Reconciling Hand-Crafted and Self-Supervised Deep Priors for Video Directional Rain Streaks Removal

Jun-Hao Zhuang, Yi-Si Luo, Xi-Le Zhao, and Tai-Xiang Jiang

***Abstract*—Removing rain streaks in videos has recently received much attention. Existing hand-crafted priors-based methods suffer from limited representation abilities, and supervised deep learning methods need high-quality training data. This paper proposes a novel video rain streaks removal method by reconciling hand-crafted and self-supervised deep priors. The hand-crafted priors include the learned gradient prior, the sparse prior, and the temporal local smooth prior. Meanwhile, a deep convolutional neural network is employed to self-supervisedly capture the deep prior of the clean video without any training data. Our method organically integrates hand-crated priors and self-supervised deep priors to achieve both high generalization abilities and representation abilities. Thus, our method can faithfully remove directional rain streaks in real world videos. To address the resulting model, we introduce an alternating direction multiplier method algorithm. Extensive experiments validate the superiority of our method over state-of-the-art methods.**

***Index Terms*—Video deraining, directional rain streaks, hand-crafted priors, self-supervised deep prior.**

I. Introduction

VIDEOS or single images obtained in outdoor environments often have rain streaks [1]–[5]. The removal of rain streaks is important before subsequent applications. The deraining problems are classed into two categories: single-image deraining and video deraining. The temporal information can be used more effectively by video deraining methods [6]–[8].

Many traditional model-based methods are proposed for deraining [9]–[14]. These methods design hand-crafted priors, transforming deraining into optimization problems and designing reasonable algorithms to solve them. However, hand-crafted priors lack representation abilities and are sometimes hard to accurately characterize rain streaks and natural scenes due to the complex rainy scenario.

In recent years, deep learning-based methods are studied for deraining [15]–[25]. The high expressive power of deep neural networks brings good performance in specific datasets. However, these methods critically depend on training datasets, which lacks generalization abilities to tackle different rainy scenarios with various rain streaks in the real world.

To address these limitations, we propose to reconcile hand-crafted priors and self-supervised deep priors for video deraining. Specifically, this paper has the following contributions:

- We propose to reconcile hand-crafted priors and self-supervised deep priors for video deraining. The **hand-crafted priors** include the learned gradient prior for removing rain streaks with different directions, the sparse prior for capturing the sparsity of rain streaks, and the temporal local smooth prior for keeping the temporal consistency. The **self-supervised deep prior** is to leverage a convolutional neural network (CNN) to capture the deep prior of the clean video without any training data. The proposed method organically integrates hand-crated priors and self-supervised deep priors to achieve both high generalization abilities and representation abilities.
- To address the resulting model, we introduce an efficient alternating direction multiplier method (ADMM) algorithm. Extensive experiments with both simulated data and real world data validate the advantage of our method as compared with state-of-the-art video deraining methods.

II. Symbolic Conventions

We use floral letters for tensors, e.g., $\mathcal{X}$ and capital boldface letters for matrices, e.g., $\mathbf{X}$. The i, j, k-th element of a third-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is denoted by $\mathcal{X}(i, j, k)$. The i-th frontal slice of $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is denoted by $\mathcal{X}^{(i)} \in \mathbb{R}^{I_1 \times I_2}$.

The inner product of two same-sized tensors is denoted by $< \mathcal{A}, \mathcal{B} >= \sum_{i,j,k} \mathcal{A}(i, j, k)\mathcal{B}(i, j, k)$. The Frobenius norm of a tensor $\mathcal{X}$ is defined as $\|\mathcal{X}\|_F = \sqrt{< \mathcal{X}, \mathcal{X} >}$. The ℓ_1-norm of a tensor $\mathcal{X}$ is defined as $\|\mathcal{X}\|_{\ell_1} = \sum_{i,j,k} |\mathcal{X}(i, j, k)|$. The Fourier transformation of a matrix $\mathbf{X}$ is denoted by $\mathcal{F}(\mathbf{X})$.

III. Proposed Method

The degradation process of a rainy video $\mathcal{O} \in \mathbb{R}^{m \times n \times t}$ is formulated as $\mathcal{O} = \mathcal{B} + \mathcal{R}$ [7]. Here, $\mathcal{B} \in \mathbb{R}^{m \times n \times t}$ and $\mathcal{R} \in \mathbb{R}^{m \times n \times t}$ respectively denote the clean video and rain streaks. We suggest the following priors for video deraining.

Manuscript received August 8, 2021; revised October 3, 2021; accepted October 11, 2021. Date of publication October 15, 2021; date of current version November 4, 2021. This work was supported in part by the NSFC under Grants 61876203, 12171072, and 12001446; in part by the Applied Basic Research Project of Sichuan Province under Grant 2021YJ0107; in part by the Key Project of Applied Basic Research in Sichuan Province under Grant 2020YJ0216; in part by the National Key Research and Development Program of China under Grant 2020YFA0714001; and in part by the Fundamental Research Funds for the Central Universities under Grant JBK2102001. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Parvaneh Saeedi. *(Jun-Hao Zhuang and Yi-Si Luo contributed equally to this work.) (Corresponding author: Xi-Le Zhao.)*

Jun-Hao Zhuang, Yi-Si Luo, and Xi-Le Zhao are with the School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu 231567, China (e-mail: junhaozhuang@foxmail.com; yisiluo1221@foxmail.com; xlzhao122003@163.com).

Tai-Xiang Jiang is with the School of Economic Information Engineering, Southwestern University of Finance and Economics, Chengdu 611130, China (e-mail: taixiangjiang@gmail.com).

Digital Object Identifier 10.1109/LSP.2021.3120598

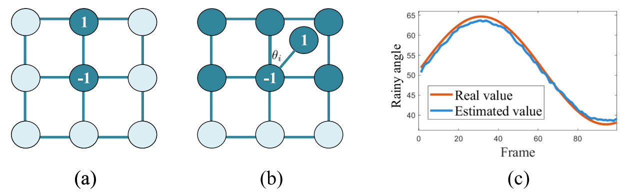

Fig. 1. (a) Illustration of the vertical differential convolutional kernel $\mathbf{\Theta}_y$. (b) Illustration of the proposed LDCK $\mathbf{\Theta}_i$. We use six discrete pixel values to interpolate a curved surface, and then take the point whose angle is θ_i to compute the gradient value. (c) The comparison of the real rain angle and the estimated rain angle θ_i using the proposed method on *highway* **Case 2**.

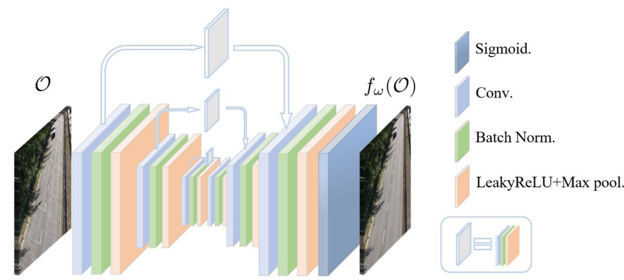

Fig. 2. The U-Net $f_\omega(\cdot)$ used in this work.

A. Priors of Rain Streaks

1) Learned Gradient Prior: Rain streaks are usually piecewise smooth in their falling directions. Thus, minimizing the ℓ_1-norm of the gradient tensor of rain streaks along their falling directions could explore the gradient prior of rain streaks for deraining. Specifically, the gradient tensor is denoted as $\nabla_\theta\mathcal{R}$, where θ denotes the rain direction. Previous methods assumed that rain streaks are vertical [6], [7], [13], where θ is fixed along the vertical direction, i.e.,

$$(\nabla_y\mathcal{R})^{(i)} = \mathbf{\Theta}_y \otimes \mathcal{R}^{(i)} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \otimes \mathcal{R}^{(i)},$$

where $i = 1, 2, \ldots, t$. Here, $\mathbf{\Theta}_y$ denotes the vertical differential convolutional kernel and $\otimes$ denotes the convolution operator. Minimizing $\|\nabla_y\mathcal{R}\|_{\ell_1}$ can remove vertical rain streaks. However, due to the affection of gravity and wind, rain streaks always have different directions in different frames, which can not be well handled by the vertical gradient prior.

To address this issue, we propose the Learned Differential Convolutional Kernel (LDCK) to adaptively capture the rain direction. Specifically, given a rain tensor $\mathcal{R} \in \mathbb{R}^{m\times n\times t}$, we suggest t LDCKs (denoted by $\mathbf{\Theta}_i \in \mathcal{R}^{3\times 3}, i = 1, 2, \ldots, t$) to obtain the gradient tensor $\nabla_\theta\mathcal{R}$:

$$(\nabla_\theta\mathcal{R})^{(i)} = \mathbf{\Theta}_i \otimes \mathcal{R}^{(i)},$$

where $\mathbf{\Theta}_i$ is the LDCK defined as

$$\begin{bmatrix} \frac{1}{2}(\sin^2\theta_i + \sin\theta_i)\cos\theta_i & \frac{1}{2}(\sin^2\theta_i + \sin\theta_i)(1-\cos\theta_i) & 0 \\ \cos^3\theta_i & \cos^2\theta_i(1-\cos\theta_i) - 1 & 0 \\ \frac{1}{2}(\sin^2\theta_i - \sin\theta_i)\cos\theta_i & \frac{1}{2}(\sin^2\theta_i - \sin\theta_i)(1-\cos\theta_i) & 0 \end{bmatrix}^{\mathrm{T}}.$$

Here, θ_i denotes the rain direction in the i-th frame. The proposed LDCK can extract the gradient value along the direction θ_i, see Fig. 1(b). Minimizing $\|\nabla_\theta\mathcal{R}\|_{\ell_1}$ can reveal the gradient prior of rain streaks along different directions θ_i so that directional rain streaks can be faithfully removed.

The LDCK $\mathbf{\Theta}_i$ is self-supervisedly learned by minimizing $\|\nabla_\theta\mathcal{R}\|_{\ell_1}$. Specifically, θ_i is the learnable parameter, i.e., we learn a direction of rain streaks θ_i for each frame so that rain streaks with different directions can be faithfully removed. When $\theta_i = 0$, the kernel $\mathbf{\Theta}_i$ degrades to $\mathbf{\Theta}_y$, i.e., the vertical gradient prior [6], [7] is just a special case of our method.

Compared with direction total variation (DTV) [26], [27], our method can self-supervisedly explore the direction of rain streaks, while DTV fixes the direction in advance. Thus, our method is more suitable for removing directional rain streaks.

2) Sparse Prior: The rain streaks are sparse [7]. Thus, we minimize ℓ_1-norm of the rain streaks $\mathcal{R}$ to depict the sparsity.

B. Priors of the Clean Video

1) Temporal Local Smooth Prior: The clean video is local smooth along its temporal dimension [7]. Thus, we minimize ℓ_1-norm of the gradient tensor of $\mathcal{B}$ along the temporal dimension, i.e., $\|\nabla_t\mathcal{B}\|_{\ell_1}$, to preserve temporal consistency.

2) Self-Supervised Deep Prior: To capture the clean rain-free video, we employ a U-Net CNN to self-supervisedly capture the rain-free video, i.e., $\mathcal{B} = f_\omega(\mathcal{O})$. Here, $f_\omega(\cdot)$ denotes the U-Net CNN parameterized by ω, whose structure is displayed in Fig. 2. We employ three layers in both encoding and decoding networks. The number of our network parameters is 11.08 K and the forward propagation requires 240.58 G FLOPs for data of size $240 \times 320 \times 100$. This strategy is motivated by the deep image prior [28]–[30], where the CNN itself is proved to have abilities to capture natural images without any training data. The powerful representation abilities of the deep CNN can effectively capture the rain-free video to better preserve the image details.

C. Deraining Model

Based on the priors, we can have the deraining model as

$$\begin{aligned} \min_{\mathcal{B},\mathcal{R},\theta_i} \ & \frac{1}{2}\|\mathcal{O}-\mathcal{B}-\mathcal{R}\|_F^2 + \alpha_1\|\nabla_\theta\mathcal{R}\|_{\ell_1} + \alpha_2\|\mathcal{R}\|_{\ell_1} \\ & + \alpha_3\|\nabla_t\mathcal{B}\|_{\ell_1} \\ \text{s.t.} \ & \mathcal{B} = f_\omega(\mathcal{O}),\ \mathbf{0} \le f_\omega(\mathcal{O}) \le \mathcal{O},\ \mathbf{0} \le \mathcal{R} \le \mathcal{O}, \end{aligned} \tag{1}$$

where $\|\mathcal{O}-\mathcal{B}-\mathcal{R}\|_F^2$ is the fidelity term.

D. Algorithm

To tackle (1), we develop an efficient ADMM algorithm. By introducing three auxiliary variables $\mathcal{V}_k \in \mathbb{R}^{m\times n\times t}$ $(k = 1, 2, 3)$, (1) can be re-written as

$$\begin{aligned} \min_{\omega,\mathcal{R},\theta_i,\mathcal{V}_k} \ & \frac{1}{2}\|\mathcal{O}-f_\omega(\mathcal{O})-\mathcal{R}\|_F^2 + \alpha_1\|\mathcal{V}_1\|_{\ell_1} + \\ & \alpha_2\|\mathcal{V}_2\|_{\ell_1} + \alpha_3\|\mathcal{V}_3\|_{\ell_1} \\ \text{s.t.} \ & \mathcal{V}_1 = \nabla_\theta\mathcal{R},\ \ \mathcal{V}_2 = \mathcal{R},\ \ \mathcal{V}_3 = \nabla_t f_\omega(\mathcal{O}), \\ & \mathbf{0} \le f_\omega(\mathcal{O}) \le \mathcal{O},\ \ \mathbf{0} \le \mathcal{R} \le \mathcal{O}. \end{aligned} \tag{2}$$

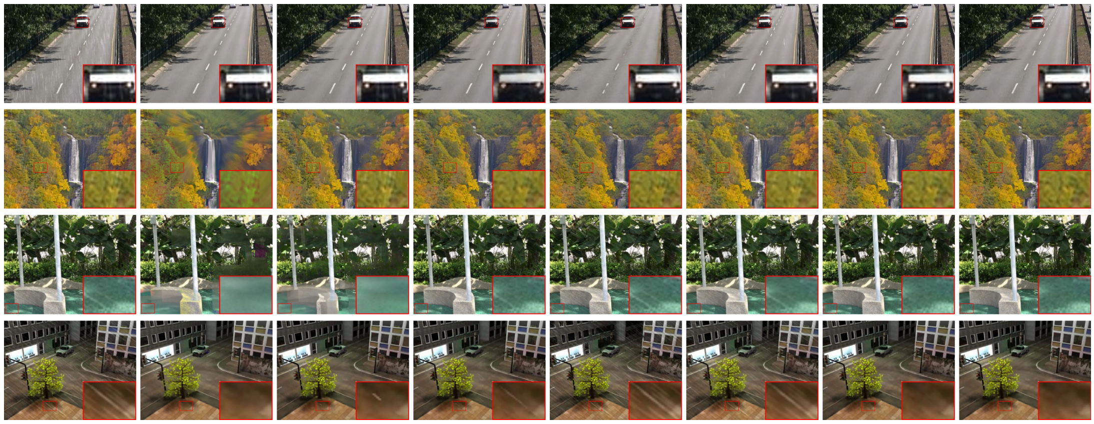

Fig. 3. The deraining results by different methods on simulated data *highway*, *waterfall* for **Case 1** and *park*, *truck* for **Case 2**.

The corresponding augmented Lagrangian function of (2) is

$$\begin{aligned}&\mathcal{L}(\omega,\mathcal{R},\theta_i,\mathcal{V}_k,\Lambda_k)=\\&\quad\frac{1}{2}\|\mathcal{O}-f_\omega(\mathcal{O})-\mathcal{R}\|_F^2+\alpha_1\|\mathcal{V}_1\|_{\ell_1}+\alpha_2\|\mathcal{V}_2\|_{\ell_1}+\\&\quad\alpha_3\|\mathcal{V}_3\|_{\ell_1}+\frac{\mu}{2}\|\nabla_\theta\mathcal{R}-\mathcal{V}_1\|_F^2+\frac{\mu}{2}\|\mathcal{R}-\mathcal{V}_2\|_F^2+\\&\quad\frac{\mu}{2}\|\nabla_t f_\omega(\mathcal{O})-\mathcal{V}_3\|_F^2+<\Lambda_1,\nabla_\theta\mathcal{R}-\mathcal{V}_1>+\\&\quad<\Lambda_2,\mathcal{R}-\mathcal{V}_2>+<\Lambda_3,\nabla_t f_\omega(\mathcal{O})-\mathcal{V}_3>,\end{aligned}$$

where μ is the penalty parameter and Λ_k $(k=1,2,3)$ are multipliers. Under the ADMM framework, the problem can be turned into the following sub-problems.

1) $\mathcal{V}$ Sub-Problems: The $\mathcal{V}_k$ $(k=1,2,3)$ sub-problems are

$$\begin{cases}\mathcal{V}_1^{t+1}=\arg\min_{\mathcal{V}_1}\frac{\mu}{2}\left\|\nabla_\theta\mathcal{R}^t+\frac{\Lambda_1^t}{\mu}-\mathcal{V}_1\right\|_F^2+\alpha_1\|\mathcal{V}_1\|_{\ell_1}\\\mathcal{V}_2^{t+1}=\arg\min_{\mathcal{V}_2}\frac{\mu}{2}\left\|\mathcal{R}^t+\frac{\Lambda_2^t}{\mu}-\mathcal{V}_2\right\|_F^2+\alpha_2\|\mathcal{V}_2\|_{\ell_1}\\\mathcal{V}_3^{t+1}=\arg\min_{\mathcal{V}_3}\frac{\mu}{2}\left\|\nabla_t f_\omega(\mathcal{O})+\frac{\Lambda_3^t}{\mu}-\mathcal{V}_3\right\|_F^2+\alpha_3\|\mathcal{V}_3\|_{\ell_1},\end{cases}$$

which can be exactly solved by

$$\begin{cases}\mathcal{V}_1^{t+1}=\mathit{Soft}_{\frac{\alpha_1}{\mu}}\left(\nabla_\theta\mathcal{R}^t+\frac{\Lambda_1^t}{\mu}\right)\\\mathcal{V}_2^{t+1}=\mathit{Soft}_{\frac{\alpha_2}{\mu}}\left(\mathcal{R}^t+\frac{\Lambda_2^t}{\mu}\right)\\\mathcal{V}_3^{t+1}=\mathit{Soft}_{\frac{\alpha_3}{\mu}}\left(\nabla_t f_\omega(\mathcal{O})+\frac{\Lambda_3^t}{\mu}\right),\end{cases}\tag{3}$$

where $\mathit{Soft}_a(\cdot)$ denotes the soft-thresholding operator [6], [7] with the threshold a.

2) ω and θ_i Sub-Problem: The ω and θ_i sub-problem is

$$\begin{aligned}&\omega,\theta_i\in\arg\min_{\omega,\theta_i}\frac{1}{2}\left\|\mathcal{O}-f_\omega(\mathcal{O})-\mathcal{R}^t\right\|_F^2+\\&\frac{\mu}{2}\left\|\nabla_t f_\omega(\mathcal{O})-\mathcal{V}_3^t+\frac{\Lambda_3^t}{\mu}\right\|_F^2+\frac{\mu}{2}\left\|\nabla_\theta\mathcal{R}^t-\mathcal{V}_1^t+\frac{\Lambda_1^t}{\mu}\right\|_F^2.\end{aligned}\tag{4}$$

We directly use the efficient adaptive moment estimation (Adam) algorithm [31] to tackle the non-convex problem (4). In each iteration of ADMM, we employ one iteration of the Adam to update ω and θ_i.

Algorithm 1: Video Directional Rain Streaks Removal.

1: **Input:** Rainy video $\mathcal{O}$;
2: **Initialization:** Randomly initialize ω, $\mathcal{R}=\mathbf{0}$, $\Lambda=\mathbf{0}$;
3: **while** not satisfy the stopping criterion **do**
4: Update $\mathcal{V}_k$ $(k=1,2,3)$ via (3);
5: Update ω and θ_i $(i=1,2,\ldots,t)$ via (4);
6: Update $\mathcal{R}$ via (5);
7: Update Λ_k $(k=1,2,3)$ via (6);
8: **end while**
9: **Output:** The estimate of the clean rain-free video $\mathcal{B}=f_\omega(\mathcal{O})$;

3) $\mathcal{R}$ Sub-Problem: The $\mathcal{R}$ sub-problem is

$$\begin{aligned}\mathcal{R}^{t+1}=&\arg\min_{\mathcal{R}}\frac{1}{2}\|\mathcal{O}-f_\omega(\mathcal{O})-\mathcal{R}\|_F^2+\\&\frac{\mu}{2}\left\|\nabla_\theta\mathcal{R}-\mathcal{V}_1^t+\frac{\Lambda_1^t}{\mu}\right\|_F^2+\frac{\mu}{2}\left\|\mathcal{R}-\mathcal{V}_2^t+\frac{\Lambda_2^t}{\mu}\right\|_F^2,\end{aligned}$$

which can be exactly solved by:

$$\begin{aligned}&\left(\mathcal{R}^{t+1}\right)^{(i)}=\mathcal{F}^{-1}\left(\left(\mathcal{F}\left(\mathcal{O}^{(i)}-(f_\omega(\mathcal{O}))^{(i)}+\mu\left(\mathcal{V}_2^t-\frac{\Lambda_2^t}{\mu}\right)^{(i)}\right)+\right.\right.\\&\left.\left.\mu\overline{\mathcal{F}(\mathbf{\Theta}_i)}\mathcal{F}\left(\left(\mathcal{V}_1+\frac{\Lambda_1^t}{\mu}\right)^{(i)}\right)\right)\left(1+\mu+\mu\overline{\mathcal{F}(\mathbf{\Theta}_i)}\mathcal{F}(\mathbf{\Theta}_i)\right)^{-1}\right),\end{aligned}\tag{5}$$

where $\mathcal{F}^{-1}$ denotes the inverse transform of $\mathcal{F}$ and $\overline{\mathbf{X}}$ denotes the conjugate matrix of $\mathbf{X}$.

4) Λ Updating: The updates of Λ_k $(k=1,2,3)$ are

$$\begin{cases}\Lambda_1^{t+1}=\Lambda_1^t+\mu(\nabla_\theta\mathcal{R}^t-\mathcal{V}_1^t)\\\Lambda_2^{t+1}=\Lambda_2^t+\mu(\mathcal{R}^t-\mathcal{V}_2^t)\\\Lambda_3^{t+1}=\Lambda_3^t+\mu(\nabla_t f_\omega(\mathcal{O})-\mathcal{V}_3^t).\end{cases}\tag{6}$$

After each iteration, we shrink the intensities of $f_\omega(\mathcal{O})$ and $\mathcal{R}$ to meet $\mathbf{0}\le f_\omega(\mathcal{O})\le\mathcal{O}$ and $\mathbf{0}\le\mathcal{R}\le\mathcal{O}$. In Algorithm 1, we summarize the above ADMM algorithm.

Fig. 4. The deraining results by different methods on real rainy videos *parking*, *wall*, and *cross*.

IV. Experiments

A. Experimental Settings

We test our method on both simulated data and real world data. Four clean videos named *highway, waterfall, park*, and *truck* are adopted to generate simulated data. Three types of rain are generated. **Case 1** contains light rain streaks with rain directions in $[-15^\circ, 15^\circ]$, where the directions are different in different frames but are the same in a single frame. In real scenarios, rain streaks always have larger angles due to the wind and gravity. Thus, we consider **Case 2**, which contains light rain streaks with larger angles in $[35^\circ, 65^\circ]$. To further test the robustness of our method, we consider **Case 3**, where the rain streaks are captured by cameras in black background [33]. For real experiments, we select three real world rainy videos named *parking* of size $288 \times 368 \times 20$, *wall* of size $480 \times 640 \times 20$, and *cross* of size $480 \times 640 \times 20$.

We use the peak signal-to-noise ratio (PSNR) and the structure similarity (SSIM) to evaluate the deraining results. The competing methods include three model-based video deraining methods (ICCV'17 [32], MS-CSC [33], and FastDeRain [7]) and two deep learning-based video deraining methods (J4R-Net [19] and S2VD [34]).

TABLE I
The Quantitative Results by Different Methods

Case	Metric	Rainy	ICCV'17	MS-CSC	FastDeRain	J4R-Net	S2VD	Ours
highway $240 \times 320 \times 100$								
Case 1	PSNR	30.95	35.09	37.78	39.75	24.10	35.42	**42.06**
	SSIM	0.8618	0.9730	0.9757	0.9875	0.8893	0.9576	**0.9883**
Case 2	PSNR	29.77	33.43	37.61	39.46	23.06	31.92	**40.64**
	SSIM	0.8350	0.9413	0.9758	0.9808	0.7645	0.8921	**0.9853**
Case 3	PSNR	22.90	23.78	24.80	29.87	21.99	25.65	**34.34**
	SSIM	0.9230	0.9530	0.9577	0.9722	0.8785	0.9564	**0.9802**
Average time (s)			776.3	466.2	8.6	245.8	6.1	1679.4
waterfall $240 \times 320 \times 100$								
Case	Metric	Rainy	ICCV'17	MS-CSC	FastDeRain	J4R-Net	S2VD	Ours
Case 1	PSNR	30.20	26.43	30.60	39.59	31.33	35.26	**41.30**
	SSIM	0.9331	0.7467	0.8350	0.9802	0.9411	0.9446	**0.9876**
Case 2	PSNR	31.41	26.42	29.95	34.37	28.11	32.00	**39.29**
	SSIM	0.9416	0.7580	0.8265	0.9368	0.8614	0.9086	**0.9802**
Case 3	PSNR	24.12	22.76	22.73	27.14	23.89	24.46	**31.38**
	SSIM	0.9463	0.7836	0.8550	0.9469	0.9395	0.9484	**0.9814**
Average time (s)			826.6	481.1	8.1	250.2	6.1	1625.9
park $240 \times 320 \times 100$								
Case	Metric	Rainy	ICCV'17	MS-CSC	FastDeRain	J4R-Net	S2VD	Ours
Case 1	PSNR	30.95	24.07	22.62	33.60	29.81	33.40	**35.76**
	SSIM	0.8803	0.8088	0.6678	0.9607	0.9529	0.9531	**0.9746**
Case 2	PSNR	29.87	23.99	22.71	33.71	29.18	32.69	**35.52**
	SSIM	0.8584	0.8062	0.6706	0.9566	0.9241	0.9471	**0.9712**
Case 3	PSNR	22.36	20.86	20.54	28.98	25.76	26.30	**33.55**
	SSIM	0.8900	0.8128	0.6541	0.9449	0.9510	0.9554	**0.9786**
Average time (s)			776.3	466.2	8.6	235.6	6.4	1542.6
truck $240 \times 320 \times 100$								
Case	Metric	Rainy	ICCV'17	MS-CSC	FastDeRain	J4R-Net	S2VD	Ours
Case 1	PSNR	30.95	33.21	36.43	41.25	29.94	36.25	**42.48**
	SSIM	0.8945	0.9714	0.9806	0.9865	0.9400	0.9674	**0.9905**
Case 2	PSNR	29.79	32.04	36.85	39.34	27.14	32.16	**40.20**
	SSIM	0.8769	0.9684	0.9814	0.9852	0.8249	0.9113	**0.9858**
Case 3	PSNR	22.36	22.68	23.64	32.92	24.33	24.63	**36.73**
	SSIM	0.9269	0.9353	0.9456	0.9758	0.9200	0.9514	**0.9865**
Average time (s)			927.5	479.5	9.1	247.3	6.2	1631.1

B. Experimental Results

The numerical results for simulated data are displayed in Table I. We can see that our method outperforms competing methods. The qualitative results are shown in Fig. 3. We can observe that model-based methods miss some image details for **Case 2** due to the lack of representation abilities. Deep learning-based methods could not remove directional rain streaks in **Case 2** because they lack generalization abilities. In contrast, our method shows better performance. The hand-crafted priors bring high generalization abilities. Especially the proposed LDCK could accurately estimate the rain directions, see Fig. 1(c). Meanwhile, the deep prior brings representation abilities to capture complex video scenarios.

The real experimental results are displayed in Fig. 4. We can observe that our method removes rain streaks well and preserves image detail better than other methods. The good performance of our method is because that LDCKs can remove rain streaks with different directions and the deep prior can capture the image details of clean videos.

V. Conclusion

In this letter, we propose to reconcile the learned gradient prior, the temporal local smooth prior, the sparse prior, and the self-supervised deep prior for video directional rain streaks removal. To address the resulting model, we introduce an ADMM algorithm. Extensive experiments verify the advantage of our method over state-of-the-art methods.

REFERENCES

[1] 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.

[2] 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 Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 1685–1694.

[3] R. Liu, Z. Jiang, L. Ma, X. Fan, H. Li, and Z. Luo, "Deep layer prior optimization for single image rain streaks removal," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2018, pp. 1408–1412.

[4] R. Liu, Z. Jiang, X. Fan, and Z. Luo, "Knowledge-driven deep unrolling for robust image layer separation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1653–1666, May 2020.

[5] H. Wang, Q. Xie, Q. Zhao, and D. Meng, "A model-driven deep neural network for single image rain removal," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2020, pp. 3100–3109.

[6] T. Jiang, T. Huang, X. Zhao, L. Deng, and Y. Wang, "A novel tensor-based video rain streaks removal approach via utilizing discriminatively intrinsic priors," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 2818–2827.

[7] T.-X. Jiang, T.-Z. Huang, X.-L. Zhao, L.-J. 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.

[8] T. Liu, M. Xu, and Z. Wang, "Removing rain in videos: A large-scale database and a two-stream ConvLSTM approach," in *Proc. IEEE Int. Conf. Multimedia Expo.*, 2019, pp. 664–669.

[9] Y. Luo, Y. Xu, and H. Ji, "Removing rain from a single image via discriminative sparse coding," in *Proc. IEEE Int. Conf. Comput. Vision*, 2015, pp. 3397–3405.

[10] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, "Rain streak removal using layer priors," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 2736–2744.

[11] W. Ren, J. Tian, Z. Han, A. Chan, and Y. Tang, "Video desnowing and deraining based on matrix decomposition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 2838–2847.

[12] L. Zhu, C.-W. Fu, D. Lischinski, and P. Heng, "Joint bi-layer optimization for single-image rain streak removal," in *Proc. IEEE Int. Conf. Comput. Vision*, 2017, pp. 2545–2553.

[13] L. Deng, T. Huang, X. Zhao, and T. Jiang, "A directional global sparse model for single image rain removal," *Appl. Math. Modelling*, vol. 59, pp. 662–679, 2018.

[14] J. Kim, J. Sim, and C. 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.

[15] 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/CVF Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 8397–8406.

[16] H. Zhang and V. M. Patel, "Density-aware single image de-raining using a multi-stream dense network," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 695–704.

[17] D. Ren, W. Zuo, Q. Hu, P. Zhu, and D. Meng, "Progressive image deraining networks: A better and simpler baseline," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 3932–3941.

[18] X. Fu, B. Liang, Y. Huang, X. Ding, and J. Paisley, "Lightweight pyramid networks for image deraining," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 6, pp. 1794–1807, Jun. 2020.

[19] J. Liu, W. Yang, S. Yang, and Z. Guo, "Erase or fill? deep joint recurrent rain removal and reconstruction in videos," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 3233–3242.

[20] Y.-T. Wang, X.-L. Zhao, T.-X. Jiang, L.-J. Deng, Y. Chang, and T.-Z. Huang, "Rain streaks removal for single image via kernel-guided convolutional neural network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 8, pp. 3664–3676, Aug. 2021.

[21] P. Mu, J. Chen, R. Liu, X. Fan, and Z. Luo, "Learning bilevel layer priors for single image rain streaks removal," *IEEE Signal Process. Lett.*, vol. 26, no. 2, pp. 307–311, Feb. 2019.

[22] 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/CVF Conf. Comput. Vision Pattern Recognit.*, 2020, pp. 1717–1726.

[23] 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/CVF Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 6286–6295.

[24] X. Fu, J. Huang, X. Ding, Y. Liao, and J. Paisley, "Clearing the skies: A deep network architecture for single-image rain removal," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2944–2956, Jun. 2017.

[25] 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 Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 1715–1723.

[26] I. Bayram and M. E. Kamasak, "Directional total variation," *IEEE Signal Process. Lett.*, vol. 19, no. 12, pp. 781–784, Dec. 2012.

[27] Y. Wang, T. Huang, X. Zhao, L. Deng, and T. Jiang, "Rain streaks removal for single image via directional total variation regularization," in *Proc. IEEE Int. Conf. Image Process.*, 2019, pp. 2801–2805.

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

[29] Y. Miao, X. Zhao, X. Fu, J. Wang, and Y. Zheng, "Hyperspectral denoising using unsupervised disentangled spatiospectral deep priors," *IEEE Trans. Geosci. Remote Sens.*, early access 2021. doi: 10.1109/TGRS.2021.3106380.

[30] Y. Luo, X. Zhao, T. Jiang, Y. Zheng, and Y. Chang, "Hyperspectral mixed noise removal via spatial-spectral constrained unsupervised deep image prior," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 9435–9449, 2021.

[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learning Representations*, 2015.

[32] 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. Vision*, 2017, pp. 2535–2544.

[33] M. Li *et al.*, "Video rain streak removal by multiscale convolutional sparse coding," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 6644–6653.

[34] Z. Yue, J. Xie, Q. Zhao, and D. Meng, "Semi-supervised video deraining with dynamical rain generator," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2021, pp. 642–652.