Learning to Remove Rain in Video With Self-Supervision

Wenhan Yang, *Member, IEEE*, Robby T. Tan, *Member, IEEE*, Shiqi Wang, *Senior Member, IEEE*, Alex C. Kot, *Fellow, IEEE*, and Jiaying Liu, *Senior Member, IEEE*

Abstract—In heavy rain video, rain streak and rain accumulation are the most common causes of degradation. They occlude background information and can significantly impair the visibility. Most existing methods rely heavily on the synthetic training data, and thus raise the domain gap problem that prevents the trained models from performing adequately in real testing cases. Unlike these methods, we introduce a self-learning method to remove both rain streaks and rain accumulation without using any ground-truth clean images in training our model, which consequently can alleviate the domain gap issue. The main idea is based on the assumptions that (1) adjacent clean frames can be aligned or warped from one frame to another frame, (2) rain streaks are distributed randomly in the temporal domain, (3) the rain streak/accumulation related variables/priors can be inferred reliably from the information within the images/sequences. Based on these assumptions, we construct an augmented Self-Learned Deraining Network (SLDNet+) to remove both rain streaks and rain accumulation by utilizing temporal correlation, consistency, and rain-related priors. For the temporal correlation, our SLDNet+ takes rain degraded adjacent frames as its input, aligns them, and learns to predict the clean version of the current frame. For the temporal consistency, a new loss is designed to build a robust mapping between the predicted clean frame and non-rain regions from the adjacent rain frames. For the rain-streak-related prior, the rain streak removal network is optimized jointly with motion estimation and rain region detection; while for the rain-accumulation-related prior, a novel non-local video rain accumulation removal method is developed to estimate the accumulation-lines from the whole input video and to offer better color constancy and temporal smoothness. Extensive experiments show the effectiveness of our approach, which provides superior results compared with the existing state of the art methods both quantitatively and qualitatively. The source code will be made publicly available at: https://github.com/flyywh/CVPR-2020-Self-Rain-Removal-Journal.

Index Terms—Multi-frame image, video rain removal, physical recovery guidance, adversarial learning

1 Introduction

Rain frequently leads to visual degradation in images and videos. The most common form of degradation caused by rain is rain streaks, which can partially obstruct a background scene, alter the image's appearance, and distort the scene. Besides rain streaks, rain produces a veiling effect, also known as rain accumulation, which is visually similar to fog/haze. Rain accumulation is observable in heavy rain, as dense distant falling raindrops cannot be seen individually, and a considerable amount of water particles are present in the atmosphere.

Based on rain spatial appearances, a few methods (*e.g.*, [26], [31], [41], [46]) focus on decomposing rain streaks from the clean background. Sparse representation [41], frequency domain representation [31], Gaussian mixture models [36], and deep networks [19], [61], provide some constraints and features to separate the rain streak layer from the background layer. Existing video-based approaches (*e.g.*, [2], [4], [5], [11], [15], [21], [23], [24], [72]) utilize temporal redundancies and contexts in addition to spatial correlations. The earliest methods [21], [23], [24] exploit the physical and photometric properties of rain streaks, such as their directional and chromatic properties. To remove rain streaks from rain frames, later methods [10], [33], [38], [39], [59] employ the temporal dynamics of videos, such as the consistency of background layers and the randomness of rain positions in the temporal domain. A few deep learning methods have also been introduced (*e.g.*, [10], [33], [38], [39], [59]). Convolutional neural networks (CNN) and other deep learning models, such as recurrent neural networks [38], [39] and convolutional sparse coding [33], are used to remove rain streaks. Explicit temporal correlation [10], scale variance of rain streaks [33], and motion contexts [38], [39] are a few examples of the developed priors.

- *Wenhan Yang and Alex C. Kot are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798. E-mail: yangwenhan@pku.edu.cn, eackot@ntu.edu.sg.*
- *Robby T. Tan is with the Yale-NUS College, Department of Electrical and Computer Engineering, National University of Singapore, Singapore 119077. E-mail: tanrobby@gmail.com.*
- *Shiqi Wang is with the Department of Computer Science, City University of Hong Kong, Hong Kong. E-mail: shiqwang@cityu.edu.hk.*
- *Jiaying Liu is with the Wangxuan Institute of Computer Technology, Peking University, Beijing 100871, China. E-mail: liujiaying@pku.edu.cn.*

Manuscript received 30 April 2021; revised 14 April 2022; accepted 11 June 2022. Date of publication 4 July 2022; date of current version 5 February 2024. This work was supported in part by NTU-PKU Joint Research Institute (a collaboration between the Nanyang Technological University and Peking University that is sponsored by a donation from the Ng Teng Fong Charitable Foundation), in part by the National Natural Science Foundation of China under Grant 62172020, and in part by the research achievement of Key Laboratory of Science, Techonology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology). The work of Wenhan Yang was supported in part by Wallenberg-NTU Presidential Postdoctoral Fellowship. The work of Robby T. Tan was supported in part by MOE2019-T2-1-130. (Corresponding author: Jiaying Liu.)
Recommended for acceptance by L. Liu, T. Hospedales, Y. LeCun, M. Long, J. Luo, W. Ouyang, M. Pietikäinen, and T. Tuytelaars.
Digital Object Identifier no. 10.1109/TPAMI.2022.3186629

0162-8828 © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

Generally, the CNN-based methods outperform existing non-deep-learning methods [62]. However, these fully supervised methods significantly rely on synthetic paired videos (rain/rain-free videos). Since, to obtain real paired videos is intractable, particularly when there are moving objects or background in the scenes. Unfortunately, there are significant domain gaps between synthetic and real rain images. As rain appearances are diverse in scales, shapes, orientations, or even forms (both rain streaks and rain accumulation), a synthetic dataset cannot include all various types of rains properly and completely. Moreover, a synthetic dataset can only provide limited background images, which makes a network tend to fail in reconstructing textures and details of clean images in unseen backgrounds. Consequently, when the training relies heavily on synthetic data, these gaps can degenerate the performance. Moreover, existing video-deraining methods do not handle rain accumulation, which is commonly present in heavy rain.

In this paper, we develop a self-learning method that does not require clean ground-truth videos during the training process. Hence, our method does not use any synthetic rain data in training our model. We jointly model the intrinsic constraints of natural video and the priors of rain streaks in a novel augmented **S**elf-**L**earned **D**eraining **Net**work (**SLDNet+**) by considering the mechanism of learning from noisy data (*i.e.*, adjacent rainy frames). Our SLDNet+ also exploits the temporal correlation and consistency in our output frames.

In SLDNet+, we also include a rain-related prior, *i.e.*, jointly optimized motion estimation and rain region detection. Using robust motion estimation, we can align the input rain frames, and make the constraint (the consistency among adjacent frames) more effective. The jointly optimized rain region detection also guides our SLDNet+ to only focus on manipulating rain regions while suppressing the effect of rain streaks in the noisy labels. To make SLDNet+ learn from the noisy labels more robustly, we design a constraint in a form of a reweighted L_1 loss that focuses more on the sparse differences, *i.e.* rain streaks, instead of scattered errors caused by inter-frame misalignment. Subsequently, we augment our model by shuffling rain input frames to obtain an enriched model that preserves the details and adapts to more diverse rain streaks in real scenarios. To remove the rain accumulation (or the rain veiling effect), we build a new non-local video rain accumulation removal model to estimate the accumulation-lines from the whole input frames, and offer better color constancy and temporal smoothness without using synthetic data.

In summary, our contributions are as follows.

- We propose a self-learning video rain removal method relying only on input rain video during both training and testing stages. To the best of our knowledge, it is the first time in the deraining literature such an effort has been made. With both temporal correlation and consistency, the proposed deep network is successful to learn from noisy labels, *i.e.* aligned adjacent rainy frames.
- We incorporate priors of rain videos, *i.e.* rain location and background motion information to suppress the influence of rain streaks in the adjacent frames, and to guide the model to only focus on manipulating the rain regions. These priors/constraints can possibly inspire further exploration in self-learning video deraining.
- To make our framework learn from noisy labels, we further develop a more robust training constraint in the form of reweighted L_1 loss. The loss leads to sparse differences, and therefore pays more attention to rain streaks instead of the scattered errors caused by the inter-frame misalignment.
- We also propose a new non-local video rain accumulation removal method that estimates the accumulation-lines from the whole input frames with better color constancy and temporal smoothness, without using synthetic data.

Extensive experiments demonstrate the effectiveness of our approach in quantitative and qualitative evaluations and the rationality of our design in various ablation studies.

This work is an extension of our paper, SLDNet, published in CVPR'2020 [63]. Unlike the conference version, our new contents are as follows: 1) We introduce a reweighted L_1 loss as a training constraint, which makes the model learning more robust to the noisy labels, *i.e.* aligned adjacent rain frames. 2) With the new loss and constraint, we revisit our model design systematically. We found the one-stage framework provides better results than our previous two-stage framework [63]. Furthermore, with the one-stage framework, we carefully tune the hyper-parameters related to the rain mask. With the newly designed architecture, more robust loss, better hyper-parameters, our new method achieves 3.47 dB and 0.0210 gains in PSNR and SSIM, respectively, with only half the parameters of our previous conference version for deraining model [63] on *NTURain* [10], which even outperforms the recent semi-supervised methods [67] using both synthetic paired and real data in the training phase. 3) Our previous method [63] handles only rain streaks. In this work, we handle both rain streaks and rain accumulation. A nonlocal-based video accumulation removal method is developed to significantly improve the visibility of our result when accumulation exists. 4) We perform more experiments, design analysis, and ablation studies to demonstrate the advantages of our SLDNet+ for video deraining, and show the effectiveness of our method over existing methods.

The rest of our paper is organized as follows. Section 2 conducts a comprehensive literature review. Section 3 presents our targeted rain synthesis model as well as the constraints and motivations that help rain removal. Section 4 proposes our self-learned deraining network SLDNet+ in detail. Section 5 develops a non-local rain accumulation removal method from videos to significantly improve visibility. In Section 6, experimental configurations and results are presented. The concluding remarks are given in Section 7.

2 Related Work

2.1 Single Image Rain Removal

The additive composite rain model considers a rain image as the composition of a clean background layer and a rain streak layer. The added rain streaks lead to visual quality degradation of the original image. The earlier methods adopt the optimization framework injected with rain and background-related priors for single-image rain removal,

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

e.g. frequency separation jointly with sparse coding [31], discriminative sparse coding [41], convolutional sparse coding [69], bi-layer optimization [75], Gaussian mixture model [36] and directional group sparse model [12]. However, due to the limited learning capacity of these hand-designed priors/constraints, the optimization-based methods cannot produce satisfying visual results, especially when heavy rain is presented.

With the development of deep learning, many deep network-based approaches have been proposed with significantly superior results. Fu *et al.* [19] introduced a three-layer convolutional neural network for rain removal. Yang *et al.* [61] injected the binary mask of the detected rain streaks to guide the learning of a multi-branch ResNet. Zhang *et al.* [70] persuaded the network to learn and perceive the rain density information and adopted a multi-path densely connected network for rain removal. Li *et al.* [35] and Ren *et al.* [43] performed rain streak removal progressively via the recurrent mechanism, and the information is shared and connected among different stages with the recurrent units. Some works [28], [50], [51], [64], [65], [68] made full use of the multi-scale redundancy to further improve the deraining performance. In [54], Wang *et al.* built a spatial attentive network to remove rain streaks from local to global. In [13], Deng *et al.* designed a two-stage context aggregation network to effectively remove rain and well restore details simultaneously. In [52], a novel deep network that is interpretable from the optimization view is introduced by integrating convolutional sparse coding and deep learning. In [7], Chen and Li introduced the feedback mechanism and created a new error detection and feature compensation method to address model errors that lead to uncertainty and degraded embedding quality. In [20], Fu *et al.* utilized two graphs for global relational modeling and reasoning to facilitate single image deraining. In [53], Wang *et al.* made efforts in removing rain streaks by exploring a more efficient way to synthesize rainy images to augment the training data. In [8], transformers are introduced for a series of low-level vision tasks and also offer impressive deraining results. Although more advanced network architectures lead to more powerful capacities of separating rain/background signals from their mixture, these methods have two limitations: 1) they heavily rely on the synthetic data, which inevitably has a domain gap with the real one, the trained models might not adapt the real rain streaks; 2) they ignore the rain accumulation, which is quite common especially when there is heavy rain.

To address the first issue, semi-supervised learning [37], [56], [66], adversarial learning [71], [74], self-supervised learning [60], neural reorganization [58], and continual learning [73] are introduced to embed the prior knowledge of real rain images to improve the generalization performance of the rain removal performance. There are also several works that focus on removing rain accumulation [25], [34], [61]. In our work, we focus on video deraining, and aim to address the generalization problem and accumulation removal in videos. Different from single-image rain removal methods, the additional temporal redundancy in videos provides the effective prior and constraint for better generalization while it is more challenging to deal with accumulation in videos as temporal inconsistency is easily incurred.

2.2 Video Rain Removal

Compared with single-image rain removal, video rain streak removal is capable of utilizing temporal correlation and dynamics to detect and remove rains. Garg and Nayar propose the seminal work of video rain modeling [23] and rain streak removal methods [21], [22], [24]. Later approaches dig deep to see the intrinsic priors rain streak and normal background signals, *i.e.* the shape, size, and orientation of rain streaks [4], [5], chromatic and temporal characteristic of rain [40], [72], Fourier domain feature [2], phase congruency features [45], directional prior of rain streaks [30], spatio-temporal redundancy of patch groups [11], Bayes rain detector [48], [49], Gaussian mixture model [9], detection and refinement in two stages based on SVM [32], matrix decomposition [44], patch-based mixtures of Gaussian [57].

Recently, deep-learning based methods arise, with significant improvements in signal manipulation capacities and their flexibility in injecting priors and constraints. In [33], a multiscale convolutional sparse coding is adopted to remove rain streaks with various scales. Chen *et al.* [10] firstly segmented superpixels from rain frames. Then, they estimated rain-free superpixels by applying the consistency constraint. After that, a CNN is further utilized to compensate for lost details and add background textures in the final results. In [38], a recurrent neural network is designed to seamlessly integrate rain degradation classification, rain removal and background detail reconstruction in a unified framework. In [39], Liu *et al.* built a hybrid rain model for modeling both rain streaks and occlusions. Then, a deep dynamic routing residue recurrent network is constructed with the motion segmentation context information. In [59], Yang *et al.* developed a two-stage recurrent network incorporated with dual-level flow regularization to estimate the related physics variables for deraining restoration by inverting the rain synthesis model. In [67], Yue *et al.* proposed a new semi-supervised video deraining method and adopted a dynamical rain generator to fit the rain layer to better depict the intrinsic characteristics of the rain.

Previous works are either model-driven, constructed with hand-crafted features/constraints, or learning-based ones, relying on synthetic paired data and ground truth clean frames. In our work, we make the exploration of the possible architectures and priors with only self-supervision and develop a trainable video deraining network not relying on synthesized paired data.

3 Rain Model and Self-Learning Constraints

3.1 Rain Model

The common rain model considers a rain image as a linear combination of rain streaks and background:

$$I = B + R, \tag{1}$$

where B is the layer without rain streaks, and R is the rain streak layer. I is the captured image with rain streaks.

In the real world, however, rain appears in the form of not only individual streaks but also the accumulation of rain water drops and particles. In a distant scene, individual rain streaks are not observable. These rain streaks of various shapes and directions overlap with each other mixed up

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

with water drops and particles, forming the appearance of rain accumulation, which can be modeled using the Koschmieder model [1], [42], [47]. Therefore, our rain model considering both rain streaks and accumulation is expressed as:

$$I = \alpha(B + R) + (1 - \alpha)A, \tag{2}$$

$$= \alpha B + R' + (1 - \alpha)A, \tag{3}$$

where α is the atmospheric transmission, measuring how much background information can go through the accumulation. R' is the rain streak in the rainy image with rain accumulation. A is the global atmospheric light.

For video, with an added temporal indicator t, a video rain synthesis model can be expressed as:

$$I_t = \alpha_t B_t + R'_t + (1 - \alpha_t)A_t, t = 1, 2, \ldots, N, \tag{4}$$

where t and N denote the current time-step and the total number of video frames, respectively. The rain streak R_t is assumed to be independent and identically distributed random samples. A_t is a global variable that changes little with different t and α_t is continuous along the temporal dimension.

In our method, to estimate B_t based on I_t, we first estimate $I_t - R'_t$ with a self-learning deraining network (Sec. 4). After that, a non-local video accumulation removal method (Sec. 5) is utilized to further infer B_t based on $I_t - R'_t$.

3.2 Temporal Cyclic Consistency for Self-Learned Rain Streak Removal

We discuss the constraints we use in our framework in the following paragraphs.

Temporal Correlation. Assuming adjacent background frames are highly correlated and thus can be aligned or warped from one to another, and rain streaks are randomly distributed along the temporal dimension, *an ideal deraining method should be able to extract the background information from the adjacent frames, while removing out the rain streaks*.This implies that if adjacent frames are all well aligned, we can change their temporal orders (by swapping one frame with another frame), our method should be able to remove rain-streaks and predict a rain-streaks free frame. With this in mind, we can augment our network by replacing the current central frames with one of its adjacent frames randomly in the training stage. In this way, more rain streak combinations are presented in the input, and the network's capacity to handle more diverse rain patterns can be improved.

Temporal Consistency. As rain-free background layers make a smooth and continuous transition in the temporal domain, aligned adjacent background layers have only small differences. On the contrary, even though good motion prediction and alignment are attained, due to the random presence of rain streaks, the well aligned rain layers will still have large differences. Hence, *if the network is constrained to produce temporally consistent outputs after alignment, it will benefit rain streak removal*.Unfortunately, when there are large movements, the alignment might not be accurate, and there might be significant differences between frames. Thus, the temporal consistency constraint might fail in this situation. As a result, we need to use motion estimation as a part of our optimization objective in our method.

Rain-Related Information. We aim to inject useful information to guide the rain streak removal process in addition to the aforementioned constraints:

- We incorporate the rain-dependent features, namely the rain mask, as a part of the loss functions. This helps our model deal with rain layers in a region-adaptive manner, *i.e.* only performing rain streak removal in rain regions of the current rain frame, while solely introducing non-rain regions of adjacent frames to form the guidance for deraining.
- Optical flow, is easily impaired by rain streaks, since it normally assumes to be extracted from clean frames. Fortunately, rain removal and optical flow can help each other, if one of them can be improved during the process. For this reason, integrating optical flow estimation in our overall optimization function will benefit rain streak removal.

Sparse and Robust Estimation. Since we do not have the paired data in the training, the training losses are created based on the temporal correlation and consistency by relying on the aligned adjacent rain frames. These losses include two kinds of errors. First, the predicted frame might include the rain streak signal. This kind of error will be distributed sparsely but sharply. Second, the predicted frame might have inaccurate background contents, which causes the predicted background contents to be misaligned. This kind of error is likely to be dense but have small values. As the commonly used L_2 norm can only lead to non-sparse errors, the results inevitably suffer from impaired backgrounds or remaining rain-streaks. In our work, we pursue a sparser estimation of the estimated errors. When the errors become sparser, the temporally misaligned errors of backgrounds will have insignificant effects.

4 Self-Learning Rain Streak Removal

Fig. 1 shows the architecture of our **S**elf-**L**earned **D**eraining **Net**work (**SLDNet+**). Our warping operation (Fig. 1a) employs optical flow [14] to obtain motion information, enabling us to align the input frames. The optical flow is optimized jointly with the rain streak removal module. Rain reMoval Network Network (RMNet) (Fig. 1b) takes as input the current rain frame and the adjacent rain frames. It outputs the predicted rain-free frames based on the inter-frame consistency loss. The loss functions (Fig. 1c) are employed with the rain mask to suppress the impact of rain streaks. The mask guides the network to pay attention to non-rain regions, making the learning process more robust. In other words, our SLDNet+ estimates optical flow, warps multiple rainy frames for performing rain removal, and imposes the temporal consistency as losses.

4.1 Optical Flow Estimation and Optimization

We estimate optical flow and utilize it to align the adjacent input rain frames to the central frame. We introduce $G(\cdot)$ to represent the optical flow estimation process:

$$C_{i \to j} = G(I_i, I_j), \tag{5}$$

where $C_{i \to j}$ represents the flow from the i-th frame to the j-th frame. Since, we do not have the clean frame, the optical

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

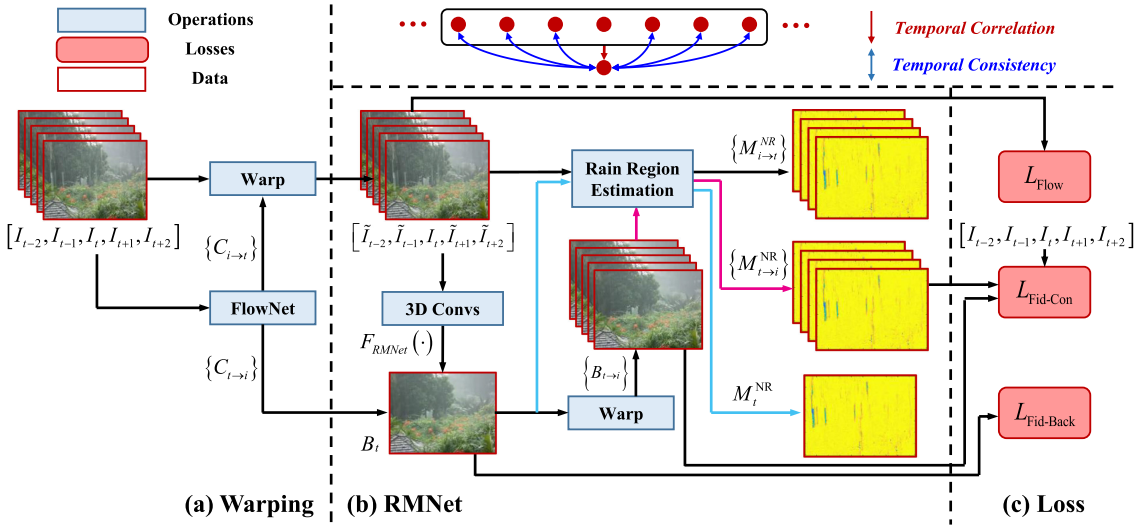


Fig. 1. The framework of our proposed augmented Self-Learning Deraining Network (SLDNet+). 1) *Warping* module aligns the neighboring frames to the central one and the central result to the adjacent frames. Successive modules utilize *temporal correlation* and *consistency* to create the mapping to restore rain video frames into clean ones. 2) *Rain Removal Network* (*RMNet*) predicts the clean central frame with the adjacent rain video frames to utilize temporal correlation. 3) RMNet is trained with the temporal consistency constraint, which persuades the generated central frame close to the adjacent frames after the alignment, with a carefully designed loss supervised by the input rain frames. The **red** arrows and **blue** arrows denote the information flow related to temporal correlation and consistency. **Black**, **cyan**, and **magenta** lines denote the data flow in our SLDNet+. The different colors of black, cyan, and magenta are used to distinguish the data flow related to the rain region estimation module.

flow is estimated from the rain frames. Subsequently, we warp the i-th frame to the j-th frame:

$$\widetilde{I}_{i\to j} = W(I_i, C_{i\to j}), \tag{6}$$

For simplicity, we adopt $\widetilde{I}_i$ to represent $\widetilde{I}_{i\to j}$ in Fig. 1 since j is set to t constantly in the whole process. Since, optical flow estimation might be affected by rain, we optimize a pre-trained optical flow network further using rain videos. The optimization criterion is that, after the warping, the non-rain regions of the aligned rain frames must be well corresponded and be identical, expressed as:

$$\begin{aligned}\mathcal{L}_{\text{Flow}} = &\sum_{i=t-s}^{t-1} \left\| M_{i\to t}^{\text{NR}} \left(\widetilde{I}_{i\to t} - I_t \right) \right\|_2^2 \\ &+ \sum_{i=t+1}^{t+s} \left\| M_{i\to t}^{\text{NR}} \left(\widetilde{I}_{i\to t} - I_t \right) \right\|_2^2,\end{aligned} \tag{7}$$

where t indexes the central frame, $2s+1$ signifies the length of the window size used by the deraining model. $M_{i\to t}^{\text{NR}}$ denotes the mask estimated by the warped versions of adjacent frames to the central rain frame and the central rain frame I_t. The calculation of $M_{i\to t}^{\text{NR}}$ will be introduced in the following section.

4.2 Rmnet

Based on the *temporal correlation*, we adopt $F_{\text{RMNet}}(\cdot)$ to represent the rain removal process of the aligned versions of successive frames $\widetilde{\Phi}_{t,s}$, including the aligned versions of adjacent frames as well as the current rain frame as the input:

$$\widetilde{\Phi}_{t,s} = \left\{ \widetilde{I}_{(t-s)\to t}, \ldots, \widetilde{I}_{(t-1)\to t}, I_t, \widetilde{I}_{(t+1)\to t}, \ldots, \widetilde{I}_{(t+s)\to t} \right\}.$$

Meanwhile, the deraining model is trained with *temporal consistency*, which enforces the estimated frame to be consistent with the estimated rain-free regions of all aligned adjacent rain frames:

$$\hat{B}_t = F_{\text{RMNet}} \left(\widetilde{\Phi}_{t,s}^{I} \right). \tag{8}$$

The consistency loss function is defined as follows,

$$\mathcal{L}_{\text{Fid-Con}} = \sum_{i=\{t-s,\ldots,t+s\}/t} \frac{1}{2s} l_{\text{fid}}^{p} \left(M_{t\to i}^{\text{NR}} \widetilde{B}_{t\to i}, M_{t\to i}^{\text{NR}} I_i \right), \tag{9}$$

$$\widetilde{B}_{t\to i} = W\left(\hat{B}_t, C_{t\to i} \right), \tag{10}$$

where $M_{t\to i}^{\text{NR}}$ is the estimated mask denoting the non-rain region of the adjacent rain frame I_i. The details of calculating $M_{t\to i}^{\text{NR}}$ are discussed in the following sections. Function $l_{\text{fid}}^{p}(\cdot)$ is our proposed robust measure focusing on sparse and large errors between the estimated rain-free frame and the adjacent rain frames after alignment.

4.3 Rain Region Estimation

Since the frames adopted in our training phase are contaminated by rain streaks, as shown in Eqs. (9) and (7), we intend to exclude the effect of this noise from the training guidance obtained from the adjacent frames with the knowledge of the estimated rain streaks. With adequate training time, we can obtain $\hat{B}_t$. Subsequently, the masks of the non-rain regions $M_{t\to i}^{\text{NR}}$ and $M_{i\to t}^{\text{NR}}$ are inferred from the forward and backward warping operations at the current time-step t and employed as soft masks, to indicate whether pixels are occluded by rain streaks. In detail, $M_{t\to i}^{\text{NR}}$ and $M_{i\to t}^{\text{NR}}$ are inferred as follows:

$$M_{t\to i}^{\text{NR}} = \exp\left\{ -\frac{\left(h_{\text{ReLU}} \left(I_i - \widetilde{B}_{t\to i} \right) \right)^2}{\omega} \right\}, \forall i \neq t, \tag{11}$$

$$M_{i\to t}^{\text{NR}} = \exp\left\{ -\frac{\left(h_{\text{ReLU}} \left(\widetilde{I}_{i\to t} - I_t \right) \right)^2}{\omega} \right\}, \forall i \neq t, \tag{12}$$

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

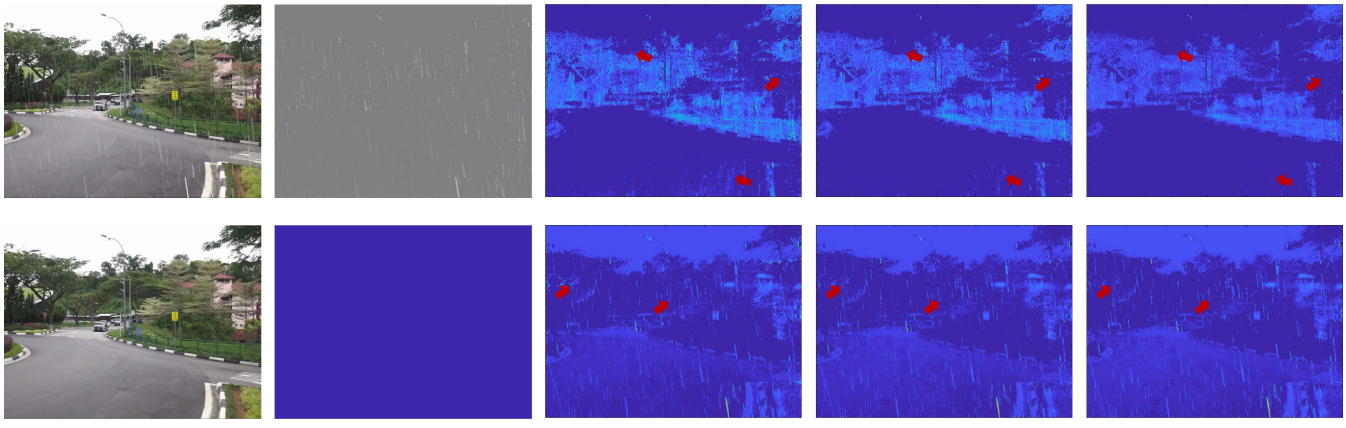


Fig. 2. Visual results of a frame in *a4* sequence of *NTURain* processed by Our SLDNet+ with different fidelity losses. Top panel: (a) rain frame; (b) rain streak map; (c)-(e): the estimated rain streak maps constrained by l_2, l_1 and reweighted l_p norms, respectively. Bottom panel: (a) clean background; (b) zero map; (c)-(e): the difference between the clean background frame and the estimated frames constrained by l_2, l_1 and reweighted l_p norms, respectively. The results show that, reweighted l_p norm leads to less background information in the estimated rain streak maps (red arrows in the top panel), and less background degradation (red arrows in the bottom panel). Zoom-in for better visualization.

where ω denotes the parameter that controls the shape of the exponential function of Eqs. (11) and (12), and can act as a threshold to decide what signals weighted by $M^{\mathrm{NR}}_{t\to i}$ and $M^{\mathrm{NR}}_{i\to t}$ can pass through. $h_{\mathrm{ReLU}}(\cdot)$ denotes the rectified linear unit function to only get the positive values past, as it is commonly observed that rain streaks are positive.

When looking into Eq. (9), one can observe that, if the mask is adopted, the effectiveness of the loss heavily relies on the motion estimation accuracy and whether the content among different frames really corresponds. However, when rain streaks are presented in video frames, the motion estimation accuracy and perfect content alignment cannot be guaranteed in most cases. The introduction of the rain streak guidance can further augment the capacity of the loss from the following two aspects.

First, as indicated in Eq. (9), we exclude the rain streaks' influence from adjacent rain frames as training labels and let the model learn useful priors from only rain-free regions. Second, the estimated mask of the current frame also guides our model to learn to manipulate only non-rain regions of the current-time frame. Based on this, the fidelity loss that leads to preserving the non-rain background structures in the current frame is defined as:

$$\mathcal{L}_{\text{Fid-Back}} = l^p_{\text{fid}}\big(M^{\mathrm{NR}}_t \hat{B}_t, M^{\mathrm{NR}}_t I_t\big), \tag{13}$$

$$M^{\mathrm{NR}}_t = \exp\left\{-\frac{\big(h_{\mathrm{ReLU}}(I_t - \hat{B}_t)\big)^2}{\omega}\right\}. \tag{14}$$

In summary, the emergence of the rain masks in the loss regularizes the model training in handling the rain and non-rain regions adaptively. For non-rain regions, it is expected that the deraining model should not change anything, preserving the information of input rain frames. For rain regions, the generated output is constrained to be consistent with the corresponding rain-free regions in the warped rain frames.

4.4 Sparse and Robust Estimation

As discussed in Sec. 3.2, a fidelity metric of $l_{\text{fid}}(\cdot)$ that pursues sparser solutions leads to a more effective constraint, focusing more on removing rain instead of enforcing the consistency of misaligned backgrounds. The commonly used norm for the fidelity metric takes the form:

$$l^p_{\text{fid}}(x, y) = \Big(\sum |x - y|^p\Big)^{\frac{1}{p}}, \tag{15}$$

where p is usually set to 1 and 2, which corresponds to mean absolute error (MAE) and mean squared error (MSE), respectively. Based on the sparse representation theory [16], if we can adopt the l_p norm with a small p, our estimation pays more attention to punishing on rain streak signals in the estimated residue (the difference between our estimated clean background and the constructed target) instead of the temporally misaligned frames.

To achieve this goal in an end-to-end optimized framework, we apply the reweighted l_p norm [17] to encourage sparsity in the estimated residue:

$$l^p_{\text{fid}}(x, y) = \omega \sum |x - y|, \tag{16}$$

$$\omega = \frac{1}{\sum |x - y|^{1-p} + \epsilon}, \tag{17}$$

where ϵ is a small positive number to stabilize the numerical calculation. Note that, ω is a given tensor that is not involved in the back-propagation optimization process. We visualize our results that are constrained by different p in Fig. 2. From the estimated rain streak maps and the difference between the clean background frame and the estimated ones, we can observe that, reweighted l_p norm leads to less background information in the estimated rain streak maps, and less background degradation.

Overall Loss Function. The whole training loss function consists of all the aforementioned loss functions:

$$\mathcal{L}_{\text{All}} = \mathcal{L}_{\text{Flow}} + \lambda_{\text{T}}\mathcal{L}_{\text{Fid-Con}} + \lambda_{\text{B}}\mathcal{L}_{\text{Fid-Back}}, \tag{18}$$

where the weighting parameters λ_{T} and λ_{B} balance the importance of three terms. This loss will lead the network to perform rain streak removal from rain videos.

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

4.5 Joint Optimization With Rain-Related Priors

Our framework performs video rain removal jointly with a rain-related prior estimation by considering the following factors:

- Optical flow provides the pixel-level motion information for our frame warping and alignment. With more accurate optical flows, the frames in $\widetilde{\Phi}_{t,s}$ will be well aligned, making $F_{\mathrm{RMNet}}(\cdot)$ work better, and $M_{t\to i}^{\mathrm{NR}}$ and M_t^{NR} more accurate. This will lead to fidelity losses in Eqs. (13) and (9) more effective. Moreover, with more accurate background estimation, $M_{i\to t}^{\mathrm{NR}}$ will be more accurate, then Eq. (7) will be more effective, rendering more accurate flow estimation. We can observe in the experimental section that, when large motion is included, it is significantly beneficial to optimize the optical flow estimation jointly with rain streak removal.
- If we can obtain more accurate rain-streak removal and optical flow estimation, the estimation of rain regions via optimizing Eq. (11) will be better. Also, if the estimation of the rain region is better, it can lead to more accurate rain removal results via Eq. (13).

5 Non-Local Rain Accumulation Removal From Videos

Without the reliance on the synthetic training data, the state-of-the-art rain-accumulation removal method (*i.e.*, non-local image dehazing method [3]) first estimates the accumulation-lines, then obtains the transmission map, and finally performs accumulation removal. Unfortunately, the method handles only a single image. If it is adopted for video accumulation removal, the estimated accumulation-lines among frames change drastically, and the accumulation removal results inevitably flicker.

To address the issue, we propose a non-local rain accumulation removal method from videos, which estimates accumulation-lines, constructs the related constraints and considers the temporal information. Our method includes four steps: 1) estimating accumulation lines, 2) estimating initial transmission, 3) transmission map refinement, 4) performing rain accumulation removal.

5.1 Accumulation Line Estimation

The global atmospheric light, $A_t = A$ is estimated based on the whole input frames. Pixels are projected onto the spherical coordinates and clustered based on the angles of pixels. The estimated rain streak-free result $H_t := I_t - R'_t$ is normalized by

$$\begin{aligned} H_t^a &= H_t - A_t = I_t - R'_t - A_t, \\ &= \alpha_t(B_t - A_t). \end{aligned} \tag{19}$$

All the normalized frames $\mathbf{H}^a = \{H_t^a\}$ are then transformed into the spherical coordinates:

$$\mathbf{H}^a = [r(p), \theta(p), \phi(p)], \tag{20}$$

where p is the pixel location at the given time-step (spatial and temporal locations), $r(p)$ is the distance to the origin. $\theta(p)$ and $\phi(p)$ are the longitude and latitude, respectively. Subsequently, the accumulation-lines are formed by grouping pixels based on $[\theta(p), \phi(p)]$.

5.2 Estimating Initial Transmission

With the clustered pixels as the accumulation-lines, we then estimate the transmission of each pixel. The basic idea is that, there are clean pixels in each accumulation line. That is to say, for those clean pixels, they are never interfered by haze and all related signal in the background B_t directly transmits to the haze image I_t, namely the transmission coefficient $\alpha = 1$ or $\alpha_t = 1$ in Eq. (4). Then, the transmission can be estimated based on the percentage of radius. The maximal radius of each accumulation-line $\hat{r}_{\max}(p)$ is estimated using:

$$\hat{r}_{\max}(p) = \max_{p\in h}\{r(p)\}, \tag{21}$$

where h is a given accumulation line. The transmission is obtained as follows:

$$\widetilde{\alpha}(p) = r(p)/\hat{r}_{\max}. \tag{22}$$

Note that, the time-step t is a part of indices in p.

5.3 Transmission Map Refinement

The transmission estimation in Eq. (22) is pixel-wise and does not consider the spatial and temporal smoothness. Then, the estimated transmission map is refined with the spatial and temporal smoothness coherency:

$$\sum_p \frac{[\hat{\alpha}(p) - \widetilde{\alpha}_{\mathrm{LB}}(p)]^2}{\sigma^2(p)} + \lambda_r \sum_p \sum_{q\in N_p} \frac{[\hat{\alpha}(p) - \hat{\alpha}(q)]^2}{\|I(p) - I(q)\|^2}, \tag{23}$$

$$\widetilde{\alpha}_{\mathrm{LB}}(p) = \max\left\{\widetilde{\alpha}(p), 1 - \min_{c\in\{R,G,B\}}\{I_c(p)/A_c\}\right\}. \tag{24}$$

where the parameter λ_r controls the trade-off between spatial and temporal smoothness. N_p is the four nearest neighbors of p in the spatial space and $\sigma(p)$ denotes the standard deviation of $\widetilde{\alpha}_{\mathrm{LB}}$, which is inferred based on each accumulation-line.

5.4 Rain Accumulation Removal

Having obtained $\widetilde{\alpha}_{\mathrm{LB}}(p)$, the estimation of transmission, the rain accumulation removal results can be computed as follows:

$$\hat{B}_t(p^*) = \{H_t(p^*) - [1 - \hat{\alpha}_t(p^*)]A\}/\hat{\alpha}_t(p^*), \tag{25}$$

where p^* indexes only the spatial locations and $p = \{p^*, t\}$.

Fig. 3 shows the accumulation removal results by a non-local image dehazing method [3] in comparison with our proposed method. The input rain-streaks frames are derained using our proposed method. As can be observed, both accumulation removal methods can significantly improve the visibility and unveil the background details. Our method, however, leads to more temporally continuous results than the single image-based method, particularly in the regions denoted by red arrows in (a)-(c) and the regions near trees in (d)-(f) of Fig. 3.

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

(a) Rain Frame (b) Non-Local (c) Ours (d) Rain Frame (e) Non-Local (f) Ours

Fig. 3. Visual comparisons of rain accumulation removal methods on two real sequences with heavy accumulation. The input rain frames are pre-processed by the above-mentioned self-learned rain streak removal method. Our proposed non-local video rain accumulation method leads to more temporally continuous results than the single image-based method, especially in the regions denoted by red arrows in (a)-(c) and the regions near trees in (d)-(f). From top to bottom: three successive video frames on two real rain videos. Zoom-in for better visualization.

TABLE 1
PSNR Results Among Different Rain Streak Removal Methods on *NTURain*

Dataset	Rain	URML	PReNet	UGSM	MS-CSC	DIP	SE	FastDeRain	J4RNet	SpacCNN	CVPR-2019	Ours	S2VD
a1	29.72	31.66	29.09	30.84	24.61	28.63	24.67	29.62	29.55	30.58	33.18	38.10	36.39
a2	29.31	25.30	25.09	29.56	25.52	29.45	23.13	30.17	26.65	31.26	33.15	36.00	33.06
a3	29.10	29.88	28.23	30.46	25.50	30.42	24.62	30.22	28.03	30.63	32.21	36.67	35.75
a4	32.63	33.20	31.26	33.46	31.62	36.19	32.30	34.07	33.43	35.33	37.42	40.44	39.53
b1	30.05	32.00	31.04	30.92	25.98	29.34	24.48	29.71	31.90	32.32	34.23	37.43	37.34
b2	30.71	33.00	31.80	31.94	27.99	32.10	27.93	31.73	32.27	35.14	36.48	38.92	40.55
b3	32.32	33.63	32.98	33.19	26.03	29.04	25.64	29.50	32.41	34.73	37.24	40.38	38.82
b4	29.42	31.96	33.42	29.96	25.85	31.13	25.54	29.26	31.60	34.90	35.18	38.96	37.53
Aver.	30.41	31.33	30.36	31.29	26.64	30.79	26.04	30.54	30.73	33.11	34.89	38.36	37.37

a1-b4 *Denote eight testing sequences in* nturain. *best results are denoted in red and the second best results are denoted in blue.*

6 Experimental Results

6.1 Datasets

Our method is compared with several different kinds of state-of-the-art methods on *NTURain* [10]. The dataset includes two sub-groups: one taken from fast moving onboard cameras, the other taken in the slow moving scenario from an unstable and panning camera. There are also other widely used video rain datasets, *e.g. RainSynLight25* and *RainSynComplex25* [38]. However, each video clip in these datasets includes too few video frames (only 7-15 frames) that we cannot conduct self-training. Thus, we do not compare different methods on these datasets. We also conduct a comparison on real rain videos widely used in the previous works and those from Youtube as well as our own rain data. We will provide more results and analysis on our project website.[1]

1. https://github.com/flyywh/CVPR-2020-Self-Rain-Removal-Journal

6.2 Implementation Details

NTURain dataset and our collected real rain videos are adopted for evaluation. *NTURain* dataset contains 25 paired training videos and 8 testing videos. However, in the quantitative evaluation, the training of our method does not rely on the training set at all, and utilizes only the rain frames in its testing set, as our method only needs to self-learn. For our qualitative evaluation, the collected real rain videos with only rain frames but without clean frames are utilized. The proposed deraining networks are trained with Adam optimizer. The learning rate is set to $1e^{-4}$. The optical flow module in our network is initialized with the existing pre-trained model, which is further finetuned on our input by setting the learning rate to $1e^{-7}$.

The training batches whose batch size is 8 are sampled from our training videos and cropped into a form of $64\times64\times5$ cubics. The PSNR and SSIM results are calculated based on the average of all frames' results for the given dataset. As for the detailed configuration, we produce the results of all compared methods following the authors'

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

TABLE 2
SSIM Results Among Different Rain Streak Removal Methods on *NTURain*

Dataset	Rain	URML	PReNet	UGSM	MS-CSC	DIP	SE	FastDeRain	J4RNet	SpacCNN	CVPR-2019	Ours	S2VD
a1	0.9161	0.9491	0.9374	0.9295	0.6843	0.9284	0.7101	0.9213	0.9372	0.9326	0.9468	0.9750	0.9658
a2	0.9255	0.9151	0.9004	0.9282	0.7412	0.9206	0.6616	0.9273	0.9098	0.9355	0.9456	0.9678	0.9519
a3	0.8971	0.9284	0.9153	0.9153	0.6585	0.9229	0.6388	0.9086	0.9134	0.9242	0.9310	0.9681	0.9564
a4	0.9386	0.9640	0.9608	0.9482	0.9300	0.9743	0.9295	0.9646	0.9624	0.9627	0.9710	0.9837	0.9779
b1	0.8971	0.9502	0.9514	0.9162	0.7537	0.9270	0.7338	0.9125	0.9478	0.9463	0.9506	0.9704	0.9712
b2	0.8880	0.9641	0.9624	0.9189	0.8535	0.9546	0.8500	0.9439	0.9580	0.9678	0.9678	0.9790	0.9821
b3	0.9305	0.9637	0.9628	0.9426	0.7614	0.9374	0.7800	0.9283	0.9542	0.9567	0.9653	0.9814	0.9754
b4	0.8938	0.9472	0.9591	0.9037	0.7461	0.9309	0.7527	0.8974	0.9426	0.9539	0.9513	0.9750	0.9657
Aver.	0.9108	0.9477	0.9437	0.9253	0.7661	0.9370	0.7571	0.9255	0.9407	0.9475	0.9540	0.9750	0.9683

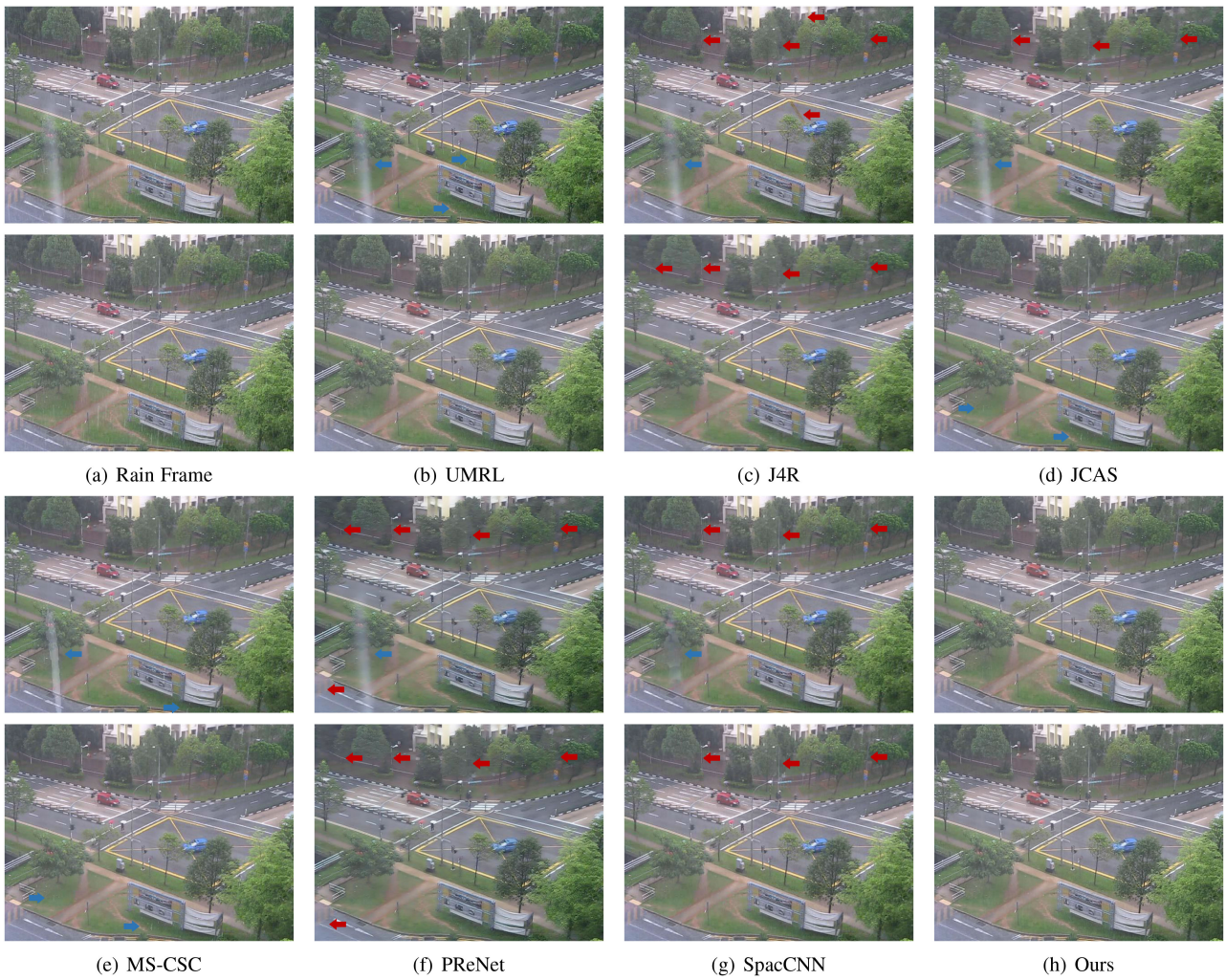
a1-b4 *Denote Eight Testing Sequences in* NTURain. *Best Results are Denoted in red and the Second Best Results are Denoted in blue.*

originally provided settings and codes: J4RNet is trained with their own datasets, PReNet and URML are trained with *Rain100H*, and others are traditional optimization-based methods not relying on the training data.

We consider that not retraining previous works and networks using *NTURain*'s training set is also fair, since: (1) during the deployment phase, practically, most of the time we know nothing about the video domain, (2) our method is not trained on *NTURain*'s training set also. Furthermore, we have SpacCNN [10] and S2VD [67] trained on *NTURain*, which has already shown the state-of-the-art performances of fully-supervised and semi-supervised categories.

6.3 Compared Methods

The proposed method is compared with state-of-the-art methods: Directional Global Sparse Model (UGSM) [12],



(a) Rain Frame (b) UMRL (c) J4R (d) JCAS

(e) MS-CSC (f) PReNet (g) SpacCNN (h) Ours

Fig. 4. Visual comparisons of different deraining methods on successive two frames in *ra4* sequence of *NTURain*. The remaining rain streaks and lost details are denoted with blue and red arrows, respectively. Zoom-in for better visualization.

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

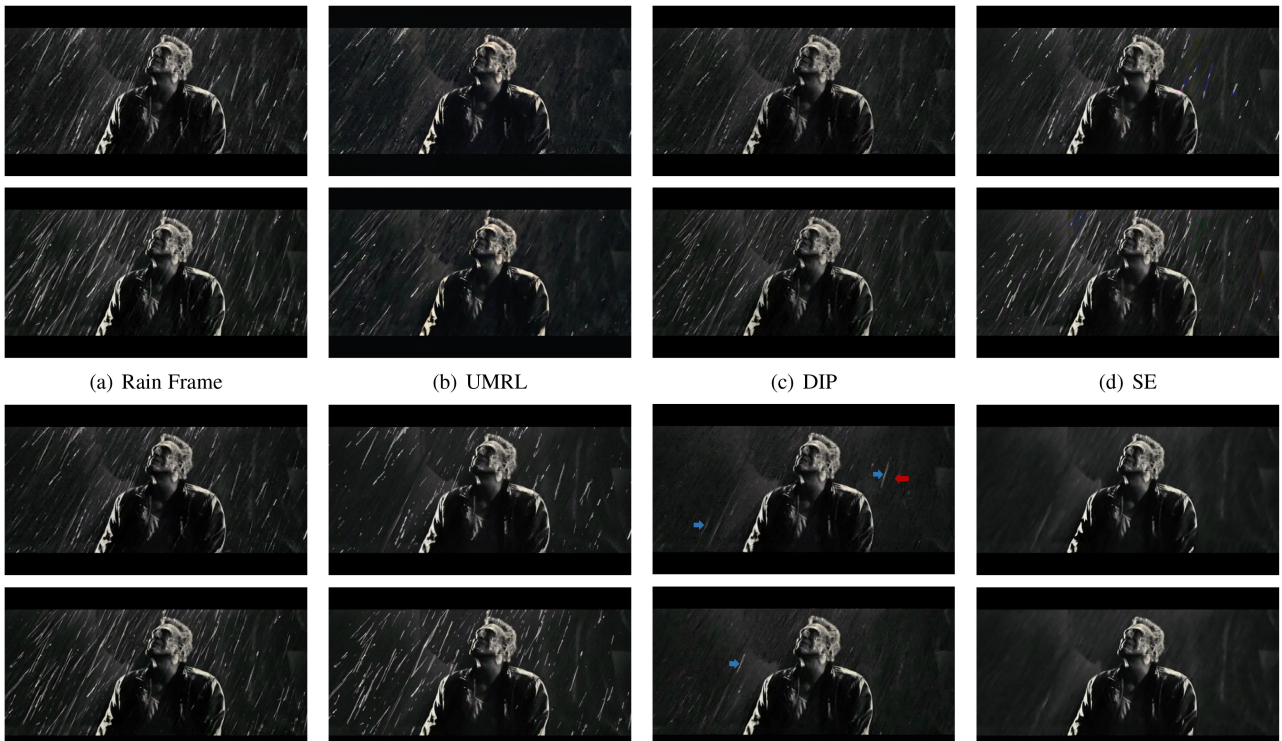


Fig. 5. Visual comparisons of different deraining methods on two successive frames in a real rain video sequence. The remaining rain streaks and lost details are denoted with blue and red arrows, respectively. Zoom-in for better visualization.

Uncertainty guided Multi-scale Residual Learning (UMRL) [65], FastDeRain [29], Progressive Recurrent Network (PReNet) [43], Stochastic Encoding (SE) [57], Discriminatively Intrinsic Priors (DIP) [30], Joint Recurrent Rain Removal and Reconstruction Network (J4RNet) [38], SuperPixel Alignment and Compensation CNN (SpacCNN) [10], Semi-Supervised Video Deraining (S2VD) [67], SuperPixel Alignment and Compensation CNN (SpacCNN) [10], Multi-Scale Convolutional Sparse Coding (MS-CSC) [33]. UGSM, UMRL, and PReNet are single-image rain removal approaches that provide state-of-the-art performance in the single-image rain removal task. DIP, FastDerain, SE, MS-CSC, J4RNet, SpacCNN, and are video derainig methods. UMRL, PReNet, J4RNet, SpacCNN, S2VD are deep-learning based methods. The former four are fully-supervised methods and S2VD is a semi-supervised method that utilizes both paired synthetic and unpaired real training videos. For all compared methods, the evaluation codes are provided kindly by the authors. We use Peak Signal-to-Noise Ratio (PSNR) [27] and Structure Similarity Index (SSIM) [55] as the comparison measures in the quantitative evaluation. Following the measure calculation in previous works, the results are only evaluated in the luminance channel, as the human visual system pays more attention to luminance than chrominance channels.

6.4 Quantitative Evaluation

We first compare the performance of different methods quantitatively in Tables 1 and 2. Comparing different methods, including both single-image methods and video rain removal methods, we reach several observations. First, the results of our methods are consistently better than previous works, including both data-driven methods or low-rank based ones, which demonstrates the rationality of our methodology. Second, in contrast to the state-of-the-art single-image rain removal method, PReNet, URML, and UGSM, we achieve more than 7 dB and 0.0270 gain in PSNR and SSIM, respectively. The results demonstrate the importance and necessity of adopting temporal modeling besides the knowledge from massive data. Third, our method outperforms SpacCNN significantly, the state-of-the-art fully-supervised learning video rain removal method, with a gain of 5.2 dB 0.0275 in PSNR and SSIM, respectively. Fourth, note that, S2VD is the state-of-the-art deraining method that utilizes both paired synthetic data and unpaired real training data. Surprisingly, our method achieves even better performance, which shows the intrinsic temporal redundancy has provided strong enough guidance for video deraining.

6.5 Qualitative Evaluation

Qualitatively, we also compare the visual results of different methods. One group of results on *NTURain* and four groups of results on real videos are presented in Figs. 4, 5, 6, 7, and 8. The testing videos contain various kinds of rain streaks in intensity, density, and scale. The results in Figs. 4, 5, 6, and 7 are processed by our self-learned rain streak removal method, while Fig. 8 is processed by our self-learned rain streak and accumulation removal methods jointly. Our results look more impressive. Fewer rain streaks remain while more abundant details with less blurring and lost

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

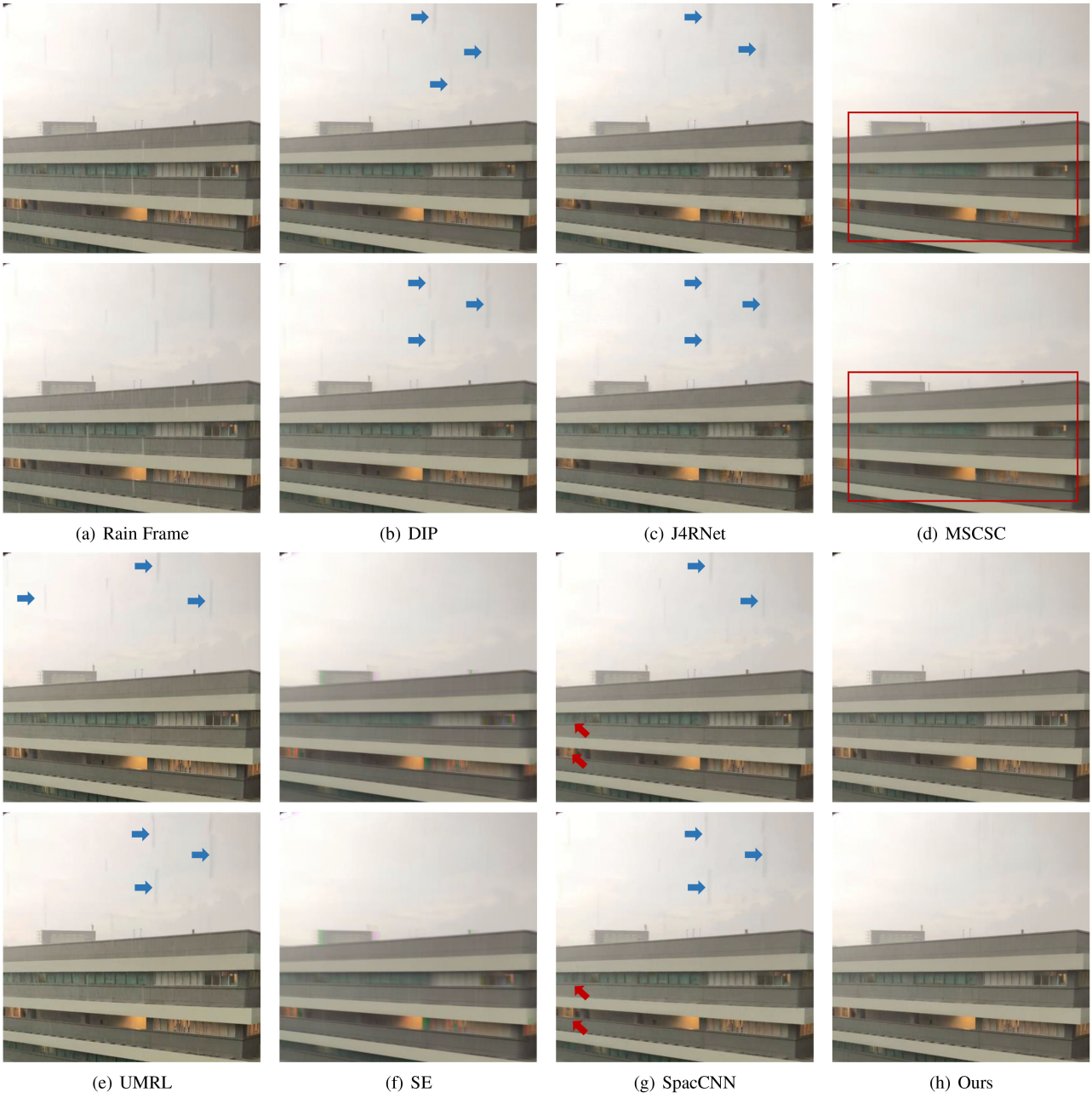


Fig. 6. Visual comparisons of different deraining methods on two successive frames in a real rain video sequence. The remaining rain streaks and lost details are denoted with blue arrows and red arrows/boxes, respectively. Zoom-in for better visualization.

details are observed. More visual comparisons will be provided on our project page[1].

6.6 Ablation Study of Losses in Model Training

To evaluate the effectiveness of loss selection, an ablation study on the training losses is shown in Tables 5 and 6. It is observed that, L_1 loss leads to much superior results than MSE loss, which is used in [63], and our reweighted L_1 loss achieves better results than L_1 loss in PSNR and SSIM. The results confirm the merits of adopting reweighted L_1 loss in our model training. We also provide the visual comparisons of our methods trained with different losses on two real rain sequences in Fig. 9. It is observed that, a sparser loss leads to better background reconstruction.

6.7 Ablation Study of Network Architectures

To evaluate the effectiveness of whether the two-stage network design [63] and the one-stage network is better, an ablation study on the network architecture is shown in Tables 3 and 4. It is observed that, with our new architecture and settings, single-stage results (only EHNet in the two-stage structure) are even better than the two-stage ones. Note that, in [63], the rain mask guidance is generated from the PreNet and cannot be imposed on the training of single-stage EHNet. Comparatively, in our method, the rain mask guidance is generated based on the final derained results, and therefore can be imposed on both kinds of architectures. With the support of the ablation study's results, we select the one-stage network as our new version, which owns about only half of the parameters compared to [63].

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

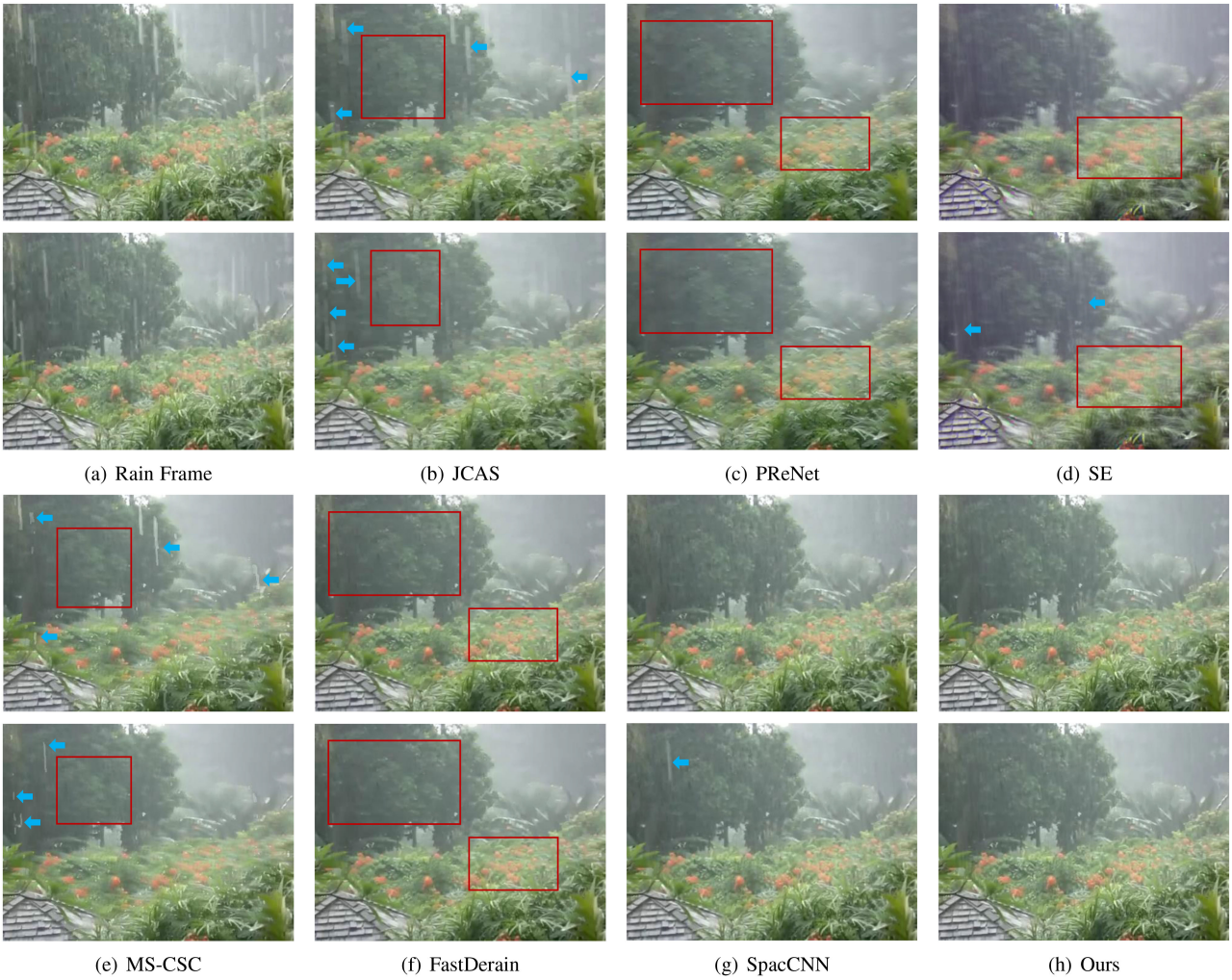


Fig. 7. Visual comparisons of different deraining methods on two successive frames in a real rain video sequence. The remaining rain streaks and lost details are denoted by blue arrows and red boxes, respectively. Zoom-in for better visualization.

6.8 Ablation Study of Rain-Related Priors

To confirm the merits of rain-related priors, several versions of our deraining network with and without rain priors, *i.e.* optical flow and rain region mask, are compared. The results are presented in Tables 7 and 8. Comparing SLDNet+v2 and SLDNet+full, we can observe that, it is very beneficial to introduce the rain region mask guidance, as the guidance lets the network know which regions are more reliable ground truths, which equivalently to refine the noisy labels in our problem.

Comparing SLDNet+v1 and SLDNet+full, it is observed that, in the static scenes (*b1*-*b4*), the versions with the finetuned optical flow leads to degraded performance. Comparatively, in the dynamic scenes (*a1*-*a4*), optical flow finetuning leads to a large performance leap. When there are no large motions, finetuning the already accurate enough optical flow model based on our input, degraded by rain streaks, inevitably leads to the degeneration of the flow model. Comparatively, when large motions are included, the gain brought by the model finetuning can surpass the loss caused by finetuning on rain contaminated input. From the results in Fig. 10, we can observe that, SLD-Net-v1 and SLD-Net-v2 both generate visual artifacts in the derained results, while SLD-Net-full provides more visually pleasing results, which shows the rationality of our model design.

6.9 Analysis on Parameters of Rain Mask Guidance

We provide the analysis on the parameter selection of rain masks in Tables 9 and 10 on *a1* and *b1* sequences, one in a static scene and the other in a dynamic scene. In general, it is observed, $\omega = 0.1$ is a satisfying option, offering competitive or superior performance in PSNR and SSIM than other options.

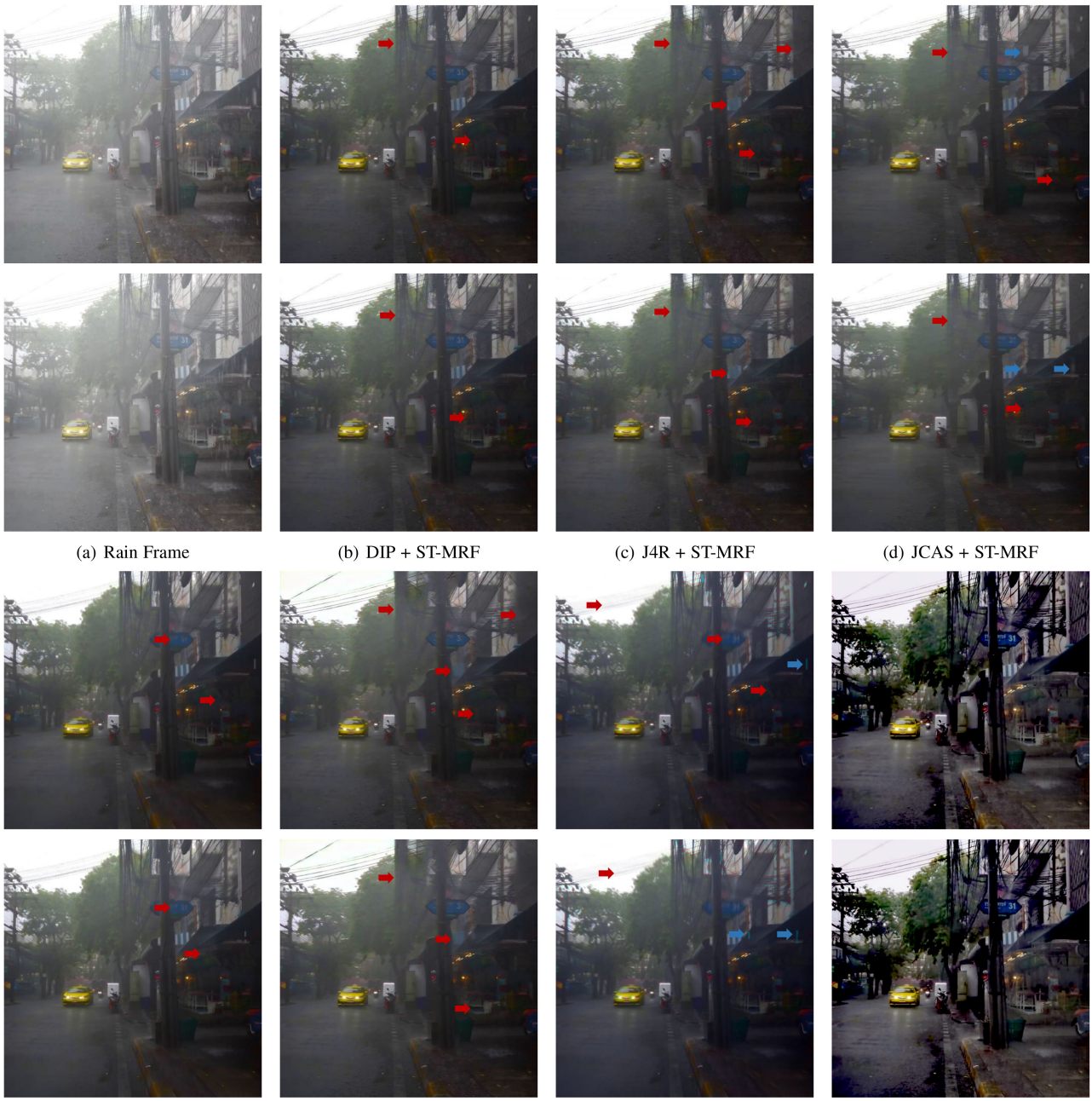
6.10 Analysis on Parameters of Sparsity-Driven Loss

We provide the analysis on the selection of the reweighted parameters in sparsity-driven losses in Tables 11 and 12 on *a1* and *b1* sequences, one in a static scene and the other in a dynamic scene. From the results, we find that, for static scenes, l_p with a smaller p leads to improved performance. While in the dynamic scenes, l_p with $p < 0.5$ leads to a rapid performance drop. In general, $l_{0.5}$ provides generally good results for both cases.

6.11 Visual Results of Rain Region Estimation

We also compare the visual results of rain region estimation on the *b1* sequence in *NTURain* dataset in Fig. 11. It is observed that, the rain/non-rain regions predicted by our

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.



(a) Rain Frame (b) DIP + ST-MRF (c) J4R + ST-MRF (d) JCAS + ST-MRF

(e) MS-CSC + ST-MRF (f) PReNet + ST-MRF (g) SE + ST-MRF (h) Ours

Fig. 8. Visual comparisons of different deraining methods on two successive frames in a real rain video sequence. Our results are produced by our self-learned rain streak and accumulation removal methods. Results of other deraining methods are post-processed by ST-MRF [6], the state-of-the-art video dehazing method. The remaining rain streaks and lost details are denoted by blue arrows and red arrows, respectively. Zoom-in for better visualization.

method successfully locate the rain streak regions and play a role to weaken the effects of these regions.

6.12 Visualization of Extracted Features

We also visualize the extracted features of our SLDNet+ and a fully supervised network, which has the same structure as SLDNet+ but trained on the training set of *NTURain* in a fully supervised way. The results are generated on *b1* sequence in *NTURain* dataset. The feature maps are ordered by their variations and the feature maps with larger variations rank first. From Fig. 12, it is observed that, the extracted features by SLDNet+ are sparser but more diverse than the fully supervised one. The first several feature maps obtained from the fully supervised method include more information related to the background and look very similar. Comparatively, our corresponding features generated by SLDNet+ capture more distinguished information.

6.13 Complexity Comparison

In Table 13, we compare the runtime of several SOTA methods. The testing frame's resolution is 832×512. The proposed rain streak removal method (Ours-S) and J4RNet are implemented in Pytorch. DetailNet is implemented in Tensorflow. Other SOTA methods, including our accumulation

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

Fig. 9. Visual comparisons of our methods with different losses on two real rain sequences. The lost details are denoted by blue arrows. Top panel: rain input frame. The second panel: results by MSE loss. The third panel: results by L1 loss. The fourth panel: results by our reweighted loss. Zoom-in for better visualization.

TABLE 3
Ablation Study for the One-Stage or Two-Stage Network Architecture in PSNR

Dataset	*a1*	*a2*	*a3*	*a4*
Two-Stage (CVPR-2020)	37.08	**36.49**	36.01	40.04
Ours	**38.10**	36.00	**36.67**	**40.44**
Dataset	*b1*	*b2*	*b3*	*b4*
Two-Stage (CVPR-2020)	36.67	38.90	39.88	38.37
Ours	**37.43**	**38.92**	**40.38**	**38.96**

a1-b4 *Denote eight testing sequences in* nturain. *the two-stage network [63] in cvpr-2020 is retrained based on our new settings,* e.g. *reweighed l_1 loss and the new threshold to produce the masks. best results are denoted in bold.*

TABLE 4
Ablation Study for the One-Stage or Two-Stage Network Architecture in SSIM

Dataset	*a1*	*a2*	*a3*	*a4*
Two-Stage (CVPR-2020)	0.9694	**0.9703**	0.9615	0.9824
Ours	**0.9750**	0.9678	**0.9681**	**0.9837**
Dataset	*b1*	*b2*	*b3*	*b4*
Two-Stage (CVPR-2020)	0.9690	0.9786	0.9805	0.9715
Ours	**0.9704**	**0.9790**	**0.9814**	**0.9750**

a1-b4 *Denote eight testing sequences in* nturain. *the two-stage network [63] in cvpr-2020 is retrained based on our new settings,* e.g. *reweighed l_1 loss and the new threshold to produce the masks. best results are denoted in bold.*

TABLE 5
Ablation Study for Losses Used in Our Work in PSRN

Network	*a1*	*a2*	*a3*	*a4*
MSE Loss	34.72	34.68	34.17	36.45
L_1 Loss	36.58	35.74	35.93	39.83
Reweighted L_1 Loss	**38.10**	**36.00**	**36.67**	**40.44**
Network	*b1*	*b2*	*b3*	*b4*
MSE Loss	33.73	35.32	34.85	34.89
L_1 Loss	36.41	38.34	38.96	37.80
Reweighted L_1 Loss	**37.43**	**38.92**	**40.38**	**38.96**

a1-b4 *Denote eight testing sequences in* nturain. *best results are denoted in bold.*

TABLE 6
Ablation Study for Losses Used in Our Work in SSIM

Network	*a1*	*a2*	*a3*	*a4*
MSE Loss	0.9582	0.9601	0.9512	0.9715
L_1 Loss	0.9698	0.9668	0.9638	0.9819
Reweighted L_1 Loss	**0.9750**	**0.9678**	**0.9681**	**0.9837**
Network	*b1*	*b2*	*b3*	*b4*
MSE Loss	0.9516	0.9588	0.9649	0.9558
L_1 Loss	0.9663	0.9755	0.9774	0.9701
Reweighted L_1 Loss	**0.9704**	**0.9790**	**0.9814**	**0.9750**

a1-b4 *Denote eight testing sequences in* nturain. *best results are denoted in bold.*

TABLE 7
Ablation Study for Rain-Related Priors Used in Our Work in PSNR

Network	*a1*	*a2*	*a3*	*a4*
SLDNet-v1 (*wo* OFF, *w* RRG)	**38.29**	**36.10**	**37.16**	**40.52**
SLDNet-v2 (*w* OFF, *wo* RRG)	34.02	34.15	33.36	38.12
SLDNet-full (*w* OFF, *w* RRG)	38.10	36.00	36.67	40.44
Network	*b1*	*b2*	*b3*	*b4*
SLDNet-v1 (*wo* OFF, *w* RRG)	36.95	38.31	39.72	38.77
SLDNet-v2 (*w* OFF, *wo* RRG)	33.83	36.69	37.26	35.95
SLDNet-full (*w* OFF, *w* RRG)	**37.43**	**38.92**	**40.38**	**38.96**

a1-b4 *Denote eight testing sequences in* nturain. *sldnet-v1: without optical flow finetuning (off), rain region guidance (rrg). sldnet-v2: with off but without rrg. sldnet-full: with off, rrg. best results are denoted in bold.*

removal method (Ours-A), are implemented in MATLAB. SpacCNN is built based on MatConvNet.[2] JORDER is built on the Caffe's Matlab wrapper.[3] TCLRM, SE, and our accumulation removal method are built based on CPU while other methods are GPU-based methods. The SpacCNN is evaluated on NVIDIA Geforce MX150 as it needs a Windows operation system in the PC while other GPU-based methods are evaluated on GeForce GTX 1080Ti in the cluster. Generally, the running speed of the proposed method is slower than other more efficient methods, such as FastDeRain and JORDER.

2. http://www.vlfeat.org/matconvnet/
3. http://caffe.berkeleyvision.org/

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

TABLE 8
Ablation Study for Rain-Related Priors Used in Our Work in SSIM

Network	*a1*	*a2*	*a3*	*a4*
SLDNet-v1 (*wo* OFF, *w* RRG)	**0.9762**	**0.9693**	**0.9701**	**0.9839**
SLDNet-v2 (*w* OFF, *wo* RRG)	0.9548	0.9582	0.9451	0.9761
SLDNet-full (*w* OFF, *w* RRG)	0.9750	0.9678	0.9681	0.9837
Network	*b1*	*b2*	*b3*	*b4*
SLDNet-v1 (*wo* OFF, *w* RRG)	0.9703	**0.9798**	**0.9820**	**0.9753**
SLDNet-v2 (*w* OFF, *wo* RRG)	0.9541	0.9735	0.9701	0.9600
SLDNet-full (*w* OFF, *w* RRG)	**0.9704**	0.9790	0.9814	0.9750

a1-b4 *Denote eight testing sequences in* nturain. *sldnet-v1: without optical flow finetuning (off), rain region guidance (rrg). sldnet-v2: with off but without rrg. sldnet-full: with off, rrg. best results are denoted in bold.*

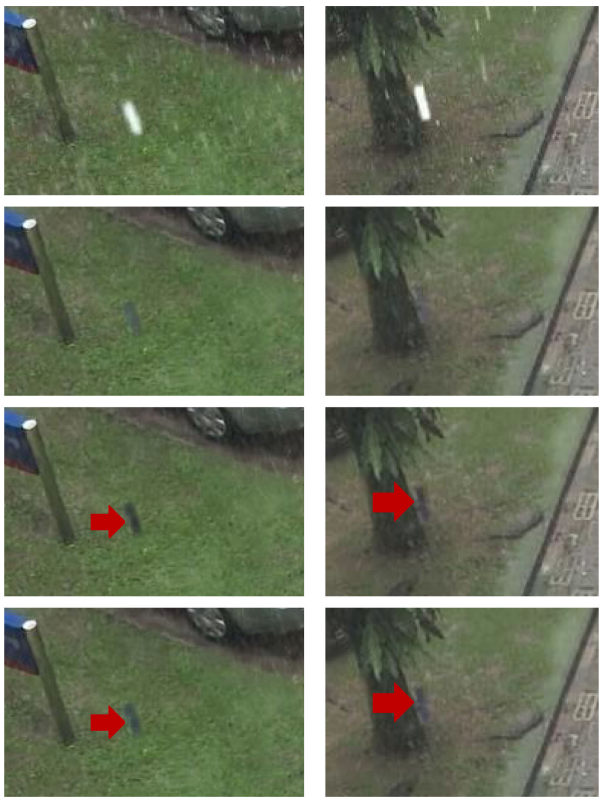


Fig. 10. Visual comparisons of our methods with and without using rain-related priors on *rb2* and *rb3* sequences in *NTURain* dataset. The lost details are denoted by red arrows. Top panel: rain input frame. The second panel: results by SLDNet-full. The third panel: results by SLDNet-v2. The fourth panel: results by SLDNet-v1. Zoom-in for better visualization.

However, our methods can be further accelerated by the following ways to make it more real-time application-driven:

- For our rain accumulation methods, we down-sample the frame resolution when estimating and the air light (1/16) and haze lines (1/8) that are derived into transmission estimation. After the operation, our accumulation removal method (Ours-A-L) offers almost the same visual results, as shown in Fig. 14, but only needs less than 1/10 original running time (0.1540 vs. 2.3234).

TABLE 9
Analysis of the Parameter Selection of Rain Masks in PSRN

ω	1	0.1	0.01	0.001	0.0001
a1	37.71	**38.10**	37.77	33.92	22.60
b1	37.30	**37.43**	36.95	31.05	18.47

a1 *and* b1 *Denote two testing sequences in* nturain. *best results are denoted in bold.*

TABLE 10
Analysis of the Parameter Selection of Rain Masks in SSIM

ω	1	0.1	0.01	0.001	0.0001
a1	0.9741	0.9750	**0.9752**	0.9711	0.8986
b1	0.9698	**0.9704**	0.9700	0.9535	0.7507

a1 *and* b1 *Denote two testing sequences in* nturain. *best results are denoted in bold.*

TABLE 11
Analysis of the Selection of the Reweighted Parameters in Sparsity-Driven Losses in PSNR

p	l_1	$l_{0.9}$	$l_{0.8}$	$l_{0.7}$	$l_{0.6}$
a1	36.58	36.96	37.19	37.44	37.55
b1	36.41	36.88	37.11	36.59	**37.45**
p	$l_{0.5}$	$l_{0.4}$	$l_{0.3}$	$l_{0.2}$	-
a1	38.10	38.28	38.40	**38.49**	-
b1	37.43	36.85	36.69	36.85	-

a1 *and* b1 *Denote two testing sequences in* nturain. *best results are denoted in bold.*

TABLE 12
Analysis of the Selection of the Reweighted Parameters in Sparsity-Driven Losses in SSIM

p	l_1	$l_{0.9}$	$l_{0.8}$	$l_{0.7}$	$l_{0.6}$
a1	0.9698	0.9712	0.9719	0.9728	0.9736
b1	0.9541	0.9673	0.9684	0.9687	0.9695
p	$l_{0.5}$	$l_{0.4}$	$l_{0.3}$	$l_{0.2}$	-
a1	0.9750	0.9757	0.9763	**0.9771**	-
b1	**0.9704**	0.9696	0.9692	0.9684	-

a1 *and* b1 *Denote two testing sequences in* nturain. *best results are denoted in bold.*

- Our rain accumulation methods can be further accelerated by GPU computation.
- For our rain streak removal methods, some more lightweight deformable convolutions might replace the time-consuming optical flow operations. Furthermore, some acceleration techniques, such as model compression and distillation can be included to further reduce the running time.

6.14 Comparisons to Using Traditional Optical Flow

In Table 14, we also compare our method with the version equipped with the traditional optical flow, *i.e.* Gunner

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

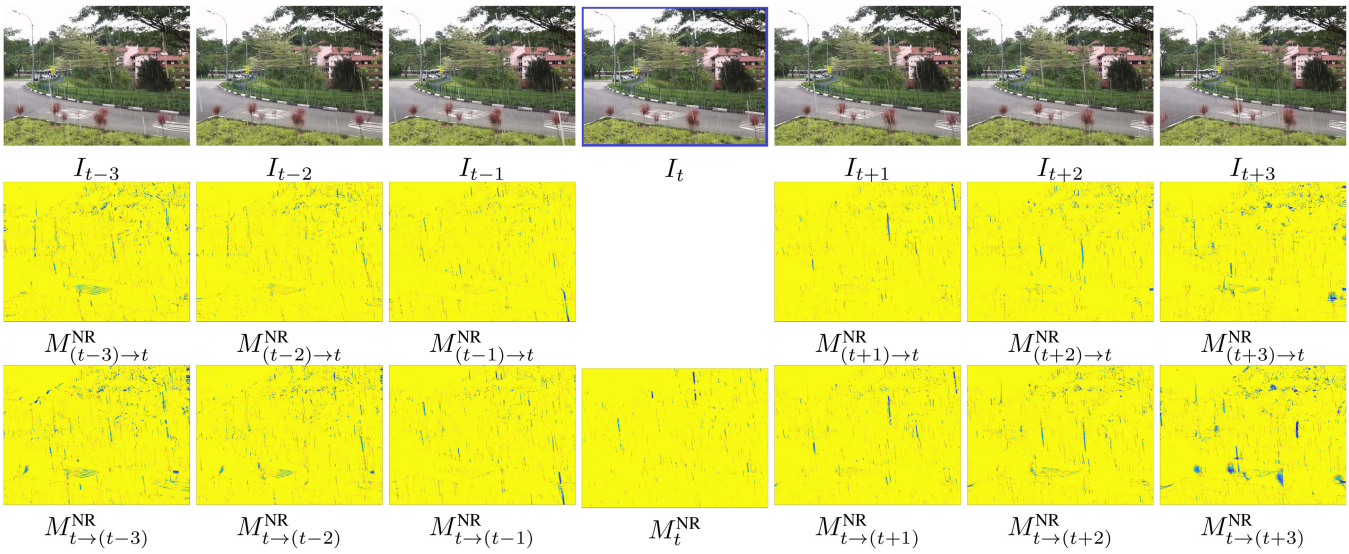
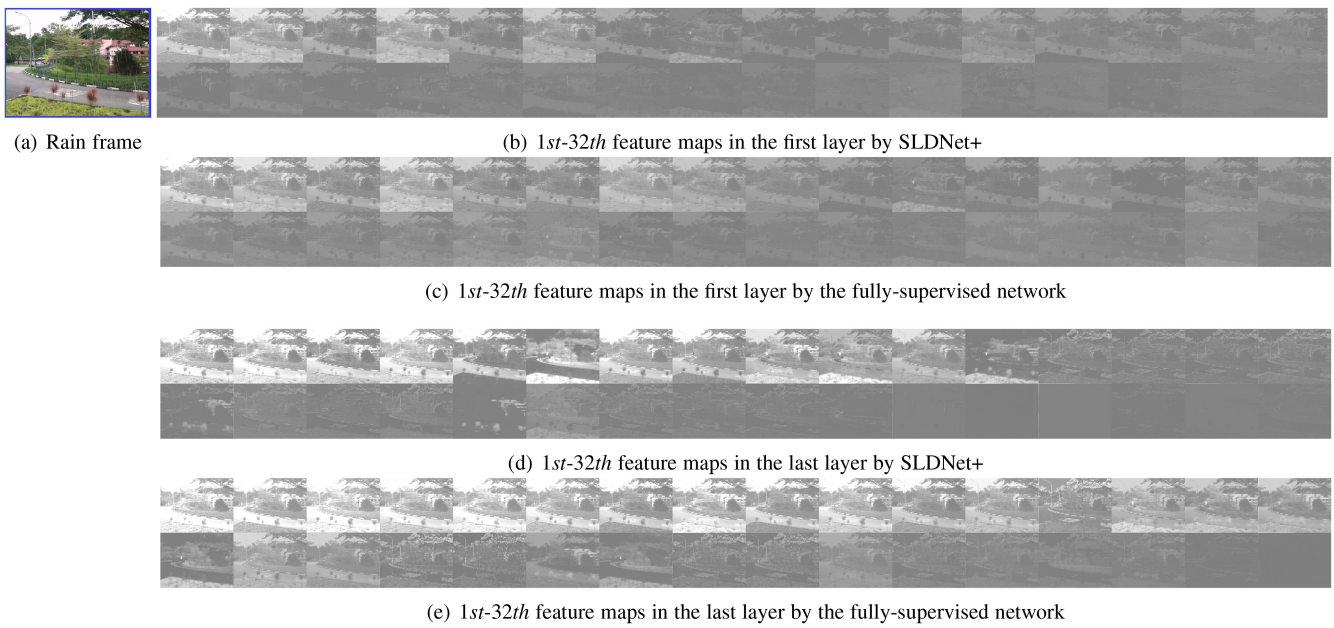


Fig. 11. visual results of rain region estimation on *b1* sequence in *NTURain* dataset. Top panel: rain input frame. Middle panel: rain masks used to train the optical flow network in Eq. (7). Bottom panel: rain masks used to train our deraining network in Eqs. (9) and (13). Yellow color denotes the pixel value is close to 1 while blue color denotes the pixel value is close to 0. Zoom-in for better visualization.



(a) Rain frame

(b) 1*st*-32*th* feature maps in the first layer by SLDNet+

(c) 1*st*-32*th* feature maps in the first layer by the fully-supervised network

(d) 1*st*-32*th* feature maps in the last layer by SLDNet+

(e) 1*st*-32*th* feature maps in the last layer by the fully-supervised network

Fig. 12. Visual comparisons of the extracted features of our SLDNet+ and a fully supervised network on *b1* sequence in *NTURain* dataset. The feature maps are ordered by their variations. Zoom-in for better visualization.

TABLE 13
Running Time Comparison (In Sec.) of Different Rain Removal Methods on a Video With the Spatial Resolution 832×512

Methods	JORDER	DetailNet	FastDeRain	SpacCNN	Ours-S	-
Time	0.6329	1.4698	0.3962	9.5075	2.0270	-
Methods	MS-CSC	SE	J4RNet	TCLRM	Ours-A	Ours-A-L
Time	15.7957	19.8516	0.8401	192.7007	2.3234	0.1540

Farneback's algorithm [18] implemented in Python. When training the version with Gunner Farneback's algorithm, the flow loss in Eq. (7) is not adopted as the algorithm is not trainable. From the results, it is observed that, our method achieves overall slightly better performance than the version using traditional Gunner Farneback optical estimation. It is surprising that, the version with Gunner Farneback's algorithm wins in PSNR on *b2*. However, for SSIM, our method still achieves superior performance to the traditional one.

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

TABLE 14
Comparisons to Using Traditional Optical Flow

Method	Metric	*a1*	*a2*	*a3*	*a4*
Traditional Flow	PSNR	36.35	34.53	35.29	39.81
Ours		**38.10**	**36.00**	**36.67**	**40.44**
Traditional Flow	SSIM	0.9691	0.9583	0.9587	0.9809
Ours		**0.9750**	**0.9678**	**0.9681**	**0.9837**
Method	Metric	*b1*	*b2*	*b3*	*b4*
Traditional Flow	PSNR	37.26	**39.12**	39.93	38.67
Ours		**37.43**	38.92	**40.38**	**38.96**
Traditional Flow	SSIM	0.9687	0.9765	0.979	0.9724
Ours		**0.9704**	**0.9790**	**0.9814**	**0.9750**

a1-b4 *Denote Eight Testing Sequences in* NTURain

(a) Rain Frame (b) Non-Local (c) Ours

Fig. 13. Visual comparisons of the rain accumulation removal methods on two real rain sequences with heavy accumulation when the rain streak removal algorithm is not employed at the first stage. Our proposed non-local video rain accumulation method leads to more temporally continuous results than the single image-based method, especially in the regions denoted by red arrows. Zoom-in for better visualization.

6.15 Evaluation on the Effect of Existence of the Previous Stage on the Later One

We evaluate the effect of the existence of the previous stage on the later one in Fig. 13. It is observed that, except for the remaining rain streaks, the previous rain streak has little impact on the second stage – rain accumulation removal. When the rain streak removal algorithm is not employed at the first stage, our method and the single image-based one can remove accumulation while the single image-based one suffers from more severe frame flicker, as denoted by the red arrows in Fig. 13.

7 Conclusion

In this paper, beyond our previous explorations on self-supervised video rain streak removal, we build an augmented one-stage Self-Learned Deraining Network (SLDNet+) to make full use of both the temporal correlation and consistency to construct the restoration mapping from rain frames to clean ones. With the temporal consistency of the generated frames in our mind, SLDNet+ enforces the generated frame to get close to the adjacent frames by using alignment. The training loss adopts a reweighted L_1 form, which helps the estimation of the degraded labels more robustly. The injection of the rain-related priors makes the network learn better, *e.g.* only manipulating the rain region and reducing the role of rain streaks in the adjacent frames as the label. Furthermore, a novel non-local video rain accumulation removal method is constructed to remove the rain accumulation, and thus improve the visibility of our results when rain is heavy. Extensive experiments show the effectiveness of our approach, which provides better results in both quantitative and qualitative evaluations.

(a) Original Version (b) Accelerated Version

Fig. 14. Visual comparisons between our accelerated method and the original one. Zoom-in for better visualization.

References

[1] C. Ancuti, C. O. Ancuti, T. Haber, and P. Bekaert, "Enhancing underwater images and videos by fusion," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 81–88.

[2] C. PeterBarnum, S. Narasimhan, and T. Kanade, "Analysis of rain and snow in frequency space," *Int. J. Comput. Vis.*, vol. 86, no. 2/3, pp. 256–274, 2010.

[3] D. Berman, T. Treibitz, and S. Avidan, "Non-local image dehazing," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1674–1682.

[4] J. Bossu, N. Hautière, and J.-P. Tarel, "Rain or snow detection in image sequences through use of a histogram of orientation of streaks," *Int. J. Comput. Vis.*, vol. 93, no. 3, pp. 348–367, 2011.

[5] N. Brewer and N. Liu, "Using the shape characteristics of rain to identify and remove rain from video," in *Proc. Joint IAPR Int. Workshops SPR SSPR*, 2008, pp. 451–458.

[6] B. Cai, X. Xu, and D. Tao, "Real-time video dehazing based on spatio-temporal mrf," in *Proc. Pacific Rim Conf. Multimedia*, 2016, pp. 315–325.

[7] C. Chen and H. Li, "Robust representation learning with feedback for single image deraining," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 7738–7747.

[8] H. Chen *et al.*, "Pre-trained image processing transformer," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 12 294–12 305.

[9] J. Chen and L. P. Chau, "A rain pixel recovery algorithm for videos with highly dynamic scenes, *IEEE Trans. Image Process.*, vol. 23, no. 3, pp. 1097–1104, Mar. 2014.

[10] J. Chen, C.-H. Tan, J. Hou, L.-P. Chau, and H. Li, "Robust video content alignment and compensation for rain removal in a CNN framework," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6286–6295.

[11] Y.-L. Chen and C.-T. Hsu, "A. generalized low-rank appearance model for spatio-temporally correlated rain streaks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 1968–1975.

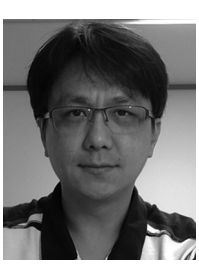
Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

[12] L.-J. Deng, T.-Z. Huang, X.-L. Zhao, and T.-X. Jiang, "A directional global sparse model for single image rain removal," *Appl. Math. Modelling*, vol. 59, pp. 662–679, 2018.
[13] S. Deng *et al.*, "Detail-recovery image deraining via context aggregation networks," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 14560–14569.
[14] A. Dosovitskiy *et al.*, FlowNet: Learning optical flow with convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2758–2766.
[15] D. Eigen, D. Krishnan, and R. Fergus, "Restoring an image taken through a window covered with dirt or rain," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 633–640.
[16] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Berlin, Germany: Springer, 2010.
[17] P. Stephen, B. Emmanuel, J., Michael, B. C., and Wakin, "Enhancing sparsity by reweighted l_1 minimization," *J. Fourier Anal. Appl.*, vol. 14, pp. 877–905, 2008.
[18] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Image Analysis*, J. Bigun and T. Gustavsson, Eds., Berlin, Germany: Springer, 2003, pp. 363–370.
[19] X. Fu, J. Huang, D. Zeng, Y. Huang, X. Ding, and J. Paisley, "Removing rain from single images via a deep detail network," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3855–3863.
[20] X. Fu, Q. Qi, Z.-J. Zha, Y. Zhu, and X. Ding, "Rain streak removal via dual graph convolutional network," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 1352–1360.
[21] K. Garg and S. K. Nayar, "Detection and removal of rain from videos," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2004, pp. I–528.
[22] K. Garg and S. K. Nayar, "When does a camera see rain?," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2005, pp. 1067–1074.
[23] K. Garg and Shree K. Nayar, "Photorealistic rendering of rain streaks," *ACM Trans. Graph.*, vol. 25, pp. 996–1002, 2006.
[24] K. Garg and S. K. Nayar, "Vision and rain,"*Int. J. Comput. Vis.*, vol. 75, no. 1, pp. 3–27, 2007.
[25] X. Hu, C.-W. Fu, L. Zhu, and P.-A. Heng, "Depth-attentional features for single-image rain removal," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8022–8031.
[26] D.-A. Huang, L.-W. Kang, Y.-C. F. Wang, and C.-W. Lin, "Self-learning based image decomposition with applications to single image denoising," *IEEE Trans. Multimedia*, vol. 16, no. 1, pp. 83–93, Jan. 2014.
[27] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment," *Electron. Lett.*, vol. 44, no. 13, pp. 800–801, 2008.
[28] K. Jiang *et al.*, "Multi-scale progressive fusion network for single image deraining," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8346–8355.
[29] T. Jiang, T. Huang, X. Zhao, L. Deng, and Y. Wang, "FastDerain: A novel video rain streak removal method using directional gradient priors," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 2089–2102, Apr. 2019.
[30] T.-X. Jiang, T.-Z. Huang, X.-L. Zhao, L.-J. Deng, and Y. Wang, "A novel tensor-based video rain streaks removal approach via utilizing discriminatively intrinsic priors," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4057–4066.
[31] L. W. Kang, C. W. Lin, and Y. H. Fu, "Automatic single-image-based rain streaks removal via image decomposition," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1742–1755, Apr. 2012.
[32] J. H. Kim, J. Y. Sim, and C. S. Kim, "Video deraining and desnowing using temporal correlation and low-rank matrix completion," *IEEE Trans. Image Process.*, vol. 24, no. 9, pp. 2658–2670, Sep. 2015.
[33] M. Li *et al.*, "Video rain streak removal by multiscale convolutional sparse coding," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6644–6653.
[34] R. Li, L.-F. Cheong, and R. T. Tan, "Heavy rain image restoration: Integrating physics model and conditional adversarial learning," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1633–1642.
[35] X. Li, J. Wu, Z. Lin, H. Liu, and H. Zha, "Recurrent squeeze-and-excitation context aggregation net for single image deraining," in *Proc. IEEE Eur. Conf. Comput. Vis.*, 2018, pp. 254–269.
[36] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, "Rain streak removal using layer priors," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2736–2744.
[37] H. Lin, Y. Li, X. Fu, X. Ding, Y. Huang, and J. Paisley, "Rain o'er me: Synthesizing real rain to derain with data distillation," *IEEE Trans. Image Process.*, vol. 29, pp. 7668–7680, 2020.
[38] J. Liu, W. Yang, S. Yang, and Z. Guo, "Erase or fill? Deep joint recurrent rain removal and reconstruction in videos," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3233–3242.
[39] J. Liu, W. Yang, S. Yang, and Z. Guo, "D3R-Net: Dynamic routing residue recurrent network for video rain removal," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 699–712, Feb. 2019.
[40] P. Liu, J. Xu, J. Liu, and X. Tang, "Pixel based temporal analysis using chromatic property for removing rain from videos," *Comput. Inf. Sci.*, vol. 2, pp. 53–60, 2009.
[41] Y. Luo, Y. Xu, and H. Ji, "Removing rain from a single image via discriminative sparse coding," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3397–3405.
[42] S. G. Narasimhan and S. K. Nayar, "Vision and the atmosphere," *Int. J. Comput. Vis.*, vol. 48, no. 3, pp. 233–254, 2002.
[43] D. Ren, W.Q. Zuo, P. H. Zhu, and D. Meng, "Progressive image deraining networks: A better and simpler baseline," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3937–3946.
[44] W. Ren, J. Tian, Z. Han, A. Chan, and Y. Tang, "Video desnowing and deraining based on matrix decomposition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4210–4219.
[45] V. Santhaseelan and K. V. Asari, "Utilizing local phase information to remove rain from video,"*Int. J. Comput. Vis.*, vol. 112, no. 1, pp. 71–89, Mar. 2015.
[46] S.-H. Sun, S.-P. Fan, and Y.-C. F. Wang, "Exploiting image structural similarity for single image rain removal," in *Proc. IEEE Int. Conf. Image Process.*, 2014, pp. 4482–4486.
[47] Y. Tian and S. G. Narasimhan, "Seeing through water: Image restoration using model-based tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2009, pp. 2303–2310.
[48] A. Kumar Tripathi and S. Mukhopadhyay, "A probabilistic approach for detection and removal of rain from videos," *IETE J. Res.*, vol. 57, no. 1, pp. 82–91, 2011.
[49] A. K. Tripathi and S. Mukhopadhyay, "Video post processing: Low-latency spatiotemporal approach for detection and removal of rain," *IET Image Process.*, vol. 6, no. 2, pp. 181–196, Mar. 2012.
[50] C. Wang, Y. Wu, Z. Su, and J. Chen, "Joint self-attention and scale-aggregation for self-calibrated deraining network," in *Proc. Conf. ACM Trans. Multimedia*, 2020, pp. 2517–2525.
[51] C. Wang, X. Xing, Y. Wu, Z. Su, and J. Chen, "DCSFN: Deep cross-scale fusion network for single image rain removal," in *Proc. Conf. ACM Trans. Multimedia*, 2020, pp. 1643–1651.
[52] H. Wang, Q. Xie, Q. Zhao, and D. Meng, "A model-driven deep neural network for single image rain removal," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 3103–3112.
[53] H. Wang, Z. Yue, Q. Xie, Q. Zhao, Y. Zheng, and D. Meng, "From rain generation to rain removal," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14 786–14 796.
[54] T. Wang, X. Yang, K. Xu, S. Chen, Q. Zhang, and R. W. H. Lau, "Spatial attentive single-image deraining with a high quality real rain dataset," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 12 270–12 279.
[55] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[56] W. Wei, D. Meng, Q. Zhao, Z. Xu, and Y. Wu, "Semi-supervised transfer learning for image rain removal," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3877–3886.
[57] W. Wei, L. Yi, Q. Xie, Q. Zhao, D. Meng, and Z. Xu, "Should we encode rain streaks in video as deterministic or stochastic?," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2516–2525.
[58] J. Xiao, M. Zhou, X. Fu, A. Liu, and Z.-J. Zha, "Improving de-raining generalization via neural reorganization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 4967–4976.
[59] W. Yang, J. Liu, and J. Feng, "Frame-consistent recurrent video deraining with dual-level flow," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1661–1670.
[60] W. Yang, J. Liu, S. Yang, and Z. Guo, "Scale-free single image deraining via visibility-enhanced recurrent wavelet learning," *IEEE Trans. Image Process.*, vol. 28, no. 6, pp. 2948–2961, Jun. 2019.
[61] W. Yang, R. T. Tan, J. Feng, J. Liu, Z. Guo, and S. Yan, "Deep joint rain detection and removal from a single image," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1357–1366.

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.

[62] W. Yang, R. T. Tan, S. Wang, Y. Fang, and J. Liu, "Single image deraining: From model-based to data-driven and beyond, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 4059–4077, Nov. 2021.
[63] W. Yang, R. T. Tan, S. Wang, and J. Liu, "Self-learning video rain streak removal: When cyclic consistency meets temporal correspondence," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 1717–1726.
[64] Y. Yang and H. Lu, "A fast and efficient network for single image deraining," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process.*, 2021, pp. 2030–2034.
[65] R. Yasarla and V. M. Patel, "Uncertainty guided multi-scale residual learning-using a cycle spinning CNN for single image de-raining," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8397–8406.
[66] R. Yasarla, A. V. Sindagi, and V. M. Patel, "Syn2real transfer learning for image deraining using gaussian processes," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2726–2736.
[67] Z. Yue, J. Xie, Q. Zhao, and D. Meng, "Semi-supervised video deraining with dynamical rain generator," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 642–652.
[68] S.W. Zamir, *et al.*, "Multi-stage progressive image restoration," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14816–14826.
[69] H. Zhang and V. M. Patel, "Convolutional sparse and low-rank coding-based rain streak removal," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2017, pp. 1259–1267.
[70] H. Zhang and M. VishalPatel, "Density-aware single image deraining using a multi-stream dense network," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 695–704.
[71] H. Zhang, V. Sindagi, and V. M. Patel, "Image de-raining using a conditional generative adversarial network," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 11, pp. 3943–3956, Nov. 2020.
[72] X. Zhang, H. Li, Y. Qi, W. K. Leow, and T. K. Ng, "Rain removal in video by combining temporal and chromatic properties," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2006, pp. 461–464.
[73] M. Zhou *et al.*, "Image de-raining via continual learning," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4905–4914.
[74] H. Zhu *et al.*, "Singe image rain removal with unpaired information: A differentiable programming perspective," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 9332–9339.
[75] L. Zhu, C. W. Fu, D. Lischinski, and P. A. Heng, "Joint bi-layer optimization for single-image rain streak removal," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2545–2553.

Wenhan Yang (Member, IEEE) received the BS and PhD (hons.) degrees in computer science from Peking University, Beijing, China, in 2012 and 2018. He is currently a presidential postdoctoral fellow with the School of Electrical and Electronic Engineering, Nanyang Technology University. His current research interests include image/video processing/restoration, bad weather restoration, human-machine collaborative coding. He has authored mort than 60 technical articles in refereed journals and proceedings, and holds 16 granted patents. He received the IEEE ICME-2020 Best Paper Award, the IEEE ISCAS-2022 MSA-TC Best Paper Award, the IFTC-2017 Best Paper Award, and the IEEE CVPR-2018 UG2 Challenge First Runner-up Award. He was the Candidate of CSIG Best Doctoral Dissertation Award in 2019. He served as the area chair of IEEE ICME-2021/2022, and the organizer of IEEE CVPR-2019/2020/2021/2022 UG2+ Challenge and Workshop.



Robby T. Tan (Member, IEEE) received the PhD degree in computer science from the University of Tokyo. He is currently an associate professor with Yale-NUS College and ECE (Electrical and Computing Engineering), National University of Singapore. Previously, he was an assistant professor with Utrecht University. His research interests include machine learning and computer vision, particularly in dealing with bad weather, physics-based, and motion analysis.

Shiqi Wang (Senior Member, IEEE,) received the BS degree in computer science from the Harbin Institute of Technology in 2008 and the PhD degree in computer application technology from Peking University in 2014. From 2014 to 2016, he was a postdoctoral fellow with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. From 2016 to 2017, he was with the Rapid-Rich Object Search Laboratory, Nanyang Technological University, Singapore, as a research fellow. He is currently an assistant professor with the Department of Computer Science, City University of Hong Kong. He has proposed more than 40 technical proposals to ISO/MPEG, ITU-T, and AVS standards, and authored/coauthored more than 150 refereed journal/conference papers. His research interests include video compression, image/video quality assessment, and image/video search and analysis. He received the Best Paper Award from IEEE ICME 2019, IEEE Multimedia 2018, PCM 2017, and is the coauthor of a paper that received the Best Student Paper Award in the IEEE ICIP 2018.

Alex C. Kot (Fellow, IEEE) has been with the Nanyang Technological University, Singapore since 1991. He was head with the Division of Information Engineering and vice dean research with the School of Electrical and Electronic Engineering. Subsequently, he served as associate dean with the College of Engineering for eight years. He is currently professor and director with Rapid-Rich Object SEarch (ROSE) Lab and NTU-PKU Joint Research Institute. He has published extensively in the areas of signal processing, biometrics, image forensics and security, and computer vision and machine learning. He served as associate editor for more than ten journals, mostly for IEEE transactions. He has served the IEEE SP Society in various capacities such as the general co-chair for the 2004 IEEE International Conference on Image Processing and the vice-president for the IEEE Signal Processing Society. He received the Best Teacher of the Year Award and is a coauthor for several best paper awards including ICPR, IEEE WIFS and IWDW, CVPR Precognition Workshop and VCIP. He was elected as the IEEE distinguished lecturer for the Signal Processing Society and the Circuits and Systems Society. He is a fellow of IES, and a fellow of Academy of Engineering, Singapore.

Jiaying Liu (Senior Member, IEEE) received the PhD (hons.) degree in computer science from Peking University, Beijing China, 2010. She is currently an associate professor, Peking University Boya Young Fellow with the Wangxuan Institute of Computer Technology, Peking University. She has authored more than 100 technical articles in refereed journals and proceedings, and holds 50 granted patents. Her current research interests include multimedia signal processing, compression, and computer vision. She is a senior member of CSIG and CCF. She was a visiting scholar with the University of Southern California, Los Angeles, from 2007 to 2008. She was a visiting researcher with the Microsoft Research Asia in 2015 supported by the Star Track Young Faculties Award. She has served as a member of Multimedia Systems and Applications Technical Committee (MSA TC), and Visual Signal Processing and Communications Technical Committee (VSPC TC) in IEEE Circuits and Systems Society. She received the IEEE ICME-2020 Best Paper Award and IEEE MMSP-2015 Top10% Paper Award. She has also served as the associate editor of *IEEE Transaction on Image Processing*, *IEEE Transaction on Circuit System for Video Technology* and *Elsevier Journal of Visual Communication and Image Representation*, the technical program chair of IEEE ICME-2021/ACM ICMR-2021, the Publicity Chair of IEEE ICME-2020/ICIP-2019, and the Area Chair of CVPR-2021/ECCV-2020/ICCV-2019. She was the APSIPA distinguished lecturer (2016-2017).

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/csdl.

Authorized licensed use limited to: Peking University. Downloaded on March 17,2024 at 06:35:49 UTC from IEEE Xplore. Restrictions apply.