

# Prompt-Based Test-Time Real Image Dehazing: A Novel Pipeline

Zixuan Chen<sup>1</sup>, Zewei He<sup>✉1,2</sup>, Ziqian Lu<sup>1</sup>,  
Xuecheng Sun<sup>1</sup>, and Zhe-Ming Lu<sup>1,2</sup>

<sup>1</sup> School of Aeronautics and Astronautics, Zhejiang University

<sup>2</sup> Huanjiang Laboratory

{zxchen, zeweihe, ziqianlu, xuechengsun, zheminglu}@zju.edu.cn

**Abstract.** Existing methods attempt to improve models’ generalization ability on real-world hazy images by exploring well-designed training schemes (*e.g.*, CycleGAN, prior loss). However, most of them need very complicated training procedures to achieve satisfactory results. For the first time, we present a novel pipeline called Prompt-based Test-Time Dehazing (PTTD) to help generate visually pleasing results of real-captured hazy images during the inference phase. We experimentally observe that given a dehazing model trained on synthetic data, fine-tuning the statistics (*i.e.*, mean and standard deviation) of encoding features is able to narrow the domain gap, boosting the performance of real image dehazing. Accordingly, we first apply a prompt generation module (PGM) to generate a visual prompt, which is the reference of appropriate statistical perturbations for mean and standard deviation. Then, we employ a feature adaptation module (FAM) into the existing dehazing models for adjusting the original statistics with the guidance of the generated prompt. PTTD is model-agnostic and can be equipped with various state-of-the-art dehazing models trained on synthetic hazy-clean pairs to tackle the real image dehazing task. Extensive experimental results demonstrate that our PTTD is effective, achieving superior performance against state-of-the-art dehazing methods in real-world scenarios. The code is available at <https://github.com/cecret3350/PTTD-Dehazing>.

**Keywords:** Image dehazing · Visual prompt · Feature adaptation

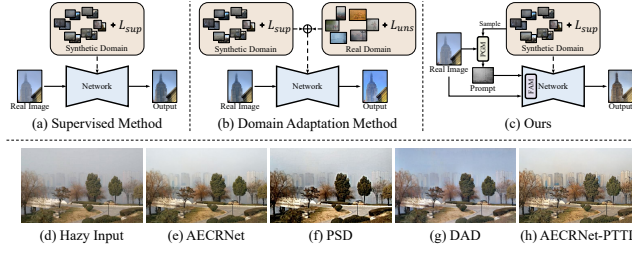
## 1 Introduction

Hazy images often suffer from low contrast, poor visibility, and color distortion [44], imposing a negative impact on the downstream high-level vision tasks, such as object detection, image classification, and semantic segmentation. According to the atmospheric scattering model (ASM) [36, 37], the hazing process is commonly formulated as:

$$I(x) = J(x)t(x) + A(1 - t(x)), \quad (1)$$

---

<sup>✉</sup> Corresponding author (zeweihe@zju.edu.cn).



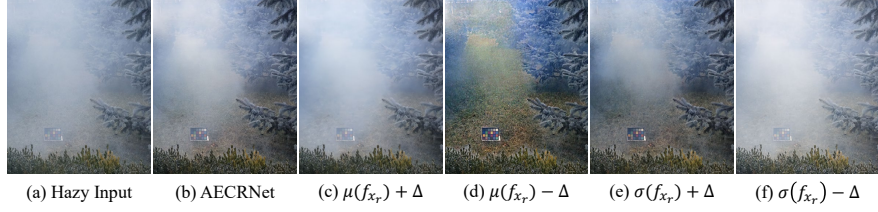
**Fig. 1:** Different frameworks and their results on real-world hazy images. (a) Directly apply model trained on synthetic data to real hazy images (*e.g.*, AECRNet [45]); (b) Use synthetic and real data together to train the model and then apply it to real hazy images (*e.g.*, PSD [10], DAD [41]); (c) Our proposed PTTD; (d) A real hazy image; and (e-h) Processing results of state-of-the-art (SOTA) methods (AECRNet [45], PSD [10], DAD [41]) and our proposed PTTD (by adopting pre-trained AECRNet [45]). It can be observed that AECRNet-PTTD achieves very promising results.

where  $I(x)$  is the observed hazy image and  $J(x)$  denotes the clean image of the same scene.  $A$  and  $t(x)$  are the global atmospheric light and the transmission map, respectively.

Image dehazing aims to recover the haze-free image from corresponding hazy input, which is a highly ill-posed problem. Early approaches tend to solve this challenge by introducing various priors, such as Dark Channel Prior (DCP) [21, 22], Non-Local Prior (NLP) [6], Color Attenuation Prior (CAP) [55], etc. However, real-world hazy images do not always satisfy the priors and artifacts may be introduced under this condition.

With the rise of deep learning, researchers have proposed a series of single image dehazing methods based on convolutional neural networks (CNNs). Most of them try to estimate the  $t(x)$  and  $A$  [8, 29, 40] in Eq. (1) or directly learn the latent haze-free image (or haze residual) [13, 24, 38, 45, 54]. The former then utilizes the estimated  $t(x)$  and  $A$  to derive the haze-free image via ASM.

Recently, existing deep learning based dehazing methods are devoted to improving performance by training sophisticated architectures on synthetic datasets [11, 54]. Though breakthrough progress has been made in the past decade, the processing results on real-captured hazy images are still unsatisfactory (as shown in Fig. 1 (e)), and it is mainly caused by the domain shift. How to bridge the gap between synthetic and real-world data is a hot research topic among the computer vision community. In order to improve the models' generalization ability on real-world hazy images, researchers start exploring CycleGAN based [41, 47] and prior loss based [10, 18] methods. However, either CycleGAN based or prior loss based methods demand very complex training procedures to guarantee the performance. Moreover, the former tends to produce results with artifacts, whereas the latter is negatively influenced by the inherent deficiencies of physical priors (Fig. 1 (f) and (g)).



**Fig. 2:** (a) is a real hazy input; (b) is the processing result of AECRNet [45], which is trained on synthetic data; (c) and (d) are the results by fine-tuning the mean  $\mu$ ; (e) and (f) are the results by fine-tuning the standard deviation  $\sigma$ . In this experiment, the perturbation  $\Delta$  is set to a small constant, *i.e.*,  $\Delta = 0.005$ .

We experimentally find an interesting phenomenon that some statistical indicators (*e.g.*, mean and standard deviation) may significantly affect the model’s output, as shown in Fig. 2. Taking pre-trained AECRNet [45] as an example, by fine-tuning the mean and standard deviation (only a small perturbation) of deep features extracted by the encoder, the predicted image reconstructed via the decoder is able to manipulate the haze distribution.

Considering this, we propose an effective remedy, called Prompt-based Test-Time Dehazing (PTTD), which attempts to avoid training process and generate visually pleasing results during the inference phase. Specifically, we first apply a prompt generation module (PGM) to create a visual prompt image, which is later employed to obtain the appropriate values of the statistics. Similar to style transfer [25], the visual prompt accepts a haze-free image as input and inherits the haze distribution from the real hazy image via the operation of image-level normalization (ILN). Then, we employ a feature adaptation module (FAM) into the encoder of a pre-trained model (PTTD is a plug-and-play pipeline, and can take various dehazing models as its backbone) for adjusting the statistics. The adjusting values are calculated by taking prompt and real hazy input into consideration, and the adjustment is realized by the operation of feature-level normalization (FLN). Both of ILN and FLN are derived from adaIN [25], which is used to align statistics of the content with those of the style in feature space. We hope the dehazing results of the real hazy images can better fit human’s perception. To achieve this ultimate goal, the ILN is further revised to avoid color distortion and the FLN adapts features with some effective and mandatory restrictions. Figure 3 shows the overall architecture of proposed PTTD.

As shown in Fig. 1, we apply the proposed PTTD to AECRNet [45] by slightly revising original structure (add PGM, FAM). We denote it as AECRNet-PTTD. The AECRNet-PTTD recovers clearer textures, more accurate color, less artifacts than the alternatives, demonstrating the superiority of proposed PTTD. Our main contributions can be summarized as follows:

- A novel dehazing pipeline called Prompt-based Test-Time Dehazing (PTTD) is proposed, which can adapt models pre-trained on synthetic data to real-world images during test-time. To the best of our knowledge, this is the first

attempt to tackle the real image dehazing task with **NO** training process (including training and test-time training).

- PTTD aims to narrow the domain gap between synthetic and real by fine-tuning the statistics of encoding features extracted by models pre-trained on synthetic data. PTTD is the first method to explore the correlation between feature statistics and the dehazing effect without modifying the CNN architecture, envisioning an innovative research direction.
- We design a Prompt Generation Module (PGM) to obtain a visual prompt via color balanced image-level normalization (CBILN). The prompt provides suitable perturbations for the Feature Adaptation Module (FAM) to adjust the statistics via feature-level normalization (FLN). PGM and FAM constitute our PTTD pipeline, which is simple (no training process), flexible (model-agnostic) and effective (SOTA performance) for real image dehazing.

## 2 Related Work

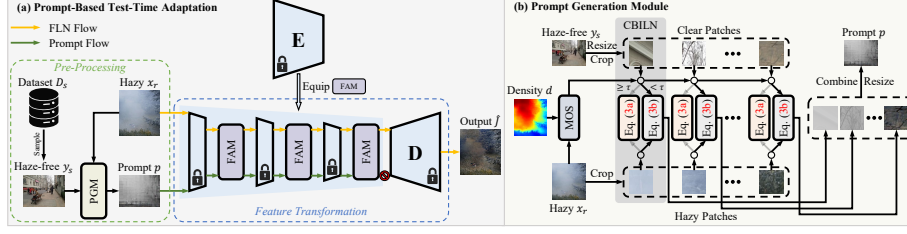
### 2.1 Single Image Dehazing

Pioneers of image dehazing [6, 15, 16, 22, 44, 55] estimate the key components of ASM (*i.e.*, transmission map and atmospheric light) by leveraging handcraft priors induced from statistics of haze-free images, and then perform image dehazing. Although prior-based methods achieve impressive dehazing performance, their effectiveness is constrained to hazy scenes which happen to satisfy the assumption they made.

With the rapid development of deep learning, learning-based methods have dominated the image dehazing domain. Early CNN-based methods [8, 29, 40, 53] utilize CNNs to estimate transmission map and atmospheric light. However, the inaccurate estimation of intermediate parameters will introduce the accumulated error. To avoid this, recent CNN-based and transformer-based methods [11, 13, 14, 20, 23, 24, 34, 38, 43, 45, 49, 54] discard the physical model and generate dehazing results or their residuals via the end-to-end paradigm. While learning-based methods demonstrate superior dehazing performance in synthetic domain, they usually suffer from performance degradation on real-world hazy scenes.

### 2.2 Real Image Dehazing

Recently, an increasing number of works have noticed the domain gap between synthetic domain and real domain, and tried to improve real-world dehazing performance. One category of these methods utilize real hazy images to reduce the domain gap [10, 17, 18, 27, 31–33, 41, 52]. DAD [41] employs CycleGAN for the translation between the synthetic domain and the real domain. PSD [10] fine-tunes pre-trained dehazing models on the real domain by establishing an unsupervised prior loss committee. However, due to the absence of paired data in real-world scenarios, complicate training strategies (*e.g.*, CycleGAN, prior



**Fig. 3:** (a) The overall architecture of our Prompt-based Test-Time Dehazing (PTTD); (b) The proposed prompt generation module (PGM).

loss) and their corresponding limitations are introduced. The other category aims to improve generalization performance of dehazing by exclusively utilizing synthetic images [42, 46, 47]. For example, Wu *et al.* [46] propose a new degradation pipeline to better align the synthetic domain with the real domain. Despite these efforts, without the aid of real hazy images, domain shift remains [46]. In order to avoid the inherent limitation of complicate training strategies and leverage the advantages of large-scale synthetic dataset, we propose a novel pipeline to perform feature adaptation on pre-trained models during only test-time. Traditional test-time adaptation (TTA) methods [19, 33, 48, 52] in low-level vision adapt models to unlabeled data by updating model parameters. Unlike these works, our method exploits the correlation between intermediate features and the model output, thus eliminating the need for parameter updating.

### 3 Methodology

Let  $D_S = \{x_s^i, y_s^i\}_{i=1}^{N_s}$  denote synthetic hazy-clean pairs and  $D_R = \{x_r^i\}_{i=1}^{N_r}$  indicate a collection of real-world hazy images, where  $N_s$  and  $N_r$  denote the number of synthetic and real-world samples, respectively. A dehazing network  $\mathcal{N}$  trained on  $D_S$  often fails to generalize well to real data  $D_R$ , which can be attributed to the domain gap between  $D_S$  and  $D_R$ . Our PTTD aims to bridge the gap during the inference phase.

#### 3.1 Observation and Motivation

Comparing a hazy image with a haze-free image, the difference is mainly observed in terms of luminance and contrast. We argue that these two image-level statistics are correlated closely with feature statistics, *i.e.*, mean  $\mu$  and standard deviation  $\sigma$ . Adjusting the values of  $\mu$  and  $\sigma$  in feature space can in turn affect the luminance and contrast of the reconstructed image, achieving haze removal effect. To validate this hypothesis, we firstly employ a dehazing model  $\mathcal{N}$  which is pre-trained on only synthetic hazy-clean pairs (without loss of generality, AECRNet [45] is adopted here) to perform dehazing on a real hazy input  $x_r$ . As shown in Fig. 2 (b), AECRNet achieves unsatisfactory dehazing result.

Then, we apply a subtle perturbation  $\Delta$  on  $\mu$  and  $\sigma$  of the extracted features through AECRNet’s encoder. Let  $f_{x_r}$  denotes the features, we adjust  $\sigma(f_{x_r})$  to  $\sigma(f_{x_r}) + \Delta / \sigma(f_{x_r}) - \Delta$  via a linear transformation. As shown in Fig. 2 (e) and (f),  $\sigma(f_{x_r}) + \Delta$  strengthens  $\mathcal{N}$ ’s dehazing effect, and  $\sigma(f_{x_r}) - \Delta$  makes the reverse optimization. Similar results can be observed in terms of  $\mu(f_{x_r})$ .

Motivated by this, we argue a possible solution which can narrow the domain gap is to adjust the statistics of  $f_{x_r}$  (with suitable perturbations) to make the features align with the synthetic domain. So the weight-fixed decoder of  $\mathcal{N}$  can reconstruct haze-free predictions.

### 3.2 Pipeline Overview

As illustrated in Fig. 3 (a), PTTD contains two major parts: pre-processing part and feature transformation part. In the pre-processing part, we employ a prompt generation module (PGM) to produce a visual prompt  $p$ , whose extracted features  $f_p$  are the target domain for guiding the adaption of  $f_{x_r}$ . As the key of our PTTD, feature transformation part can adopt most dehazing models as the backbone. In the feature transformation part, the feature adaption module (FAM) is employed into the encoder of  $\mathcal{N}$  to conduct the adaptation process for narrowing domain gap. We denote the new model with FAM as  $\mathcal{N}^\dagger$ .  $\mathcal{N}^\dagger$  takes  $x_r$  and  $p$  as inputs and outputs the reconstructed result  $\hat{J}$ .

### 3.3 Prompt Generation Module

Prompt generation module (PGM) is the key component of the pre-processing part. The goal of PGM is to generate a visual prompt  $p$ , which can guide the adjustment of statistics inside the FAM (offer appropriate values of mean  $\mu$  and standard deviation  $\sigma$ ). To fulfill this objective, the prompt  $p$  must meet two criteria: (1) it should be similar to the synthetic domain, so the encoder part of a certain pre-trained model can correctly extract the features. (2) it must have similar haze distribution with  $x_r$ , so the adaption could be stable to avoid the collapse of the whole pipeline. These principles incline us to sample a haze-free image  $y_s$  from  $D_S$ , and then synthesize similar haze as  $x_r$  on  $y_s$ . For simplicity, the  $y_s$  is randomly chosen from the clean images of synthetic hazy-clean pairs.

Adaptive instance normalization (adaIN) [25] tries to enable arbitrary style transfer by aligning the channel-wise mean and variance of the content features with those of the style features. Accordingly, we adopt adaIN by taking  $x_r$  and  $y_s$  as the style input and content input, respectively. We denote it as image-level normalization (ILN) by aligning  $\mu$  and  $\sigma$  of  $y_s$  with those of  $x_r$ :

$$p = \text{ILN}(y_s, x_r) = \sigma_c(x_r) \frac{y_s - \mu_c(y_s)}{\sigma_c(y_s)} + \mu_c(x_r), \quad (2)$$

where  $\mu_c(\cdot) \in \mathbb{R}^C$  and  $\sigma_c(\cdot) \in \mathbb{R}^C$  are mean and standard deviation, computed across the spatial dimensions for R,G,B channels.



**Fig. 4:** (a) A real hazy image  $x_r$ ; (b) A haze-free image  $y_s$ ; (c) Prompt via direct ILN (adaIN); (d) Prompt via ILN with partition; (e) Prompt via CBILN with partition.

As shown in Fig. 4 (c), simply adopting the adaIN scheme would fail to transfer the haze distribution from  $x_r$  to  $y_s$ , since light-haze and dense-haze regions may compromise each other and generate average results. We argue that  $\mu$  and  $\sigma$  are global statistics, and the haze distribution (*e.g.*, density) is inconsistent across the whole image. Therefore, we uniformly crop  $x_r$  and  $y_s$  into small patches without overlap<sup>3</sup> and apply ILN on patches. With this partition strategy, our ILN can effectively transfer the haze distribution from  $x_r$  to  $y_s$ . Fig. 4 (d) shows the generated prompt, sharing very similar haze with  $x_r$ .

The criterion (2) is definitely satisfied under this setting. We surprisingly find that criterion (1) is also met. We reveal that ILN with partition strategy can be viewed as the process of haze simulation based on ASM. By comparing Eq. (1) and Eq. (2), if the transmission map  $t(x)$  is assumed to be a constant, both equations can be regarded as the linear transformation. With a sufficient number of partition regions, the transmission map  $t(x)$  of a certain patch can be approximately described as a fixed value, which satisfies the key assumption. In this situation, prompt generated via ILN with partition shares the same characteristics with the synthetic domain, since the prompt strictly follows the ASM.

Inspired by [32], we also take intra-domain gap (*i.e.*, varicolored hazy scenes) into consideration to calibrate the color distortion. As illustrated in Fig. 4, the real hazy image is often varicolored (*i.e.*, bluish here) and the generated prompt via ILN has similar color distribution. We argue that the prompt must be color balanced to avoid introducing color distortion in the predicted haze-free image. There have been many methods developed to provide color constancy, for example, the widely used gray world assumption [7, 39] hypothesizes that the average scene captured in an image is gray. We accordingly modify the ILN into color balanced ILN (CBILN) by replacing  $\mu_c(x_r)$  and  $\sigma_c(x_r)$  with corresponding mean values among R,G,B channels, *i.e.*,  $\bar{\mu} = \text{mean}(\mu_c(x_r))$  and  $\bar{\sigma} = \text{mean}(\sigma_c(x_r))$ .

$$p = \text{CBILN}(y_s, x_r) = \begin{cases} \text{ILN}(y_s, x_r), & \text{MOS}(x_r) \geq \tau \\ \bar{\sigma} \frac{y_s - \mu_c(y_s)}{\sigma_c(y_s)} + \bar{\mu}, & \text{otherwise} \end{cases} \quad (3a)$$

$$(3b)$$

Following [26], we utilize the measure of the spread of hue space (MOS) as our criteria to indicate whether a hazy region (which is determined by haze

<sup>3</sup> In practice, the resolutions of  $x_r$  and  $y_s$  may be different, we first resize  $y_s$  to the resolution of  $x_r$  to improve the flexibility.

**Algorithm 1:** Feature-level normalization - FLN

---

**Input:**  $f_p, f_{x_r}$   
**Output:**  $\hat{f}_{x_r}$ ; //  $\hat{f}_{x_r} = \text{FLN}(f_{x_r}, f_p)$

1 compute  $\mu_c(f_p), \mu_c(f_{x_r}), \sigma_c(f_p), \sigma_c(f_{x_r})$  ;  
   */\* mean adaptation \*/*  
 2 **if**  $\mu_c(f_p) \cdot \mu_c(f_{x_r}) > 0$  **then**  $\mu_c(f_p, f_{x_r}) \leftarrow \min(\mu_c(f_p), \mu_c(f_{x_r}))$  ;  
 3 **else**  $\mu_c(f_p, f_{x_r}) \leftarrow \mu_c(f_{x_r})$  ;  
   */\* std adaptation \*/*  
 4 **if**  $\frac{\sigma_c(f_p) - \text{mean}(\sigma_c(f_{x_r}))}{\text{std}(\sigma_c(f_{x_r}))} < \alpha$  **then**  $\sigma_c(f_p, f_{x_r}) \leftarrow \max(\sigma_c(f_p), \sigma_c(f_{x_r}))$  ;  
 5 **else**  $\sigma_c(f_p, f_{x_r}) \leftarrow \sigma_c(f_{x_r})$  ;  
 6  $\hat{f}_{x_r} \leftarrow \sigma_c(f_p, f_{x_r}) \frac{f_{x_r} - \mu_c(f_{x_r})}{\sigma_c(f_{x_r})} + \mu_c(f_p, f_{x_r})$

---

density  $d$  computed via DCP) is varicolored or not. Smaller MOS value means this image tends to be varicolored and CBILN should be adopted, and vice versa. As illustrated in Fig. 3 (b), we utilize a threshold  $\tau = 0.005$  for choosing Eq. (3a) or Eq. (3b) to generate the prompt.

### 3.4 Feature Adaptation Module

As stated before, our PTTD can be easily applied on dehazing models pre-trained on synthetic data. Typically, we adopt the encoder-decoder-like architecture which is quite prevalent among the pre-trained models [11, 13, 45]. In the feature transformation part, we keep the decoder part unchanged and modify the encoder part by inserting the feature adaptation module (FAM) after the basic blocks in every level (different spatial sizes indicate different levels). Taking the generated prompt  $p$  and a real hazy image  $x_r$  as inputs, the encoder part of a certain pre-trained dehazing model  $\mathcal{N}$  extracts corresponding features of different levels, denoted as  $f_p^l$  and  $f_{x_r}^l$ . We omit the level indicator for simplification, and one-level case is presented in this section. It is not laborious to extend to multiple levels. According to the key observation in Fig. 2, a naive idea is to align the channel-wise statistics of  $f_{x_r}$  with those of  $f_p$  (we denote this as feature-level normalization - FLN), meanwhile make sure the  $\sigma_c(f_p)$  is larger than  $\sigma_c(x_r)$  and the  $\mu_c^+(f_p)/\mu_c^-(f_p)$  is smaller than  $\mu_c^+(x_r)/\mu_c^-(x_r)$ . However, since the visual prompt  $p$  is derived from a selected image, the values of  $\sigma_c(f_p)$  and  $\mu_c(f_p)$  may fluctuate. We need to add some restrictions on the statistical calculation of  $f_p$ , and the FLN can be formulated as:

$$\text{FLN}(f_{x_r}, f_p) = \sigma_c(f_p, f_{x_r}) \frac{f_{x_r} - \mu_c(f_{x_r})}{\sigma_c(f_{x_r})} + \mu_c(f_p, f_{x_r}), \quad (4)$$

where  $\mu_c(f_p, f_{x_r})$  and  $\sigma_c(f_p, f_{x_r})$  calculate the statistics of the target domain, providing appropriate perturbations for haze removal. The calculations are described by the pseudo code (see in Algorithm 1).

In step 2 and 3, the computed  $\mu$  is less than or equal to  $\mu_c(f_{x_r})$ . The  $\mu_c(f_p) \cdot \mu_c(f_{x_r}) \leq 0$  situation means these two items belong to different quadrants, and



in this situation, we abort the feature adaptation from  $f_{x_r}$  to  $f_p$  to achieve more natural dehazing results.

In step 4 and 5, the computed  $\sigma$  is larger than or equal to  $\sigma_c(f_{x_r})$ . Although applying a positive perturbation  $+\Delta$  on  $\sigma$  can yield positive gains for hazy regions, it may be overly aggressive if there is a significant magnitude of difference between the  $\sigma_c(f_p)$  and  $\text{mean}(\sigma_c(f_{x_r}))$ . The condition  $\frac{\sigma_c(f_p) - \text{mean}(\sigma_c(f_{x_r}))}{\text{std}(\sigma_c(f_{x_r}))} < \alpha$  avoids significant fluctuation, restricting the distance from the mean of  $\sigma_c(f_{x_r})$ . In our implementation, the hyper-parameter  $\alpha$  is set to 2.

## 4 Experiments

### 4.1 Experimental Configuration

**Datasets.** Our proposed PTTD pipeline is a test-time approach and model-agnostic, which can be regarded as a basic module to be plugged into most state-of-the-art dehazing models. Therefore, we don’t need training data, and PTTD is free of training. Moreover, we adopt total six real-world datasets for evaluation. Four datasets with ground truths (O-HAZE [3], I-HAZE [5], NH-HAZE [2] and Dense-Haze [1]), and two datasets without ground truths (RTTS [30] and Fattal’s [16]). O-HAZE, I-HAZE, NH-HAZE and Dense-Haze consist of 45, 35, 55 and 55 pairs of real hazy and corresponding haze-free images captured in various scenes. RTTS contains over 4000 real hazy images with diverse scenes, and haze-free images are not provided. In addition, Fattal’s dataset with 31 classical real hazy images is also included for evaluation.

**Implementation Details.** We select three state-of-the-art models pre-trained on synthetic data as the backbone to comprehensively show the flexibility of our PTTD pipeline, including a CNN-based (*i.e.*, AECRNet [45]), a transformer-based (*i.e.*, Dehazeformer [43]), and a general image restoration model (*i.e.*, NAFNet [9]). For fair comparisons, we re-train these three models on the same data built by Wu *et al.* [46]<sup>4</sup>, and apply the proposed PTTD to narrow the domain gap from synthetic to real (**We also perform experiments on their original models in our supplementary materials**). Typically, we traverse the haze-free images in Wu’s dataset [46] (take it as  $y_s$ ) and calculate the PSNR values (of AECRNet-PTTD model) on exclusive NH-HAZE2 dataset [4] to search the top-performing  $y_s$ <sup>5</sup>. The chosen  $y_s$  is illustrated in Fig. 4 (b). In PGM, we cropped square patches from resized  $y_s$  by setting side length to  $\frac{W}{10}$ , where  $W$  denotes the width of the real hazy input  $x_r$ . Another hyper-parameter is the  $\alpha$  in Algorithm 1, and we set  $\alpha = 2$ .

<sup>4</sup> The models re-trained on Wu’s dataset [46] outperform the original.

<sup>5</sup> NH-HAZE2 dataset is adopted in the famous New Trends in Image Restoration and Enhancement (NTIRE) competition [4], and has no intersection with the testing datasets (including NH-HAZE).



**Fig. 5:** Dehazing results of various methods on O-HAZE. We choose AECRNet-PTTD to compare with SOTA dehazing methods. Please zoom in on screen for a better view.

**Table 1:** Benchmark results of various dehazing methods on O-HAZE, I-HAZE, NH-HAZE, and Dense-Haze datasets. **Bold** numbers indicate that the proposed PTTD profile can successfully help boost the performance.

Method	O-HAZE		I-HAZE		NH-HAZE		Dense-Haze	
	PSNR $\uparrow$	SSIM $\uparrow$	PSNR $\uparrow$	SSIM $\uparrow$	PSNR $\uparrow$	SSIM $\uparrow$	PSNR $\uparrow$	SSIM $\uparrow$
(TIP'20) DeepDCP	16.92	0.6789	14.92	0.7171	13.04	0.4603	11.37	0.4505
(IJCV'21) YOLY	15.83	0.6640	15.21	0.6870	12.37	0.4561	11.60	0.4534
(CVPR'21) Zero-Restore	16.65	0.7536	16.56	0.7909	11.29	0.5151	12.29	0.4411
(CVPR'20) DAD	18.36	0.7484	18.02	0.7982	14.34	0.5564	13.51	0.4627
(CVPR'21) PSD	11.66	0.6831	13.79	0.7379	10.62	0.5246	9.74	0.4311
(CVPR'22) D4	16.96	0.7229	15.64	0.7294	12.67	0.5043	11.50	0.4500
(CVPR'23) RIDCP	16.52	0.7154	16.88	0.7794	12.32	0.5341	9.853	0.4525
(TIP'23) Dehazeformer	15.23	0.7437	17.42	0.8184	11.78	0.5423	9.25	0.4425
(Ours) Dehazeformer-PTTD	<b>17.84</b> <sup>+2.61</sup>	<b>0.7925</b> <sup>+0.0488</sup>	<b>17.98</b> <sup>+0.56</sup>	<b>0.8186</b> <sup>+0.0002</sup>	<b>13.26</b> <sup>+1.48</sup>	<b>0.5807</b> <sup>+0.0384</sup>	<b>11.99</b> <sup>+2.74</sup>	<b>0.4681</b> <sup>+0.0256</sup>
(ECCV'22) NAFNet	18.16	0.7783	17.03	0.8095	13.17	0.5367	10.45	0.4514
(Ours) NAFNet-PTTD	<b>19.63</b> <sup>+1.47</sup>	<b>0.8094</b> <sup>+0.0311</sup>	<b>17.94</b> <sup>+0.81</sup>	<b>0.8280</b> <sup>+0.0185</sup>	<b>14.08</b> <sup>+0.91</sup>	<b>0.5765</b> <sup>+0.0398</sup>	<b>13.23</b> <sup>+2.78</sup>	<b>0.4787</b> <sup>+0.0273</sup>
(CVPR'21) AECRNet	17.24	0.7640	16.72	0.8098	12.65	0.5326	8.59	0.4325
(Ours) AECRNet-PTTD	<b>20.23</b> <sup>+2.99</sup>	<b>0.8145</b> <sup>+0.0505</sup>	<b>18.47</b> <sup>+1.75</sup>	<b>0.8198</b> <sup>+0.0100</sup>	<b>14.54</b> <sup>+1.89</sup>	<b>0.5932</b> <sup>+0.0606</sup>	<b>13.67</b> <sup>+5.08</sup>	<b>0.4440</b> <sup>+0.0115</sup>

## 4.2 Experiments on Real-captured Datasets with Labels

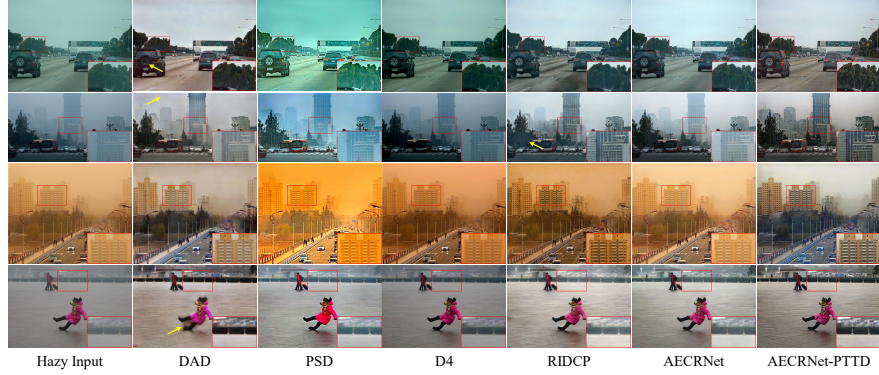
We first employ O-HAZE [3], I-HAZE [5], NH-HAZE [2] and Dense-Haze [1] datasets to evaluate our PTTD profile. With the labels (*i.e.*, haze-free images), reference-based metrics PSNR and SSIM are utilized to measure the performance. The quantitative comparisons on these four datasets are summarized in Tab. 1. The model with PTTD is denoted with suffix ‘-PTTD’. We observe that by employing PTTD, robust improvements can be achieved on selected models, proving the effectiveness of this novel pipeline.

In addition, we also compare the models with ‘-PTTD’ suffix with some recent real image dehazing methods (DAD [41], PSD [10], D4 [47], and RIDCP [46]), an unsupervised method (DeepDCP [18]), and two zero-shot methods (YOLY [31], Zero-Restore [27]). AECRNet-PTTD achieves state-of-the-art performance on O-HAZE, I-HAZE and NH-HAZE. Specifically, compared with DAD [41], AECRNet-PTTD achieves 1.87 dB and 0.0661 gains in terms of PSNR and SSIM on O-HAZE. Note that, the same  $y_s$  is utilized for all models and for all datasets. We believe better performance can be achieved by searching optimal  $y_s$  for each model and each dataset.

We provide some qualitative comparisons of various methods on O-HAZE in Fig. 5. Note that, there is still a significant amount of haze, remaining in the predicted images of {PSD, Zero-Restore, D4, RIDCP, AECRNet}. DAD removes

**Table 2:** Quantitative comparisons of various dehazing methods on RTTS dataset and Fattal’s dataset. **Bold** numbers indicate the proposed PTTD profile can successfully boost the performance.

Method	RTTS				Fattal’s dataset			
	FADE↓	BRISQUE↓	PAQ2PIQ↑	MUSIQ↑	FADE↓	BRISQUE↓	PAQ2PIQ↑	MUSIQ↑
Hazy Input	2.576	37.01	66.05	53.77	1.061	21.08	71.54	63.25
(CVPR’20) DAD	1.131	32.93	66.79	49.88	0.4838	29.64	71.56	58.64
(CVPR’21) PSD	1.044	22.22	70.43	52.80	0.4161	23.61	76.02	63.04
(CVPR’22) D4	1.406	34.52	66.84	53.57	0.4109	20.33	73.13	63.27
(CVPR’23) RIDCP	0.9180	21.38	70.82	59.38	0.4083	20.05	74.64	66.88
(TIP’23) Dehazeformer	1.047	21.62	69.90	58.46	0.4399	21.70	74.72	67.15
(Ours) Dehazeformer-PTTD	<b>0.7905</b> <sup>-0.2565</sup>	<b>17.34</b> <sup>-4.28</sup>	<b>71.45</b> <sup>+1.55</sup>	<b>61.04</b> <sup>+2.68</sup>	<b>0.4164</b> <sup>-0.0235</sup>	<b>19.68</b> <sup>-2.02</sup>	<b>75.70</b> <sup>+0.98</sup>	<b>69.52</b> <sup>+2.37</sup>
(ECCV’22) NAFNet	1.121	26.25	70.08	58.89	0.4183	19.80	74.12	65.91
(Ours) NAFNet-PTTD	<b>0.8267</b> <sup>-0.2943</sup>	<b>22.72</b> <sup>-3.53</sup>	<b>70.97</b> <sup>+0.89</sup>	<b>59.79</b> <sup>+0.90</sup>	0.4289 <sup>+0.0106</sup>	<b>18.75</b> <sup>-1.05</sup>	<b>74.65</b> <sup>+0.53</sup>	<b>66.98</b> <sup>+1.07</sup>
(CVPR’21) AECRNet	1.285	23.97	70.07	58.30	0.4319	21.44	74.41	66.83
(Ours) AECRNet-PTTD	<b>0.7120</b> <sup>-0.5740</sup>	<b>16.63</b> <sup>-7.34</sup>	<b>72.04</b> <sup>+1.97</sup>	<b>62.11</b> <sup>+3.81</sup>	<b>0.3825</b> <sup>-0.0494</sup>	<b>19.31</b> <sup>-2.13</sup>	<b>75.01</b> <sup>+0.60</sup>	<b>67.69</b> <sup>+0.86</sup>



**Fig. 6:** Dehazing results of various methods on RTTS. We choose AECRNet-PTTD to compare with SOTA dehazing methods. Please zoom in on screen for a better view.

the haze successfully. However, the overall predictions tend to be warm-toned and some high-frequency information is also lost. Our proposed AECRNet-PTTD restores images with clearer details and less color distortion, which are closer to the ground-truths. **More experimental results can be found in the supplementary materials.**

### 4.3 Experiments on RTTS and Fattal’s

We then employ RTTS [30] and Fattal’s [16] datasets to evaluate our PTTD profile. Since there is no clean images in these two datasets, we embed some non-reference metrics (*e.g.*, FADE [12], BRISQUE [35], PAQ2PIQ [50], and MUSIQ [28]) to evaluate the quality of dehazing results without reference images [51].

The quantitative results are presented in Tab. 2. We also take the real image dehazing methods {DAD, PSD, D4, RIDCP} into the comparison. With PTTD profile, the selected models achieve robust performance improvements on RTTS. When it comes to Fattal’s dataset, only NAFNet-PTTD fails to boost the performance in terms of FADE. Note that, RTTS has relatively larger number of

images than Fattal’s (4332 *vs.* 31). The metrics measured on latter tend to be unstable and biased. AECRNet-PTTD achieves state-of-the-art performance on RTTS dataset in terms of all non-reference metrics. Similarly, the same  $y_s$  is utilized for all models and for both datasets.

Figure 6 shows the qualitative comparisons on RTTS. When it comes to varicolored scenes, all the comparative methods fail to eliminate color cast except DAD. However, DAD suffers from undesired artifacts, which may cause some distortions or color patches in certain regions. Only AECRNet-PTTD can restore the visibility and tackle the color shift simultaneously. In addition, we notice that our PTTD is capable of removing haze for faraway regions in the images, which is usually hard for previous methods. **More experimental results can be found in the supplementary materials.**

#### 4.4 Ablation Study

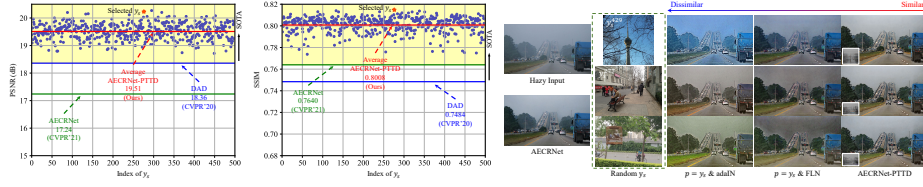
In this section, we perform ablation study to verify the effectiveness of (1) PGM, and (2) FAM. We utilize pre-trained AECRNet [45] as the backbone and measure the PSNR and SSIM values on O-HAZE dataset. Since the performance of our PTTD correlates to the haze-free image  $y_s$  sampled from  $D_S$ , we divide ablation experiments into two groups: (1) selected prompt: elaborately search through the clean images in Wu’s dataset [46] and evaluate the performance on NH-HAZE2 dataset [4] to

determine the top-performing  $y_s$ . The selected  $y_s$  is then utilized for all experiments. (2) random prompt: randomly select one clean image from Wu’s dataset [46] and the experimental results are averaged on all involved  $y_s$ .

**Discussion on the selection of  $y_s$ .** The selection of the haze-free image is technically unlimited. As shown in Tab. 3, though the model using the selected  $y_s$  performs better than that with random  $y_s$ , the latter still reaches state-of-the-art performance. As illustrated in Fig. 7, even the worst-performing  $y_s$  from the ‘Random Prompt’ achieves higher PSNR values than DAD and AECRNet. By grouping 5% best-performing and 5% worst-performing clean images, we find that images with rich textures (hierarchical depth information) will fit our PTTD. When it comes to certain images with large homogeneous region (few textures), the performance somewhat decreases. Different  $y_s$  would lead to different results. It provides the possibility that the performance can be further improved by searching optimal  $y_s$  for every single input. **More discussions can be found in the supplementary materials.** We also experimentally find that by choosing varied  $y_s$ , the final recovered results are very similar (see in Fig. 8), though the intermediate results may be different. There is no doubt

**Table 3:** Ablation study of PGM and FAM on O-HAZE. The metrics in ‘Random Prompt’ are averaged on all involved  $y_s$ .

	PGM			FAM		Selected Prompt		Random Prompt	
	ILN	CBILN	partition	adaIN	FLN	PSNR↑	SSIM↑	PSNR↑	SSIM↑
	-	-	-	-	-	17.24	0.7643	-	-
①	-	-	-	✓	-	16.70	0.7638	14.58	0.7272
②	✓	-	-	✓	-	16.90	0.7777	13.92	0.7211
③	✓	-	✓	✓	-	18.28	0.7862	16.11	0.7501
④	-	✓	✓	✓	-	19.46	0.8068	17.81	0.7826
⑤	-	✓	✓	-	✓	<b>20.23</b>	<b>0.8145</b>	<b>19.51</b>	<b>0.8008</b>

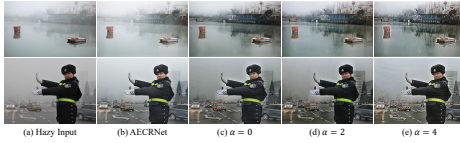


**Fig. 7:** We randomly choose a clean image in Wu’s dataset [46] as  $y_s$ , and plot the performance scatter graph of each  $y_s$  on O-HAZE  $y_s^i$  denotes the index number of Wu’s [46] dataset.

that the model with random  $y_s$  works relatively well, indicating the robustness of our proposed PTTD profile.

**Effectiveness of PGM.** The pre-trained AECRNet is regarded as the baseline (the first row of Tab. 3). PGM is designed to generate a visual prompt, and consists of image-level normalization (ILN), color balanced ILN (CBILN), and partition strategy. To analyze the role of each part, four variants are adopted: (1) PGM is not applied and  $p = y_s$ ; (2)  $p = \text{ILN}(y_s, x_r)$ ; (3)  $p = \text{ILN}(y_s, x_r) + \text{partition strategy}$ ; (4)  $p = \text{CBILN}(y_s, x_r) + \text{partition strategy}$ . The experimental results are summarized in Tab. 3. Directly taking  $y_s$  as  $p$  or simply adopting ILN to generate  $p$  can not help the haze removal and may cause performance drop (16.90 dB and 16.70 dB). Partition strategy can guarantee the haze distribution of  $p$  to be consistent with  $x_r$  and similar to synthetic domain, boosting the performance of ILN to 18.28 dB. By taking the intra-domain gap into consideration, CBILN pushes the performance forward to 19.46 dB.

**Effectiveness of FAM.** The function of FAM is to adapt the features  $f_{x_r}$  and make them match with the target domain (*i.e.*,  $f_p$ ). We freeze the design of PGM as CBILN with partition, and compare proposed FLN with adaIN. The main difference is based on the calculations of statistics of the target domain. The experimental results are summarized in the bottom two rows of Tab. 3. With more restrictions, FLN outperforms adaIN by 0.77 dB. In addition, we also conduct ablation study on the hyper-parameter  $\alpha$ , and the qualitative results are illustrated in Fig. 9. The value of  $\alpha$  and the haze removal effect are proportional to each other. Over-enhanced results may be produced when  $\alpha$  is greater than 2, and we can obtain under-enhanced results if  $\alpha$  is less than 2. We set  $\alpha$  to 2 to balance the removal effect and visual perception.



**Fig. 9:** Ablation study of different values of  $\alpha$ .

**Table 4:** Inference time comparisons of various dehazing methods.

Method	YOLY [31]	DAD [41]	PSD [10]	RIDCP [46]	AECRNet [45]	AECRNet-PTTD
# Param. (M)	39.99	54.59	33.11	28.72	2.611	2.611
Runtime (ms)	21852	19.16	29.96	173.2	23.48	38.03

#### 4.5 Inference Time

Table 4 demonstrates the inference time comparisons of various dehazing methods. The results are measured on color images with  $512 \times 512$  spatial resolution. YOLY [31] is time-consuming, since it requires training procedure for every single input. Due to the calculation procedures of PGM and FAM, PTTD profile inevitably increases the inference time of the adopted backbone. It is worth mentioning that the increase in inference time of our AECRNet-PTTD against the backbone (*i.e.*, AECRNet [45]) is acceptable (around 60%).

### 5 Limitation and Conclusion



**Fig. 10:** An imperfect case with artifacts for our PTTD. Our pipeline fails to eliminate the artifacts (*e.g.*, color patches on the building).

**Limitation.** While our PTTD is capable of generating clean images for distant regions, it may produce artifacts in these areas. As shown in Fig. 10, all the competitors fail to remove the dense haze covering the building, except our AECRNet-PTTD. However, some artifacts degrade the quality of the recovered result. Due to limited time, we leave the challenge here and hope future work can address it well.

**Conclusion.** In this paper, we propose a PTTD pipeline to adapt models pre-trained on synthetic data to real-world images during the inference phase. We reveal that fine-tuning the statistics (with suitable perturbations) of encoding features extracted by a certain pre-trained model can help narrow the domain gap between synthetic and real. Accordingly, a PGM is employed to generate a visual prompt by transferring the haze distribution from the real hazy image to the haze-free image (via the operation of ILN), providing the perturbations for subsequent features adaptation. Then we modify the encoder of a pre-trained model by adding FAMs to adjust the statistics of extracted features (via the operation of FLN). Extensive experimental results clearly show the superiority of our PTTD.

## Acknowledgment

This work was supported in part by National Natural Science Foundation of China under Grant No. 52305590, Zhejiang Provincial Natural Science Foundation of China under Grant No. LQ24F010004, China Postdoctoral Science Foundation funded project under Grant No. 2022M712792.

## References

1. Ancuti, C.O., Ancuti, C., Sbert, M., Timofte, R.: Dense-Haze: A Benchmark for Image Dehazing with Dense-Haze and Haze-Free Images. In: ICIP. pp. 1014–1018. IEEE (sep 2019). <https://doi.org/10.1109/ICIP.2019.8803046>, <https://ieeexplore.ieee.org/document/8803046/>
2. Ancuti, C.O., Ancuti, C., Timofte, R.: Nh-haze: An image dehazing benchmark with non-homogeneous hazy and haze-free images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (June 2020)
3. Ancuti, C.O., Ancuti, C., Timofte, R., De Vleeschouwer, C.: O-haze: A dehazing benchmark with real hazy and haze-free outdoor images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (June 2018)
4. Ancuti, C.O., Ancuti, C., Vasluianu, F.A., Timofte, R.: Ntire 2021 nonhomogeneous dehazing challenge report. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 627–646 (2021)
5. Ancuti, C., Ancuti, C.O., Timofte, R., De Vleeschouwer, C.: I-HAZE: A Dehazing Benchmark with Real Hazy and Haze-Free Indoor Images. In: CVPRW. pp. 620–631 (2018). [https://doi.org/10.1007/978-3-030-01449-0\\_52](https://doi.org/10.1007/978-3-030-01449-0_52), [http://link.springer.com/10.1007/978-3-030-01449-0\\_{\\_}52](http://link.springer.com/10.1007/978-3-030-01449-0_{_}52)
6. Berman, D., Treibitz, T., Avidan, S.: Non-local image dehazing. In: CVPR. pp. 1674–1682 (2016). <https://doi.org/10.1109/CVPR.2016.185>
7. Buchsbaum, G.: A spatial processor model for object colour perception. *Journal of the Franklin institute* **310**(1), 1–26 (1980)
8. Cai, B., Xu, X., Jia, K., Qing, C., Tao, D.: Dehazenet: An end-to-end system for single image haze removal. *IEEE Transactions on Image Processing* **25**(11), 5187–5198 (2016)
9. Chen, L., Chu, X., Zhang, X., Sun, J.: Simple baselines for image restoration. In: European Conference on Computer Vision. pp. 17–33. Springer (2022)
10. Chen, Z., Wang, Y., Yang, Y., Liu, D.: Psd: Principled synthetic-to-real dehazing guided by physical priors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7180–7189 (June 2021)
11. Chen, Z., He, Z., Lu, Z.M.: Dea-net: Single image dehazing based on detail-enhanced convolution and content-guided attention. *arXiv preprint arXiv:2301.04805* (2023)
12. Choi, L.K., You, J., Bovik, A.C.: Referenceless prediction of perceptual fog density and perceptual image defogging. *IEEE Transactions on Image Processing* **24**(11), 3888–3901 (2015). <https://doi.org/10.1109/TIP.2015.2456502>
13. Dong, H., Pan, J., Xiang, L., Hu, Z., Zhang, X., Wang, F., Yang, M.H.: Multi-scale boosted dehazing network with dense feature fusion. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2157–2167 (2020)

14. Dong, J., Pan, J.: Physics-based feature dehazing networks. In: European Conference on Computer Vision. pp. 188–204. Springer (2020)
15. Fattal, R.: Single image dehazing. *ACM transactions on graphics (TOG)* **27**(3), 1–9 (2008)
16. Fattal, R.: Dehazing using color-lines. *ACM transactions on graphics (TOG)* **34**(1), 1–14 (2014)
17. Gandelsman, Y., Shocher, A., Irani, M.: "double-dip": Unsupervised image decomposition via coupled deep-image-priors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
18. Golts, A., Freedman, D., Elad, M.: Unsupervised single image dehazing using dark channel prior loss. *IEEE Transactions on Image Processing* **29**, 2692–2701 (2020). <https://doi.org/10.1109/TIP.2019.2952032>
19. Gou, Y., Zhao, H., Li, B., Xiao, X., Peng, X.: Test-time degradation adaptation for open-set image restoration. In: Forty-first International Conference on Machine Learning (Jul 2024)
20. Guo, C.L., Yan, Q., Anwar, S., Cong, R., Ren, W., Li, C.: Image dehazing transformer with transmission-aware 3d position embedding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5812–5820 (2022)
21. He, K., Sun, J., Tang, X.: Single image haze removal using dark channel prior. In: CVPR. pp. 1956–1963 (2009)
22. He, K., Sun, J., Tang, X.: Single image haze removal using dark channel prior. *IEEE transactions on pattern analysis and machine intelligence* **33**(12), 2341–2353 (2010)
23. He, Z., Chen, Z., Lu, Z., Sun, X., Lu, Z.M.: Accurate and lightweight dehazing via multi-receptive-field non-local network and novel contrastive regularization. *arXiv preprint arXiv:2309.16494* (2023)
24. Hong, M., Liu, J., Li, C., Qu, Y.: Uncertainty-driven dehazing network. *Proceedings of the AAAI Conference on Artificial Intelligence* **36**(1), 906–913 (Jun 2022). <https://doi.org/10.1609/aaai.v36i1.19973>, <https://ojs.aaai.org/index.php/AAAI/article/view/19973>
25. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (Oct 2017)
26. Kanti Dhara, S., Roy, M., Sen, D., Kumar Biswas, P.: Color cast dependent image dehazing via adaptive airlight refinement and non-linear color balancing. *IEEE Transactions on Circuits and Systems for Video Technology* **31**(5), 2076–2081 (2021). <https://doi.org/10.1109/TCSVT.2020.3007850>
27. Kar, A., Dhara, S.K., Sen, D., Biswas, P.K.: Zero-shot single image restoration through controlled perturbation of koschmieder’s model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16205–16215 (2021)
28. Ke, J., Wang, Q., Wang, Y., Milanfar, P., Yang, F.: Musiq: Multi-scale image quality transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5148–5157 (2021)
29. Li, B., Peng, X., Wang, Z., Xu, J., Feng, D.: Aod-net: All-in-one dehazing network. In: Proceedings of the IEEE international conference on computer vision. pp. 4770–4778 (2017)
30. Li, B., Ren, W., Fu, D., Tao, D., Feng, D., Zeng, W., Wang, Z.: Benchmarking single-image dehazing and beyond. *IEEE Transactions on Image Processing* **28**(1), 492–505 (2018)



31. Li, B., Gou, Y., Gu, S., Liu, J.Z., Zhou, J.T., Peng, X.: You only look yourself: Unsupervised and untrained single image dehazing neural network. *International Journal of Computer Vision* **129**, 1754–1767 (2021)
32. Li, Y., Chang, Y., Gao, Y., Yu, C., Yan, L.: Physically disentangled intra- and inter-domain adaptation for varicolored haze removal. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 5841–5850 (June 2022)
33. Liu, H., Wu, Z., Li, L., Salehkalaibar, S., Chen, J., Wang, K.: Towards Multi-domain Single Image Dehazing via Test-time Training. In: *CVPR*. pp. 5831–5840 (2022)
34. Liu, X., Ma, Y., Shi, Z., Chen, J.: Griddehazenet: Attention-based multi-scale network for image dehazing. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 7314–7323 (2019)
35. Mittal, A., Moorthy, A.K., Bovik, A.C.: No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing* **21**(12), 4695–4708 (2012)
36. Narasimhan, S.G., Nayar, S.K.: Contrast restoration of weather degraded images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **25**(6), 713–724 (2003). <https://doi.org/10.1109/TPAMI.2003.1201821>
37. Narasimhan, S.G., Nayar, S.K.: Vision and the atmosphere. *International journal of computer vision* **48**, 233–254 (2002)
38. Qin, X., Wang, Z., Bai, Y., Xie, X., Jia, H.: Ffa-net: Feature fusion attention network for single image dehazing. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. vol. 34, pp. 11908–11915 (2020)
39. Reinhard, E., Adhikhmin, M., Gooch, B., Shirley, P.: Color transfer between images. *IEEE Computer graphics and applications* **21**(5), 34–41 (2001)
40. Ren, W., Liu, S., Zhang, H., Pan, J., Cao, X., Yang, M.H.: Single image dehazing via multi-scale convolutional neural networks. In: *European conference on computer vision*. pp. 154–169. Springer (2016)
41. Shao, Y., Li, L., Ren, W., Gao, C., Sang, N.: Domain adaptation for image dehazing. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2020)
42. Shyam, P., Yoon, K.J., Kim, K.S.: Towards domain invariant single image dehazing. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. vol. 35, pp. 9657–9665 (2021)
43. Song, Y., He, Z., Qian, H., Du, X.: Vision transformers for single image dehazing. *IEEE Transactions on Image Processing* **32**, 1927–1941 (2023). <https://doi.org/10.1109/TIP.2023.3256763>
44. Tan, R.T.: Visibility in bad weather from a single image. In: *2008 IEEE conference on computer vision and pattern recognition*. pp. 1–8. IEEE (2008)
45. Wu, H., Qu, Y., Lin, S., Zhou, J., Qiao, R., Zhang, Z., Xie, Y., Ma, L.: Contrastive learning for compact single image dehazing. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 10551–10560 (2021)
46. Wu, R.Q., Duan, Z.P., Guo, C.L., Chai, Z., Li, C.: Ridep: Revitalizing real image dehazing via high-quality codebook priors. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 22282–22291 (2023)
47. Yang, Y., Wang, C., Liu, R., Zhang, L., Guo, X., Tao, D.: Self-augmented unpaired image dehazing via density and depth decomposition. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 2037–2046 (2022)

48. Yang, Y., Wu, H., Aviles-Rivero, A.I., Zhang, Y., Qin, J., Zhu, L.: Genuine knowledge from practice: Diffusion test-time adaptation for video adverse weather removal. arXiv preprint arXiv:2403.07684 (2024)
49. Ye, T., Jiang, M., Zhang, Y., Chen, L., Chen, E., Chen, P., Lu, Z.: Perceiving and modeling density is all you need for image dehazing. In: European Conference on Computer Vision. pp. 130–145 (2022), <http://arxiv.org/abs/2111.09733>
50. Ying, Z., Niu, H., Gupta, P., Mahajan, D., Ghadiyaram, D., Bovik, A.: From patches to pictures (paq-2-piq): Mapping the perceptual space of picture quality. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3575–3585 (2020)
51. You, Z., Li, Z., Gu, J., Yin, Z., Xue, T., Dong, C.: Depicting beyond scores: Advancing image quality assessment through multi-modal language models. arXiv preprint arXiv:2312.08962 (2023)
52. Yu, H., Huang, J., Liu, Y., Zhu, Q., Zhou, M., Zhao, F.: Source-free domain adaptation for real-world image dehazing. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 6645–6654 (2022)
53. Zhang, H., Patel, V.M.: Densely connected pyramid dehazing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3194–3203 (2018)
54. Zheng, Y., Zhan, J., He, S., Dong, J., Du, Y.: Curricular contrastive regularization for physics-aware single image dehazing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5785–5794 (June 2023)
55. Zhu, Q., Mai, J., Shao, L.: A fast single image haze removal algorithm using color attenuation prior. IEEE transactions on image processing **24**(11), 3522–3533 (2015)