

Rethinking Real-world Image Deraining via An Unpaired Degradation-Conditioned Diffusion Model

Yiyang Shen^{1,3}, Mingqiang Wei¹, Yongzhen Wang¹, Xueyang Fu², and Jing Qin³

¹ Nanjing University of Aeronautics and Astronautics

² University of Science and Technology of China

³ The Hong Kong Polytechnic University

Abstract. Recent diffusion models have exhibited great potential in generative modeling tasks. Part of their success can be attributed to the ability of training stable on huge sets of paired synthetic data. However, adapting these models to real-world image deraining remains difficult for two aspects. First, collecting a large-scale paired real-world clean/rainy dataset is unavailable while regular conditional diffusion models heavily rely on paired data for training. Second, real-world rain usually reflects real-world scenarios with a variety of unknown rain degradation types, which poses a significant challenge for the generative modeling process. To meet these challenges, we propose **RainDiff**, the first real-world image deraining paradigm based on diffusion models, serving as a new standard bar for real-world image deraining. We address the first challenge by introducing a stable and non-adversarial unpaired cycle-consistent architecture that can be trained, end-to-end, with only unpaired data for supervision; and the second challenge by proposing a degradation-conditioned diffusion model that refines the desired output via a diffusive generative process conditioned by learned priors of multiple rain degradations. Extensive experiments confirm the superiority of our RainDiff over existing unpaired/semi-supervised methods and show its competitive advantages over several fully-supervised ones.

Keywords: RainDiff · Unpaired learning · Degradation-conditioned diffusion model

1 Introduction

Image deraining is an ill-posed problem. By learning from massive synthetic clean/rainy image pairs, the performance of learning-based techniques is substantially improved [11, 51], compared to the traditional prior wisdom, such as sparse coding [29], Gaussian Mixture Model [27], and low-rank representation [3]. Despite their successes, these fully supervised methods achieve sub-optimal performance on real-world rainy images, because of i) the existence of the domain gap between synthetic and real-world rainy images, and ii) the difficulty to collect large-scale real-world clean/rainy image pairs.

To alleviate the aforementioned problems, semi-supervised deraining techniques, leverage paired synthetic data for good initialization and unpaired real-world data for generalization [45, 52, 53]. But the transferability is still limited since the rain patterns

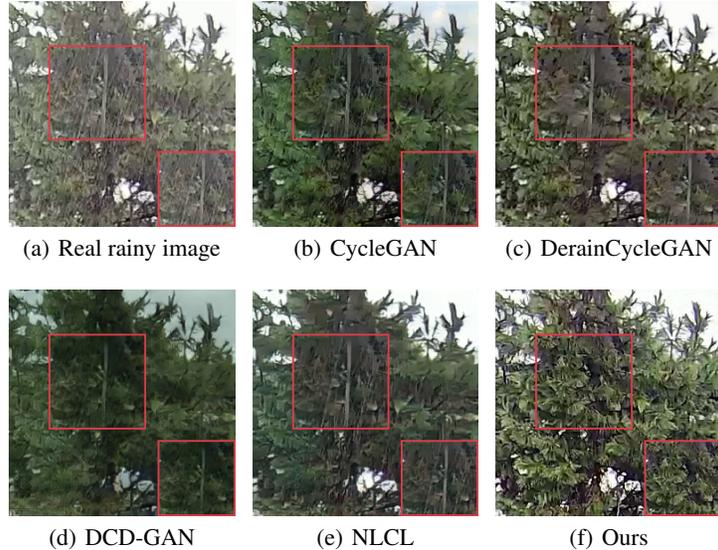


Fig. 1: Image deraining results on a real-world rainy image. From (a) to (f): (a) the real-world rainy image, the deraining results of (b) CycleGAN [59], (c) DerainCycleGAN [46], (d) DCD-GAN [5], (e) NLCL [54] and (f) our RainDiff. RainDiff generates both rain-free and perceptually more pleasing results.

of synthetic images are fixed while the rain patterns of real-world images are dynamically changing. Furthermore, the introduction of CycleGAN [59] makes Generative Adversarial Networks (GANs) the preferred model family for real-world image deraining tasks, as they avoid the need for paired data [5, 16, 22, 47, 58]. However, these unpaired deraining methods are known as being difficult to train due to their complex adversarial objectives. As a result, they are susceptible to a series of problems, such as premature convergence, model collapse, and optimization instability. Moreover, these methods, limited only to single rain degradation cases, may not be optimal for multiple degradations in real-world rainy conditions, leading to image degradation including the loss of image details, remnant rain, halo artifacts, and/or color distortion.

More recently, diffusion models [19, 37, 39] have garnered significant attention for their effectiveness in a wide range of generative modeling tasks, such as image inpainting [30], image restoration [31], and image super-resolution [35]. Compared to GANs, diffusion models offer a stable training process and exhibit greater efficacy in modeling the pixel distribution of images. However, no work to-date explores what will happen when unpaired learning meets diffusion models for real-world image deraining. We identify two major obstacles to their practical application in real-world image deraining. First, real-world rainy images lack corresponding clean images, which poses a challenge for existing diffusion models that typically prioritize synthetic degradation scenarios, where generating large-scale paired synthetic data is easier than for real-world examples. Models trained on paired synthetic data struggle to effectively handle

unpaired real-world data. Second, real-world rain presents diverse degradation types, including but not limited to rain streaks, raindrops, rainy haze, and a combination of them (also called the mixture of rain), which usually changes over time, especially in heavy rain conditions. Thus, models primarily designed for single-degradation image processing may not be well generalized to multiple real-world rain degradations.

We propose **RainDiff**, a new standard bar for real-world image deraining. RainDiff utilizes an effective unpaired cycle-consistent architecture with a degradation-conditioned diffusion model that achieves the desired deraining results under real-world rain scenarios with multiple rain degradations (see Fig. 1).

Why is the unpaired cycle-consistent architecture? The unpaired cycle-consistent architecture fully takes advantage of the circulatory architecture to overcome the challenge of training without paired data. Instead of popular adversarial learning architectures for unpaired data, the proposed method offers a stable and non-adversarial training process that better facilitates real-world rain removal.

Why is the degradation-conditioned diffusion model? The degradation-conditioned diffusion model adds additional degradation-conditioned controls to the diffusion model, making it possible to handle diverse rain degradations. Such degradation-conditioned controls precisely express the space differences among various types of rain degradations, that enable finer diffusive generative processes in real-world rain conditions.

Why does RainDiff serve as a new standard bar for real-world image deraining? RainDiff is an implementation of the idea of “solving real-world image deraining in an unpaired learning manner”, such as DerainCycleGAN [46] and DCD-GAN [5]. At the macro level, it is the first time to form an idea of applying the popular diffusion model to real-world image deraining; it solves several challenges encountered during the practical application process. At the micro level, RainDiff introduces stable training of unpaired real-world data, rather than weakly adversarial training; It also learns priors of multiple rain degradations to enhance its performance in real-world deraining.

Extensive experiments show that RainDiff outperforms existing unpaired/semi-supervised methods and achieves comparable performance against fully-supervised ones. Overall, our contributions are as follows:

- We propose a novel unpaired learning paradigm via a degradation-conditioned diffusion model, called RainDiff, to generate quality real-world deraining results.
- We propose an unpaired cycle-consistent architecture to provide a non-adversarial training process for unpaired data, where rain-related and clean-cue features can facilitate rain removal.
- We propose a degradation-conditioned diffusion model to provide powerful diffusive progress for image deraining, where the learned priors of multiple rain degradations boost the generalization of deraining for diverse real-world rain scenarios.

2 Related Work

Single Image Deraining. Images captured under complicated rainy scenarios inevitably suffer from the noticeable degradation of visual quality. This degradation causes detrimental impacts on many vision tasks, including segmentation [28], object detection

[49], and video surveillance [40]. Thus, it is indispensable to develop effective algorithms to recover quality rain-free images, which is referred to as *image deraining*. Early single image deraining methods employ hand-crafted priors, such as low-rank representation [3, 6], sparse coding [12, 43], and Gaussian mixture model [27], to restore rainy images. Recently, deep learning based methods have been substantiated to be effective in image deraining [9]. The pioneering work [9] introduces an end-to-end residual convolutional neural network (CNN) for simplifying the learning process. Network modules, such as dense block [25, 41], recursive block [8, 26] and dilated convolution [7], and structures, such as RNN [26, 33], GAN [5, 57] and multi-stream networks [51, 56], are validated to be effective in image deraining. Despite the promising deraining results on synthetic datasets, these methods trained on such synthetic images generalize poorly to real-world images, typically because of the obvious domain gap between synthetic and real-world rainy images. To solve this issue, several semi-supervised frameworks have been proposed [45, 52, 53] to achieve improved generalization performance. However, these supervised/semi-supervised methods still require paired data, which is challenging or even impossible to obtain in real-world rainy scenes. Motivated by the success of CycleGAN [59], a popular image-to-image translation architecture, recent works [5, 22, 46, 58] attempt to exploit the improved CycleGAN architecture and constrained transfer learning to jointly learn the rainy and rain-free image domains. However, these unpaired deraining methods heavily rely on complex adversarial objectives to develop their algorithms, which makes it difficult to achieve stable training. Additionally, they are limited in their ability to handle specific degradations.

Denoising Diffusion Probabilistic Models. Recently, denoising diffusion probabilistic models (DDPM) [19, 37, 39] have exhibited their powerful ability in various vision tasks, such as image super-resolution [35], text-to-image generation [13], and image segmentation [1]. More recently, Özdenizci et al. [31] propose patch-based denoising diffusion models to demonstrate how diffusion models can be used for image restoration. However, these architectures have not been applied to real-world image deraining. They still require paired data for training and are designed for a specific degradation only. This observation serves as a motivation for us to propose the first diffusion-based model for real-world image deraining.

3 RainDiff

We start by discussing the necessary background and notation on diffusion models in Sec. 3.1, and then we introduce our method in Sec. 3.2 and Sec. 3.3.

3.1 Denoising Diffusion Probabilistic Models

Denoising diffusion probabilistic models (DDPM) slowly corrupt the training data with Gaussian noise and learn to reverse this corruption as a generative model [19, 37, 39]. In the forward process, Gaussian noise is added sequentially onto an input image $x_0 \sim q(x_0)$ over T time steps according to the Markovian process:

$$q(x_{1:T}|x_0) = \prod_{t=1}^T q(x_t|x_{t-1}) \quad (1)$$

where $q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t I)$, $\mathcal{N}(\cdot)$ is Gaussian distribution, I is an identity covariance matrix with the same dimensions as the input image x_0 . An important property of this forward process is its ability to directly sample any x_t from x_0 :

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \epsilon_t\sqrt{1 - \bar{\alpha}_t} \quad (2)$$

where $\epsilon_t \sim \mathcal{N}(0, I)$, $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^t \alpha_i$. Similarly, reverse diffusion also adopts a Markov chain from x_T onto x_0 , albeit each step aims to gradually denoise the samples. Even though the reverse transition probability between x_t and x_{t-1} can be approximated as a Gaussian distribution under small β_t and large T :

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_\theta(x_t, t)) \quad (3)$$

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^T p_\theta(x_{t-1}|x_t) \quad (4)$$

where the reverse process is parameterized by a network to estimate $\mu_\theta(x_t, t)$ and $\sigma_\theta(x_t, t)$. Common parametrization focuses on $\mu_\theta(x_t, t)$ while ignoring $\sigma_\theta(x_t, t)$ [19]:

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(x_t, t) \right) \quad (5)$$

In this case, the network is used to estimate the added noise ϵ_t by minimizing the loss:

$$L_{err} = \mathbb{E}_{x_0, t, \epsilon_t \sim \mathcal{N}(0, I)} [\|\epsilon_t - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \epsilon_t\sqrt{1 - \bar{\alpha}_t}, t)\|^2] \quad (6)$$

During inference, reverse diffusion steps are performed starting from a random sample $x_T \sim \mathcal{N}(0, I)$. For each step $t \in \{T, \dots, 1\}$, μ is derived by Eq. 5 based on the estimated ϵ_θ , and x_{t-1} is sampled based on Eq. 3 as:

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(x_t, t) \right) + \sigma_t z \quad (7)$$

where $z \sim \mathcal{N}(0, I)$ resembling one step of sampling via Langevin dynamics [48].

To achieve high-quality image deraining, we need to learn a conditional reverse process $p_\theta(x_{0:T}|\tilde{x})$ without modifying the diffusion process $q(x_{1:T}|x_0)$ for x , where x_0 and \tilde{x} represent clean and rainy images, respectively. During the training phase, we sample $(x_0, \tilde{x}) \sim q(x_0, \tilde{x})$ from a paired data distribution and learn a conditional diffusion model. We input \tilde{x} to the reverse process as:

$$p_\theta(x_{0:T}|\tilde{x}) = p(x_T) \prod_{t=1}^T p_\theta(x_{t-1}|x_t, \tilde{x}) \quad (8)$$

The noise estimators in Eqs. 5-7 are also replaced by $\epsilon_\theta(x_t, \tilde{x}, t)$.

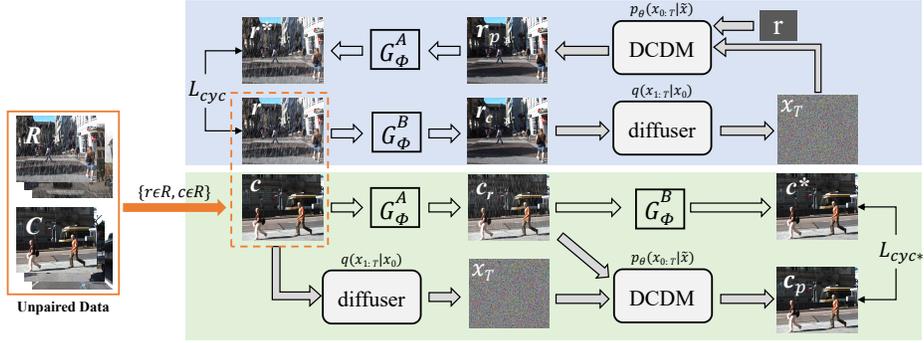


Fig. 2: The pipeline of RainDiff. It takes unpaired clean/rainy data $\{C, R\}$ as input and trains an unpaired cycle-consistent architecture with a degradation-conditioned diffusion model (DCDM). Once trained, the model can produce high-quality real-world image deraining results, without access to paired clean images. Please refer to Sec. 3 for details.

3.2 Unpaired Cycle-consistent Architecture

Despite the impressive performance of diffusion models in image-conditional data synthesis and restoration [31, 35], their implementation still requires paired clean/rainy images. We address this issue by designing a new unpaired cycle-consistent architecture without requiring adversarial training. It incorporates two cycle-consistent circuits with a degradation-conditioned diffusion model (DCDM) for unpaired training (see Fig. 2).

Given an unpaired rainy image $\{r \in R\}$ and clean image $\{c \in C\}$, we first employ two non-diffusive generators with parameters $\phi^{A,B}$ to obtain the initial translation:

$$c_r = G_\phi^A(c), r_c = G_\phi^B(r) \quad (9)$$

where c_r and r_c refer to the generated rainy image and rain-free image, respectively. Then, we use such clean/rainy image pairs to train DCDM for a conditional reverse process $p_\theta(x_{0:T} | \tilde{x})$, $\tilde{x} \in \{c_r, r\}$ without modifying the diffusion process $q(x_{1:T} | x_0)$, $x_0 \in \{c, r_c\}$. Finally, we adopt two cycle-consistency loss functions to constrain the unpaired training procedure of the above cycle-consistent circuits:

$$L_{cyc} = E_{r \sim P_{data}(r)} [\|r^* - r\|_1] \quad (10)$$

$$L_{cyc^*} = E_{c_p \sim P_{data}(c_p)} [\|c^* - c_p\|_1] \quad (11)$$

where $r^* = G_\phi^A(r_p)$ and $c^* = G_\phi^B(c_r)$, c_p and r_p refer to sampling results from the DCDM. Especially, we adopt U-Net [34] as our non-diffusive generators. The training phase of RainDiff is outlined in Algorithm 1.

Difference between Existing Circulatory Structures and Ours. In unpaired learning, the circulatory structures with cycle-consistency loss functions are commonly used for model training [5, 46, 59]. The differences between existing circulatory structures

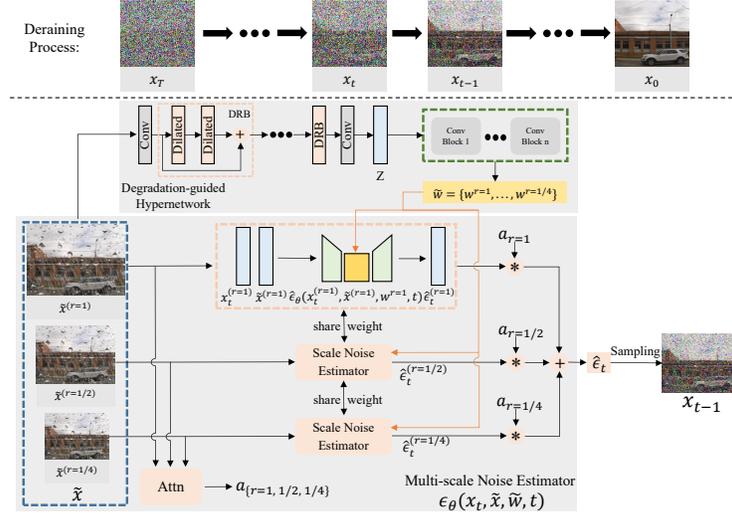


Fig. 3: The degradation-conditioned diffusion model.

and ours lie in two aspects: 1) RainDiff requires no discriminators for adversarial training and provides a reliable non-adversarial training process. 2) It involves a degradation-conditioned diffusion model, which is primarily designed for image deraining. Unlike generators in existing circulatory structures, our G_ϕ^A and G_ϕ^B are both only used in the training phase and remain uninvolved in the testing phase for image deraining.

3.3 Degradation-Conditioned Diffusion Model

In the real world, rain exhibits multiple degradations, e.g., rain streaks, raindrops, rainy haze, and the mixture of rain, which may change over time. It presents a considerable challenge in directly implementing diffusion models to perform real-world image deraining. Beyond existing fixed-degradation diffusion models, we design a degradation-conditioned diffusion model (DCDM) to address this problem (see Fig. 3).

Degradation-guided Hypernetwork. As shown in Fig. 3, our idea is to design a degradation-guided hypernetwork (DG-Hyper) that controls the reverse diffusion process using the learned degradation representation. Given a rainy image \tilde{x} , we first feed it into f_D to learn the latent degradation representation $Z = f_D(\tilde{x})$, where f_D represents two convolutional layers with five dilated residual blocks (DRBs) [55] that helps enlarge the receptive field to capture more comprehensive characteristic of different rain degradations. We set the dilation rates of these five DRBs as $\{1, 2, 4, 2, 1\}$.

As a single type of conditional control for the diffusion model, Z should possess adaptability to different rain degradations. It means that Z will dynamically change in accordance with the variations in types of rain degradations, even with the same image contents. Inspired by contrastive learning [15, 17, 24, 50], we leverage a contrastive loss function to make Z pull the representation with the same degradation Z^+ (called positive samples), and push apart the representation between negative samples Z^- with

different degradations. The proposed contrastive loss L_{cl} is reformulated as:

$$L_{cl} = \sum_{i=1}^n \omega_i \cdot \frac{\|\varphi_i(Z^+) - \varphi_i(Z)\|^2}{\|\varphi_i(Z^-) - \varphi_i(Z)\|^2}, \quad (12)$$

where $\varphi_i(\cdot)$, $i = 1, 2, \dots, n$, refers to extracting the i -th hidden features from the pre-trained VGG-16 network. We choose the 2-nd, 3-rd, and 5-th max-pooling layers. Similar to [50], ω_i are weight coefficients with $\omega_1 = 0.2$, $\omega_2 = 0.5$, and $\omega_3 = 1$. Note that Z^+ and Z^- are generated by feeding patches from \tilde{x} and other images, respectively.

Based on the learned latent degradation representation Z , we aim to enhance the adaptability of diffusion models to multiple rain degradations. To achieve it, we use a meta-learning technique, often referred to as hypernetwork [14], to control the diffusion model using Z . Especially, we input Z into a series of convolution blocks, consisting of a 1×1 convolutional layer with channel groups, to generate the part of kernels' parameters (weights) $\tilde{w} = \{w^{r=1}, \dots, w^{r=1/4}\}$ of the scale noise estimators in the multi-scale noise estimator (primary network) $\hat{\epsilon}_\theta(x_t^r, \tilde{x}^r, w^r, t)$. Notably, if U-Net [34] is employed as the scale noise estimator, only two up-convolution and down-convolution layers at the highest resolution are optimized using these kernel weights. In summary, the introduction of DG-Hyper that learns to modulate the weights of the noise estimator in order to accurately represent latent degradation enables enhanced generalization of the diffusion model across various rain degradations.

Multi-scale Noise Estimator. We observe that rain degradations, particularly rain streaks, exhibit a wide range of characteristics, including but not limited to varying directions, densities, and sizes. These diverse rain patterns demonstrate minimal variation across different scales. Consequently, we incorporate this correlation of rain across multiple scales into the diffusion generative process to develop a multi-scale noise estimator for image deraining (see Fig. 3).

At each time step t , when provided with an intermediate sample x_t and a rainy image \tilde{x} , our initial step involves downsampling the original images into various scales, such as $1/2$ and $1/4$, as represented by:

$$\{x_t^r, \tilde{x}^r\} = \text{down}(x_t, \tilde{x}), r \in \{1, 1/2, 1/4\} \quad (13)$$

where r represents the scale of the image. To make full use of multi-scale information for deraining, we then use attention heads [4] to learn all attention masks $\alpha_{r=\{1,1/2,1/4\}}$ for each of a fixed set of scales, which can be used to weight the multi-scale features at each pixel location. For each scale branch, the scale noise estimator $\hat{\epsilon}_\theta$ receives the intermediate variable x_t^r and rainy image \tilde{x}^r from a single (lower) scale r along with the time step t as input to predict the noise map $\hat{\epsilon}_t^r$, and its part of kernels' weights are optimized with w^r from DG-Hyper:

$$\hat{\epsilon}_t^r = \hat{\epsilon}_\theta(x_t^r, \tilde{x}^r, w^r, t) \quad (14)$$

We combine the noise maps from multiple scales, i.e., $\hat{\epsilon}_t^{r \in \{1,1/2,1/4\}}$, by multiplying the attention masks $\alpha_{r=\{1,1/2,1/4\}}$ with the maps in a pixel-wise manner, and then summing the results across different scales to obtain the final noise map $\hat{\epsilon}_t$. The whole multi-scale

Algorithm 1 The training of RainDiff**Require:** Unpaired clean image x , rainy image y .

- 1: **repeat**
- 2: $c_r = G_\phi^A(c), r_c = G_\phi^B(r)$
- 3: $t \sim \text{Uniform}(\{1, \dots, T\})$
- 4: $\epsilon \sim \mathcal{N}(0, I)$
- 5: $x_T = q(x_{1:T}|x_0), x_0 \in \{c, r_c\}$
- 6: $\{c_p, r_p\} = p_\theta(x_{0:T}|\tilde{x}), \tilde{x} \in \{c_r, r\}$
- 7: $c^* = G_\phi^B(c_r), r^* = G_\phi^A(r_p)$
- 8: **Take gradient descent step on**
- 9: $\nabla_{\theta, \phi} [|\epsilon - \epsilon_\theta(\sqrt{\alpha_t}c + \epsilon_t\sqrt{1-\alpha_t}, c_r, \tilde{w}_t, t)|^2$
- 10: $+ \nabla_{\theta, \phi} [|\epsilon - \epsilon_\theta(\sqrt{\alpha_t}r_c + \epsilon_t\sqrt{1-\alpha_t}, r, \tilde{w}_t, t)|^2$
- 11: $+ \lambda_{cyc} L_{cyc} + \lambda_{cyc^*} L_{cyc^*} + \lambda_{cl} L_{cl}]$
- 12: **until** converged
- 13: **return** θ

Algorithm 2 The testing of RainDiff**Require:** Rainy image \tilde{x} , multi-scale noise estimator $\epsilon_\theta(x_t, \tilde{x}, \tilde{w}, t)$, number of implicit sampling iterations T , and DG-Hyper $f(\cdot)$.

- 1: $x_t \sim \mathcal{N}(0, I)$
- 2: $\tilde{w} = f(\tilde{x})$
- 3: **for** each $i = S, \dots, 1$ **do**
- 4: $t = (i - 1) \cdot T/S + 1$
- 5: $t_{next} = (i - 2) \cdot T/S + 1$ **if** $i > 1$ **else** 0
- 6: $\hat{\epsilon}_t = \epsilon_\theta(x_t, \tilde{x}, \tilde{w}, t)$
- 7: $x_{t-1} = \sqrt{\alpha_{t_{next}}} \left(\frac{x_t - \sqrt{1-\alpha_t} \cdot \hat{\epsilon}_t}{\sqrt{\alpha_t}} \right) + \sqrt{1-\alpha_{t_{next}}} \cdot \hat{\epsilon}_t$
- 8: **end for**
- 9: **return** x_0

noise estimator $\epsilon_\theta(x_t, \tilde{x}, \tilde{w}, t)$ is expressed as:

$$\epsilon_\theta(x_t, \tilde{x}, \tilde{w}, t) = \hat{\epsilon}_t = \sum_{i=1}^N a_r * \hat{\epsilon}_t^i \quad (15)$$

where N denotes the total number of different scales r . Especially, our multi-scale noise estimator incorporates multi-scale rain information into the diffusion generative process and introduces an attention mechanism that softly weights the multi-scale features at each pixel location.

The testing phase of RainDiff is outlined in Algorithm 2. A large T inevitably leads to costly sampling, e.g., when $T = 1000$. To address this problem, we use an implicit sampling strategy [38] to accelerate our sampling process (lines 4-5 in Alg. 2). Implicit sampling with a multi-scale noise estimator $\epsilon_\theta(x_t, \tilde{x}, \tilde{w}, t)$ can be performed by:

$$x_{t-1} = \sqrt{\alpha_{t-1}} \left(\frac{x_t - \sqrt{1-\alpha_t} \cdot \epsilon_\theta(x_t, \tilde{x}, \tilde{w}, t)}{\sqrt{\alpha_t}} \right) + \sqrt{1-\alpha_{t-1}} \cdot \epsilon_\theta(x_t, \tilde{x}, \tilde{w}, t) \quad (16)$$

Table 1: Descriptions of the established mixture dataset.

Type	Synthetic					Total
	RS	RH	RD	RDS	RHS	
Train-Set	RainDS	OTS	RainDS	RainDS	RainCityscapes	-
Num	1000	1000	1000	1000	1000	5000
Test-Set	RainDS	OTS	RainDS	RainDS	RainCityscapes	-
Num	200	200	200	200	200	1000
Type	Real-world					Total
	RS	RH	RD	RDS	RHS	
Train-Set	RainDS	RTTS	RainDS	RainDS	GT-Rain	-
Num	150	150	150	150	150	750
Test-Set	RainDS	RTTS	RainDS	RainDS	GT-Rain	-
Num	98	98	98	98	98	490

during accelerated sampling we only needs a subsequence τ_1, \dots, τ_S of the complete $1, \dots, T$ timestep indices, which can be performed by:

$$\tau_i = (i - 1) \cdot T/S + 1 \quad (17)$$

At any denoising time step t , according to Eq. 15, we utilize the multi-scale noise estimator $\epsilon_\theta(x_t, \tilde{x}, \tilde{w}, t)$ from lines 9-10 in Alg. 1 to estimate the noise map $\hat{\epsilon}_t$ (line 6 in Alg. 2). Subsequently, we perform an implicit sampling update utilizing the noise map $\hat{\epsilon}_t$ (line 7 in Alg. 2).

4 Experiment

4.1 Experimental Settings

Implementation Details. We implement RainDiff using Pytorch 1.6 on an Nvidia GeForce RTX 3090 GPU. For optimizing RainDiff, we use the Adam optimizer with a min-batch size of 4 to train the paradigm, where the momentum parameters β_1 and β_2 take the values of 0.5 and 0.999, respectively. The initial learning rate is set to $1e^{-4}$. For training, a 128×128 patch is randomly cropped from the original image (or its horizontal flipped version). The balance weights λ_{cyc} , λ_{cyc^*} and λ_{cl} are both set to 1.

Datasets. We adopt four challenging benchmark datasets to create a mixture dataset with five different rain degradations for evaluation (see Table 1), i.e., RainDS [32] includes synthetic and real-world images of rain streaks, raindrops, and the combination of them as well as their corresponding clean images, RESIDE- β [23] includes synthetic and real-world hazy images (OTS set and RTTS set), RainCityscapes [20] and GT-Rain [2] include synthetic and real-world images of the combination of rain streaks and rainy haze, respectively. Take a synthetic train set as an example, each train set of different rain degradations contains 1000 images to build a whole mixture train set (including 5000 images). Similarly, the total number of images of synthetic test sets

Table 2: Comparisons of different deraining methods on the synthetic datasets. **Bold** and underline indicate the best and second-best results.

Method	Type	Synthetic				
		RS	RH	RD	RDS	RHS
		PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
DSC [29]	Prior-based	13.18/0.456	17.46/0.727	14.54/0.607	13.49/0.482	15.33/0.629
GMM [27]	Prior-based	14.64/0.485	18.21/0.718	16.35/0.659	15.92/0.585	16.75/0.683
DDN [10]	Supervised	23.79/0.692	22.42/0.882	22.92/0.836	19.74/0.644	18.66/0.790
DID-MDN [56]	Supervised	25.26/0.758	22.95/0.887	24.86/0.890	21.56/0.718	20.15/0.803
SPA-Net [42]	Supervised	31.09/0.906	25.38/0.915	27.98/ 0.920	25.82/0.847	23.38/0.898
DRD-Net [7]	Supervised	29.97/0.893	25.22/0.904	28.15/0.907	22.43/0.696	21.49/0.864
WeatherDiffusion [31]	Supervised	32.15/0.924	26.94/0.922	30.22/0.913	<u>28.09/0.882</u>	<u>23.84/0.892</u>
SIRR [45]	Semi-super	29.45/0.878	23.93/0.874	25.93/0.897	24.39/0.833	20.59/0.873
Syn2Real [52]	Semi-super	30.33/0.908	24.19/0.918	26.44/0.911	26.97/0.858	<u>21.31/0.901</u>
JRGR [53]	Semi-super	30.92/0.909	25.39/0.916	25.58/0.902	26.76/0.864	22.56/0.896
CycleGAN [59]	Unpaired	25.08/0.764	23.22/0.911	21.19/0.784	20.58/0.663	22.60/0.888
DerainCycleGAN [46]	Unpaired	25.40/0.770	20.14/0.812	20.69/0.799	20.06/0.660	19.06/0.865
DCD-GAN [5]	Unpaired	22.80/0.737	21.46/0.789	21.43/0.689	21.07/0.651	22.90/0.829
NLCL [54]	Unpaired	24.12/0.808	21.17/0.846	22.37/0.832	21.14/0.719	22.45/0.891
Ours	Unpaired	<u>31.30/0.911</u>	27.32/0.928	<u>30.54/0.915</u>	28.41/0.878	24.28/0.906

and real-world train and test sets are 1000, 750, and 490, respectively. To ensure a balanced representation of diverse rain degradations, we carefully select a specific number of images from RESIDE- β [23], RainCityscapes [20], and GT-Rain [2] based on the aforementioned criteria to construct the mixture dataset. Notably, we also collect some real-world rainy images without ground truth from the Internet for testing.

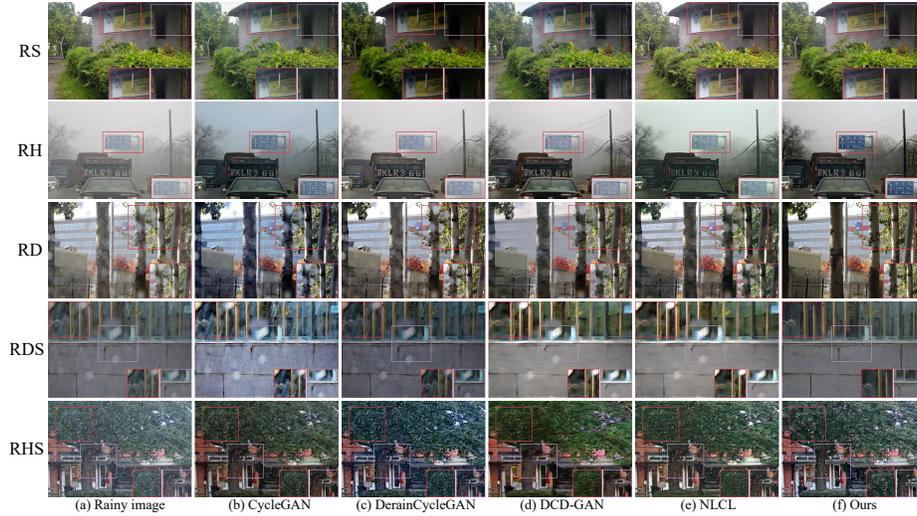
Comparison Methods. We qualitatively and quantitatively compare our method with two prior-based algorithms (i.e., DSC [29], GMM [27]), five paired supervised methods (i.e., DDN [10], DID-MDN [56], SPA-Net [42], DRD-Net [7], and WeatherDiffusion [31]), three semi-supervised methods (i.e., SIRR [45], Syn2Real [52], and JRGR [53]), as well as four unpaired methods (i.e., CycleGAN [59], DerainCycleGAN [46], DCD-GAN [5] and NLCL [54]). Two popular metrics are used for quantitative comparisons, i.e., Peak Signal-to-Noise Ratio (PSNR) [21] and Structure Similarity (SSIM) [44]. Higher value of these metrics indicates better performance of the methods. For fair comparisons, we re-train these methods on the mixture dataset that consists of the above five distinct rain degradations.

4.2 Comparison with State-of-the-arts

Comparison on Synthetic Datasets. Table 2 presents the quantitative results of different methods on five synthetic test sets. We make the following observations: 1) Compared with unpaired deraining methods, our method obtains higher values of PSNR and SSIM, which verifies the excellent performance of RainDiff. 2) There is an obvious performance gap between semi-supervised and supervised methods, which can be even more significant than the gap between RainDiff and supervised ones. 3) Our method is capable of achieving competitive results to existing supervised ones even without the paired data for supervision.

Table 3: Comparisons of different deraining methods on the real-world datasets. **Bold** and underline indicate the best and second-best results.

Method	Type	Real-world			
		RS	RD	RDS	RHS
		PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
DSC [29]	Prior-based	15.14/0.496	14.35/0.469	13.51/0.443	14.98/0.517
GMM [27]	Prior-based	17.48/0.527	14.77/0.481	14.24/0.475	15.18/0.532
DDN [10]	Supervised	18.39/0.608	17.23/0.525	15.67/0.522	17.41/0.590
DID-MDN [56]	Supervised	19.74/0.645	17.82/0.554	17.36/0.563	18.98/0.617
SPA-Net [42]	Supervised	21.54/0.672	18.95/0.563	19.12/0.608	21.28/ <u>0.656</u>
DRD-Net [7]	Supervised	22.49/0.716	19.34/0.636	18.96/0.590	20.82/0.629
WeatherDiffusion [31]	Supervised	<u>23.78/0.739</u>	20.57/0.648	19.90/0.622	21.08/0.618
SIRR [45]	Semi-super	23.42/0.718	20.66/0.610	17.98/0.576	21.04/0.630
Syn2Real [52]	Semi-super	23.17/0.691	19.27/0.621	18.54/0.583	<u>21.65/0.642</u>
JRGR [53]	Semi-super	23.49/0.712	<u>20.74/0.652</u>	18.13/0.560	20.39/0.618
CycleGAN [59]	Unpaired	21.07/0.612	18.24/0.568	17.55/0.515	18.71/0.572
DerainCycleGAN [46]	Unpaired	21.08/0.615	18.43/0.565	17.99/0.523	21.06/0.648
DCD-GAN [5]	Unpaired	18.81/0.692	17.81/0.553	<u>20.74/0.654</u>	17.84/0.537
NLCL [54]	Unpaired	21.71/0.699	19.24/0.606	19.88/0.580	20.18/0.603
Ours	Unpaired	24.73/0.770	21.89/0.685	22.18/0.681	22.66/0.694

**Fig. 4:** Comparison of deraining performance on real-world rainy images. Our method is more successful to remove different rain degradations and obtains the cleanest result with clear details.

Comparison on Real-world Datasets. For further general verification in practical use, we compare RainDiff with different methods on five real-world test sets. Notably,

Table 4: Ablation study for different models on the real-world RDS test set. $L_{err}(c, c_r)$ indicates line 9 in Alg. 1.

Model	w/o $L_{err}(c, c_r)$	w/o $L_{err}(r_c, r)$	w/o L_{cyc}	w/o L_{cyc}^*	w/o Up	w/o Low	w/o DG-Hyper	Ours
PSNR	20.66	20.98	19.57	19.85	17.48	17.63	18.98	22.18
SSIM	0.671	0.675	0.649	0.654	0.534	0.547	0.634	0.681

Table 5: Ablation analysis for different scales on the real-world RDS test set.

Settings	V1	V2	V3	V4
$r = 1$	✓	✓	✓	✓
$r = 1/2$	w/o	✓	w/o	✓
$r = 1/4$	w/o	w/o	✓	✓
PSNR	20.68	21.75	21.46	22.18
SSIM	0.658	0.679	0.670	0.681

these test sets, except for the hazy set, both contain real-world rainy images along with their corresponding ground truth for evaluation using the numerical metrics. And we visualize the real-world haze removal results in Fig. 4. Table 3 demonstrates that our method achieves superior performance gains to all compared methods. The incorporation of additional constraints provided by DG-Hyper enables RainDiff to effectively handle diverse rain degradations in real-world rainy images.

In Fig. 4, we visualize the rain removal results on the collected real-world rainy images. RainDiff successfully handles diverse rain degradations and achieves superior visual results than all compared methods. Notably, the deraining results of unpaired methods may induce color and structure distortion, whereas our method can better preserve the color and structure of the image. More qualitative results are provided in the supplementary material.

4.3 Ablation Study

Effect of Loss Function. We evaluate the effectiveness of our hybrid loss function on the real-world RDS test set. Especially, we remove one component to each configuration at one time. For fair comparison, the same training settings are kept for all models testing. As depicted in Table 4, the full structure of RainDiff exhibits the highest performance in both PSNR and SSIM metrics, suggesting that all the components of RainDiff are advantageous for proficient rain removal.

Effect of DG-Hyper. To show the effectiveness of our DG-Hyper, we conduct an ablation study on the real-world RDS test set by removing DG-Hyper with the corresponding loss function L_{cl} . From Table 4, we can see that DG-Hyper can add an additional constraint for multiple rain degradations, which further improves the real-world deraining performance of the proposed method. Furthermore, we train it with different combinations of multiple degradations to analyze how the performance is influenced by different degradations. The results are provided in the supplementary material.

Table 6: Ablation study for the choice of noise estimator on the real-world RDS test set.

Estimator	ResNet-50	VGG-16	U-Net
PSNR / SSIM	21.07/0.629	21.64/0.673	22.18/0.681

Effect of Cycle-consistent Circuits. As shown in Fig. 3, there are two cycle-consistent circuits in the unpaired cycle-consistent architecture. To further validate their effectiveness, we remove the upper circuit of rainy to rainy and the lower circuit of rain-free to rain-free, which are denoted as “w/o Up” and “w/o Low” in Table 4. The result indicates that both cycle-consistent circuits can enhance clean exemplars and offer supplementary constraints for improved rain removal.

Effect of Different Scale Settings. The multi-scale learning strategy provides the capability of scale-robust rain removal. We perform an ablation analysis of different scale settings as shown in Table 5. It is observed that the combination of scales $r \in \{1, 1/2, 1/4\}$ yields the best results.

Choice of Different Noise Estimators. Apart from U-Net [34], we adopt other baselines as our scale noise estimators in DCDM, such as ResNet-50 [18] and VGG-16 [36]. Notably, all the kernels’ parameters of ResNet-50 [18] and VGG-16 [36] are generated by our DG-Hyper. Table 6 indicates that U-Net is a fitting candidate for noise estimation.

5 Limitations

Although RainDiff shows superiority in five different rain degradations, it is unclear how its performance in diverse weather conditions, such as snow, low light, etc. In addition, similar to existing diffusion models [31, 35], it requires comparably longer runtime compared with end-to-end image restoration models which only require a single forward pass for processing without requisite steps involved during sampling. The time efficiency relies on the choice of algorithm hyper-parameters (e.g., a higher value of sampling steps increases image quality but also the inference time).

6 Conclusion

In this paper, we propose a new unpaired learning paradigm based on the diffusion model, called RainDiff, to tackle the unfavorable prevailing problem of real-world image deraining. The core of our method is a non-adversarial unpaired cycle-consistent architecture that can be trained using only unpaired data. Furthermore, we propose a degradation-conditioned diffusion model that learns multiple rain degradations for the diffusive generative process to improve the performance in image deraining for multiple degradations. Experiments on both synthetic and real-world rainy images demonstrate the superiority of the proposed framework.

References

1. Amit, T., Nachmani, E., Shaharbany, T., Wolf, L.: Segdiff: Image segmentation with diffusion probabilistic models. arXiv preprint arXiv:2112.00390 (2021) [4](#)
2. Ba, Y., Zhang, H., Yang, E., Suzuki, A., Pfahnl, A., Chandrappa, C.C., de Melo, C.M., You, S., Soatto, S., Wong, A., et al.: Not just streaks: Towards ground truth for single image deraining. In: European Conference on Computer Vision. pp. 723–740. Springer (2022) [10](#), [11](#)
3. Chang, Y., Yan, L., Zhong, S.: Transformed low-rank model for line pattern noise removal. In: Proceedings of the IEEE international conference on computer vision. pp. 1726–1734 (2017) [1](#), [4](#)
4. Chen, L.C., Yang, Y., Wang, J., Xu, W., Yuille, A.L.: Attention to scale: Scale-aware semantic image segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3640–3649 (2016) [8](#)
5. Chen, X., Pan, J., Jiang, K., Li, Y., Huang, Y., Kong, C., Dai, L., Fan, Z.: Unpaired deep image deraining using dual contrastive learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2017–2026 (2022) [2](#), [3](#), [4](#), [6](#), [11](#), [12](#)
6. Chen, Y.L., Hsu, C.T.: A generalized low-rank appearance model for spatio-temporally correlated rain streaks. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1968–1975 (2013) [4](#)
7. Deng, S., Wei, M., Wang, J., Feng, Y., Liang, L., Xie, H., Wang, F.L., Wang, M.: Detail-recovery image deraining via context aggregation networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14560–14569 (2020) [4](#), [11](#), [12](#)
8. Fan, Z., Wu, H., Fu, X., Huang, Y., Ding, X.: Residual-guide network for single image deraining. In: Proceedings of the 26th ACM international conference on Multimedia. pp. 1751–1759 (2018) [4](#)
9. Fu, X., Huang, J., Ding, X., Liao, Y., Paisley, J.: Clearing the skies: A deep network architecture for single-image rain removal. *IEEE Transactions on Image Processing* **26**(6), 2944–2956 (2017) [4](#)
10. Fu, X., Huang, J., Zeng, D., Huang, Y., Ding, X., Paisley, J.W.: Removing rain from single images via a deep detail network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1715–1723 (2017) [11](#), [12](#)
11. Fu, X., Qi, Q., Zha, Z.J., Zhu, Y., Ding, X.: Rain streak removal via dual graph convolutional network. In: Proc. AAAI Conf. Artif. Intell. pp. 1–9 (2021) [1](#)
12. Gu, S., Meng, D., Zuo, W., Zhang, L.: Joint convolutional analysis and synthesis sparse representation for single image layer separation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1708–1716 (2017) [4](#)
13. Gu, S., Chen, D., Bao, J., Wen, F., Zhang, B., Chen, D., Yuan, L., Guo, B.: Vector quantized diffusion model for text-to-image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10696–10706 (2022) [4](#)
14. Ha, D., Dai, A., Le, Q.: Hypernetworks (2016) [8](#)
15. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: 2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06). vol. 2, pp. 1735–1742. IEEE (2006) [7](#)
16. Han, K., Xiang, X.: Decomposed cyclegan for single image deraining with unpaired data. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 1828–1832. IEEE (2020) [2](#)
17. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9729–9738 (2020) [7](#)

18. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016) [14](#)
19. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems* **33**, 6840–6851 (2020) [2](#), [4](#), [5](#)
20. Hu, X., Fu, C.W., Zhu, L., Heng, P.A.: Depth-attentional features for single-image rain removal. In: Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. pp. 8022–8031 (2019) [10](#), [11](#)
21. Huynh-Thu, Q., Ghanbari, M.: Scope of validity of psnr in image/video quality assessment. *Electronics letters* **44**(13), 800–801 (2008) [11](#)
22. Jin, X., Chen, Z., Lin, J., Chen, Z., Zhou, W.: Unsupervised single image deraining with self-supervised constraints. In: 2019 IEEE International Conference on Image Processing (ICIP). pp. 2761–2765. IEEE (2019) [2](#), [4](#)
23. 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) [10](#), [11](#)
24. Li, B., Liu, X., Hu, P., Wu, Z., Lv, J., Peng, X.: All-in-one image restoration for unknown corruption. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17452–17462 (2022) [7](#)
25. Li, G., He, X., Zhang, W., Chang, H., Dong, L., Lin, L.: Non-locally enhanced encoder-decoder network for single image de-raining. In: Proceedings of the 26th ACM international conference on Multimedia. pp. 1056–1064 (2018) [4](#)
26. Li, X., Wu, J., Lin, Z., Liu, H., Zha, H.: Recurrent squeeze-and-excitation context aggregation net for single image deraining. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 254–269 (2018) [4](#)
27. Li, Y., Tan, R.T., Guo, X., Lu, J., Brown, M.S.: Rain streak removal using layer priors. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2736–2744 (2016) [1](#), [4](#), [11](#), [12](#)
28. Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., Pietikäinen, M.: Deep learning for generic object detection: A survey. *International journal of computer vision* **128**(2), 261–318 (2020) [3](#)
29. Luo, Y., Xu, Y., Ji, H.: Removing rain from a single image via discriminative sparse coding. In: Proceedings of the IEEE international conference on computer vision. pp. 3397–3405 (2015) [1](#), [11](#), [12](#)
30. Nichol, A.Q., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., Chen, M.: Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In: International Conference on Machine Learning. pp. 16784–16804. PMLR (2022) [2](#)
31. Özdenizci, O., Legenstein, R.: Restoring vision in adverse weather conditions with patch-based denoising diffusion models. arXiv preprint arXiv:2207.14626 (2022) [2](#), [4](#), [6](#), [11](#), [12](#), [14](#)
32. Quan, R., Yu, X., Liang, Y., Yang, Y.: Removing raindrops and rain streaks in one go. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9147–9156 (2021) [10](#)
33. Ren, D., Zuo, W., Hu, Q., Zhu, P., Meng, D.: Progressive image deraining networks: a better and simpler baseline. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3937–3946 (2019) [4](#)
34. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015) [6](#), [8](#), [14](#)

35. Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D.J., Norouzi, M.: Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022) [2](#), [4](#), [6](#), [14](#)
36. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014) [14](#)
37. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: *International Conference on Machine Learning*. pp. 2256–2265. *PMLR* (2015) [2](#), [4](#)
38. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502* (2020) [9](#)
39. Song, Y., Ermon, S.: Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems* **32** (2019) [2](#), [4](#)
40. Sultani, W., Chen, C., Shah, M.: Real-world anomaly detection in surveillance videos. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 6479–6488 (2018) [4](#)
41. Wang, G., Sun, C., Sowmya, A.: Erl-net: Entangled representation learning for single image de-raining. In: *Proceedings of the IEEE International Conference on Computer Vision*. pp. 5644–5652 (2019) [4](#)
42. Wang, T., Yang, X., Xu, K., Chen, S., Zhang, Q., Lau, R.W.: Spatial attentive single-image deraining with a high quality real rain dataset. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 12270–12279 (2019) [11](#), [12](#)
43. Wang, Y., Liu, S., Chen, C., Zeng, B.: A hierarchical approach for rain or snow removing in a single color image. *IEEE Transactions on Image Processing* **26**(8), 3936–3950 (2017) [4](#)
44. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* **13**(4), 600–612 (2004) [11](#)
45. Wei, W., Meng, D., Zhao, Q., Xu, Z., Wu, Y.: Semi-supervised transfer learning for image rain removal. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 3877–3886 (2019) [1](#), [4](#), [11](#), [12](#)
46. Wei, Y., Zhang, Z., Wang, Y., Xu, M., Yang, Y., Yan, S., Wang, M.: Deraincyclegan: Rain attentive cyclegan for single image deraining and rainmaking. *IEEE Transactions on Image Processing* **30**, 4788–4801 (2021) [2](#), [3](#), [4](#), [6](#), [11](#), [12](#)
47. Wei, Y., Zhang, Z., Wang, Y., Zhang, H., Zhao, M., Xu, M., Wang, M.: Semi-deraingan: A new semi-supervised single image deraining. In: *2021 IEEE International Conference on Multimedia and Expo (ICME)*. pp. 1–6. *IEEE* (2021) [2](#)
48. Welling, M., Teh, Y.W.: Bayesian learning via stochastic gradient langevin dynamics. In: *Proceedings of the 28th international conference on machine learning (ICML-11)*. pp. 681–688 (2011) [5](#)
49. Wojna, Z., Ferrari, V., Guadarrama, S., Silberman, N., Chen, L.C., Fathi, A., Uijlings, J.: The devil is in the decoder: Classification, regression and gans. *International Journal of Computer Vision* **127**(11), 1694–1706 (2019) [4](#)
50. 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) [7](#), [8](#)
51. Yang, W., Tan, R.T., Feng, J., Liu, J., Guo, Z., Yan, S.: Deep joint rain detection and removal from a single image. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 1357–1366 (2017) [1](#), [4](#)
52. Yasarla, R., Sindagi, V.A., Patel, V.M.: Syn2real transfer learning for image deraining using gaussian processes. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 2726–2736 (2020) [1](#), [4](#), [11](#), [12](#)

53. Ye, Y., Chang, Y., Zhou, H., Yan, L.: Closing the loop: Joint rain generation and removal via disentangled image translation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2053–2062 (2021) [1](#), [4](#), [11](#), [12](#)
54. Ye, Y., Yu, C., Chang, Y., Zhu, L., Zhao, X.L., Yan, L., Tian, Y.: Unsupervised deraining: Where contrastive learning meets self-similarity. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5821–5830 (2022) [2](#), [11](#), [12](#)
55. Yu, F., Koltun, V., Funkhouser, T.: Dilated residual networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 472–480 (2017) [7](#)
56. Zhang, H., Patel, V.M.: Density-aware single image de-raining using a multi-stream dense network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 695–704 (2018) [4](#), [11](#), [12](#)
57. Zhang, H., Sindagi, V., Patel, V.M.: Image de-raining using a conditional generative adversarial network. IEEE transactions on circuits and systems for video technology **30**(11), 3943–3956 (2019) [4](#)
58. Zhu, H., Peng, X., Zhou, J.T., Yang, S., Chandrasekh, V., Li, L., Lim, J.H.: Single image rain removal with unpaired information: A differentiable programming perspective. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 9332–9339 (2019) [2](#), [4](#)
59. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision. pp. 2223–2232 (2017) [2](#), [4](#), [6](#), [11](#), [12](#)