Communications in Information and Systems
Volume 22, Number 1, 79–102, 2022

Low-rank appearance-preserving rain streak removal from single images

Xiaoge He, Yidan Feng, Qian Xie, and Jun Wang

Rain streak removal (RSR) enables the restoration of images affected by rain, facilitating outdoor vision-based tasks. However, conventional wisdoms lead to image degradations when rain is heavy, while learning-based techniques that learn mappings from specific and synthetic datasets hardly generalize and adapt to real-world scenes with unseen patterns. This paper presents a low-rank appearance-preserving RSR algorithm (LA-RSR) for single images. To fully consider the multi-type real-world rainy images, we for the first time formulate a four-prior based optimization function (FPOF) to ensure the performance of both RSR and preserving intrinsic properties (i.e., low-frequency structures and high-frequency details). FPOF is effectively solved by a two-stage decomposition strategy in an iterative way, in which we utilize low-rank matrix recovery and unidirectional total variation (UTV) in the high-frequency component of the rainy image to better separate the rain streak layer and the detail layer. The detail layer is combined with the low-frequency component to yield the final rain-free image. Qualitative and quantitative results show that our approach consistently outperforms the conventional RSR methods and is comparable to the deep learning based methods, without requiring training.

1. Introduction

Severe weather conditions (rain, fog, snow, etc.) unpredictably damage the visual quality of images captured by outdoor vision systems [1–3], which negatively impact on a broad range of vision applications such as monitoring, tracking and navigation. Among them, rain streak remains the most common reason for such kind of image degradations. Through the years, many efforts have been put into the study of image rain streak removal [4–7], which a lot of promising results are obtained for various vision-based applications. However, it is still challenging to recover clean images without streak remnants and/or image over-smoothing, from their real-world rainy counterparts with complex backgrounds.

Figure 1: Comparison of rain streak removal (RSR) methods on a real-world rainy image. From (a) to (g): (a) The input, (b-g): the RSR results of (b) DSC [8]; (c) GMM [9]; (d) UGSM [10]; (e) DetailNet [11]; (f) Hu [12]; and (g) our LA-RSR. The marked regions of interest are enlarged to observe the visual difference. Our LA-RSR consistently outperforms the traditional methods and is comparable to the deep learning based methods.

The challenge of the ill-posed rain streak removal (RSR) problem is how to completely eliminate rain streaks yet with a minimal disturbance of the image's intrinsic properties, such as the low-frequency structures and the high-frequency details (textures). According to the presence of temporal information, RSR algorithms can be divided into single-image based approaches and video-based approaches. Early research mainly focuses on video streak removal, and most of them emphasize rain streak detection by analyzing the complex optical and physical characteristics of rain streaks [1, 13, 14]. This detection-and-RSR operation can be much easier compared with the single-image streak removal problem, thanks to the redundant information given by adjacent frames.

In this work, we focus on RSR from single images, which requires less information for restoration but benefits a broader range of applications. The performance of single-image RSR commonly relies on the hand-crafted image priors. For example, Kang et al. [15] utilized dictionary-based sparse coding to decompose the input image into high- and low-frequency components. Li et al. [9] formulated an energy minimization model, in which the corresponding patch-based priors of rain layer and background layer are learned by Gaussian mixture models. In particular, Chen and Hsu [16] introduced an RSR method based on the low-rank prior revealing that rain streaks in disparate patches share similar patterns. The low-rank property captures

the similarity of rain streak patches across different background contexts, which is a simple and effective way to remove rain streaks. However, there mainly exist two drawbacks of only enforcing the low-rank constraint: first, not only rain streaks but also background textures have similarity in different patches, leading to an overblurred result especially in texture regions; second, similarity of rain streaks among image patches does not always exist in every type of rainy images, such that a single low-rank prior fails to ensure the rain removal quality in real-world scenarios. Either the streak remnant results or the blurry results with missing details (as shown in Fig. 1) resulted from existing prior-based [8–10] or network-based formulations [11], Hu [12] leave room for possible improvement with better formulations. Therefore, our method attempts to incorporate multiple priors of not only rain streaks but also image intrinsic properties which tend to be overlooked by existing algorithms. Multiple priors can deal with different types of real-world rainy images, which shows a superior ability in both RSR and image intrinsic property preservation.

Concurrently, RSR methods [17–19] exploiting deep neural networks are making their debuts with impressive results, which undoubtedly exceed many traditional optimization- and filter-based approaches in removing synthetic rain streaks. However, these methods, with the help of powerful GPUs, heavily depend on expensive training over massive datasets. Thus, they are often impaired when the rainy images being processed deviate significantly from the training ones. For example, when it comes to real-world rainy images, the lack of corresponding real clean images prevents deep learning based methods from extending their superiority to real scenes.

We propose a low-rank appearance-preserving RSR algorithm (LA-RSR) for single images. To adapt the LA-RSR to different types of real-world rainy images to ensure the performance of both RSR and image intrinsic property preservation, we enforce four different priors based on a two-stage decomposition strategy. In the first stage, based on the observation that rain streaks mainly exist in the high-frequency image component, we decompose the input image I into a high-frequency part H and a low-frequency part L using low-pass filtering techniques. As a result, we can focus on the decomposition of rain-streak layer R and high-frequency background layer S in a much smaller problem domain. Inspired by unidirectional total variation (UTV) proposed to remove the stripe noise, we impose two sparse priors on both the rain-streak layer R and the background layer S. To utilize the directional property of rain streaks, the sparse priors are enforced on the variation along the vertical direction of R and the variation along the horizontal direction of S. In practice, we adopt a rotational strategy to accommodate skewed rain

directions. Moreover, the low-rank prior [16] is also enforced on R, which enhances the rain removal ability by characterizing rain streaks from a different point of view. Meanwhile, an L_0 prior is imposed on the gradients of L, which helps to avoid overblurring structures by enforcing sharp edges in the output image. Combining the four priors, the final optimization model is formulated and then solved by an iterative minimization strategy.

Our approach consistently outperforms the conventional RSR methods in terms of RSR and image intrinsic property preservation (see Fig. 1(b–d)). Meanwhile, our LA-RSR does not rely on training, which can generalize and adapt to multi-type scenarios. Thus, it is very comparable to or even better than the deep learning based ones (see Fig. 1(e–g)). The major contributions of this work are three-fold:

(1) We for the first time propose a low-rank appearance-preserving rain streak removal algorithm for single images. In detail, we introduce a comprehensive optimization objective within a two-stage decomposition architecture to achieve RSR with the image's intrinsic properties preserved. An alternating iterative strategy is leveraged to efficiently solve the resulting optimization problem.
(2) In our model, rain streaks are characterized by both directional sparsity and structural similarity, while another sparse prior is imposed on the high-frequency background layer to protect background details.
(3) Our proposed LA-RSR approach has been validated on a large number of real-world rainy images, with no prior knowledge of image patterns. Meanwhile, a user study is further given by evaluating the RSR results of all the compared methods.

2. Related work

Single image rain removal Early methods concerning image rain removal are mainly video-based [14, 20–24]. These methods rely on the information provided by successive frames, which is not always available in lots of application scenarios. Accordingly, a number of deraining algorithms [8, 9, 15, 25, 26] have proposed to simultaneously extract rain patterns and preserve background details while requiring no temporal information from dynamic video sequences. There are many methods that strive to design better prior models based on sparse coding or low-rank representation [26–28]. For example, Zhu et al. [26] analyze the main direction and the similarity of rain patterns by collecting rain streak dominated patches. They employ the centralized sparse representation to remove rain patterns by enforcing

three disparate priors on both the background layer and the rain streak layer. Chen et al. [16] formulate the problem as a low-rank model and thus get rid of rain detection and the dictionary learning process. In addition, Kim et al. [25] integrate physical features into nonlocal mean filtering to smooth out rain streaks. Nevertheless, all these methods suffer from degradation of image details. To better maintain details in the original image, Kang et al. [15] first decompose the image into a low frequency layer and a high frequency layer. Since rain streaks mainly appear as high frequency components, this decomposition step avoids unnecessary loss of detail information in the image recovering process, but it can makes no contribution to thoroughly removing rain streaks. From [29], studies on rain streak removal via deep learning have emerged and draw much attention [11, 18, 19, 29]. Recently, Liang et al. [30] proposed a novel incremental random wired network (IRWN), which can effectively model rain streaks during encoding and remove multiple rain spots after decoding. These methods achieve efficiency and effectiveness counting on a massive database of labeled images yet most of the natural rainy scenes do not come with a clear ground truth. Therefore, it is still a challenging problem to reach an appropriate balance between sufficient rain streak removal and detail preservation.

Structure-preserving image smoothing The goal of structure-preserving image smoothing is to maintain or sharpen the semantically important structures while smoothing out image details, which complies with our main purpose in the first-stage decomposition. Literature under this topic is vast [31–36]. [37] is a representative filtering-based method, which suppresses high frequency signals via a spatial kernel while maintaining large-amplitude signal discontinuities via a range kernel. It is simple but effective in separating low frequency components from low-magnitude high frequency noise or other interference. Another perspective in solving the edge-preserving smoothing problem is based on mathematical optimization. Rudin et al. [36] introduce statistic property to constrain the objective function, which inspired a number of approaches dealing with detexturing [38], destriping [10], deraining [10], etc. Farbman et al. [34] propose a popular mechanism based on the weighted least squares framework. They attempt to preserve structures by minimizing the distance between the input and the output images while ensuring smoothness by penalizing the gradient amplitudes. A different regularizer is proposed by Xu et al. [35], who utilize the sparsity of significant gradients as a regularization term and succeed in sharpening the main structures within a smooth background.

Destriping of remote sensing images Stripe noise caused by inconsistent responses of different detectors seriously affect the quality of remote sensing images and it possesses strong directional and structural characteristics, which is consistent with the rotated rain streaks. Many efforts have been put into smoothing out stripe noise without degrading the original image structures [10, 39–43]. Among them, methods combining the directional characteristic of stripe noise into the total variation optimization framework have shown superiority compared the other approaches. The notion of unidirectional total variation (UTV) was first proposed by Bouali et al. [10], which minimizes the energy function regularized by the sparsity of gradients in the noisy image along the vertical direction and in the structure image along the horizonal direction. Recently, Chen et al. [39] explored more on the group sparsity of stripe noise and further improved the performance of the original UTV model.

3. Problem formulation

3.1. Motivation

The visual effects of rain streaks are quite complex [1], as rain contains a group of randomly distributed raindrops falling at high velocities. It is a challenging task to accurately detect rain streaks in a single image, especially for real rainy images. For this problem, we propose a new model based on multi-component decomposition and low-rank matrix recovery for single image rain streak removal.

3.2. Rain streak removal

Dividing the image into multiple components makes it possible to represent the image scales in an efficient manner. Suppose that an input image $I \in [0,1]^{p\times q\times 3}$ of size $p \times q$ is composed of a high-frequency layer $H \in [0,1]^{p\times q\times 3}$ and a low-frequency layer $L \in [0,1]^{p\times q\times 3}$. Mathematically, it can be expressed as:

$$\mathrm{I} = H + L \tag{1}$$

where L mainly includes most of the information in the background image, and H mainly contains both the detail and rain streaks information. On this basis, only the high-frequency part H is subjected to a rain removing operation. An image I is passed through a low-pass filter to obtain a low-frequency component L, the implementation process will be described in detail below.

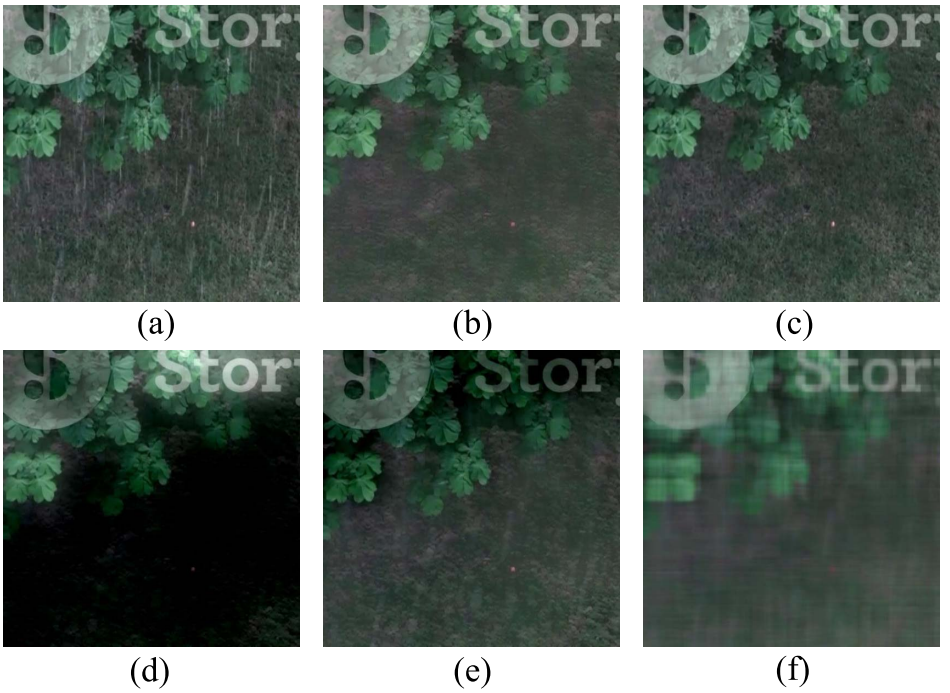

Figure 2: The results of ablation experiments with each constraint. (a) is the original rainy image; (b) is our LA-RSR result; (c) is the ground truth; (d) shows the result of processing only the high-frequency part; (e) shows the result without the δ_1 term; and (f) shows the result without both the δ_1 and δ_2 terms.

3.2.1. Approximate estimation of the rain streak direction In natural images, the rain streaks are generally roughly present in the same direction from the top to the bottom. As can be seen from Figure 2(a), a single rainy image usually contains similar patterns of rain streaks in different local patches. Therefore, we need to approximately estimate the direction of the area containing the rain streaks. First of all, we use the local gradient distribution for a given rainy image I to determine the rain-dominated areas. Then we use a Canny edge detector to find edges in each of the rain-dominated area and employ the Hough transform to detect the longest line. The direction of the longest line of the local patch is determined as the direction of the rain streaks.

3.2.2. High-frequency rain streak removal Rain images usually contain similar rain streaks in different local patchs, here we assume that the

rain streaks are linearly related. At this time we decompose the high frequency component H into the rain streak layer R and the background layer S. Based on this consideration, we introduce a low-rank approximation. Our direct employment of a simple low-rank model significantly reduces the rain streaks.

$$\begin{aligned}(\mathrm{L},\mathrm{R}) =& arg\min_{\mathrm{L},\mathrm{R}}(\|I-H-L\|_{\mathrm{F}}^2+\delta_1\|P(R)\|_*),\\ & s.t. \quad H=S+R\end{aligned} \tag{2}$$

where $\|.\|_*$ refers to the sum of the singular values of the matrix. δ_1 is the scalar parameter that controls the nuclear norm constraint term. $P(.)$ is the patch map. As mentioned in [16], the patch map function using overlapping patches may bring additive bias to the objective costs. However, such bias could be neglected by proper normalization in the algorithm. During the above-mentioned rain removal process, although the rain streaks are largely removed, the details would be lost to some extent.

Similar to the stripe noise, rain streaks are generally distributed throughout the image, and both of them contain direction information and detail information. Chen [39] suggested that the stripe image is decomposed into two components: image component and stripe component. In this case, the stripe component is parallel to the Y axis and the TV constraints are imposed on the image component. The role of the TV term in image restoration and denoising is to maintain the smoothness of the image and eliminates the artifacts of image restoration. In most circumstances the direction of the rain streaks is not parallel to the Y axis. Therefore, we rotate the direction of the rain streaks by θ angle to make it perpendicular to the X axis. Hence, the characteristics of the rain streaks after rotation are similar to the stripe noise, which motivates us to use the unidirectional total variation model to solve the RSR problem. This method has a wide range of applications in digital imaging, and preserves sharp edges to restore clean images. The formula is as follows:

$$\begin{aligned}(\mathrm{L},\mathrm{S},\mathrm{R}) =& arg\min_{\mathrm{L},\mathrm{S},\mathrm{R}}(\|I-H-L\|_{\mathrm{F}}^2+\delta_1\|P(R)\|_*\\ &+\delta_2\|\Delta_X(S)\|_1+\delta_3\|\Delta_Y(S)\|_1\\ &+\delta_4\|\Delta_Y(R)\|_1), s.t. \quad H=S+R\end{aligned} \tag{3}$$

where $\|I-H-L\|_{\mathrm{F}}^2$ is the data-fidelity term, which denotes that the sum of high-frequency component H and low-frequency component L is close to the rainy image I. δ_2 and δ_3 are the regularization parameters. $S \in [0,1]^{p\times q\times 3}$

and $R \in [0,1]^{p\times q\times 3}$ denote a rain-free background and a rain-streak layer respectively. With the evaluated rain streak direction, $\Delta_X(.)$ and $\Delta_Y(.)$ represent the rotation operations on the X-axis and the Y-axis, respectively.

3.3. Appearance structure optimization

Since the rain streaks exist mostly on the high-frequency component, we merely need to obtain the appearance structure information about the low-frequency component. Xu et al. [35] found that the L_0 norm can extract the main structural information and eliminate the inconspicuous details. Accordingly, we enforce the structural detail (texture) decomposition constraint in the low-frequency component, and it is an effective tool for distinguishing structural elements:

$$\tilde{\mathrm{L}} = arg\min_{\tilde{\mathrm{L}}} \sum_i (\|\tilde{L}_i - L_i\|_{\mathrm{F}}^2 + \delta_5 \|C(\tilde{L})\|_0) \tag{4}$$

where $\|.\|_0$ is the sparsity measure by counting the number of nonzero components, $C(\tilde{L})$ counts pixel p whose magnitude $|\delta x L_i| + |\delta y L_i|$ is not zero. δ_5 term is a weight parameter that controls the structural texture term. The larger the value, the more seriously the structural information is smoothed. Thus $\tilde{L}$ is the final appearance structure information.

Combining with the above two formulas, we can obtain the general formula of our algorithm:

$$\begin{aligned}(\mathrm{L},\mathrm{S},\mathrm{R}) =& arg\min_{\mathrm{L},\mathrm{S},\mathrm{R}} (\|I - H - L\|_{\mathrm{F}}^2 + \delta_1 \|P(R)\|_* \\ &+ \delta_2 \|\Delta_X(S)\|_1 + \delta_3 \|\Delta_Y(S)\|_1 \\ &+ \delta_4 \|\Delta_Y(R)\|_1 + \sum_i \|\tilde{L}_i - L_i\|_{\mathrm{F}}^2 \\ &+ \delta_5 \|C(\tilde{L})\|_0), s.t. \quad H = S + R\end{aligned} \tag{5}$$

where δ_1 is used to suppress the rain streak, and δ_2, δ_3, δ_4 are to separate the background layer and the rain streak layer in the high-frequency part, in order to obtain a desirable background layer. δ_5 applied to the low-frequency part is used to preserve the appearance structure information on the image.

Since the proposed model (5) is not differentiable, thus we make variable substitutions and solve the following equivalent problem:

$$\begin{aligned}(\mathrm{L},\mathrm{S},\mathrm{R}) =& arg\min_{\mathrm{L},\mathrm{S},\mathrm{R}} (\|I - H - L\|_2^2 + \mu/2 \|H - S - R\|_2^2 \\ &+ < p, H - S - R > + \delta_1 \|P(R)\|_* + \delta_2 \|\Delta_X(S)\|_1\end{aligned}$$

$$
\begin{aligned}
&+ \delta_3\|\Delta_Y(S)\|_1 + \delta_4\|\Delta_Y(R)\|_1 \\
&+ \sum_i \|\tilde{L}_i - L_i\|_{\mathrm{F}}^2 + \delta_5\|C(\tilde{L})\|_0)
\end{aligned} \tag{6}
$$

where μ is regularization parameter, and p represent Lagrange multiplier.

To demonstrate the effect of each constraint, we perform an ablation experiment, and its results are shown in Figure 2. We can observe from Figure 2(d), that the function of the appearance structure optimization is to preserve the structure information of the image. It can be seen from Figure 2(e) that a small amount of rain streaks are left in the image. And the rain streaks in Figure 2(f) are effectively suppressed, but the details are seriously lost. Figure 2(b) is our LA-RSR result, thus it provides the overall algorithm with all the terms is most effective.

4. Optimization algorithm

In this section, we give a detailed description of solving the optimization problem, i.e., Equation (6), by an iterative update strategy.

4.1. Low-rank matrix recovery

H can be obtained by fixing L, R, and S. Note that there is a challenge if we solve it directly by the alternating minimization. To alleviate this problem, we use Z instead of $P(R)$:

$$
Z = arg\min_{\mathrm{Z}}(\|P(R) - Z\|_2^2 + \delta_1\|Z\|_*) \tag{7}
$$

Obviously this is a problem with a kernel norm solution. We adopt the singular value thresholding algorithm (SVTA) [44] to yield the Z.

4.2. Rain streak layer and background layer decomposition

By simplifying the irrelevant regular terms, the following formulas can be obtained:

$$
\begin{aligned}
(S, R) = &arg\min_{\mathrm{S,R}}(\mu/2\|H - S - R\|_2^2 + < p, H - S - R > \\
&+ \delta_1\|P(R)\|_* + \delta_2\|\Delta_X(S)\|_1 + \delta_3\|\Delta_Y(S)\|_1 \\
&+ \delta_4\|\Delta_Y(R)\|_1)
\end{aligned} \tag{8}
$$

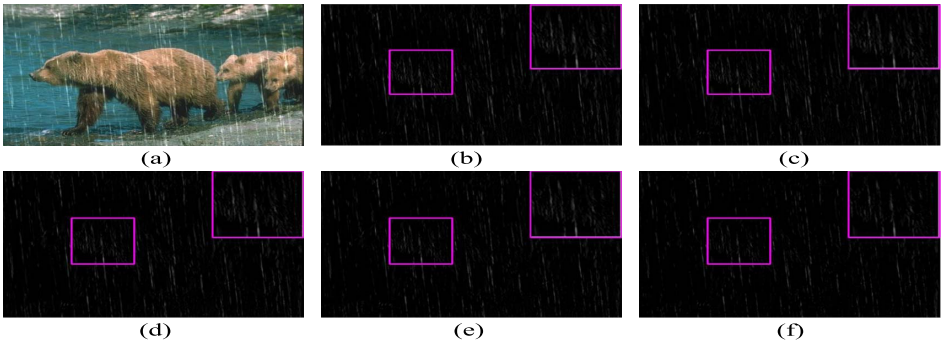

Figure 3: As the number of iterations increases, the changes of the residual images are illustrated. (b)–(f) represent the residual images obtained after each iteration. As can be seen from the magnified areas, the detail information in the residual images are gradually reduced.

For the rain streak layer and the background layer:

$$S = arg\min_{S}(\mu/2\|H-S-R\|_2^2 + < p, H-S-R > + \delta_2\|\Delta_X(S)\|_1 + \delta_3\|\Delta_Y(S)\|_1 \tag{9}$$

$$R = arg\min_{R}(\mu/2\|H-S-R\|_2^2 + < p, H-S-R > + \delta_1\|P(R)\|_* + \delta_4\|\Delta_Y(R)\|_1) \tag{10}$$

It is an L_1 norm minimization problem, that can be effectively solved by the method developed in [10].

$$H = arg\min_{H}(\|I-H-L\|_2^2 + < p, H-S-R > + \mu/2\|H-S-R\|_2^2) \tag{11}$$

This is a simple L_2-norm problem which has the following closed-form solution. Then, we can get H by keeping Z, L, S and R fixed.

In order to better separate the background layer and rain-streak layer of the high-frequency component, we restore the background structure residual information, which often exists on the rain-streak layer.

Thus we introduce a refinement step to further retrieve these background structure residues from the rain-streak layer. Based on this idea, we use the guided image smoothing framework [34], Hu [12], with the background layer as guidance to further clean the rain streak output R. Specifically, we

attempt to obtain the background structure residues $\tilde{S}$ by optimizing the following objective function:

$$\tilde{S} = arg\min_{\tilde{S}} \sum_{i} \sum_{j=N_4(i)} \Theta_{(i,j)}(S)(\tilde{S}_i - \tilde{S}_j)^2 \tag{12}$$

where i is the pixel index. $N_4(i)$ represents the four neighbors for pixel i. $\Theta_{(i,j)}$ measures how similar two pixels i and j are at the guidance image S. The term encourages the smoothness under the guidance of S. We use the fast guided image smoothing techniques Hu [12] to solve Equation (9). The final rain streak layer can be computed as:

$$R = \max(R - \tilde{S}; 0) \tag{13}$$

Figure 3 is the result of RSR, which shows the background structure residue recovery. As can be seen, this step can more successfully recover the background residues and make the rain streaks clearer.

4.3. Optimization of the low-frequency component

By incorporating both the fidelity terms and simplifying the formula (6), we can get:

$$\mathrm{L} = arg\min_{\mathrm{L}}(\|I - H - L\|_{\mathrm{F}}^2 + \sum_{i} \|\tilde{L}_i - L_i\|_{\mathrm{F}}^2) \tag{14}$$

$$\tilde{\mathrm{L}} = arg\min_{\tilde{\mathrm{L}}} \sum_{i} (\|\tilde{L}_i - L_i\|_{\mathrm{F}}^2 + \delta_4 \|C(\tilde{L})\|_0) \tag{15}$$

where Equation (15) is an L_0 norm problem, so we use [35] to solve it.

A rain-free image consists of a high-frequency part and a low-frequency part. Consequently, the final rainless image can be obtained by superposition of the rain-free high-frequency component and the appearance low-frequency component. After the iteration, the low-frequency component is added to the high-frequency background layer, which forms the final rain-free image:

$$F = S + L \tag{16}$$

where $F \in [0, 1]^{p \times q \times 3}$ is of size $p \times q$.

During this process, we use an iterative update strategy to solve it. Furthermore, the detail information of the source image is greatly preserved, while the rain streaks are effectively suppressed. This property of the method is very conducive to enhancing the visual effects of deraining results.

Figure 4: The RSR results when the number of iterations k is set to 1, 3 and 5 in our algorithm. The first row shows the low-frequency (background) layers, the second row shows the rain streak layers, and the third row shows the final RSR results.

5. Experimental verification

Our LA-RSR has been tested in both synthetic and real-world rainy images. In order to better verify the correctness and advantages of the proposed algorithm, we compare it with three model-based algorithms and two methods based on the deep neural networks. The details of these five algorithms are listed as follows.

1) The discriminative sparse coding approach [8] (denoted as DSC)
2) GMM-based layer prior [9] (denoted as GMM)
3) Directional global sparse model [10] (denoted as UGSM)
4) Deep detail network [11] (denoted as DetailNet)
5) Depth-attentional features [12] (denoted as Hu)

For DSC, GMM, USGM and DetailNet, we use the codes provided by the authors with their default parameter settings. And for Hu's method, the comparison images shown are from the provided dataset. Similar to the most rain-removal algorithms, ours also has some fixed parameters in the experiment, shown in Algorithm 1. Empirically, δ_1 ranges within $[0.1, 1.5]$.

Algorithm 1 The optimization framework of LA-RSR

Require: Rainy image I

Ensure: Background layer S and low-frequency L

Initialization:

$H = L = Z = 0$, $\mu = 10^6, max\mu = 10^{15}, \rho = 1.1$, $\delta_1 = 1.3$, $\delta_2 = 0.08$, $\delta_3 = 1$, $\delta_4 = 0.95$, $\delta_5 = 1$, and maximum number of iterations k

while not converged and within K iterations **do**

update matrices R, S, H, L:

(1) Update S and R respectively via Equation (9) and (13);

(2) Update H by solving Equation (11);

(3) Update L via Equation (14);

(4) Update μ, $\mu = \min(\rho * \mu, max\mu)$.

end while

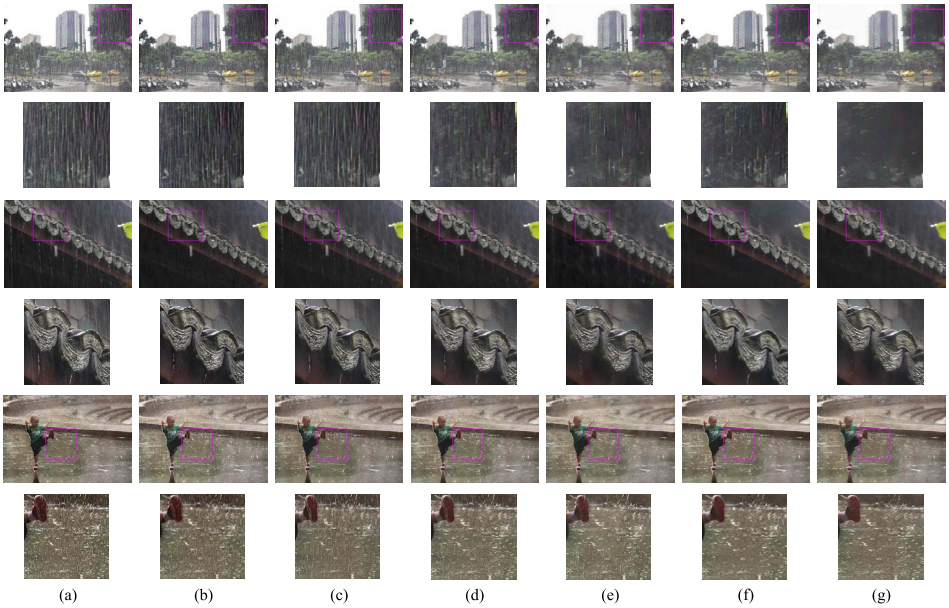

Figure 5: The test on real rainy images with complex structures. (a) shows the original rainy images, (b)–(g) represent the results of the comparison methods, which are DSC [8], GMM [9], UGSM [10], DetailNet [11], Hu [12], and our LA-RSR respectively. One part of each image has been magnified to clearly observe the differences of various deraining results.

The iteration number K is set to be 5. The parameters in the comparison methods are consistent with those described in the literature. Figure 4 shows the rain-streak layers and the background layers which are from the results of the first iteration, the third iteration and the fifth iteration, respectively. As can be seen from Figure 4, when the iteration number is set to 1, there are some shortcomings of image detail loss, leading to image blurring. The results produced by the second and last columns indicate that the final performance is better as the number of iterations increases.

For the synthetic data, experiments have shown that both traditional rain removal methods and deep learning-based rain removal methods have greatly improved the evaluation indicators and visual effects. It introduces briefly the visual effects of composite images for different contrast methods. However, because of lacking the ground truths of real rainy images, we simply visually evaluate the performance of deraining images. In [7], Wang et al. present a new dataset including rainy images and their corresponding ground truths. Therefore, we test the performance of our algorithm on this dataset, which is described in detail below.

5.1. Test on real rainy images

In order to prove the validity of the algorithm, we test the real images with different methods, as shown in Figure 5. The first row is a building with dense rain streaks (Building for short), which destroys the quality of the image. The third row shows the part of the roof (Roof for short). The fifth row is a rainy image of a Child (Child for short). The second and fourth rows, and the sixth row are their corresponding enlarged areas. In this case, Figure 5(b)–Figure 5(g) are results from DSC, GMM, UGSM, DetailNet, Hu and our LA-RSR.

Building shows an example of light rain removal. Here, our goal is to remove the dense rain streaks in the entire scene, including buildings, trees, cars, pedestrians, etc. For removing the dense rain streaks, the overall effect of the contrast methods is not ideal. It can be seen that there are still a small amount of residual rain streaks in DSC. In GMM, UGSM and DetailNet, although the rain streaks are well-removed, many details are lost and the image quality is slightly degraded. The rainy image of Roof is characterized in dense rain streaks on the top of the roof and thicker rain drops under the tiles. The comparison experiments demonstrate that some methods have a good effect on the dense rain streaks on the roof but ignore the rain streaks under the tile. Our method balances the two parts and achieves promising results.

Child depicts a little boy standing in the heavy rain. Through the enlarged area of the test results, it can be clearly seen that the DSC method has a poor removal effect, and a lot of rain information remains in the image. For GMM and DetailNet, although the overall effect of rain streak removal been improved, the rain streaks around the little boy in the image still exist. The UGSM method retains relatively complete background information, but fails to clean residual rain streaksand the image background obtained from Hu's method is overblurred and a lot of information is lost.

Experimental results in Figure 5 verify that our proposed algorithm can effectively remove the rain streaks on the real rainy images while maximizing the information of the background images. In comparison, deep learning based approaches suffer from background detail degradation, and single prior based methods have a limited ability in dense rain removal. In contrast, our algorithm is more versatile, since it not only removes rain streaks, but also retains a lot of background details.

At the same time, a user study (Figure 8) is further given by evaluating the deraining results of all the compared methods. We borrow used Google API to test the real rain image and our derained image method, as well as two comparison methods [11], Hu [12], and the results are illustrated in Figure 8(h). As we can see, in Figure 8(a) and Figure 8(b), the ability to identify objects in derained images generated from Hu and DetailNet is greatly reduced.

To make comparison possible, we evaluate our model on a new real rainy dataset [7], as shown in Figure 6. Our proposed method achieves state-of-the-art results on this dataset compared to other approaches. Moreover, we enlarge the areas of interest to see the comparison results more clearly. As can be seen from Figures 6(b)–(g), DSC, GMM, UGSM, and DetailNet all cannot effectively remove the rain streaks, while the rain streaks are effectively removed by Hu, which however suffers from poor visual quality and smoothing of detailed information.

Furthermore, we use peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) to evaluate the quality of target images in Table 1. The largest values are highlighted in **bold**. It can be seen that our method produces the highest values for all of the tested real-world rainy images.

5.2. Test on synthetic rainy images

Our method is also run on synthetic images. We only show an example for subjective assessment, due to the limited pages. The enlarged area of Figure 7 provides a clearer view to compare the performance of both rain

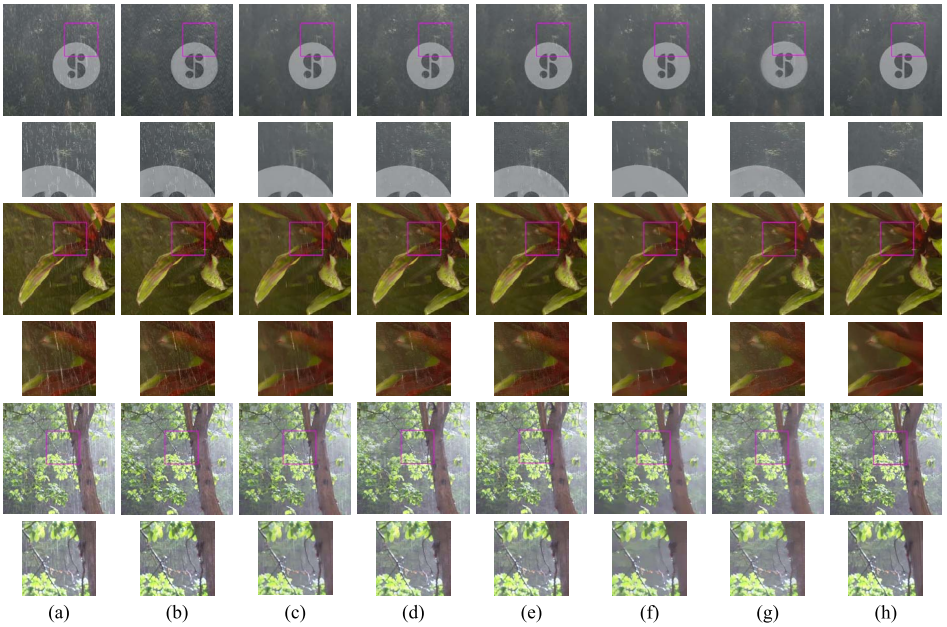

Figure 6: The test on the real rainy images with complex details. (a) shows the original rainy images, (b)–(g) illustrate the results of the comparison methods, (b) is DSC [8], (c) is GMM [9], (d) is UGSM [10], (e) is DetailNet [11], (f) is Hu [12], (g) is our LA-RSR, respectively; and (h) is the ground truth. One part of each image has been magnified to clearly observe the differences of various deraining results.

removal and detail preservation. Therefore, compared with the traditional algorithms, our method is superior to other algorithms in terms of qualitative visual effects. Our result is similar to the ones produced by the deep learning methods. However, our method does not require any training data (as opposed to a learning-based approach).

5.3. Failure case

Although our method has a good performance for the majority of cases, there are still instances that are more challenging to predict (see Figure 9). Sometimes, the original rain streak map contains not only rain streaks, but also raindrops. When the image is mixed with raindrops, our algorithm treats the raindrops as background information, which would be preserved in the derained image. It is also a common problem for other deraining methods.

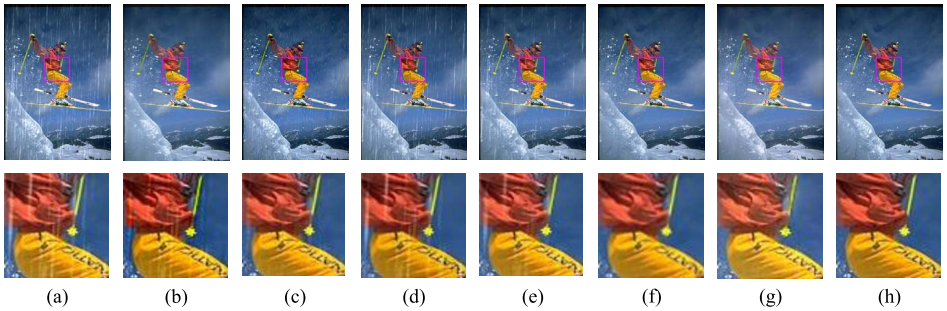

Figure 7: Results on synthetic rainy images. (a) and (h) represent the original rainy image and the ground-truth image, (b)–(g) represent the results of the comparison methods, which are DSC [8], GMM [9], UGSM [10], DetailNet [11], Hu [12], respectively, and (g) is our LA-RSR. The area of interest is partially enlarged in the box.

Figure 8: The rain removal effect tested on Google Vision API. The first column, the second column and the third column show the results of DetailNet, Hu and ours, respectively. (d) is the result of recognition object generated by the real rain image. (h) represents the average confidence in identifying rainwater from the real-world rainy image of DetailNet [11], Hu [12] and our LA-RSR.

6. Conclusion

In this paper, we have proposed an image rain streak removal algorithm based on multi-component decomposition and low rank matrix recovery. According to the characteristic that the rain streaks mostly exist in the

Table 1: Quantitative results of Figure 6

	Top: Environment					
Index	DSC	GMM	USCG	DetailNet	Hu	Ours
PSNR	28.8176	36.1899	35.3611	35.7234	36.2380	**37.2120**
SSIM	0.7775	0.9347	0.9089	0.9345	0.9269	**0.9411**
	Middle: Leaf					
Index	DSC	GMM	USCG	DetailNet	Hu	Ours
PSNR	30.6845	31.9663	32.2162	33.7337	32.5473	**34.7213**
SSIM	0.8386	0.9069	0.9115	0.9275	0.9273	**0.9302**
	Bottom: Tree					
Index	DSC	GMM	USCG	DetailNet	Hu	Ours
PSNR	29.8570	29.7738	29.5014	30.3190	28.8473	**30.8354**
SSIM	0.9010	0.9059	0.9139	0.9144	0.8579	**0.9186**

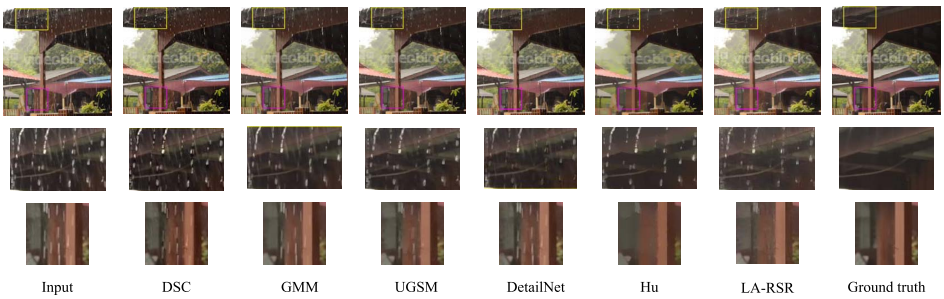

Figure 9: Raindrops are commonly preserved when removing rain streaks.

high-frequency component of the image, we first decompose the original rainy image into the high-frequency component and the low-frequency component. For the high-frequency component, we apply the low-rank matrix approximation to remove the rain streaks. To be more effectively, we adopt the unidirectional total variation to remove the rain streaks and restore more background details. By this means, the detail information of the background layer in the high-frequency component is extracted. At the same time, L_0 norm is used in the low-frequency component to retain more base layer appearance information. Finally, the final rain-free image is reconstructed by

combining low-frequency base layer and high-frequency background layer information. It shows better performance than the previous traditional methods and deep learning based methods on real rainy images. The proposed method is practically useful, since it does not require any training data (as opposed to the learning-based methods), as obtaining a large number of real-world rainy/non-rainy image pairs are difficult.

References

[1] K. Garg and S. K. Nayar, *Vision and Rain*. International Journal of Computer Vision, **75**(1):3–27, 2007.

[2] S. G. Narasimhan and S. K. Naya, *Vision and the Atmosphere*. International Journal of Computer Vision, **48**(3):233–254, 2002.

[3] Z. Deng, L. Zhu, X. Hu, C. W. Fu and P. A. Heng, *Deep Multi-Model Fusion for Single-Image Dehazing*. International Conference on Computer Vision (ICCV), 2019.

[4] Z. Fan, H. Wu, X. Fu, Y. Hunag and X. Ding, *Residual-Guide Feature Fusion Network for Single Image Deraining*. arXiv:1804.07493, 2019.

[5] S. Du, Y. Liu, M. Ye, Z. Xu, J. Li and J. Liu, *Single image deraining via decorrelating the rain streaks and background scene in gradient domain*. Pattern Recognition, **79**:303–317, 2018.

[6] G. Li, X. He, W. Zhang, H. Chang, L. Dong, and L. Lin, *Non-locally Enhanced Encoder-Decoder Network for Single Image De-raining*. In "Proceedings of the 26th ACM international conference on Multimedia", pages 1056–1064, 2018.

[7] T. Wang, X. Yang, K. Xu, S. Chen, Q. Zhang and R. W. Lau, M. Dyer, P. Gritzmann and A. Hufnagel, *Spatial Attentive Single-Image Deraining with a High Quality Real Rain Dataset*. In "Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition", pages 12270–12279, 2019.

[8] Y. Luo, Y. Xu and H. Ji, *Removing rain from a single image via discriminative sparse coding*. In "Proceedings of the IEEE International Conference on Computer Vision", pages 3397–3405, 2017. MR3386149

[9] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, *Rain streak removal using layer priors*. In "Proceedings of the IEEE conference on computer vision and pattern recognition", pages 2736–2744, 2016.

[10] M. Bouali and S. Ladjal, *Toward optimal destriping of MODIS data using a unidirectional variational model*. IEEE Transactions on Geoscience and Remote Sensing, **49**(8):2924–2935, 2011.

[11] X. Fu, J. Huang, D. Zeng, Y. Huang, X. Ding and J. Paisley, *Removing rain from single images via a deep detail network*. In "Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition", pages 3855–3863, 2017.

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

[13] J. Bossu, N. Hautiere and J. P. Tarel, *Rain or snow detection in image sequences through use of a histogram of orientation of streaks*. International journal of computer vision, **93**(3):348–367, 2011.

[14] S. You, R. T. Tan, R. Kawakami, Y. Mukaigawa and K. Ikeuchi, *Adherent raindrop modeling, detectionand removal in video*. IEEE transactions on pattern analysis and machine intelligence, **38**(9):1721–1733, 2015.

[15] L. W. Kang, C. W. Lin and Y. H. Fu, *Automatic single-image-based rain streaks removal via image decomposition*. IEEE transactions on image processing, **21**(4):1742–1755, 2011. MR2959485

[16] Y. L. Chen and C. T. Hsu, *A generalized low-rank appearance model for spatio-temporally correlated rain streaks*. In "Proceedings of the IEEE International Conference on Computer Vision", pages 1968–1975, 2013.

[17] L. Shen, Z. Yue, Q. Chen, F. Feng and J. Ma, *Deep joint rain and haze removal from a single image*. In 2018 24th International Conference on Pattern Recognition (ICPR), pages 2821–2826, August. 2018.

[18] X. Fu, J. Huang, X. Ding, Y. Liao and J. Paisley, *Clearing the skies: A deep network architecture for single-image rain removal*. IEEE Transactions on Image Processing, **26**(6):2944–2956, 2017. MR3648682

[19] D. Eigen, D. Krishnan and R. Fergus, *Restoring an image taken through a window covered with dirt or rain*. In "Proceedings of the IEEE international conference on computer vision", pages 633–640, 2013.

[20] K. Garg and S. K. Nayar, *Detection and removal of rain from videos*. In "Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition", pages I–I, June. 2004.

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

[22] X. Zhang, H. Li, Y. Qi, W. K. Leow and T. K. Ng, *Rain removal in video by combining temporal and chromatic properties*. In 2006 IEEE International Conference on Multimedia and Expo, pages 461–464, July. 2006.

[23] P. Barnum, T. Kanade and S. Narasimhan, *Spatio-Temporal Frequency Analysis for Removing Rain and Snow from Videos*. In "Proceedings of the First International Workshop on Photometric Analysis For Computer Vision-PACV", pages 8–p, October. 2007.

[24] N. Brewer and N. Liu, *Using the Shape Characteristics of Rain to Identify and Remove Rain from Video*. In "Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)", pages 451–458. Springer, Berlin, Heidelberg. December. 2008.

[25] J. H. Kim, C. Lee, J. Y. Sim and C. S. Kim, *Single-image deraining using an adaptive nonlocal means filter*. In 2013 IEEE International Conference on Image Processing, pages 914–917, September. 2013.

[26] L. Zhu, C. W. Fu, D. Lischinski and P. A. Heng, *Joint bi-layer optimization for single-image rain streak removal*. In "Proceedings of the IEEE International Conference on Computer Vision", pages 2526–2534, 2017.

[27] D. Y. Chen, C. C. Chen and L. W. Kang, *Visual depth guided image rain streaks removal via sparse coding*. In 2012 International Symposium on Intelligent Signal Processing and Communications Systems, pages 151–156, November. 2012.

[28] D. A. Huang, L. W. Kang, M. C. Yang, C. W. Lin, and Y. C. F. Wang, *Context-aware single image rain removal*. In 2012 IEEE International Conference on Multimedia and Expo, pages 164–169, July. 2012.

[29] C. Dong, C. C. Loy, K. He and X. Tang, *Image super-resolution using deep convolutional networks*. IEEE Transactions on Pattern Analysis and Machine Intelligence, **38**(2):295–307, 2015.

[30] X. Liang, B. Qiu, Z. Su, C. Gao, X. Shi and R. Wang, *Rain wiper: An incremental randomly wired network for single image deraining*. In Computer Graphics Forum, **38**(7):159–169, October. 2019.

[31] R. Fattal, *Edge-avoiding wavelets and their applications*. ACM Transactions on Graphics (TOG), **28**(3):1–10, 2009.

[32] E. S. Gastal and M. M. Oliveira, *Domain transform for edge-aware image and video processing*. In "ACM SIGGRAPH 2011 papers", pages 1–12, 2011.

[33] K. Subr, C. Soler and F. Durand, *Edge-preserving multiscale image decomposition based on local extrema*. ACM Transactions on Graphics (TOG), **28**(5):1–9, 2009.

[34] Z. Farbman, R. Fattal, D. Lischinski and R. Szeliski, *Edge-preserving decompositions for multi-scale tone and detail manipulation*. ACM Transactions on Graphics (TOG), **27**(3), 1–10, 2008.

[35] L. Xu, C. Lu, Y. Xu and J. Jia, *Image smoothing via L_0 gradient minimization*. In "Proceedings of the 2011 SIGGRAPH Asia conference", pages 1–12, December. 2011. MR2835578

[36] L. I. Rudin, S. Osher and E. Fatemi, *Nonlinear total variation based noise removal algorithms*. Physica D: Nonlinear Phenomena, **60**(1–4):259–268, 1992. MR3363401

[37] C. Tomasi and R. Manduchi, *Bilateral filtering for gray and color images*. In "Sixth International Conference on Computer Vision", pages 839–846, January. 1998.

[38] L. Xu, Q. Yan, Y. Xia and J. Jia, *Structure extraction from texture via relative total variation*. ACM Transactions on Graphics (TOG), **31**(6):1–10, 2012.

[39] Y. Chen, T. Z. Huang, X. L. Zhao, L. J. Deng and J. Huang, *Stripe noise removal of remote sensing images by total variation regularization and group sparsity constraint*. Remote Sensing, **9**(6):559, 2017.

[40] Y. Chang, H. Fang, L. Yan and H. Liu, *Robust destriping method with unidirectional total variation and framelet regularization*. Optics Express, **21**(20):23307–23323, 2013.

[41] Y. Zhang, Z. Gang, L. Yan, and T. Zhang, *A destriping algorithm based on tv-stokes and unidirectional total variation model*. Optik – International Journal for Light and Electron Optics, **127**(1):428–439, 2016.

[42] Y. Chang, L. Yan, H. Fang and H. Liu, *Simultaneous destriping and denoising for remote sensing images with unidirectional total variation and sparse representation*. IEEE Geoscience and Remote Sensing Letters, **11**(6):1051–1055, 2014.

[43] H. Carfantan, and J. Idier, *Statistical linear destriping of satellite-based pushbroom-type images*. IEEE Transactions on Geoscience and Remote Sensing, **48**(4):1860–1871, 2010.

[44] J. F. Cai, J. C. Emmanuel, Z. W. Shen, *A singular value thresholding algorithm for matrix completion*. SIAM Journal on Optimization, **20**(4):1956–1982, 2010. MR2600248

Xiaoge He
College of Computer Science and Technology
Nanjing University of Aeronautics and Astronautics
Nanjing 210016
China
E-mail address: Hexiaoge@nuaa.edu.cn

Yidan Feng
College of Computer Science and Technology
Nanjing University of Aeronautics and Astronautics
Nanjing 210016
China
E-mail address: fengyidan1995@126.com

Qian Xie
College of Mechanical and Electrical Engineering
Nanjing University of Aeronautics and Astronautics
Nanjing 210016
China
E-mail address: qianxie@nuaa.edu.cn

Jun Wang
College of Computer Science and Technology
Nanjing University of Aeronautics and Astronautics
Nanjing 210016
China
E-mail address: wjun@nuaa.edu.cn

Received July 16, 2020