RAIN STREAKS REMOVAL FOR SINGLE IMAGE VIA DIRECTIONAL TOTAL VARIATION REGULARIZATION

Yugang Wang, Ting-Zhu Huang, Xi-Le Zhao, Liang-Jian Deng, Tai-Xiang Jiang

School of Mathematical Sciences/Research Center for Image and Vision Computing, University of Electronic Science and Technology of China, Chengdu, Sichuan, 611731, P. R. China.

ABSTRACT

Images captured in rainy conditions are often corrupted by unexpected rain streaks, which severely degrade the performance of subsequent processes in outdoor computer vision systems. In this paper, we exploit the directional smoothness of rain streaks for the single-image rain streaks removal and propose a convex model that uses the directional total variation (DTV) to characterize the smoothness of rain streaks in arbitrary orientations. The proposed model consists of four terms: the fidelity term, the $\ell_1$ norm for the sparsity of rain streaks, and two DTV regularization terms for the directional smoothness and the piecewise smoothness of rain streaks and rain-free backgrounds, respectively. To solve the proposed model, we develop an efficient algorithm based on the alternating direction method of multipliers (ADMM) framework. Extensive experimental results on both synthetic and real rainy images show that our method outperforms the recent state-of-the-art methods visually and quantitatively.

Index Terms— Single-image rain streaks removal, directional total variation (DTV), alternating direction method of multipliers

1. INTRODUCTION

Outdoor images taken in rainy weather usually suffer from the undesirable interference caused by the inevitable falling raindrops. The captured rain images often contain bright rain streaks which may obstruct the background [1], distort the colors, and blur the true scenes. These types of visibility degradation would severely affect the performance of many algorithms in outdoor computer vision systems. Thus the removal of rain streaks is an essential and urgent issue and has received much attention in recent years.

Garg et al. [2, 3] first proposed to detect and remove the rain streaks from videos by adopting the photometric appearance of raindrops or altering camera settings. In the last few decades, abundant literatures aimed at video rain streaks removal using the spatial-temporal redundancy of videos [4, 5]. Comparing with removing rain from videos, single image rain streaks removal is a more challenging task since only one rainy frame is given. To effectively estimate the rain-free content from a rainy image, many existing methods had explored various priors for both rain streaks and rain-free backgrounds and achieved impressive results. The dictionary-based sparse prior [6–8] was used via finding the discriminative sparse representations of clean images or rain streaks under the given or learned dictionary. Chen et al. [9] leveraged a low-rank prior for the rain streaks patches. Kim et al. [10] and Chang et al. [11] incorporated the nonlocal self-similarity property into their rain removal methods. Li et al. [1] removed the rain streaks using Gaussian Mixed Model (GMM) to characterize the rain layer. The methods based on deep networks were recently proposed for the deraining task by learning deep features of both rain streaks and clean images [12–16]. Some approaches [17–19] took the directional smooth property of
rain streaks into account. These methods assumed the rain direction to be vertical and used the unidirectional total variation to characterize the directional smoothness of rain streaks. However, in most real-world applications, images are often corrupted by oblique rain streaks so that the vertical smoothness of rain streaks would not be satisfied (see Fig. 1). Even though Deng et al. [17] handled the oblique rain streaks by rotating the image to get the vertical rain streaks, the operations of rotation often cause the resampling of image pixels and inevitable degrade the images.

To deal with the oblique rain streaks in more general case, in this paper, we employ the directional total variation (DTV) regularization, which can characterize the directional smoothness in arbitrary directions, to regularize the oblique rain streaks and the rain-free backgrounds. The contributions of this paper can be summarized as follows:

- We consider the directional smoothness of oblique rain streaks, and propose a convex model utilizing DTV to characterize the smoothness of rain streaks in arbitrary directions. To the best of our knowledge, our work is the first attempt to bring DTV regularization in rain streaks removal task.
- To tackle the proposed model, we develop an efficient algorithm based on the alternating direction method of multipliers (ADMM) framework where the convergence can be theoretically guaranteed. Extensive experiments demonstrate the superiority of our method visually and quantitatively.

The rest of this paper is organized as follows. In Section 2, we introduce the DTV and present the proposed method. Section 3 shows the experimental results on both synthetic and real rain images. The conclusion is given in Section 4.

2. RAIN STREAKS REMOVAL VIA DTV REGULARIZATION

2.1. Directional Total Variation

We define $\nabla_1$ and $\nabla_2$: $\mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \times n}$ as the first-order difference operator, respectively, along horizontal and vertical directions with circulant boundary condition. Then for a discrete-space image $X \in \mathbb{R}^{m \times n}$, the directional total variation (DTV) [20, 21] of $X$ can be formulated as

$$\text{DTV}_{\theta,a}(X) = \sum_{i,j} \|\nabla_{\theta,a} X(i,j)\|_2,$$

where $a \in [0, 1]$ is a positive parameter and $0 < \theta < 180^\circ$ can be an arbitrary direction. $\nabla_{\theta,a} X(i,j)$ can be written as

$$\nabla_{\theta,a} X(i,j) = \Lambda_a R_{\theta} \left( \begin{array}{c} \nabla_1 X(i,j) \\ \nabla_2 X(i,j) \end{array} \right),$$

where

$$\Lambda_a = \begin{pmatrix} 1 & 0 \\ 0 & a \end{pmatrix}, \quad R_{\theta} = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix}$$

are the translation matrix and rotation matrix, respectively. The DTV has been widely used for modeling the images whose texture mainly has one direction $\theta$ in image restoration literatures [20, 21]. Form Fig. 1, we can observe that the vertical gradient fails to model the oblique rain streaks, while DTV can effectively characterize the directional smoothness and recover the better rain-free background.

2.2. The Proposed Model for Rain Streaks Removal

The degradation model [1] of a rainy image can be written as

$$O = B + R,$$

where $O$, $B$, and $R \in \mathbb{R}^{m \times n}$ are the observed rain image, the desired clean image, and the rain streaks layer, respectively. To recover $B$ from $O$, we use the DTV regularizer to characterize the directional smoothness of $R$. In addition, an $\ell_1$ norm is also utilized to enhance the sparsity of rain streaks. To model the clean background $B$, we consider the piece-wise smoothness of natural images and also utilize the the DTV regularization for $B$ along rain-perpendicular direction. Thus, the proposed model can be formulated as follows

$$\min_{O,R} \frac{1}{2} \|O - B - R\|_F^2 + \lambda_1 \sum_{i,j} \|\nabla_{\theta,a_1} R(i,j)\|_2$$

$$+ \lambda_2 \|R\|_1 + \lambda_3 \sum_{i,j} \|\nabla_{\theta,a_2} B(i,j)\|_2$$

$$s.t. \quad 0 \leq B \leq O, 0 \leq R \leq O,$$

where $\theta$ is the rain direction and $\theta^\perp = \theta + 90^\circ$ is the orthogonal direction of $\theta$. $\lambda_1$, $\lambda_2$, and $\lambda_3$ are positive parameters. In the next section, we develop an efficient algorithm based on ADMM framework [22] to solve this proposed model.

2.3. The Numerical Algorithm

Based on the ADMM principle, we introduce several auxiliary variables and set the constraints $(U_1^T, U_2^T)^T = \nabla_{\theta,a_1} R, (M_1^T, M_2^T)^T = \nabla_{\theta,a_2} B, V = R, P = R$ and $Q = B$. Then the augmented Lagrangian function of (5) can be written as

$$\mathcal{L}(\Phi, \Gamma_1, \Gamma_2, \Gamma_3, K_2, K_2, L_1, L_2) =$$

$$\frac{1}{2} \|O - B - R\|_F^2 + \lambda_1 \sum_{i,j} \|(U_1(i,j), U_2(i,j))\|_2^2$$

$$+ \lambda_2 \|V\|_1 + \lambda_3 \sum_{i,j} \|(M_1(i,j), M_2(i,j))\|_2^2$$

$$+ \frac{\beta}{2} \|(U_1^T, U_2^T)^T - \nabla_{\theta,a} R + \frac{1}{\beta} (K_1^T, K_2^T)^T\|_F^2,$$

where $\lambda_a = \begin{pmatrix} 1 & 0 \\ 0 & a \end{pmatrix}, R_{\theta} = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix}$.
\[
+ \frac{\beta}{2} \| (M_1^T, M_2^T)^T - \tilde{\nabla}_{\theta+\alpha_2} B + \frac{1}{\beta} (L_1^T, L_2^T)^T \|_F^2 \\
+ \frac{\beta}{2} \| V - R + \frac{1}{\beta} \Gamma_1 \|_F^2 + \frac{\beta}{2} \| P - R + \frac{1}{\beta} \Gamma_2 \|_F^2 \\
+ \frac{\beta}{2} \| Q - B + \frac{1}{\beta} \Gamma_3 \|_F^2 + \rho(P) + \rho(Q).
\]

where \( \Phi = \{ B, R, U_1, U_2, M_1, M_2, P, V, Q \} \) is the set of variables, \( K_1, K_2, L_1, L_2, \Gamma_1, \Gamma_2, \Gamma_3 \) are the Lagrangian multipliers and \( \beta \) is a positive scalar. \( \rho(\cdot) \) is the indicator function defined as

\[
\rho(X(i,j)) = \begin{cases} 
0, & 0 \leq X(i,j) \leq O(i,j), \\
+\infty, & \text{otherwise}.
\end{cases}
\]

The sub-problems of \((U_1^T, U_2^T)^T\) and \((M_1^T, M_2^T)^T\) have closed-form solutions which can be calculated by the shrinkage formula [22]

\[
\begin{align*}
(U_1^{k+1}(i,j)) & = \max \left( \| d_{ij}^k \|_2 - \frac{\lambda_1}{\beta}, 0 \right) \frac{d_{ij}^k}{\| d_{ij}^k \|_2}, \\
(M_1^{k+1}(i,j)) & = \max \left( \| w_{ij}^k \|_2 - \frac{\lambda_3}{\beta}, 0 \right) \frac{w_{ij}^k}{\| w_{ij}^k \|_2},
\end{align*}
\]

where \( d_{ij}^k \) and \( w_{ij}^k \) are vectors defined as

\[
\begin{align*}
d_{ij}^k & = \nabla_{\theta+\alpha_2} R^k(i,j) - \frac{1}{\beta} (K_1^k(i,j), K_2^k(i,j))^T, \\
w_{ij}^k & = \nabla_{\theta+\alpha_2} B^k(i,j) - \frac{1}{\beta} (L_1^k(i,j), L_2^k(i,j))^T.
\end{align*}
\]

The variable \( V \) can be solved by a shrinkage operator

\[
V^{k+1}(i,j) = \text{shrink}(R^k(i,j) - \frac{\Gamma_1^k(i,j)}{\beta}, \frac{\lambda_2}{\beta}),
\]

where \( \text{shrink} \) is the soft-shrinkage operator [23].

\( P^{k+1} \) and \( Q^{k+1} \) can be respectively updated by projecting \( R^k - 1/\beta \Gamma_1^k \) and \( B^k - 1/\beta \Gamma_3^k \) between 0 and \( O \).

\( B \) and \( R \) should be jointly solved. Their sub-problem is a least squares problem which could be efficiently solved by fast Fourier transforms (FFTs) and Cramer’s rule [24].

At last, each Lagrangian multiplier are updated by gradient ascent optimization with the other variables fixed [22].

We set \( \alpha_1 = 0 \) and \( \alpha_2 = 0.5 \) in all experiments. To estimate the rain direction \( \theta \), we employ the strategy of [25] to first search the rain-dominated patches of a rain image, which contain little background details. Then we find the directions in which the smallest DTV values of these rain-dominated patches can be achieved. At last, we calculate the average of these directions to approximate \( \theta \).

3. EXPERIMENTAL RESULTS

In this section, we test our method on both synthetic and real-world rainy images. Three state-of-the-art methods: GMM layer prior (GMMLP) based method [1], convolutional neural network (CNN) method [26], and the unidirectional global sparse method (UGSM) [17] are compared with the proposed DTV regularized sparse (DTVRS) method. All experiments are implemented in MATLAB 2017b on an Intel 3.70 GHz PC with 16GB RAM, and a GTX1060 GPU. We efficiently implement DTVRS on the GPU device. The computing time for our method is about 14 seconds for running a color image of size 366 × 550 × 3.

3.1. Synthetic Experiments

We generate the synthetic rain streaks using the same strategy as [18], which first adds the “salt and pepper” noise on a zero image and then convolutes it with a motion blur kernel to form the rain streaks. The density of “salt and pepper” noise is taken randomly in the range of \([0.05, 0.1]\) and the length of motion is set in \([10, 30]\). Empirically, we set \( \beta = 0.1 \), and select \( \lambda_1, \lambda_2, \) and \( \lambda_3 \) in the sets of \([0.1, 0.2], [0.001, 0.002, 0.003], \) and \([0.02, 0.03, 0.04]\), respectively. For quantitative assessment, the peak signal-to-noise ratio (PSNR) and the structural similarity (SSIM) are calculated for all the result images.

We firstly adopt three nature images named “umbrella”, “girl”, and “Sydney” which are widely used in image derain-
Table 1. Quantitative evaluation on synthetic images with rain streaks of different directions.

<table>
<thead>
<tr>
<th>Images</th>
<th>umbrella</th>
<th>girl</th>
<th>Sydney</th>
</tr>
</thead>
<tbody>
<tr>
<td>Directions</td>
<td>θ°</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>θ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>30°</td>
<td>75°</td>
<td>120°</td>
</tr>
<tr>
<td>Rainy</td>
<td>23.8405</td>
<td>26.0357</td>
<td>23.8180</td>
</tr>
<tr>
<td>GMMLP</td>
<td>29.4895</td>
<td>31.5672</td>
<td>29.4545</td>
</tr>
<tr>
<td>CNN</td>
<td>24.1248</td>
<td>27.2630</td>
<td>24.3583</td>
</tr>
<tr>
<td>UGSM</td>
<td>28.6841</td>
<td>30.8983</td>
<td>29.0354</td>
</tr>
<tr>
<td>DTVRS</td>
<td>31.3372</td>
<td>32.2719</td>
<td>32.0719</td>
</tr>
<tr>
<td></td>
<td>45°</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rainy</td>
<td>25.2197</td>
<td>24.8202</td>
<td>25.2197</td>
</tr>
<tr>
<td>GMMLP</td>
<td>31.8517</td>
<td>33.4776</td>
<td>31.8517</td>
</tr>
<tr>
<td>CNN</td>
<td>25.6772</td>
<td>26.6789</td>
<td>25.4605</td>
</tr>
<tr>
<td>UGSM</td>
<td>28.9564</td>
<td>33.3201</td>
<td>29.3947</td>
</tr>
<tr>
<td>DTVRS</td>
<td>35.0720</td>
<td>33.5534</td>
<td>30.7475</td>
</tr>
<tr>
<td></td>
<td>90°</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rainy</td>
<td>24.8202</td>
<td>24.9191</td>
<td>24.8202</td>
</tr>
<tr>
<td>GMMLP</td>
<td>32.3116</td>
<td>32.116</td>
<td>32.3116</td>
</tr>
<tr>
<td>CNN</td>
<td>24.3659</td>
<td>24.5482</td>
<td>24.3659</td>
</tr>
<tr>
<td>UGSM</td>
<td>29.8990</td>
<td>29.9268</td>
<td>27.3833</td>
</tr>
<tr>
<td>DTVRS</td>
<td>32.4554</td>
<td>32.3067</td>
<td>31.5733</td>
</tr>
<tr>
<td></td>
<td>135°</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rainy</td>
<td>24.9191</td>
<td>24.9191</td>
<td>24.9191</td>
</tr>
<tr>
<td>GMMLP</td>
<td>32.3116</td>
<td>32.3116</td>
<td>32.3116</td>
</tr>
<tr>
<td>CNN</td>
<td>24.3659</td>
<td>24.5482</td>
<td>24.3659</td>
</tr>
<tr>
<td>UGSM</td>
<td>29.8990</td>
<td>29.9268</td>
<td>27.3833</td>
</tr>
<tr>
<td>DTVRS</td>
<td>32.4554</td>
<td>32.3067</td>
<td>31.5733</td>
</tr>
<tr>
<td></td>
<td>150°</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rainy</td>
<td>25.2197</td>
<td>25.4605</td>
<td>25.2197</td>
</tr>
<tr>
<td>GMMLP</td>
<td>31.8517</td>
<td>33.4776</td>
<td>31.8517</td>
</tr>
<tr>
<td>CNN</td>
<td>25.6772</td>
<td>26.6789</td>
<td>25.4605</td>
</tr>
<tr>
<td>UGSM</td>
<td>28.9564</td>
<td>33.3201</td>
<td>29.3947</td>
</tr>
<tr>
<td>DTVRS</td>
<td>35.0720</td>
<td>33.5534</td>
<td>30.7475</td>
</tr>
</tbody>
</table>

Table 2. Quantitative evaluation on BSD500 dataset.

<table>
<thead>
<tr>
<th>Metrics</th>
<th>rainy</th>
<th>GMMLP</th>
<th>CNN</th>
<th>UGSM</th>
<th>DTVRS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSIM</td>
<td>0.8405 ± 0.0690</td>
<td>0.9259 ± 0.0326</td>
<td>0.8675 ± 0.0580</td>
<td>0.8966 ± 0.0435</td>
<td>0.9306 ± 0.0291</td>
</tr>
</tbody>
</table>

Fig. 3. The visual comparison on two synthetic rain images of BSD500 dataset.

Fig. 4. The visual comparison on two real-world rain images.

3.2. Real Experiments

In this section, we test all the methods on two real images. One image is corrupted by oblique light rain and another contains heavy rain in nearly vertical direction. Fig. 4 shows the visual comparison results. We can observe that some shadow rain streaks (in the first row) or dense rain streaks (in the second row) are not removed completely by the other compared methods. While our method removes most of the rain streaks and recovers the rain-free background successfully.

4. CONCLUSION

In this paper, we addressed the single-image deraining task by exploiting the directional smoothness of rain streaks in arbitrary directions. We proposed a convex model using DTV to characterize the discriminative smoothness of rain streaks and rain-free backgrounds, respectively. To solve the proposed model, we developed an efficient algorithm based on ADMM framework. Extensive experimental results demonstrate the superiority of our method visually and quantitatively.
5. REFERENCES


