.06	0.8504	13.13	0.7948
GFN [23]	Supervised	22.30	0.8800	13.15	0.7957
EPDN [7]	Supervised	21.55	0.9071	14.44	0.8189
FAMED-Net [22]	Supervised	23.63	0.9209	12.55	0.7999
DisentGAN [35]	Weakly	22.12	0.8991	13.55	0.7724
RefineDNet (Ours)	Weakly	**24.23**	**0.9431**	**15.44**	0.8267

qualitatively. Among those approaches, prior-based ones are FVR [13], DCP [8], BCCR [3], CAP [50] and NLD [19]. Supervised ones are DehazeNet [4], MSCNN [21], AOD-Net [5], GFN [23], PQC [24], EPDN [7], GridDehazeNet [27], and FAMED-Net [22]. The weakly supervised one is Disent-GAN [35]. In this subsection, the champion and the runner-up for each metric in the tables are highlighted in boldface and underline, respectively.

1) Results on Indoor Datasets: Table I shows the quantitative results of different methods on both SOTS and D-HAZY. As shown, although RefineDNet is weakly supervised, it still outperforms the competitors including supervised ones in terms of most metrics, which demonstrates the superiority of our framework. Note that some methods, e.g., EPDN [7] and GridDehazeNet [27], adopt complex multi-scale structures to improve their performance, while RefineDNet only makes use of basic backbone networks provided by cycleGAN [31]. Therefore, RefineDNet might be further enhanced with specially designed multi-scale structures.

Additionally, RefineDNet achieves consistently excellent performance on both two datasets, whereas FAMED-Net [22] is the runner-up method on SOTS but fails to remain competitive on the cross-domain part of D-HAZY. Such a result indicates that RefineDNet is more robust with a good generalization ability. Fig. 4(a) and 4(b) visualize the dehazing results of different methods using examples from SOTS and D-HAZY, respectively. As shown, RefineDNet is capable of removing haze without introducing obvious artifacts or distortions.

2) Results on Outdoor Datasets: In Table II, we present the quantitative evaluation results on the real-world benchmark dataset BeDDE. All the supervised models were trained on indoor datasets, and the weakly supervised DisentGAN [35] and RefineDNet were trained on RESIDE-unpaired. It seems unfair at first glance, but since there are no paired outdoor images to train supervised models, it is one of the important advantages of weakly supervised approaches to be able to train with unpaired outdoor images. Thus, the comparisons are reasonable.

According to Table II, RefineDNet still achieves comparable performance, and it's the sole method that achieves high performance in terms of both VI and RI. Since VI and

TABLE II

COMPARISONS OF STATE-OF-THE-ART METHODS ON THE REAL-WORLD BENCHMARK DATASET, BEDDE

Method	Type	VSI	VI	RI
FVR [13]	Prior	0.8858	0.8054	0.9511
DCP [8]	Prior	0.9462	**0.9111**	0.9654
BCCR [3]	Prior	0.9251	0.8907	0.9588
CAP [50]	Prior	0.9156	0.8507	0.9482
NLD [19]	Prior	0.8959	0.8278	0.9557
DehazeNet [4]	Supervised	0.9522	0.8902	**0.9718**
MSCNN [21]	Supervised	0.9471	0.8920	0.9702
AODNet [5]	Supervised	0.9540	0.8961	0.9703
GFN [23]	Supervised	0.9389	0.8659	0.9651
PQC [24]	Supervised	0.9541	0.8923	0.9694
EPDN [7]	Supervised	0.9498	0.8960	0.9648
GridDehazeNet [27]	Supervised	0.9422	0.8837	0.9687
FAMED-Net [22]	Supervised	0.9303	0.8669	0.9691
DisentGAN [35]	Weakly	0.9424	0.8655	0.9613
RefineDNet (Ours)	Weakly	**0.9597**	0.9070	0.9708

Fig. 6. The dehazing results of DCP and corresponding results refined by RefineDNet. The hazy and haze-free images comes from SOST. As shown, RefineDNet effectively removes the artifacts and refines the dehazing results generated by DCP.

RI evaluate the visibility and realness of dehazing methods, respectively, it can be concluded that RefineDNet can not only remove haze but also avoid artifacts. Moreover, it is practical to collect unpaired training images for RefineDNet in a large amount. Thus, RefineDNet might receive further improvement with more high-quality training samples. Fig. 5 illustrates the visual results of different dehazing methods for real-world hazy samples as qualitative evaluations. The samples of the first 3 rows come from BeDDE, and the others are from the Internet. As we can see, RefineDNet works well in those cases by generating dehazing results with high realness and visibility, whereas others might involve artifacts or fail to remove haze. Considering those results together, we can conclude that RefineDNet boasts the advantages of both DCP and learning-based methods.

C. Ablation Study

1) Analysis of the Two-Stage Dehazing: We aim to justify the effectiveness of RefineDNet's main idea, i.e., restoring visibility first with priors and then improving the realness of the results via learning-based refinement. Therefore, we compare RefineDNet with three baselines, DCP [8], CycleGAN [31] and BasicNet. DCP is the prior-based method

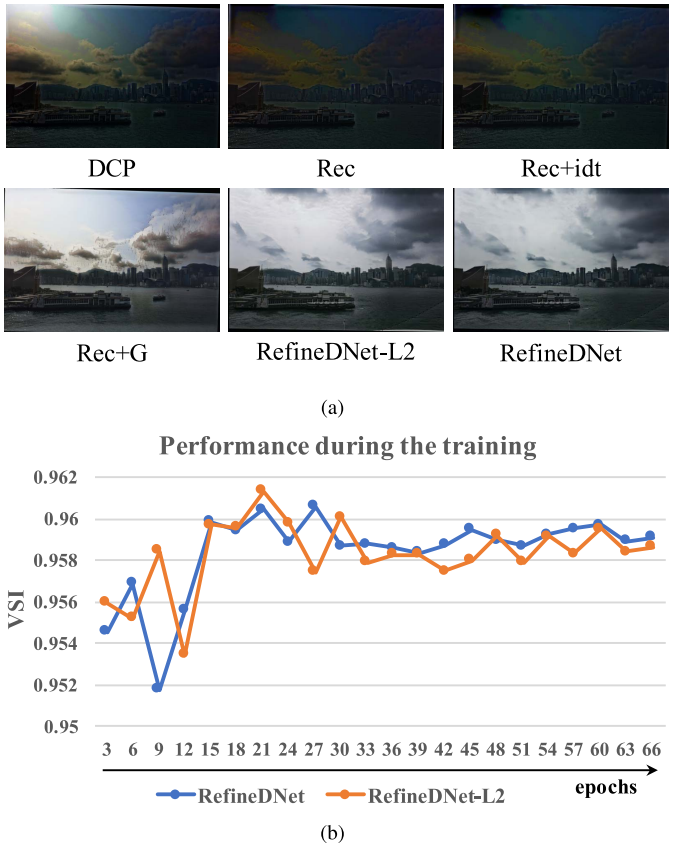


Fig. 7. (a) Visual results of DCP [8] and our dehazing models trained with different loss terms. RefineDNet involves all the three loss terms. (b) VSI scores of RefineDNet and RefineDNet-L2 on BeDDE. The performance for every three epochs is exhibited.

TABLE III

EVALUATION OF THE EFFECTIVENESS OF THE TWO-STAGE DEHAZING STRATEGY ON SOTS

	DCP [8]	CycleGAN [31]	BasicNet	RefineDNet
PSNR	16.62	17.51	21.68	**24.23**
SSIM	0.8179	0.7705	0.8883	**0.9431**

chosen for the first stage of RefineDNet. cycleGAN is a general unpaired image-to-image transformation framework. BasicNet has exactly the same structure as the second stage of RefineDNet but takes I_{real} as the input rather than T_{DCP} and J_{DCP}. We trained cycleGAN, BasicNet, and RefineDNet on ITS and evaluated them on SOTS. Table III provides the evaluation results.

As shown in Table III, RefineDNet outperforms the others with a large margin. Since the only difference between BasicNet and RefineDNet is whether to dehaze with two stages, it is clearly testified that dehazing is highly plausible and effective by restoring visibility with priors first and then improving realness via the learning-based refinement. To better demonstrate how RefineDNet improves the results of DCP, we present several samples from SOST in Fig. 6. Apparently, RefineDNet effectively removes artifacts generated by DCP and produces very natural images that are highly close to the haze-free ground-truth.

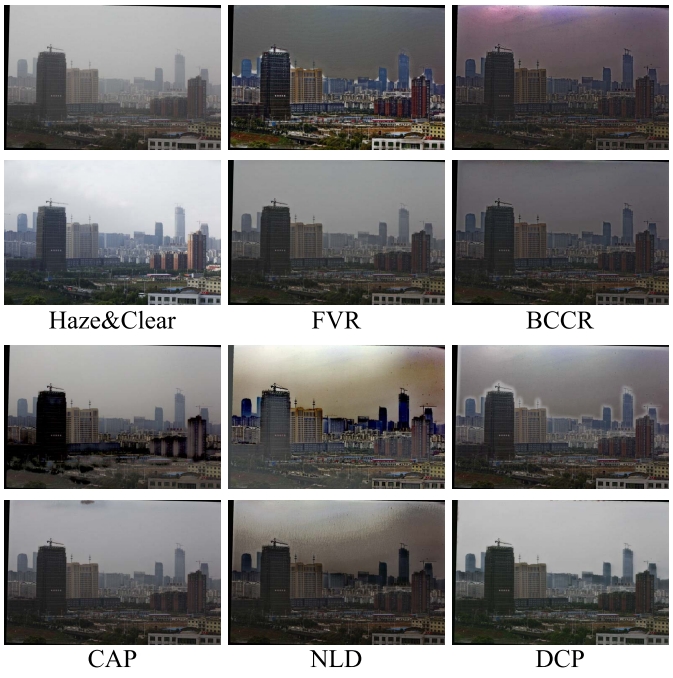


Fig. 8. Visual results of RefineDNet with various prior-based methods. For each method, the first row presents the preliminary result of this method, and the second row presents the result after the refinement. Images in the column of "Haze&Clear" are the hazy image and the clear reference.

TABLE IV

PERFORMANCE OF DCP AND DEHAZING MODELS TRAINED WITH VARIOUS COMBINATIONS OF LOSS TERMS ON BEDDE

	DCP [8]	Rec	Rec+idt	Rec+G	RefineDNet-L2	RefineDNet
VSI	0.9462	0.9179	0.9192	0.9488	0.9595	**0.9597**
VI	**0.9111**	0.8861	0.8774	0.8935	0.9066	0.9070
RI	0.9654	0.9595	0.9569	0.9694	0.9690	**0.9708**

2) Analysis of Loss Terms: We demonstrate how loss terms in our framework affect the results by comparing RefineDNet with four baselines trained with different loss terms on RESIDE-unpaired. Those baselines are 1) **Rec**: The model trained with L_{rec} only; 2) **Rec+G**: The model trained with L_{rec} and L_G; 3) **Rec+idt**: The model trained with L_{rec} and L_{idt}; 4) **RefineDNet-L2**: We adopted L_2 distance in the loss function and trained the model with all loss terms. Table IV provides the quantitative evaluations on BeDDE. Fig. 7(a) illustrates the visual results of DCP, the four baselines, and RefineDNet. Fig. 7(b) displays the VSI scores of RefineDNet and **RefineDNet-L2** evaluated on BeDDE every three epochs during the training.

From those results, we have several interesting findings. First, supported by the results of **Rec** and **Rec+G**, the GAN loss contributes to considerable improvement quantitatively, which is probably because when L_G is absent, **Rec**'s refiner R_J has no idea on how to improve the result and produces similar outputs as DCP's. Second, the GAN loss seems to introduce unwanted structures that don't exist in the original scene, whereas the identity loss can depress those structures and further improves the performance of **Rec+G**. This can be explained by the fact that the identity loss encourages R_J to generate similar outputs as its inputs, and thus the unwanted structures are eliminated to some extent. Third, **RefineDNet-L2** and RefineDNet achieve similar performance. Considering the only difference between them is the distance used in the training loss, we can conclude that both L_1 and L_2 are qualified to train the proposed framework.

TABLE V

PERFORMANCE OF OUR MODELS WITH VARIOUS PRIORS ON BEDDE. "VSI" ("VI" OR "RI") AND "+R/VSI" ("+R/VI" OR "+R/RI") REFER TO THE VSI (VI OR RI) SCORES BEFORE AND AFTER THE REFINEMENT, RESPECTIVELY

	FVR [13]	BCCR [3]	CAP [50]	NLD [19]	DCP [8]
VSI	0.8858	0.9251	0.9156	0.8959	0.9462
+R/VSI	0.9524	0.9423	0.9489	0.9341	0.9597
VI	0.8054	0.8907	0.8507	0.8278	0.9111
+R/VI	0.8872	0.8864	0.8876	0.8867	0.9070
RI	0.9654	0.9588	0.9482	0.9557	0.9654
+R/RI	0.9720	0.9680	0.9714	0.9593	0.9708

TABLE VI

EVALUATION OF THE EFFECTIVENESS OF THE PERCEPTUAL FUSION ON DIFFERENT DATASETS

		J_{rec}	J_{ref}	J_{fused}
SOST	PSNR	24.13	23.85	**24.23**
	SSIM	0.9401	0.9379	**0.9431**
D-HAZY	PSNR	14.27	14.74	**15.44**
	SSIM	0.8252	0.8214	**0.8267**
BeDDE	VSI	0.9537	0.9499	**0.9597**
	VI	0.8918	0.8805	**0.9070**
	RI	0.9683	0.9678	**0.9708**

3) Analysis of RefineDNet With Different Priors: We perform an ablation study to demonstrate how RefineDNet works with different preliminary dehazing results generated by various prior-based methods. This study involves 5 representative prior-based methods, namely FVR [13], BCCR [3], CAP [50], NLD [19] and DCP. All models were trained on RESIDE-unpaired and evaluated on BeDDE. Table V shows the quantitative evaluations of different preliminary results and evaluations of the refined counterparts from RefineDNet. In Table V, for preliminary results of each prior, "VSI", "VI", and "RI" refer to the VSI, VI, and RI scores, respectively. "+R/VSI", "+R/VI", and "+R/RI" refer to the corresponding scores of the refined results. For example, "+R/VSI" means the VSI score after the refinement. Fig. 8 visually illustrates the results before and after the refinement.

As we can see, RefineDNet is effective with various prior-based methods and can be regarded as a general dehazing framework. Moreover, RefineDNet with DCP achieves the best performance, and a probable explanation for this outcome is that DCP can generate better preliminary results and simplify the task of refinement in the second stage of our framework. In this sense, with better priors, RefineDNet can be further improved.

4) Analysis of the Perceptual Fusion: In Table VI, we present different outputs of RefineDNet, i.e., J_{rec}, J_{ref}, and J_{fused}, to evaluate the effectiveness of our perceptual fusion strategy. As shown, J_{fused} outperforms the other two on all the three test datasets in terms of various metrics. Considering

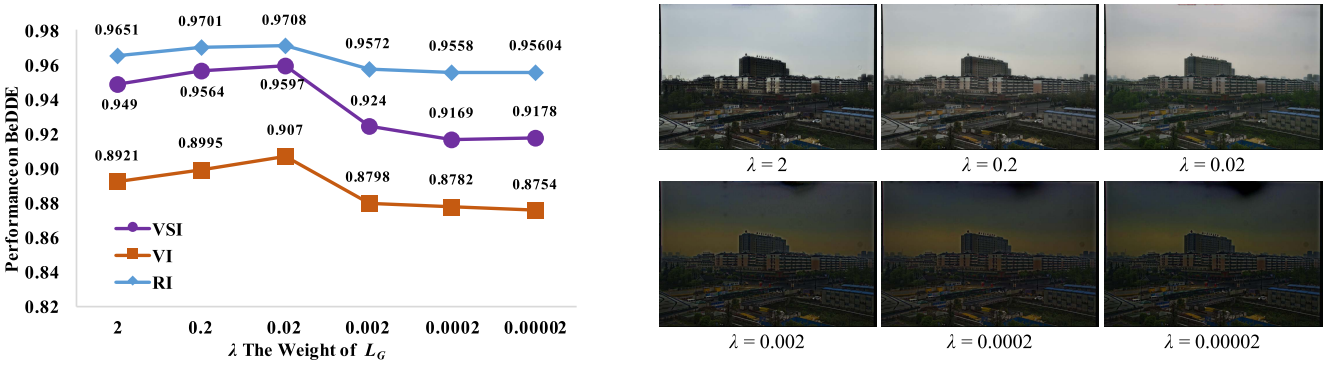


Fig. 9. (a) Performance plot for RefineDNet models trained with different λ values. All models were trained on RESIDE-unpaired and evaluated on BeDDE. (b) Dehazing results of all models in (a). The value of λ for each model is provided under the image.

that J_{fused} is the combination of J_{rec} and J_{ref} without further enhancement, we can conclude that our perceptual fusion strategy contributes to better dehazing results. Note that we only fuse two intermediate dehazing results in RefineDNet, but the perceptual fusion can be easily adapted to fuse more results. Therefore, we can expect its applications in other multi-output dehazing methods.

5) Analysis of L_G's Weight: Besides the two-stage strategy, loss terms, priors, and the fusion scheme, we further demonstrate the impact of λ on the dehazing results. Note that λ in Eq. (19) is the sole parameter in our loss function. We trained RefineDNet on RESIDE-unpaired with λ as 2, 2×10^{-1}, 2×10^{-2}, 2×10^{-3}, 2×10^{-4}, and 2×10^{-5}, respectively. Then, all models were evaluated on BeDDE. Fig. 9(a) displays the performance of RefineDNet models trained with different λ values.

Apparently, as λ gets smaller, the performance first increases and then decreases. When λ is between 0.2 and 0.02, the results remain satisfactory. However, if λ gets less than 0.02, the performance suffers a huge drop. Probably, with a too large weight for L_G, the training of RefineDNet behaves the same as that of the baseline **Rec+G**. In other words, R_J is encouraged to generate unpleasant structures to cheat the discriminator D. On the other hand, if the weight λ is too small, RefineDNet deteriorates as the baseline **Rec+idt**. Therefore, R_J learns nothing about the refinement and produces dark and distorted outputs as DCP does. To support our analysis, we exhibit several dehazing results of RefineDNet trained with different λ values in Fig. 9(b). As shown, when λ equals 0.2 or 0.02, the results are visually pleasing. However, others are unacceptable due to either getting too dark or involving plenty of artifacts.

V. Conclusion

In this work, we propose a simple yet effective two-stage weakly supervised dehazing framework RefineDNet for two purposes, i.e., merging the merits of prior-based and learning-based methods and addressing the lack of paired training images. To get more qualified results, we also propose a perceptual fusion strategy to fuse different outputs of RefineDNet. According to the experimental results, RefineDNet can achieve state-of-the-art performance using basic backbone networks on both indoor and outdoor datasets. Its components are thoroughly studied and demonstrated to be effective. Additionally, we construct an unpaired dataset with 6,480 outdoor images which can benefit further studies of weakly supervised dehazing. In the future, we are going to explore customized structures and priors to ameliorate RefineDNet.

Appendix A

In Section I, it is illustrated that the prior-based dehazing methods are relatively better for restoring visibility whereas learning-based methods are preferable for improving image realness. In this appendix, we provide some theoretical explanations for this phenomenon.

A. Prior-Based Methods

The visibility is decided by the contrast of objects against the ambient light A. Fog reduces the contrast and leads to low visibility, and thus, many prior-based methods restore the visibility by improving the contrast. For example, DCP [8] generates the dehazed image with the least upper bound of the contrast that follows Koschmieder's law [9], which can be proved as follows.

On one hand, based on Eq. (5), the transmission of the image $I(\mathbf{x})$ at pixel $\mathbf{x}$ can be defined as,

$$t(\mathbf{x}) = \frac{A - I^{dark}(\mathbf{x}) + J^{dark}(\mathbf{x})t(\mathbf{x})}{A}. \tag{20}$$

Here, $J^{dark}(\mathbf{x}) \geq 0$, and according to Eq. (2), $t(\mathbf{x}) = e^{-\beta d(\mathbf{x})} \geq 0$. Hence,

$$t(\mathbf{x}) \geq \frac{A - I^{dark}(\mathbf{x})}{A}. \tag{21}$$

According to DCP's assumption, $J^{dark}(\mathbf{x})$ is close to or equals to zero. Therefore, in some cases, $t(\mathbf{x}) = (A - I^{dark}(\mathbf{x}))/A$ that is the greatest lower bound of $\mathbf{x}$.

On the other hand, the contrast of the dehazed image $J(\mathbf{x})$ against A at $\mathbf{x}$, $C_J(\mathbf{x})$, is defined as,

$$C_J(\mathbf{x}) = \frac{|J(\mathbf{x}) - A|}{A}. \tag{22}$$

According to Koschmieder's law, $J(\mathbf{x})$ can be defined as,

$$J(\mathbf{x}) = \frac{I(\mathbf{x}) - A}{t(\mathbf{x})} + A. \tag{23}$$

Applying Eq. (23) to Eq. (22), $C_J(\mathbf{x})$ can be written as,

$$C_J(\mathbf{x}) = \frac{|I(\mathbf{x}) - A|}{A \cdot t(\mathbf{x})}. \tag{24}$$

Therefore, when $t(\mathbf{x})$ is at its minimum, $C_J(\mathbf{x})$ gets its maximum. Since DCP adopts the greatest lower bound of $t(\mathbf{x})$ to generate the dehazing result with Koschmieder's law, the dehazed image gets its least upper bound of the contrast. As a result, the dehazing results of DCP enjoy high contrast, and the visibility is satisfactory.

However, it is inevitable that the DCP's assumption does not hold in some regions. In this case, $t(\mathbf{x})$ is underestimated, and the contrast is overly enhanced. Thus, the noises of those regions are magnified, which leads to obvious artifacts. To conclude, DCP can restore visibility effectively but introduce artifacts.

B. Learning-Based Methods

On one hand, according to Deep Image Prior [10], the structure of a neural network is sufficient to capture a great deal of low-level image statistics prior to any learning. Thus, neural networks can work well in many reverse problems such as denoising, super-resolution, and inpainting. In other words, neural networks are inborn to generate natural images with few artifacts.

On the other hand, learning-based dehazing methods lack real-world foggy and clear image pairs as supervision. Instead, most studies adopt simulated indoor image pairs to train and evaluate their dehazing models. However, there are considerable data gaps between the simulated indoor and real-world outdoor foggy images. Therefore, supervised dehazing models are not fully trained to handle the real-world fog. Then, those models may perform well on their own test set but fail to deal with real-world foggy images.

In conclusion, the learning-based dehazing methods are able to generate high realness results with CNNs but may fail to remove the fog of the real-world outdoor images due to the lack of suitable training data.

APPENDIX B

In Section III-A, we claim that inaccurate T_{ref} values of the sky, which are larger than the true values as shown in Fig. 2, do not affect the dehazing results in sky regions. In this appendix, we explain the reason for our claim in detail.

A. The Observed Color of the Sky

Since there is no object in the sky, the intrinsic luminance of the sky J^{sky} equals to zero, i.e. $J^{sky} = 0$, which is not what we observe in the clear weather. Thus, unlike other objects, the intrinsic luminance of the sky cannot be regarded as its color in the dehazing result. Actually, the observed color of the sky is close to the ambient light (A) in the atmosphere, which can be explained as follows.

Denote by β_1 the extinction coefficient of the clear day. The color of the sky I^{sky}_{clear} can be formulated as,

$$\begin{aligned} I^{sky}_{clear} &= J^{sky}(\mathbf{x}) \cdot e^{-\beta_1 \cdot d(\mathbf{x})} + A \cdot (1 - e^{-\beta_1 \cdot d(\mathbf{x})}) \\ &= 0 + A \cdot (1 - e^{-\beta_1 \cdot d(\mathbf{x})}) \end{aligned} \tag{25}$$

where $\mathbf{x}$ refers to the location of one pixel in sky regions. Since $d(\mathbf{x}) \to \infty$, the transmission $t_1(\mathbf{x}) = e^{-\beta_1 \cdot d(\mathbf{x})}$ is close to zero. Thus, $I^{sky}_{clear} \approx A$ and $I^{sky}_{clear} \leq A$. When haze presents in the scene, the extinction coefficient increases from β_1 to a much larger value. In this case, for the color of the sky with haze denoted by I^{sky}_{haze}, we have similar conclusions that $I^{sky}_{haze} \approx A$ and $I^{sky}_{haze} \leq A$. Therefore, the color of sky regions in either of clear and hazy days is close to and less than the ambient light.

B. The Color of the Sky After Dehazing

Supposing t_2 is the estimated transmission, the color of the dehazed sky ($I^{sky}_{dehazed}$) is formulated as,

$$I^{sky}_{dehazed}(\mathbf{x}) = \frac{I^{sky}_{haze}(\mathbf{x}) - A}{t_2(\mathbf{x})} + A \tag{26}$$

where $\mathbf{x}$ refers to the location of one pixel in sky regions. Supposing that $\Delta(\mathbf{x}) = A - I^{sky}_{haze}(\mathbf{x})$, which is for sure a quite small number, then,

$$I^{sky}_{dehazed}(\mathbf{x}) = -\frac{\Delta(\mathbf{x})}{t_2(\mathbf{x})} + A. \tag{27}$$

Therefore, once t_2 is not a small value, $I^{sky}_{dehazed} \approx A$, which is a good estimation of I^{sky}_{clear}, regardless of the value of t_2.

APPENDIX C

In Section III-A, it is mentioned that the refiner network R_T is updated by minimizing the distance of I_{real} and I_{rec}. In this appendix, we provide the reason why R_T can be trained appropriately with this loss function.

As we know, there is no ground truth to supervise R_T, and thus, we have to update it in an unsupervised way. To this end, we leverage Koschmieder's law [9] to construct the unsupervised training. There are four parameters in Koschmieder's law, namely, the foggy image, clear image, ambient light and transmission, and any one of them can be derived by the other three. Since the refiner network R_J of RefineDNet is updated by adversarial learning with unpaired data, its output J_{ref} is a dehazed image with high visibility and realness. Moreover, a satisfactory estimation of the ambient light A can be obtained by the estimation method of DCP [8]. Therefore, with J_{ref}, A, and I_{real}, we can get the pseudo ground truth of transmission denoted as T_{pse}, according to Koschmieder's law, i.e.,

$$T_{pse} = \frac{I_{real} - A}{J_{ref} - A}. \tag{28}$$

Then, R_T can be updated by minimizing the distance between T_{pse} and T_{ref} as,

$$\begin{aligned}\|T_{pse} - T_{ref}\| &= \|\frac{I_{real} - A}{J_{ref} - A} - T_{ref}\| \\ &= \frac{\|I_{real} - [J_{ref} \odot T_{ref} + A(1 - T_{ref})]\|}{\|J_{ref} - A\|}.\end{aligned} \tag{29}$$

Applying Eq. (1) to Eq. (29), we can get the following formula,

$$\|T_{pse} - T_{ref}\| = \frac{\|I_{real} - I_{rec}\|}{\|J_{ref} - A\|} \propto \|I_{real} - I_{rec}\|. \tag{30}$$

Therefore, we can update R_T by minimizing the distance of I_{real} and I_{rec}.

REFERENCES

[1] R. Fattal, "Single image dehazing," *ACM Trans. Graph.*, vol. 27, no. 3, p. 72, Aug. 2008.

[2] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1956–1963.

[3] G. Meng, Y. Wang, J. Duan, S. Xiang, and C. Pan, "Efficient image dehazing with boundary constraint and contextual regularization," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 617–624.

[4] 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.

[5] 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.

[6] 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.

[7] Y. Qu, Y. Chen, J. Huang, and Y. Xie, "Enhanced Pix2pix dehazing network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8160–8168.

[8] 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.

[9] W. E. K. Middleton and V. Twersky, "Vision through the atmosphere," *Phys. Today*, vol. 7, no. 3, p. 21, Mar. 1954.

[10] V. Lempitsky, A. Vedaldi, and D. Ulyanov, "Deep image prior," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9446–9454.

[11] B. Li *et al.*, "Benchmarking single-image dehazing and beyond," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 492–505, Jan. 2019.

[12] R. T. Tan, "Visibility in bad weather from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[13] J.-P. Tarel and N. Hautiere, "Fast visibility restoration from a single color or gray level image," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 2201–2208.

[14] S. Salazar-Colores, E. Cabal-Yepez, J. M. Ramos-Arreguin, G. Botella, L. M. Ledesma-Carrillo, and S. Ledesma, "A fast image dehazing algorithm using morphological reconstruction," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2357–2366, May 2019.

[15] Q. Liu, X. Gao, L. He, and W. Lu, "Single image dehazing with depth-aware non-local total variation regularization," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 5178–5191, Oct. 2018.

[16] K. Nishino, L. Kratz, and S. Lombardi, "Bayesian defogging," *Int. J. Comput. Vis.*, vol. 98, no. 3, pp. 263–278, Jul. 2012.

[17] J. Kim and R. Zabih, "Factorial Markov random fields," in *Proc. Eur. Conf. Comput. Vis.*, 2002, pp. 321–334.

[18] R. Fattal, "Dehazing using color-lines," *ACM Trans. Graph.*, vol. 34, no. 1, pp. 13:1–13:14, Dec. 2014.

[19] 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.

[20] T. M. Bui and W. Kim, "Single image dehazing using color ellipsoid prior," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 999–1009, Feb. 2018.

[21] G. Tang, L. Zhao, R. Jiang, and X. Zhang, "Single image dehazing via lightweight multi-scale networks," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2019, pp. 154–169.

[22] J. Zhang and D. Tao, "FAMED-net: A fast and accurate multi-scale end-to-end dehazing network," *IEEE Trans. Image Process.*, vol. 29, pp. 72–84, 2020.

[23] 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.

[24] S. Santra, R. Mondal, and B. Chanda, "Learning a patch quality comparator for single image dehazing," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4598–4607, Sep. 2018.

[25] A. Wang, W. Wang, J. Liu, and N. Gu, "AIPNet: Image-to-image single image dehazing with atmospheric illumination prior," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 381–393, Jan. 2019.

[26] Y. Liu, J. Pan, J. Ren, and Z. Su, "Learning deep priors for image dehazing," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2492–2500.

[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. 7314–7323.

[28] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2472–2481.

[29] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[30] 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. 1125–1134.

[31] 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. 2223–2232.

[32] C. Ledig *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4681–4690.

[33] H. Zhu, X. Peng, V. Chandrasekhar, L. Li, and J.-H. Lim, "DehazeGAN: When image dehazing meets differential programming," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 1234–1240.

[34] R. Li, J. Pan, Z. Li, and J. Tang, "Single image dehazing via conditional generative adversarial network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8202–8211.

[35] X. Yang, Z. Xu, and J. Luo, "Towards perceptual image dehazing by physics-based disentanglement and adversarial training," in *Proc. AAAI Conf. Artifi. Intell.*, 2018, pp. 7485–7492.

[36] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.

[37] 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.

[38] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, Aug. 2011.

[39] W. Xue, L. Zhang, X. Mou, and A. C. Bovik, "Gradient magnitude similarity deviation: A highly efficient perceptual image quality index," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 684–695, Feb. 2014.

[40] L. Zhang, Y. Shen, and H. Li, "VSI: A visual saliency-induced index for perceptual image quality assessment," *IEEE Trans. Image Process.*, vol. 23, no. 10, pp. 4270–4281, Oct. 2014.

[41] S. Zhao, L. Zhang, S. Huang, Y. Shen, and S. Zhao, "Dehazing evaluation: Real-world benchmark datasets, criteria, and baselines," *IEEE Trans. Image Process.*, vol. 29, pp. 6947–6962, 2020.

[42] J.-M. Geusebroek, R. Van Den Boomgaard, A. W. M. Smeulders, and A. Dev, "Color and scale: The spatial structure of color images," in *Proc. Eur. Conf. Comput. Vis.*, 2000, pp. 331–341.

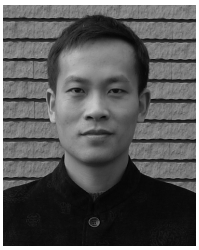
[43] J. M. Geusebroek, R. van den Boomgaard, A. W. M. Smeulders, and H. Geerts, "Color invariance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 12, pp. 1338–1350, Dec. 2001.

[44] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 746–760.

[45] C. Ancuti, C. O. Ancuti, and C. De Vleeschouwer, "D-HAZY: A dataset to evaluate quantitatively dehazing algorithms," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 2226–2230.

[46] D. Scharstein *et al.*, "High-resolution stereo datasets with subpixel-accurate ground truth," in *Proc. German Conf. Pattern Recognit.*, 2014, pp. 31–42.
[47] 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.
[48] S. Zhao, L. Zhang, S. Huang, Y. Shen, S. Zhao, and Y. Yang, "Evaluation of defogging: A real-world benchmark dataset, a new criterion and baselines," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2019, pp. 1840–1845.
[49] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980
[50] 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.

Shiyu Zhao received the B.S. degree from the School of Software Engineering, Tongji University, Shanghai, China, in 2017, where he is currently pursuing the master's degree. His research interests include visibility enhancement for bad weather images, scene understanding, and machine learning.



Lin Zhang (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees from the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China, in 2003 and 2006, respectively, and the Ph.D. degree from the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, in 2011. From March 2011 to August 2011, he was a Research Associate with the Department of Computing, The Hong Kong Polytechnic University. In August 2011, he joined the School of Software Engineering, Tongji University, Shanghai, where he is currently a Full Professor. His current research interests include environment perception of intelligent vehicle, pattern recognition, computer vision, and perceptual image/video quality assessment. He serves as an Associate Editor for IEEE ROBOTICS AND AUTOMATION LETTERS, and *Journal of Visual Communication and Image Representation*. He was awarded as a Young Scholar of Changjiang Scholars Program, Ministry of Education, China.

Ying Shen (Member, IEEE) received the B.S. and M.S. degrees from the Software School, Shanghai Jiao Tong University, Shanghai, China, in 2006 and 2009, respectively, and the Ph.D. degree from the Department of Computer Science, City University of Hong Kong, Hong Kong, in 2012. In 2013, she joined the School of Software Engineering, Tongji University, Shanghai, where she is currently an Associate Professor. Her research interests include bioinformatics and pattern recognition.

Yicong Zhou (Senior Member, IEEE) received the B.S. degree in electrical engineering from Hunan University, Changsha, China, and the M.S. and Ph.D. degrees in electrical engineering from Tufts University, Medford, MA, USA. He is currently an Associate Professor and the Director of the Vision and Image Processing Laboratory, Department of Computer and Information Science, University of Macau, Macau. His research interests include chaotic systems, multimedia security, computer vision, and machine learning. He is a Senior Member of the International Society for Optical Engineering (SPIE). He was a recipient of the Third Price of Macau Natural Science Award in 2014. He is the Co-Chair of the Technical Committee on Cognitive Computing in the IEEE Systems, Man, and Cybernetics Society. He serves as an Associate Editor for *Neurocomputing*, *Journal of Visual Communication and Image Representation*, and *Signal Processing: Image Communication*.