Self-Aligned Video Deraining with Transmission-Depth Consistency

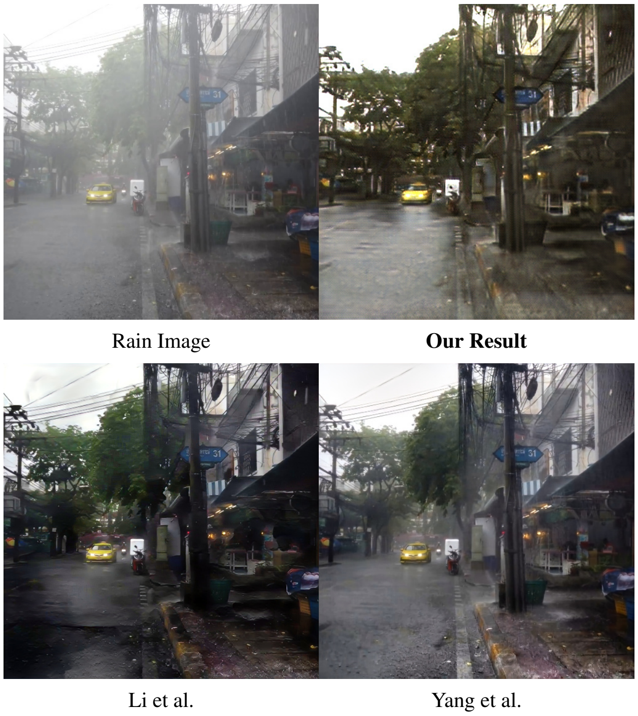
Wending Yan[1], Robby T. Tan[1,2], Wenhan Yang[3] and Dengxin Dai[4]
[1]National University of Singapore, [2]Yale-NUS College, [3]City University of Hong Kong ,[4]ETH Zurich
eleyanw@nus.edu.sg, robby.tan@{nus,yale-nus}.edu.sg, wyang34@cityu.edu.hk, dai@vision.ee.ethz.ch

Abstract

In this paper, we address the problem of rain streaks and rain accumulation removal in video, by developing a self-alignment network with transmission-depth consistency. Existing video based deraining methods focus only on rain streak removal, and commonly use optical flow to align the rain video frames. However, besides rain streaks, rain accummulation can considerably degrade visibility; and, optical flow estimation in a rain video is still erroneous, making the deraining performance tend to be inaccurate. Our method employs deformable convolution layers in our encoder to achieve feature-level frame alignment, and hence avoids using optical flow. For rain streaks, our method predicts the current frame from its adjacent frames, such that rain streaks that appear randomly in the temporal domain can be removed. For rain accumulation, our method employs a transmission-depth consistency loss to resolve the ambiguity between the depth and water-droplet density. Our network estimates the depth from consecutive rain-accumulation-removal outputs, and calculates the transmission map using a commonly used physics model. To ensure photometric-temporal and depth-temporal consistencies, our method estimates the camera poses, so that it can warp one frame to its adjacent frames. Experimental results show that our method is effective in removing both rain streaks and rain accumulation, outperforming those of state-of-the-art methods quantitatively and qualitatively.

1. Introduction

Rain is a common outdoor weather condition, and degrades visibility in video. The degradation are mainly caused by: rain streaks and rain accumulation (or rain veiling effect). Rain streaks partially occlude a background scene, change image appearance, cause the scene to look blurred. Rain accumulation, which is like fog or mist, washes out the scene colors, reduces the overall contrast and generates a veiling effect. Both rain streaks and accumulation are visibly present and thus degrades the visibility of a scene. Hence, to obtain better background scene visual information, we need to remove both rain streaks and rain accumulation in videos.


Rain Image

Our Result

Li et al.

Yang et al.

Figure 1: Top left: Input image. Top right: Our result. Bottom left: Li et al.'s result [30]. Bottom right: Yang et al.'s result [51]. Zoom-in for better visualization.

A series of rain removal methods for videos have been proposed [45, 39, 52, 53, 49, 18, 30, 33, 48, 51]. Most of them focus on rain streaks alone, and thus cannot deal with rain accumulation, which is unfortunately commonly present in any rainy situations, particularly in heavy rain. Many of these methods, e.g., [25, 33, 48, 51] rely on optical flow for aligning adjacent frames. Yet, optical flow estimation in rainy conditions is still unstable and challenging, since its main constraint (the brightness constancy con-

†This work is supported by MOE2019-T2-1-130.

straint) generally does not hold.

For rain accumulation, to our knowledge, there is no video-based method dealing with the problem. There are rain-accumulation removal methods for single images, e.g. [49, 30, 18]. Unfortunately, they suffer from the ambiguity between depth and water-droplet density, i.e.: a thick veiling effect can be triggered either by a relatively sparse droplet density but a distant scene, or by a relatively dense droplet but a nearby scene. This ambiguity applies to all surfaces, yet particularly to achromatic surfaces (i.e., white, gray), causing the over-saturation or under-saturation effect in the rain removal results.

In this paper, we address the problem of daytime rain removal from video by focusing on both rain streaks and rain accumulation. To accomplish this task, first, we align a few consecutive input frames using a feature-based alignment network; and thus unlike many existing methods, we do not rely on optical flow. Second, our network removes rain-steaks in every frame based on the aligned features of its adjacent frames, as most likely rain streaks randomly appear along the temporal domain. We train our network with both synthetic rain videos with ground-truths and real rain videos without ground-truths. Third, to deal with rain accumulation, we exploit the depth cues that can be obtained from the input video. Given the estimated rainstreak-free images from the previous step, our accumulationNet generates a rain-accumulation free images, which is our final output. To train the accumulationNet, we employ a few losses: depth-transmission consistency loss, depth-temporal consistency loss, and photo-temporal consistency loss.

As a summary, our contributions are as follows:

- We introduce a video deraining method that can remove both rain streaks and rain accumulation in one end-to-end framework. To our knowledge, this is the first attempt in video deraining dealing with both rain streaks and rain accumulation.

- We provide a video-based deraining method with feature-level alignment. Many existing deraining methods use optical flow, which brings many issues due to the degradation in the input video. By using deformable convolution layers in encoder, we avoid using optical flow in our method.

- We propose a few losses that combines depth, transmission map, and camera pose to deal with rain accumulation. The use of depth and camera pose enable our method to handle the depth and water-droplet ambiguity problem, and thus improving our results.

Using these novel ideas, our experimental results show the effectiveness of our method compared to the-state-of-the-art methods qualitatively and quantitatively.

2. Related Works

Recently, many methods have been proposed for rain removal in images. Many of them try to capture the pattern signal differences between rain streaks and background texture, and then remove all detected rain streaks[7, 23, 37, 24, 32, 4, 57]. More recently, deep learning methods become the main trend for rain streak removal. Many of them remove rain streaks from single image by developing advanced networks [28, 45, 39, 52, 31, 43, 11] or by utilizing more effective priors [4, 60, 53, 9, 36, 44].

Some methods deal with rain streak and accumulation issues together. Yang et al. embed wavelet tranform to remove multiscale rain streaks, and further enhance visibility for rain accumulation and darkness [49]. Hu et al. designed a depth-guided attention network to remove rain accumulation [18]. Li et al. proposed a two-stage method that has a physical-model deraining first and follows a GAN refinement stage [30]. Wang et al. reformulated rain streaks as transmission medium together with vapors for rain imaging modeling, and an encoder-decoder CNN is used to learn the transmission map of rain streaks [46]. Yasarla employed the Gaussian prcesses to predict pseudo-GT of real rainy images at the latent space by jointly modeling the labeled and unlabeled latent space vectors [54]. Yan et al. [50] provides a comprehensive survey on single image deraining methods.

Usually, a video contains more information and temporal correlation than a single image, so many methods process deraining on video. Unlike in a single image method, the video-based methods attempt to exploit the temporal domain. This temporal domain can be useful, since due to the dynamic of rain streaks, there is different information in different frames. Garg and Nayar firstly propose the video a rain model [14] and rain streak removal methods [12, 13, 15]. Later, more methods are proposed with more intrinsic priors on difference between rain streak and normal background signals [35, 58, 2, 3, 1, 7, 21, 5, 42, 41, 25, 47, 40]. Many deep learning methods are also proposed for video rain streak removal. Li et al. proposed a multiscale method with multiscale convolutional spares coding [29].

Chen et al. applied segment superpixels on video frames and predicted clean background through aligned superpixels [6]. Liu et al. proposed a recurrent neural network contains rain level classification [34]. Liu et al. combined the motion segmentation context information into a dynamic routing residue recurrent network, to solve the rain streaks and occlusions together [33]. Yang et al. proposed a two-stage recurrent network with dual-level regularization and physical model [48]. Yang et al. introduce a self-learning method with temporal correspondence, which is free from supervised training data [51]. While this method is elegant and provides good results, it relies on optical flow, which can be vulnerable in heavy rain videos.

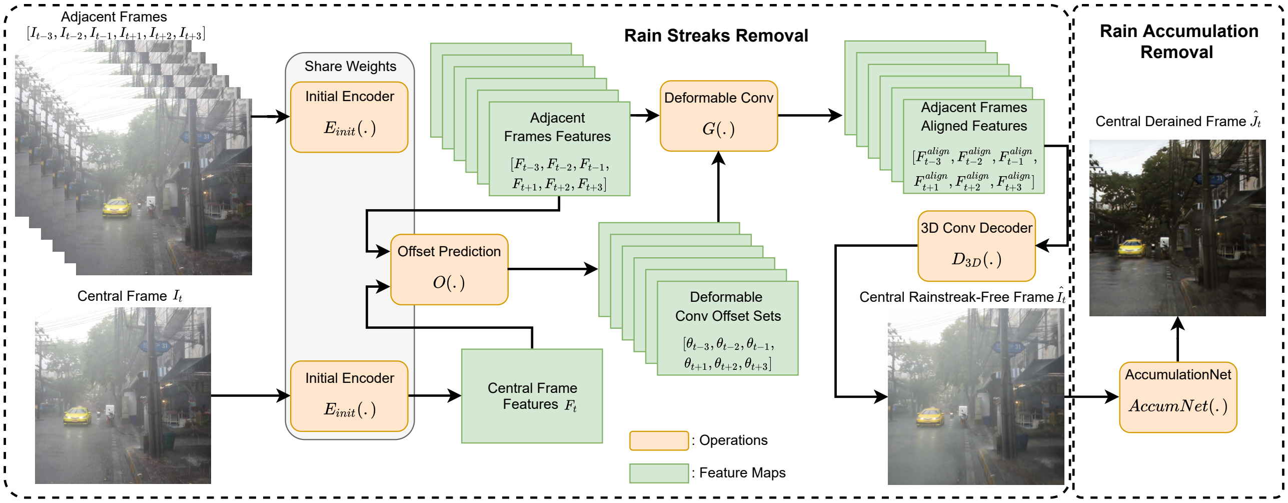

Adjacent Frames
$[I_{t-3}, I_{t-2}, I_{t-1}, I_{t+1}, I_{t+2}, I_{t+3}]$
Share Weights
Initial Encoder
$E_{init}(.)$
Adjacent Frames Features
$[F_{t-3}, F_{t-2}, F_{t-1}, F_{t+1}, F_{t+2}, F_{t+3}]$
Rain Streaks Removal
Deformable Conv
$G(.)$
Adjacent Frames Aligned Features
$[F^{align}_{t-3}, F^{align}_{t-2}, F^{align}_{t-1}, F^{align}_{t+1}, F^{align}_{t+2}, F^{align}_{t+3}]$
Rain Accumulation Removal
Central Derained Frame $\hat{J}_t$
Offset Prediction
$O(.)$
3D Conv Decoder
$D_{3D}(.)$
Deformable Conv Offset Sets
$[\theta_{t-3}, \theta_{t-2}, \theta_{t-1}, \theta_{t+1}, \theta_{t+2}, \theta_{t+3}]$
Central Frame I_t
Central Rainstreak-Free Frame $\hat{I}_t$
Initial Encoder
$E_{init}(.)$
Central Frame Features F_t
AccumulationNet
$AccumNet(.)$
: Operations
: Feature Maps

Figure 2: The pipeline of our framework that consists of two main components: Rain Streaks Removal Component (left) and Rain Accumulation Removal Component (right). For the images, zoom-in for better visualization.

3. Proposed Method

Figure 2 shows the pipeline of our method, which consists of two parts: rain streak removal and rain accumulation removal. Our input is a video, which we process in batches. Each batch consist of N consecutive frames, which in our experiments $N = 7$: $(I_{t-3}, I_{t-2}, I_{t-1}, I_t, I_{t+1}, I_{t+2}, I_{t+3})$, where I_t indicates the current frame (or central frame). Our output for each batch is J_t, which is the derained result of the current frame. J_t is free from both rain streaks and rain accumulation.

3.1. Rain Streaks Removal

In a rain video, most likely rain streaks appear in adjacent video frames randomly, since rain streaks move independently from the camera. It means that the locations of a rain streak will be different from frame to frame. It further implies that, in a set of frames, we can obtain a background scene/area that are occluded by rain streaks in some frames, but not in other frames. For adjacent frames, we can also assume that the images of the background scene are highly correlated. Meaning, the background scene in a few consecutive frame largely overlap. Thus, if we have inputs: $(I_{t-3}, I_{t-2}, I_{t-1}, I_{t+1}, I_{t+2}, I_{t+3})$, which exclude the current frame I_t, we can have the information of the rainstreak-free background scene for I_t. The following paragraphs discuss the details on how we can exploit the adjacent frames to remove rain streaks.

Feature-Alignment Encoder Our method takes a few adjacent frames as input, and our first step is to align them. Many video deraining methods (e.g., [25, 33, 48, 51]) employ optical flow to obtain the correlations between frames. Unfortunately, in rain conditions, estimating optical flow is problematic, since the main assumption, which is the brightness constancy, is largely violated. Thus, enforcing the assumption likely causes the estimation to be erroneous.

To address the issue, our method align the features of the input images, instead of aligning the input image directly. The key idea of the feature-level alignment is the deformable convolution layers [8]. Unlike the common convolutional layers with fixed kernel configuration, in the deformable convolution layers, the grid is deformable, similar to dilated convolution layers [55]. Unlike the offsets in the dilated convolution layers, which are fixed, the offsets in the deformable convolution layers are learnable. With proper training on these learnable offsets, the deformable convolution layers can align features in the feature domain.

More specifically, our initial encoder E_{init} extracts feature maps from the central frame I_t and all adjacent frames $(I_{t-3}, I_{t-2}, I_{t-1}, I_{t+1}, I_{t+2}, I_{t+3})$.

$$F_i = E_{init}(I_i), \tag{1}$$

where $i \in \{t-3, t-2, t-1, t, t+1, t+2, t+3\}$. Note that those extracted feature maps $(F_{t-3}, F_{t-2}, F_{t-1}, F_{t+1}, F_{t+2}, F_{t+3})$ are not yet aligned to the central frame's feature map F_t.

We predict the offsets θ_i from each unaligned feature map. All offsets are predicted with respect to the the central feature map F_t. For each convoluted feature map, the number of offsets is the same as the number of sampled pixels in this feature map. The function for offsets prediction O contains a few convolutional layers:

$$\theta_i = O(F_i, F_t). \tag{2}$$

Once we obtain $(\theta_{t-3}, \theta_{t-2}, \theta_{t-1}, \theta_{t+1}, \theta_{t+2}, \theta_{t+3})$, we extract the aligned feature maps $(F^{align}_{t-3}, F^{align}_{t-2}, F^{align}_{t-1}, F^{align}_{t+1}, F^{align}_{t+2}, F^{align}_{t+3})$ from the unaligned feature maps using the deformation convolution layers, G:

$$F^{align}_i = G(F_i, \theta_i). \tag{3}$$

Note that, the central image I_t and its features F_i are merely used as the reference for calculating the offset θ_i. Thus, all the aligned adjacent features do not receive any information from the central feature map, F_t.

Rainstreak Removal Decoder Since adjacent video frames are correlated, their aligned features should contain the same texture information except the random rain streaks. We use 3D convolution layers, D_{3D}, to decode a rainstreek-free output S_t from a set of the aligned feature maps $(F^{align}_{t-3}, F^{align}_{t-2}, F^{align}_{t-1}, F^{align}_{t+1}, F^{align}_{t+2}, F^{align}_{t+3})$:

$$\begin{aligned} S_t = D_{3D}(&F^{align}_{t-3}, F^{align}_{t-2}, F^{align}_{t-1} \\ &, F^{align}_{t+1}, F^{align}_{t+2}, F^{align}_{t+3}). \end{aligned} \tag{4}$$

If in a feature map, one location is covered by a rain streak, the same location is likely to have background information in any of the other feature maps. Inspired by [51] , we employ 3D convolutional layers, where our decoder has an extra temporal dimension. Once trained with proper losses, our decoder can ignore the rain streak features in the temporal dimension. It can output rain-streak-free images by choosing frames' features that represent clean background features.

In summary, the whole network of our rain streaks removal $StreakNet(.)$ is expressed as:

$$\begin{aligned} S_t = StreakNet((&I_{t-3}, I_{t-2}, I_{t-1}, \\ &I_{t+1}, I_{t+2}, I_{t+3}), I_t), \end{aligned} \tag{5}$$

where $StreakNet$ consists of D_{3D}, G, O, and E_{init} operations. To train the whole networks require a few losses, which in our case include supervised and unsupervised losses (semi-supervised learning). We discuss the details of the losses in the following section.

3.1.1 Rain-Streak Removal Loss Functions

To train the networks in rain-streak removal module, we apply semi-supervised learning by combining synthetic rain-streak images with ground-truths and real rain-streak images without ground-truths.

Clean L1 Loss Using paired clean synthetic data, we train our initial encoder E_{init} in Eq. (1) and deformable convolution layers O in Eq. (2) and G in Eq. (3) using the following loss:

$$\mathcal{L}_{L1\_clean} = \left\| \hat{J}^{synt}_t, J^{synt-gt}_t \right\|_1, \tag{6}$$

where $\hat{J}^{synt}_t = StreakNet((J^{synt-gt}_{t-3}, J^{synt-gt}_{t-2}, J^{synt-gt}_{t-1}, J^{synt-gt}_{t+1}, J^{synt-gt}_{t+2}, J^{synt-gt}_{t+3}), J^{synt-gt}_t)$. Once trained, we freeze our 3 networks: E_{init}, O, G; and, we further train the 3D-conv decoder D_{3D} using the following semi-supervised training strategy.

Rain-Streak L1 Loss For paired synthetic data with rain-streak-free ground-truths, we apply L1 loss between the output S^{synt}_t and the corresponding rain-streak-free ground-truth $S^{synt\_gt}_t$:

$$\mathcal{L}_{L1\_streak} = \left\| S^{synt}_t, S^{synt\_gt}_t \right\|_1. \tag{7}$$

where $S^{synt}_t = StreakNet([I^{synt}_{t-3}, I^{synt}_{t-2}, I^{synt}_{t-1}, I^{synt}_{t+1}, I^{synt}_{t+2}, I^{synt}_{t+3}], I^{synt}_t)$ using Eq. (5). I^{synt}_t is the synthetic image at frame i. This loss is a supervised loss.

Self-Learned Consistency Loss As for real rain-streak images without ground-truths, we design a self-learned consistency loss:

$$\mathcal{L}_{self\_consis} = \|S_t, I_t\|_1 \tag{8}$$

where $S_t = StreakNet((I_{t-3}, I_{t-2}, I_{t-1}, I_{t+1}, I_{t+2}, I_{t+3}), I_t)$. In this loss, we basically enforce S_t to be consistent with I_t. Note that, S_t is generated by the adjacent frames of I_t, but not directly from I_t itself. Moreover, the loss tries to ensure that S_t to be consistent with the clean background of I_t. While this loss cannot fully ensure this, since I_t contains rain-streaks, we rely on the other losses to handle the rain streaks regions.

Once our whole rain-streaks removal network $StreakNet$ is properly trained, we freeze the whole network, and begin to train our rain accumulation removal network.

3.2. Rain Accumulation Removal

Once rain streaks are removed, we turn our attention to the rain accumulation removal module. In this module, our inputs are $(S_{t-3}, S_{t-2}, S_{t-1}, S_t, S_{t+1}, S_{t+2}, S_{t+3})$, which are the adjacent frames which are free from rain-streaks. Our output is J_t, which is the estimated clean background, free from both rain streaks and rain accumulation. To obtain this output, we employ a deep network we call $AccumNet$, which takes a single image as input, and output J_t:

$$J_t = AccumNet(S_t). \tag{9}$$

To train this network we apply a few losses: transmission-depth consistency loss, photo temporal consistency loss and depth-temporal consistency loss, as shown in Figure 3.

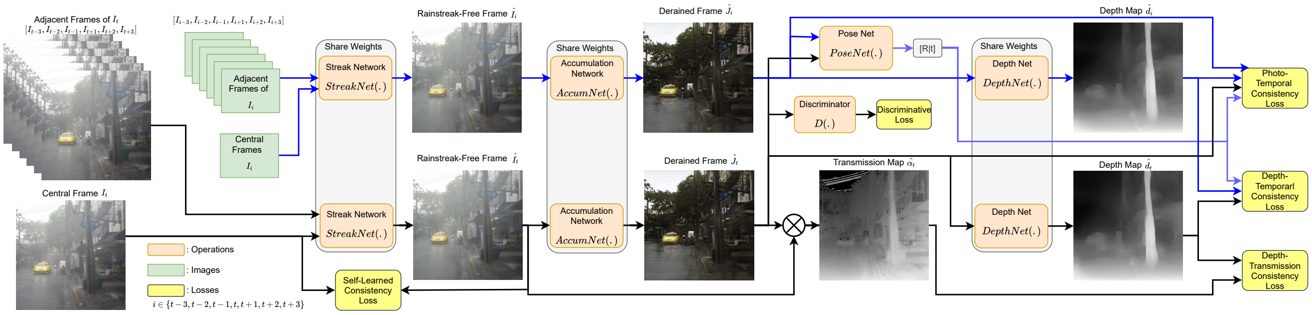

Adjacent Frames of I_t
$[I_{t-3}, I_{t-2}, I_{t-1}, I_{t+1}, I_{t+2}, I_{t+3}]$
$[I_{i-3}, I_{i-2}, I_{i-1}, I_{i+1}, I_{i+2}, I_{i+3}]$
Adjacent Frames of I_i
Central Frames I_i
Central Frame I_t
Share Weights
Streak Network $StreakNet(.)$
Streak Network $StreakNet(.)$
Rainstreak-Free Frame $\hat{I}_i$
Rainstreak-Free Frame $\hat{I}_t$
Share Weights
Accumulation Network $AccumNet(.)$
Accumulation Network $AccumNet(.)$
Derained Frame $\hat{J}_i$
Derained Frame $\hat{J}_t$
: Operations
: Images
: Losses
$i \in \{t-3, t-2, t-1, t, t+1, t+2, t+3\}$
Self-Learned Consistency Loss
Pose Net $PoseNet(.)$
[R|t]
Discriminator $D(.)$
Discriminative Loss
Share Weights
Depth Net $DepthNet(.)$
Depth Net $DepthNet(.)$
Depth Map $\hat{d}_i$
Depth Map $\hat{d}_t$
Transmission Map $\hat{\alpha}_t$
Photo-Temporal Consistency Loss
Depth-Temporarl Consistency Loss
Depth-Transmission Consistency Loss

Figure 3: The flow of our loss functions. For the images, zoom-in for better visualization.

3.2.1 Transmission-Depth Consistency Loss

A rain image suffering only from the rain accumulation looks similar to fog or mist. Hence, we can apply the Koschmieder law:

$$S_t(x) = \alpha_t(x)J_t(x) + (1 - \alpha_t(x))A_t, \tag{10}$$

where J_t is the derained image, free from rain-streak and rain accumulation. α_t is the transmission map and A_t is the atmospheric light. x is the pixel spatial location. To estimate the transmission map, we need to estimate the atmospheric light A_t from $S_t(x)$. In our implementation, we utilize the commonly used technique of the brightest region [16]. Given S_t, J_t and the estimated A_t, we can calculate the transmission map, α_t. From the physics model, the transmission map α_t is defined as:

$$\alpha_t(x) = \exp(-\beta d_t(x)), \tag{11}$$

where d_t is the depth (the distance between the camera and the scene). β is the water particle attenuation factor.

Hence, based on the physics model, we can enforce the consistency loss between the transmission map and depth map:

$$\mathcal{L}_{trans-depth} = \left\| Norm(-log(\alpha_t)), Norm(d_t) \right\|_1, \tag{12}$$

where $Norm(.)$ is the normalization function that normalize map value to be [0, 1], defined as $Norm(u) = (u_{max} - u)/(u_{max} - u_{min})$. u is the normalized input, u_{max} and u_{min} are the maximal and minimum values of u. By employing this normalization function, we cancel out the presence of the attenuation factor, β.

Depth Estimation To compute the depth map, d_t, we create a network $DepthNet$ adopted from [38], which is pretrained using clear monocular videos. $DepthNet$ estimates the depth map for every input frame J_t:

$$d_t = DepthNet(J_t). \tag{13}$$

We do not freeze $DepthNet$, instead we further train this network and $AccumNet$ in Eq. (9) together using the transmission-depth consistency loss in Eq. (12) and other losses, as shown in Figure 3. To our knowledge, this is a first method that integrate depth estimation and deraining jointly. Here, $AccumNet$ not only gets benefit from the depth information, but also $DepthNet$ can learn sharper depth map from the transmission map, α_t, which is computed from J_t, the output of $AccumNet$. In other words, the transmission map and the depth map support each other to have better performance.

3.2.2 Temporal Consistency Losses

While we consider that the extra depth information can benefit our rain accumulation removal, all the information we use so far is only from one single frame. In this section, since our method is a video-based method, we exploit the availability of adjacent frames by designing temporal consistency losses.

Although our features from different frames are aligned as discussed in Section 3.1, our image frames themselves are not. Thus, we employ a camera-pose estimation network, $PoseNet$ [56], to estimate the camera pose from adjacent frames to the central frame:

$$\{R, t\}_{i \to t} = PoseNet(J_i, J_t), \tag{14}$$

where t is the index of the central frame, and i are index of the adjacent frames. As defined before, $i \in \{t-3, t-2, t-1, t+1, t+2, t+3\}$.

Given the depth map, d_t, and the camera pose $\{R, t\}_{i \to t}$, we can obtain a projection function that warps frame i to frame t: $\hat{J}_t = \pi_{i \to t}(J_i)$. Using this projection function, we can form two temporal consistency losses: photo-temporal consistency and depth-temporal consistency.

Photo-Temporal Consistency Loss We define the loss as:

$$\mathcal{L}_{photo-temp} = \sum_i \left\| J_t, \pi_{i \to t}(J_i) \right\|_1, \tag{15}$$

where $i \in \{t-3, t-2, t-1, t, t+1, t+2, t+3\}$. J_t is the derained central frame, and J_i is one of the derained adjacent frames. Here, we enforce photo consistency for every pixels along the temporal domain after warping.

Depth-Temporal Consistency Loss Similarly, we apply the temporal consistency for our depth estimations:

$$\mathcal{L}_{depth-temp} = \sum_{i} \|d_t, \pi_{i \to t}(d_i)\|_1 , \qquad (16)$$

where d_t is the depth map for the central frame, and d_i is the depth map from an adjacent frame.

3.2.3 Other Losses in Rain Accumulation Network

Besides the 3 losses in in Eqs. (12,15,16), we also train $AccumNet$ using the following losses:

Rain-Accumulation L1 Loss We apply the L1 loss on paired rain-accumulation synthetic data I_t^{synt} with the clean ground-truth $J_t^{synt\_gt}$:

$$\mathcal{L}_{L1\_accum} = \left\|AccumNet(I_t^{synt}), J_t^{synt-gt}\right\|_1 . \qquad (17)$$

Discriminative Loss As for real rain-accumulation data without ground-truths, we apply a discriminative loss:

$$\mathcal{L}_{dis} = -\log(D(J^{ref}) - \log(1 - D(J_t))), \qquad (18)$$

where J^{ref} as an unpaired clean reference image.

4. Experimental Results

In section, we discuss the details of our implementation, and evaluate our method by comparing them with those of the existing methods quantitatively and qualitatively.

4.1. Implementation Details

In our feature-alignment encoder, the initial encoder is based on the ResNet architecture [17], which has five resnet blocks. Our rainstreak removal decoder is formed by 19 layers of 3D convolution layers. In our rain accumulation removal module, our accumulation network is ResNet too, but with nine resnet blocks. The discriminator is a multi-layer network consists of three stride layers[19, 27].

In our method, we need a set of synthetic data to train the networks. We estimate the depth maps of clear video frames using a single image depth estimation method [38, 59]. Then we render both rain streaks and accumulation on clear frames based on estimated depth maps. To render rain accumulation, we set the range of $\beta = [4.6, 6.6]$, and we choose atmospheric light values randomly between $A = [178, 255]$. Note that, β and atmospheric light are constant in one continuous frames sequence. During the training, the network is optimized using the Adam method [26] with learning rate 2×10^{-4} and $\beta_1 = 0.9$.

Table 1: Quantitative results on our synthetic rainy data.

	PSNR	SSIM
Input Image	14.51	0.5189
HRRestorer [30]	16.57	0.7033
DualFlow [48]	15.13	0.6755
Syn2Real [54]	14.94	0.5274
FastDeRain+MSBDN [22]	14.65	0.6206
MSPFN+MSBDN [20]	14.54	0.5499
SLDNet+MSBDN [51]	14.63	0.5411
Without Temporal Consistency	14.56	0.5387
Without DepthNet	16.15	0.6408
With Frozen DepthNet	16.43	0.6874
Without PoseNet	17.28	0.7071
Our Result	**17.52**	**0.7284**

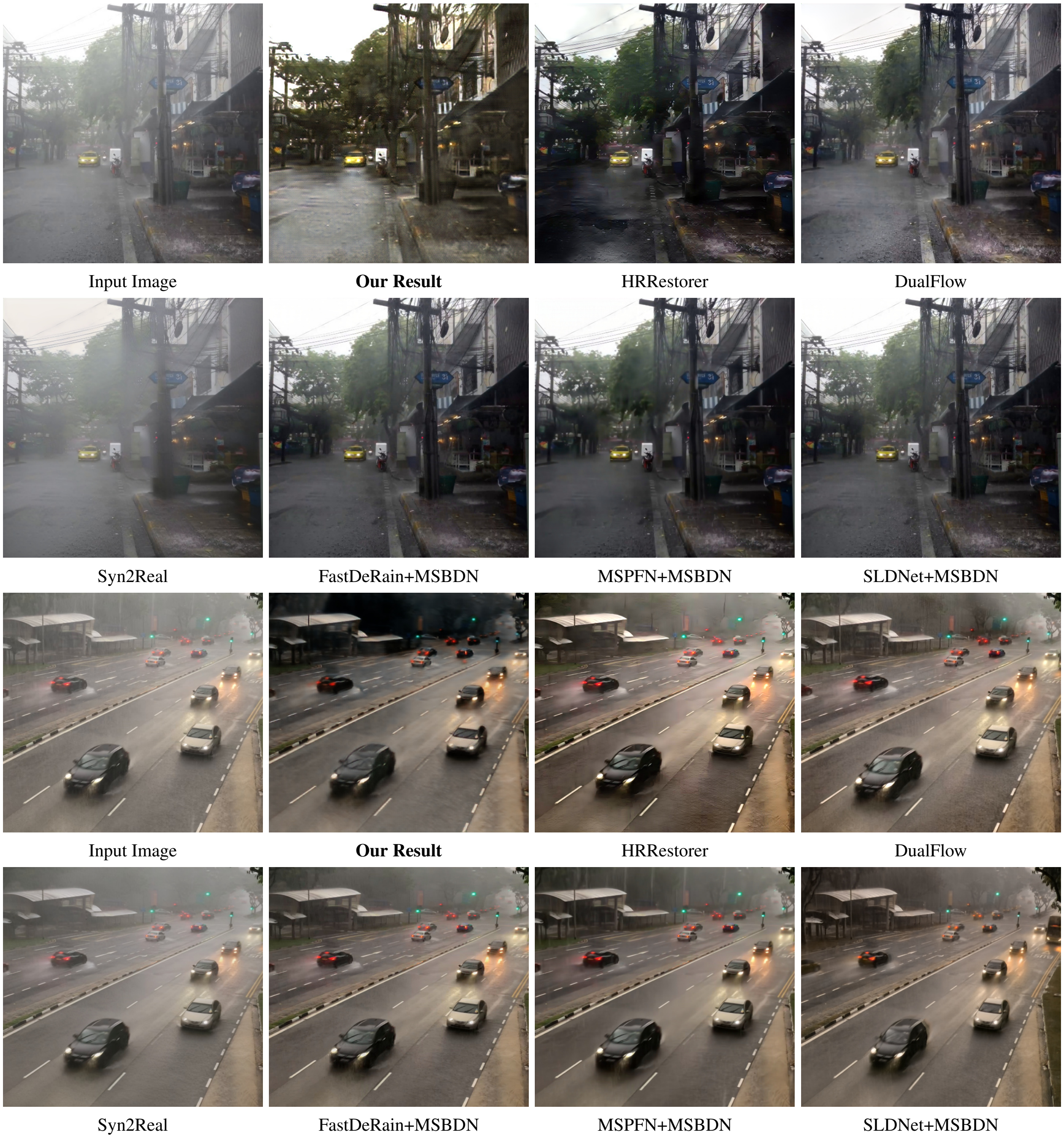
4.2. Comparison Results

We evaluate our method against the state-of-the-art deraining methods: HRRestorer [30], DualFlow [48] and Syn2Real [54]. All above methods consider both rain streak and rain accumulation effects. For the state-of-the-art deraining methods which only consider rain streak removal: FastDeRain [22], MSPFN [20] and SLDNet [51], we apply the state-of-the-art dehazing method MSBDN [10] on the outputs from them.

Fig. 5 shows the qualitative evaluation results on real rainy images with only rain streaks. As one can notice that our method qualitatively provides clear results compared with the results of other baseline methods. Not only rain streaks, but also splashes on the ground are removed. Fig. 4 shows the qualitative evaluation results on real rainy images with both rain streaks and accumulation. Results of some baseline methods still has residual rain streaks, and all baseline methods suffer from the rain accumulation. Our method gives more clear streak-removal results. Also, our method recovers background trees well with extra depth information. For the quantitative evaluation, we use 420 pairs of synthetic data. The quantitative evaluation is shown in Table 1, where our method shows better performance on both PSNR and SSIM compared to all the baseline methods.

5. Ablation Studies

To show the effectiveness of our DepthNet, we remove DepthNet from the rain accumulation removal module. The first column and second row of Fig. 6 shows the results trained without transmission-depth consistency loss. As can be seen, the rain accumulation effect is still considerably noticeable, particularly trees on the background. To prove that our DepthNet also learns from the transmission-depth consistency loss, we freeze DepthMap during training and show the depth map and the derained output in the second column of Fig. 6. Obviously, a learnable DepthNet provides



Input Image

Our Result

HRRestorer

DualFlow

Syn2Real

FastDeRain+MSBDN

MSPFN+MSBDN

SLDNet+MSBDN

Input Image

Our Result

HRRestorer

DualFlow

Syn2Real

FastDeRain+MSBDN

MSPFN+MSBDN

SLDNet+MSBDN

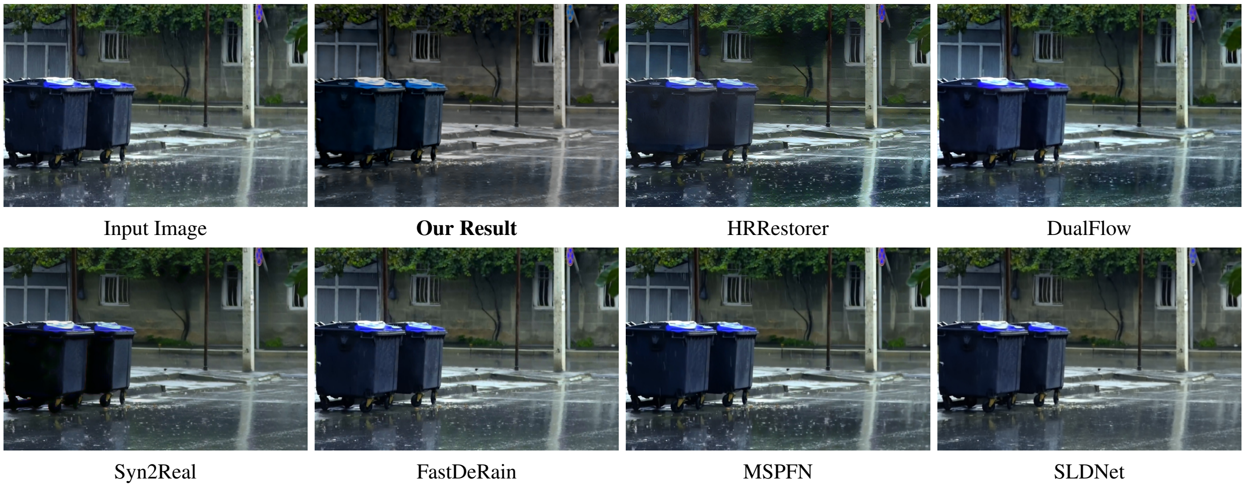
Figure 4: Qualitative comparisons with the state of the art methods on real rainy images. Zoom-in for better visualization.

a better depth map. This better depth map helps our method repress the ambiguity between depth and water-droplet density.

The third column of Fig. 6 shows results without PoseNet (hence there is no temporal consistency), and DepthNet only learns from single frame. From the results, DepthNet is benefited from the extra information of video. With the temporal consistency, the depth map has shaper edges on the yellow car and motocycle. As a result, the rained output has more natural colors.

6. Conclusion

We have introduced a video deraining method with feature-level alignment . To our knowledge, this is the first time, a video-based method is dedicated to handle both rain streaks and accumulation problems. Due to the instability of


Input Image

Our Result

HRRestorer

DualFlow

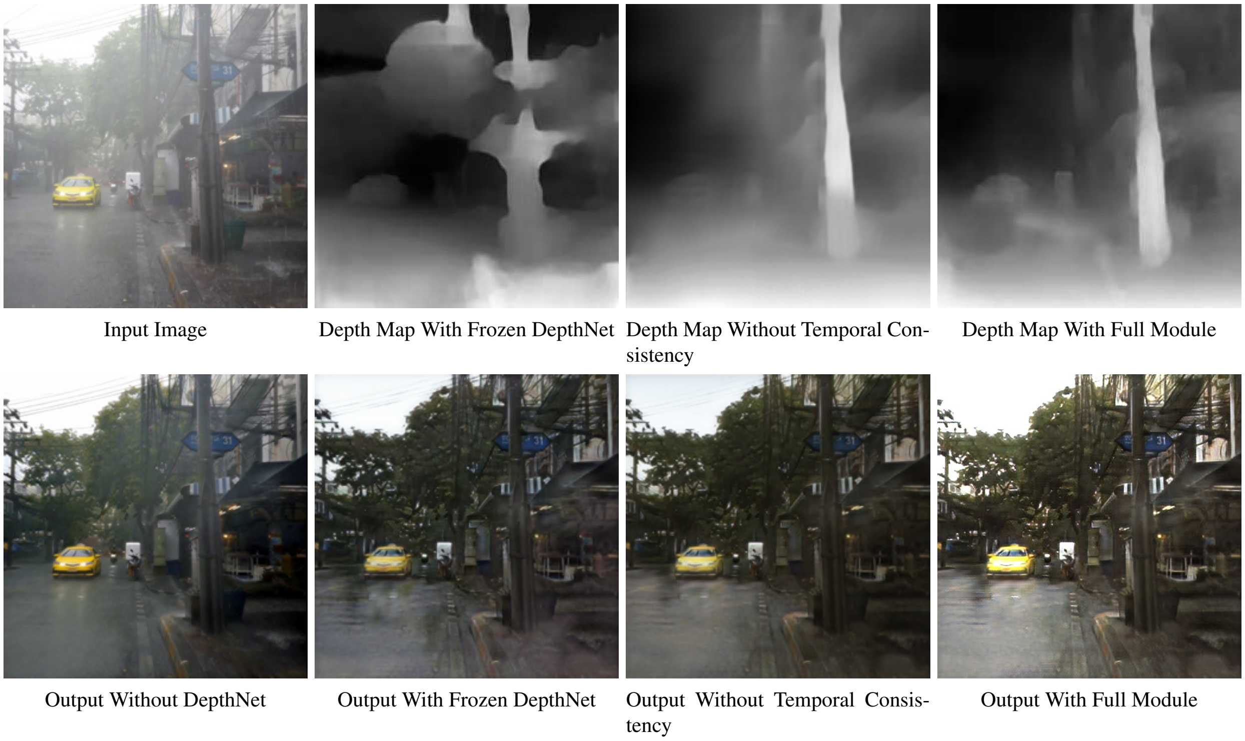
Syn2Real

FastDeRain

MSPFN

SLDNet

Figure 5: Qualitative comparisons with the state of the art methods on real rainstreak images. Zoom-in for better visualization.


Input Image

Depth Map With Frozen DepthNet

Depth Map Without Temporal Consistency

Depth Map With Full Module

Output Without DepthNet

Output With Frozen DepthNet

Output Without Temporal Consistency

Output With Full Module

Figure 6: Ablation studies on DepthNet, frozen DepthNet, and temporal consistency.

optical flow in rain videos, our method use deformable convolution layers to achieve alignment in the feature domain. To solve the depth and water-droplet ambiguity problem, we employ DepthNet and PoseNet to provide few novel losses to improve our results. Experimental results and evaluations, both quantitative and qualitative, show the effectiveness of our method.

References

[1] Peter C Barnum, Srinivasa Narasimhan, and Takeo Kanade. Analysis of rain and snow in frequency space. *International journal of computer vision*, 86(2-3):256, 2010. 2

[2] Jérémie Bossu, Nicolas Hautière, and Jean-Philippe Tarel. Rain or snow detection in image sequences through use of a histogram of orientation of streaks. *International journal of computer vision*, 93(3):348–367, 2011. 2

[3] Nathan Brewer and Nianjun Liu. Using the shape characteristics of rain to identify and remove rain from video. In *Joint*

IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR), pages 451–458. Springer, 2008. 2

[4] Yi Chang, Luxin Yan, and Sheng Zhong. Transformed low-rank model for line pattern noise removal. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1726–1734, 2017. 2

[5] Jie Chen and Lap-Pui Chau. A rain pixel recovery algorithm for videos with highly dynamic scenes. *IEEE transactions on image processing*, 23(3):1097–1104, 2013. 2

[6] Jie Chen, Cheen-Hau Tan, Junhui Hou, Lap-Pui Chau, and He Li. Robust video content alignment and compensation for rain removal in a cnn framework. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6286–6295, 2018. 2

[7] Yi-Lei Chen and Chiou-Ting Hsu. A generalized low-rank appearance model for spatio-temporally correlated rain streaks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1968–1975, 2013. 2

[8] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017. 3

[9] Sen Deng, Mingqiang Wei, Jun Wang, Yidan Feng, Luming Liang, Haoran Xie, Fu Lee Wang, and Meng Wang. Detail-recovery image deraining via context aggregation networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[10] Hang Dong, Jinshan Pan, Lei Xiang, Zhe Hu, Xinyi Zhang, Fei Wang, and Ming-Hsuan Yang. Multi-scale boosted dehazing network with dense feature fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2157–2167, 2020. 6

[11] X. Fu, B. Liang, Y. Huang, X. Ding, and J. Paisley. Lightweight pyramid networks for image deraining. *IEEE Transactions on Neural Networks and Learning Systems*, 31(6):1794–1807, 2020. 2

[12] Kshitiz Garg and Shree K Nayar. Detection and removal of rain from videos. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 1, pages I–I. IEEE, 2004. 2

[13] Kshitiz Garg and Shree K Nayar. When does a camera see rain? In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, pages 1067–1074. IEEE, 2005. 2

[14] Kshitiz Garg and Shree K Nayar. Photorealistic rendering of rain streaks. *ACM Transactions on Graphics (TOG)*, 25(3):996–1002, 2006. 2

[15] Kshitiz Garg and Shree K Nayar. Vision and rain. *International Journal of Computer Vision*, 75(1):3–27, 2007. 2

[16] Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *IEEE transactions on pattern analysis and machine intelligence*, 33(12):2341–2353, 2010. 5

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. pages 770–778, 2016. 6

[18] Xiaowei Hu, Chi-Wing Fu, Lei Zhu, and Pheng-Ann Heng. Depth-attentional features for single-image rain removal. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8022–8031, 2019. 1, 2

[19] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. pages 5967–5976, 2017. 6

[20] Kui Jiang, Zhongyuan Wang, Peng Yi, Baojin Huang, Yimin Luo, Jiayi Ma, and Junjun Jiang. Multi-scale progressive fusion network for single image deraining. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 6

[21] Tai-Xiang Jiang, Ting-Zhu Huang, Xi-Le Zhao, Liang-Jian Deng, and Yao Wang. A novel tensor-based video rain streaks removal approach via utilizing discriminatively intrinsic priors. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 4057–4066, 2017. 2

[22] Tai-Xiang Jiang, Ting-Zhu Huang, Xi-Le Zhao, Liang-Jian Deng, and Yao Wang. Fastderain: A novel video rain streak removal method using directional gradient priors. *IEEE Transactions on Image Processing*, 28(4):2089–2102, 2018. 6

[23] Li-Wei Kang, Chia-Wen Lin, and Yu-Hsiang Fu. Automatic single-image-based rain streaks removal via image decomposition. *IEEE transactions on image processing*, 21(4):1742–1755, 2011. 2

[24] Jin-Hwan Kim, Chul Lee, Jae-Young Sim, and Chang-Su Kim. Single-image deraining using an adaptive nonlocal means filter. In *2013 IEEE International Conference on Image Processing*, pages 914–917. IEEE, 2013. 2

[25] Jin-Hwan Kim, Jae-Young Sim, and Chang-Su Kim. Video deraining and desnowing using temporal correlation and low-rank matrix completion. *IEEE Transactions on Image Processing*, 24(9):2658–2670, 2015. 1, 2, 3

[26] D Kinga and J Ba Adam. A method for stochastic optimization. volume 5, 2015. 6

[27] Wei-Sheng Lai, Jia-Bin Huang, and Ming-Hsuan Yang. Semi-supervised learning for optical flow with generative adversarial networks. In *Advances in Neural Information Processing Systems*, pages 354–364, 2017. 6

[28] Guanbin Li, Xiang He, Wei Zhang, Huiyou Chang, Le Dong, and Liang Lin. Non-locally enhanced encoder-decoder network for single image de-raining. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1056–1064, 2018. 2

[29] Minghan Li, Qi Xie, Qian Zhao, Wei Wei, Shuhang Gu, Jing Tao, and Deyu Meng. Video rain streak removal by multiscale convolutional sparse coding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6644–6653, 2018. 2

[30] Ruoteng Li, Loong-Fah Cheong, and Robby T Tan. Heavy rain image restoration: Integrating physics model and conditional adversarial learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1633–1642, 2019. 1, 2, 6

[31] Siyuan Li, Wenqi Ren, Jiawan Zhang, Jinke Yu, and Xiaojie Guo. Single image rain removal via a deep decomposition–composition network. *Computer Vision and Image Understanding*, 186:48 – 57, 2019. 2

[32] Yu Li, Robby T Tan, Xiaojie Guo, Jiangbo Lu, and Michael S Brown. Rain streak removal using layer priors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2736–2744, 2016. 2

[33] Jiaying Liu, Wenhan Yang, Shuai Yang, and Zongming Guo. D3r-net: Dynamic routing residue recurrent network for video rain removal. *IEEE Transactions on Image Processing*, 28(2):699–712, 2018. 1, 2, 3

[34] Jiaying Liu, Wenhan Yang, Shuai Yang, and Zongming Guo. Erase or fill? deep joint recurrent rain removal and reconstruction in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3233–3242, 2018. 2

[35] Peng Liu, Jing Xu, Jiafeng Liu, and Xianglong Tang. Pixel based temporal analysis using chromatic property for removing rain from videos. *Computer and information science*, 2(1):53–60, 2009. 2

[36] R. Liu, Z. Jiang, X. Fan, and Z. Luo. Knowledge-driven deep unrolling for robust image layer separation. *IEEE Transactions on Neural Networks and Learning Systems*, 31(5):1653–1666, 2020. 2

[37] Yu Luo, Yong Xu, and Hui Ji. Removing rain from a single image via discriminative sparse coding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3397–3405, 2015. 2

[38] Anurag Ranjan, Varun Jampani, Lukas Balles, Kihwan Kim, Deqing Sun, Jonas Wulff, and Michael J Black. Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 12240–12249, 2019. 5, 6

[39] Dongwei Ren, Wangmeng Zuo, Qinghua Hu, Pengfei Zhu, and Deyu Meng. Progressive image deraining networks: A better and simpler baseline. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3937–3946, 2019. 1, 2

[40] Weihong Ren, Jiandong Tian, Zhi Han, Antoni Chan, and Yandong Tang. Video desnowing and deraining based on matrix decomposition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4210–4219, 2017. 2

[41] AK Tripathi and S Mukhopadhyay. Video post processing: low-latency spatiotemporal approach for detection and removal of rain. *IET image processing*, 6(2):181–196, 2012. 2

[42] Abhishek Kumar Tripathi and Sudipta Mukhopadhyay. A probabilistic approach for detection and removal of rain from videos. *IETE Journal of Research*, 57(1):82–91, 2011. 2

[43] G. Wang, C. Sun, and A. Sowmya. Erl-net: Entangled representation learning for single image de-raining. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5643–5651, 2019. 2

[44] Hong Wang, Qi Xie, Qian Zhao, and Deyu Meng. A model-driven deep neural network for single image rain removal. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[45] Tianyu Wang, Xin Yang, Ke Xu, Shaozhe Chen, Qiang Zhang, and Rynson WH Lau. Spatial attentive single-image deraining with a high quality real rain dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12270–12279, 2019. 1, 2

[46] Yinglong Wang, Yibing Song, Chao Ma, and Bing Zeng. Rethinking image deraining via rain streaks and vapors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2

[47] Wei Wei, Lixuan Yi, Qi Xie, Qian Zhao, Deyu Meng, and Zongben Xu. Should we encode rain streaks in video as deterministic or stochastic? In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2516–2525, 2017. 2

[48] Wenhan Yang, Jiaying Liu, and Jiashi Feng. Frame-consistent recurrent video deraining with dual-level flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1661–1670, 2019. 1, 2, 3, 6

[49] Wenhan Yang, Jiaying Liu, Shuai Yang, and Zongming Guo. Scale-free single image deraining via visibility-enhanced recurrent wavelet learning. *IEEE Transactions on Image Processing*, 28(6):2948–2961, 2019. 1, 2

[50] Wenhan Yang, Robby T Tan, Shiqi Wang, Yuming Fang, and Jiaying Liu. Single image deraining: From model-based to data-driven and beyond. *IEEE Transactions on pattern analysis and machine intelligence*, 2020. 2

[51] Wenhan Yang, Robby T Tan, Shiqi Wang, and Jiaying Liu. Self-learning video rain streak removal: When cyclic consistency meets temporal correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1720–1729, 2020. 1, 2, 3, 4, 6

[52] Wenhan Yang, Shiqi Wang, Dejia Xu, Xiaodong Wang, and Jiaying Liu. Towards scale-free rain streak removal via self-supervised fractal band learning. In *AAAI*, pages 12629–12636, 2020. 1, 2

[53] Rajeev Yasarla and Vishal M Patel. Uncertainty guided multi-scale residual learning-using a cycle spinning cnn for single image de-raining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8405–8414, 2019. 1, 2

[54] Rajeev Yasarla, Vishwanath A Sindagi, and Vishal M Patel. Syn2real transfer learning for image deraining using gaussian processes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2726–2736, 2020. 2, 6

[55] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015. 3

[56] Huangying Zhan, Ravi Garg, Chamara Saroj Weerasekera, Kejie Li, Harsh Agarwal, and Ian Reid. Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 340–349, 2018. 5

[57] He Zhang and Vishal M Patel. Density-aware single image de-raining using a multi-stream dense network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 695–704, 2018. 2

[58] Xiaopeng Zhang, Hao Li, Yingyi Qi, Wee Kheng Leow, and Teck Khim Ng. Rain removal in video by combining temporal and chromatic properties. In *2006 IEEE international conference on multimedia and expo*, pages 461–464. IEEE, 2006. 2

[59] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1851–1858, 2017. 6

[60] Lei Zhu, Chi-Wing Fu, Dani Lischinski, and Pheng-Ann Heng. Joint bi-layer optimization for single-image rain streak removal. In *Proceedings of the IEEE international conference on computer vision*, pages 2526–2534, 2017. 2