A Novel Tensor-based Video Rain Streaks Removal Approach via Utilizing Discriminatively Intrinsic Priors

Anonymous CVPR submission

Paper ID 1636

Abstract

Rain streaks removal is an important issue of the outdoor vision system and has been recently investigated extensively. In this paper, we propose a novel tensor based video rain streaks removal approach by fully considering the discriminatively intrinsic characteristics of rain streaks and clean videos, which needs neither rain detection nor time-consuming dictionary learning stage. In specific, on the one hand, rain streaks are sparse and smooth along the raindrops' direction, and on the other hand, the clean videos possess smoothness along the rain-perpendicular direction and global and local correlation along time direction. We use the l_1 norm to enhance the sparsity of the underlying rain, two unidirectional Total Variation (TV) regularizers to guarantee the different discriminative smoothness, and a tensor nuclear norm and a time directional difference operator to characterize the exclusive correlation of the clean video along time. Alternation direction method of multipliers (ADMM) is employed to solve the proposed concise tensor based convex model. Experiments implemented on synthetic and real data substantiate the effectiveness and efficiency of the proposed method. Under comprehensive quantitative performance measures, our approach outperforms other state-of-the-art methods.

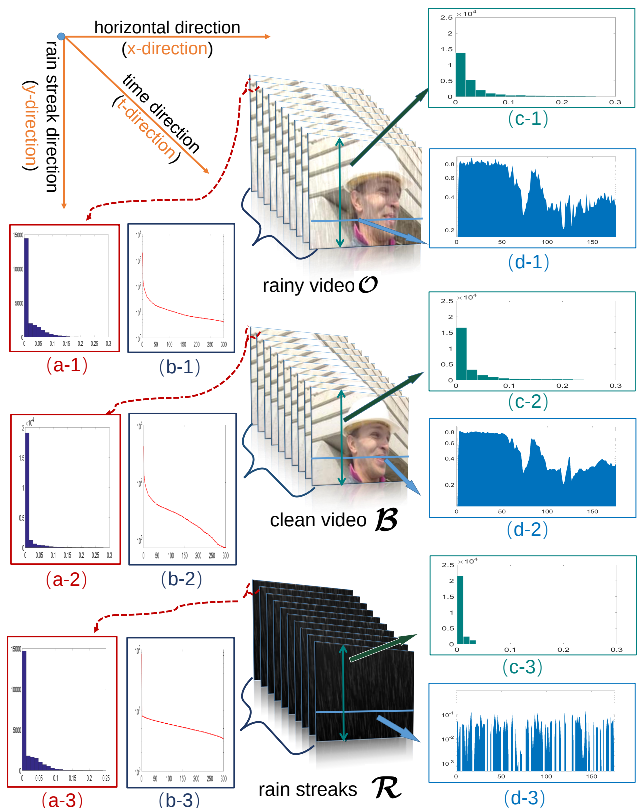

Figure 1. From left to right: 1) the histograms of difference of the 1st and 2nd frame from the rainy video, clean video and rain streaks, respectively; 2) the singular values of $\boldsymbol{O}_{(3)}$, $\boldsymbol{B}_{(3)}$ and $\boldsymbol{R}_{(3)}$ in decreasing order, severally; 3) some example frames of rainy video, clean video and rain streaks; 4) the histograms (c-1,2,3) of rain directional difference of the 10th frame, and the intensities of a row (d-1,2,3) of the rainy video, clean video and rain streaks, respectively.

1. Introduction

Outdoor vision system is frequently affected by bad weather, one of which is the rain. Due to its scattering light out and into the complementary metal oxide semiconductor of cameras and its high velocities, raindrops usually bring the bright streaks to the images or videos. Moreover, rain streaks also interfere with the nearby pixels, because of their specular highlights, scattering, and blurring effect [6]. This undesirable interference will degrade the performance of various subsequent computer vision algorithms, such as event detection [7], object detection [8, 9], tracking[10], and recognition [11], and scene analysis [12]. Therefore, removal of rain streaks is indeed considerable and essential, and has recently received much attention[1, 2, 3, 4, 5].

In general, the observation model of rainy image is formulated as $\boldsymbol{O} = \boldsymbol{B} + \boldsymbol{R}$ [6, 1, 2], which can be generalized

Table 1. Comparison of related recent works on rain removal

Method	Priors or assumptions	Detection or dictionary learning
Kang et al.[1]	Rain streaks exist only in the HF part and can be decomposed by MCA based dictionary learning and sparse coding	Both
Yi-Lei Chen et al.[2]	Rain streaks are spatio-temporally correlated, and TV regularization is discriminative for image content from highly-patterned rain streaks	Nor
Hakim et al.[3]	Rain streaks are sparse and the clean video is low-rank	Nor
Kim et al.[4]	Rain streaks are temporally correlated and the clean video is low-rank	Both
Luo et al.[5]	Local patches from both image and rain can be sparsely modeled in a learned dictionary, and their sparse codes are sufficiently discriminative	Dictionary learning
Li et al.[6]	GMM patch priors and gradient sparsity of background	Dictionary learning

to the video case: $\mathcal{O} = \mathcal{B} + \mathcal{R}$, where $\mathcal{O}$, $\mathcal{B}$ and $\mathcal{R} \in \mathbb{R}^{m\times n\times t}$ are three 3-mode tensors, indicating the observed rainy video, the unknown rain-free video and rain streaks, respectively. Rain streaks removal methods aim at separating clean video and rain streaks from the input rainy video. As we know, it is an ill-posed problem, which is traditionally coped with by enforcing priors with corresponding regularizations, in low-level computer vision. Therefore, from this perspective, the most significant issue is to rationally extract and fully utilize the prior knowledge, which is discriminative for separating the to-be-reconstructed rain-free video and rain streaks. Meanwhile, as shown in Table 1, many recent state-of-the-art rain streaks removal methods can also be viewed as conducting the separation based on some priors or assumptions.

These approaches mentioned in the Talbe. 1 are demonstrated to be effective, however there are a few drawbacks. To begin with, some of their priors or assumptions are not instinct sufficiently. Second, they focus on the rain streaks more than the rain-freed part. Actually, the rain-free part maintains a lot of useful information, which is not fully utilized. At last, most of them involve the time-consuming dictionary learning stage. Therefore, it still has room to further enhance the potential capacity and efficiency of the rain streaks removal model.

To alleviate these problems, this paper proposes a new rain streaks removal model, which fully takes the discriminatively intrinsic characteristics of rain and rain-free part into consideration. More specifically, the spatial and temporal, global and local prior knowledge is analyzed. In the spatial aspect, the directional property of the raindrops causes two different effects on the rainy video, along the raindrops' direction and its perpendicular direction respectively, which can be seen from (c-1,2,3) and (d-1,2,3) of the Fig. 1. Practically, the traditional TV regularization is applied in [6, 2], but it is not capable of handling these two different effects. Fortunately, the unidirectional TV, introduced in [13, 14], is naturally suitable, so that we adopt it to utilize the spatial priors. As for the temporal aspect, the rain-free part maintains a quite different situation with comparisons to the rain streaks and rainy part. (a-2) and (b-2) in Fig. 1 show the tighter correlation along the time axis, comparing with (a-1,3)and (b-1,3) respectively. Therefore, a tensor nuclear norm and a time directional difference operator are applied to simultaneously boost the global and local correlation of the underlying clean video along the time direction. Finally, we consider the sparsity of the rain streaks, and use an l_1 norm to guarantee it.

Our method is convex and concise, and it is easier to implement and more efficiently generates considerably better results qualitatively and quantitatively, compared with existing state-of-the-art methods. In addition, our method is practical, since it is not limited by the rain streak orientations and the dynamic/stastic of the camera or scene (see more details in Section 4.2). For all we know, this is the first method to rationally extract such priors together for the task of rain streak removal.

The outline of this paper is given as follows. In Section 2, some preliminary knowledge of tensor is given. Section 3 discusses the related works. In Section 4, the formulation of our model as well as the ADMM solver are proposed. Experimental results are reported in Section 5. Finally, we draw some conclusions in Section 6.

2. Notations and preliminaries

Following [15], we use low-case letters for vectors, e.g., $\boldsymbol{a}$, upper-case letters for matrices, e.g., $\boldsymbol{A}$, and calligraphic letters for tensors, e.g., $\mathcal{A}$. An N-mode tensor is defined as $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$, and $x_{i_1 i_2\cdots i_N}$ is its $(i_1, i_2, \cdots, i_N)$-th component.

Fibers are defined by fixing every index but one. Third-order tensors have column, row, and tube fibers, denoted by $\boldsymbol{x}_{:jk}$, $\boldsymbol{x}_{i:k}$, and $\boldsymbol{x}_{ij:}$, respectively. When extracted from the tensor, fibers are always assumed to be oriented as column vectors.

Slices are two-dimensional sections of a tensor, defined by fixing all but two indices. The horizontal, lateral, and frontal slides of a third-order tensor $\mathcal{X}$, denoted by $\boldsymbol{X}_{i::}$,

Table 2. Tensor Notations

Notations	Explanations
$\boldsymbol{\mathcal{X}}, \boldsymbol{X}, \boldsymbol{x}, x$	Tensor, matrix, vector, scalar.
$\boldsymbol{x}(:i_2 i_3 \cdots i_N)$	**Fiber** of tensor $\boldsymbol{\mathcal{X}}$ defined by fixing every index but one.
$\boldsymbol{X}(::i_3 \cdots i_N)$	**Slice** of a tensor defined by fixing all but two indices.
$\langle \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}} \rangle$	**Inner product** of two same-sized tensors $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$.
$\|\boldsymbol{\mathcal{X}}\|_F$	**Frobenius** norm of tensor $\boldsymbol{\mathcal{X}}$.
$\boldsymbol{X}_{(n)}$, $\text{unfold}_n(\boldsymbol{\mathcal{X}})$	**Mode-n unfolding** of a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ denoted as $\boldsymbol{X}_{(n)} \in \mathbb{R}^{I_n \times \prod_{i \neq n} I_i}$.
$(r_1, r_2, \cdots, r_N)$	**N-rank**, where $r_n = \textbf{rank}(\boldsymbol{X}_{(n)})$, $n = 1, 2, \cdots, N$.

$\boldsymbol{X}_{:j:}$, and $\boldsymbol{X}_{::k}$, respectively. Alternatively, the k-th frontal slice of a third-order tensor, $\boldsymbol{X}_{::k}$, may be denoted more compactly as $\boldsymbol{X}_k$.

The **inner product** of two same-sized tensors $\mathcal{X}$ and $\mathcal{Y}$ is defined as $\langle \mathcal{X}, \mathcal{Y} \rangle := \sum_{i_1, i_2, \cdots, i_N} x_{i_1 i_2 \cdots i_N} \cdot y_{i_1 i_2 \cdots i_N}$. The corresponding norm (**Frobenius norm**) is then defined as $\|\mathcal{X}\|_F := \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle}$.

The **mode-n unfolding** of a tensor $\mathcal{X}$ is denoted as $\boldsymbol{X}_{(n)} \in \mathbb{R}^{I_n \times \Pi_{i \neq n} I_i}$, where the tensor element $(i_1, i_2, \cdots, i_N)$ maps to the matrix element (i_n, j) satisfying $j = 1 + \sum_{k=1, k\neq n}^{N} (i_k - 1) J_k$ with $J_k = \prod_{m=1, m\neq n}^{k-1} I_m$. The inverse operator of unfolding is denoted as "fold", i.e., $\mathcal{X} = \text{fold}_n(\boldsymbol{X}_{(n)})$.

The **n-rank**, which we adopt in our work, is defined as an array $n\text{-rank}(\mathcal{X}) = [\text{rank}(\boldsymbol{X}_{(1)}), \text{rank}(\boldsymbol{X}_{(2)}), \cdots, \text{rank}(\boldsymbol{X}_{(N)})]$. The tensor $\boldsymbol{\mathcal{X}}$ is low-rank, if $\boldsymbol{X}_{(n)}$ is low-rank for all n.

Please refer to [15] for a more extensive overview.

3. Related work

Numerous methods are proposed to improve the visibility of images/videos captured with rain streak interference. They can be split into two categories: multiple image/video-based and single image methods.

For single image de-raining task, Kang et al.[1] decomposed the rainy image into low frequency (LF) and high frequency (HF) part, and applied an MCA based dictionary learning and sparse coding to separate the rain streaks, in the HF part. Following this elegant decomposition idea, Sun et al. [16] take the structure information into account. However, the background estimated by these methods tends to be blurry. Chen et al.[2] considered the pattern of rain streaks and the smoothness of background, but the constraints in their objective function are not sufficiently strong. Discriminative sparse coding was adopt by Luo et al.[5], but its performance is not desirable. The recent work by Li et al.[6], firstly utilizing the Gaussian mixture model (GMM) patch priors for rain streaks removal, was able to handle orientations and scales of rain streaks. Nevertheless, there is still over smooth in their results.

For video cases, Abdel-Hakim et al.[3] applied robust principle components analysis (RPCA) for rain streaks removal. Their method is limited for the statical camera and statical background. Kim et al.[4] took the temporal correlation of rain streaks and the low-rankness of clean video into account, but its effectiveness is still somehow weak for some dynamic video recorded by dynamic camera. Please refer to [17], for a more comprehensive review on the existing video-based methods. In Table 1, characteristics of recent related works are briefly introduced.

4. Tensor based video rain removal model

In general, from the point of image processing, a rainy video $\boldsymbol{\mathcal{O}} \in \mathbb{R}^{m \times n \times t}$ can be modeled as a linear superimposition:

$$\boldsymbol{\mathcal{O}} = \boldsymbol{\mathcal{B}} + \boldsymbol{\mathcal{R}},$$

where $\boldsymbol{\mathcal{B}}$ and $\boldsymbol{\mathcal{R}} \in \mathbb{R}^{m \times n \times t}$ are the unknown rain-free video and rain steaks, respectively. These three tensors are illustrated in the third column of Figure 1. Our goal is to decompose the rain-free video $\boldsymbol{\mathcal{B}}$ and the rain streaks $\boldsymbol{\mathcal{R}}$ from the input rainy video $\boldsymbol{\mathcal{O}}$. To solve this ill-posed problem, we need to analyze the priors of both $\boldsymbol{\mathcal{B}}$ and $\boldsymbol{\mathcal{R}}$, and then introduce the corresponding regularizers, which will be discussed in the next subsection.

4.1. Priors and regularizers

Sparsity of rain streaks When the rain is light, the rain streaks can naturally be considered as being sparse approximately. We can also obtain the sparsity of rain streaks from the instantiated example in Fig. 1. Hence, the enhancement of the sparsity of underlying rain streaks is helpful to the separation. To boost the sparsity of rain streaks, l_0 norm, which indicates the number of nonzero elements, is an ideal choice. Meanwhile, we can tune the parameter of the sparsity term to handle the scene with heavy rain, since that the rain streaks are always intrinsically sparser than the background clean video.

Smoothness along the rain-perpendicular direction In Fig. 1, (d-1),(d-2) and (d-3) display the pixel intensity of a fixed row in the rain-perpendicular direction, from the 10th frame of rainy video, clean video and rain streaks, respectively. It is obvious that only the variation of pixel intensity in (d-2) is piecewise smooth while burrs appear frequently in (d-1) and (d-3). Therefore, as previously mentioned, an

l_0 norm of the rain-perpendicularly unidirectional TV regularizer for $\mathcal{B}$ is a suitable candidate.

Peculiarity of the rain along the rainy direction It can be found in Fig. 1 that (c-3), which exhibits the histogram of the intensity of rain directional difference of a rain streaks frame, maintains a particular distribution with respect to (c-1) and (c-2). More zeros and smaller non-zeros values indicate the smoothness of the rain streaks along the rain direction. Naturally, we apply l_1 norm to the rain directional unidirectional TV regularizer , or said differently the rain directional difference operator, of the rain streaks $\mathcal{R}$.

Correlation along time direcion It can be found that, clean video maintains different type of correlation along the time direction from the first and second columns of the Fig. 1, compared with the rainy video and rain streaks.

On the one hand, the sub-figures (a-1), (a-2) and (a-3), which present the distributions of the magnitudes of the difference of two adjacent frames, illustrate that the difference of clean video possesses more zero values and smaller non-zero values, while the differences of the rainy video and rain streaks tend to have more and larger non-zero values. Therefore, the l_1 norm is naturally selected for the time directional difference of clean video $\mathcal{B}$.

On the other hand, (b-1), (b-2) and (b-3) respectively show the singular values of the $\boldsymbol{O}_{(t)}$, $\boldsymbol{B}_{(t)}$ and $\boldsymbol{R}_{(t)}$ in declining order, where the matrix $\boldsymbol{X}_{(t)}$ is the time mode unfolding of a tensor $\mathcal{X}$. What noteworthy is that the singular valves of $\boldsymbol{B}_{(t)}$ finally descend approximately to zeros, yet the singular values of $\boldsymbol{O}_{(t)}$ and $\boldsymbol{R}_{(t)}$ do not share this property. Thus we can conclude that, the rank minimization of $\boldsymbol{B}_{(t)}$ would promote the separation of rain streaks and clean video, although the clean video is not extremely low-rank, i.e. dynamic background and moving camera. By the way, if the video is taken by static camera or with static background, the rank minimization is more forceful. Meanwhile, as discussed in [18], there is weak correlations in video frames or natural images. Hence, we consider to minimize the rank of $\mathcal{B}$.

4.2. Formulation

As a summary of the discussion of the prior and regularization, our model can be succinctly formulated as:

$$\begin{aligned} \min_{\mathcal{B},\mathcal{R}} \quad & \alpha_1\|\nabla_1\mathcal{R}\|_0 + \alpha_2\|\mathcal{R}\|_0 + \alpha_3\|\nabla_2\mathcal{B}\|_1 \\ & + \alpha_4\|\nabla_t\mathcal{B}\|_1 + \text{rank}(\mathcal{B}), \\ \text{s.t.} \quad & \mathcal{O} = \mathcal{B} + \mathcal{R}, \quad \mathcal{B},\mathcal{R} \geqslant 0, \end{aligned} \tag{1}$$

where ∇_1 and ∇_2 are the unidirectional TV operators of rain direction and the perpendicular direction, respectively, and ∇_t indicates the time directional difference operator.

Nevertheless, the l_0 and rank terms in (1) can only take discrete values, and lead to combinatorial optimization problem in applications which is hard to solve. We thus relax them as l_1 norm and tensor nuclear norm, the definition of which is selected form [19] as $\|\mathcal{X}\|_* = \sum_{i=1}^{n}\|\boldsymbol{X}_i\|_*$, where $\boldsymbol{X}_i = \text{Unfold}_i(\mathcal{X})$.

Moreover, in real rainfall scene, the raindrops generally fall from top to bottom, so that the rain streaks' direction can be approximately counted as the mode-1 (column) direction of the video tensor. Thus rain streaks direction is denoted as y-direction while the perpendicular direction (horizontal direction) denoted as x-direction, for convenience. Commonly, there would be an angle between the y-direction and the real falling direction of raindrops. The priors, corresponding to the unidirectional TV regularizers, still exist, when the angle is small. Actually, the rain streaks in Fig. 1 is not strictly vertical, and there is a 5-degree angle. For the large angle cases, we can handle them by rotating the frames of rainy videos.

Instead of solving (1), our goal then turns to solving the following convex optimization problem:

$$\begin{aligned} \min_{\mathcal{R}} \quad & \alpha_1\|\nabla_y(\mathcal{R})\|_1 + \alpha_2\|\mathcal{R}\|_1 + \|\mathcal{O}-\mathcal{R}\|_* \\ & + \alpha_3\|\nabla_x(\mathcal{O}-\mathcal{R})\|_1 + \alpha_4\|\nabla_t(\mathcal{O}-\mathcal{R})\|_1. \end{aligned} \tag{2}$$

where $\mathcal{R} \in \mathbb{R}^{m\times n\times t}$ is the rain streaks.

An efficient algorithm is then proposed in the following section to solve the problem.

4.3. optimization

Since the proposed model (2) is a convex model, many state-of-the-art solvers are available. Here we apply the ADMM [20, 21, 22, 23], an effective strategy for solving large scale optimization problems. Firstly, five auxiliary tensors $\mathcal{Y},\mathcal{S},\mathcal{X},\mathcal{T}$ and $\mathcal{L}$ are introduced and the proposed model (2) is reformulated as the following equivalent constrained problem:

$$\begin{aligned} \min_{\mathcal{R},\mathcal{Y},\mathcal{S},\mathcal{X},\mathcal{T},\mathcal{L},} \quad & \alpha_1\|\mathcal{Y}\|_1 + \alpha_2\|\mathcal{S}\|_1 \\ & +\alpha_3\|\mathcal{X}\|_1 + \alpha_4\|\mathcal{T}\|_1 + \|\mathcal{L}\|_* \\ \text{s.t.} \quad \mathcal{Y} \; & = \nabla_y\mathcal{R}, \quad \mathcal{S} = \mathcal{R}, \\ \mathcal{X} \; & = \nabla_x(\mathcal{O}-\mathcal{R}), \\ \mathcal{T} \; & = \nabla_t(\mathcal{O}-\mathcal{R}), \\ \mathcal{L} \; & = \mathcal{O}-\mathcal{R}, \quad \mathcal{O} \geqslant \mathcal{R} \geqslant 0, \end{aligned} \tag{3}$$

where $\mathcal{S},\mathcal{Y},\mathcal{X},\mathcal{T}$ and $\mathcal{L} \in \mathbb{R}^{m\times n\times t}$.

Then the augmented Lagrangian function of (3) is:

$$
\begin{aligned}
&L_\beta(\mathcal{R},\mathcal{Y},\mathcal{S},\mathcal{X},\mathcal{T},\mathcal{L},\mathbf{\Lambda}) = \alpha_1\|\mathcal{Y}\|_1 + \alpha_2\|\mathcal{S}\|_1 \\
&\quad + \alpha_3\|\mathcal{X}\|_1 + \alpha_4\|\mathcal{T}\|_1 + \|\mathcal{L}\|_* + \langle \mathbf{\Lambda}_1, \nabla_y(\mathcal{R}) - \mathcal{Y}\rangle \\
&\quad + \frac{\beta_1}{2}\|\nabla_y(\mathcal{R}) - \mathcal{Y}\|_F^2 + \langle \mathbf{\Lambda}_2, \mathcal{R} - \mathcal{S}\rangle + \frac{\beta_2}{2}\|\mathcal{R} - \mathcal{S}\|_F^2 \\
&\quad + \langle \mathbf{\Lambda}_3, \nabla_x(\mathcal{O} - \mathcal{R}) - \mathcal{X}\rangle + \frac{\beta_3}{2}\|\nabla_x(\mathcal{O} - \mathcal{R}) - \mathcal{X}\|_F^2 \\
&\quad + \langle \mathbf{\Lambda}_3, \nabla_t(\mathcal{O} - \mathcal{R}) - \mathcal{T}\rangle + \frac{\beta_4}{2}\|\nabla_t(\mathcal{O} - \mathcal{R}) - \mathcal{T}\|_F^2 \\
&\quad + \langle \mathbf{\Lambda}_5, (\mathcal{O} - \mathcal{R}) - \mathcal{L}\rangle + \frac{\beta_5}{2}\|\mathcal{O} - \mathcal{R} - \mathcal{L}\|_F^2,
\end{aligned}
$$

where $\mathbf{\Lambda} = [\mathbf{\Lambda}_1, \mathbf{\Lambda}_2, \cdots, \mathbf{\Lambda}_5]$ is the Lagrange Multipliers and $\beta = [\beta_1, \beta_2, \cdots, \beta_5]$ are five positive scalars. Now this joint minimization problem, which can be decomposed into six easier and smaller subproblems, is able to be solved within the ADMM framework.

$\mathcal{Y}$, $\mathcal{S}$, $\mathcal{X}$, and $\mathcal{T}$ sub-problems With other parameters fixed, $\mathcal{Y}$, $\mathcal{S}$, $\mathcal{X}$ and $\mathcal{T}$ sub-problems all turn to the same format equivalent problem:

$$
\mathcal{A}^+ = \arg\min_{\mathcal{A}} \quad \alpha\|\mathcal{A}\|_1 + \frac{\beta}{2}\|\mathcal{A} - \mathcal{B}\|_F^2,
$$

which has a closed-form solution by soft thresholding:

$$
\mathcal{A}^+ = \mathcal{S}\text{hrink}_{\frac{\alpha}{\beta}}(\mathcal{B}).
$$

Here, the tensor nonnegative **soft-thresholding operator** $\mathcal{S}\text{hrink}_v(\cdot)$ is defined as

$$
\mathcal{S}\text{hrink}_v(\mathcal{B}) = \bar{\mathcal{B}}
$$

with

$$
\bar{b}_{i_1 i_2 \cdots i_N} = \begin{cases} b_{i_1 i_2 \cdots i_N} - v, & b_{i_1 i_2 \cdots i_N} > v, \\ 0, & \text{otherwise.} \end{cases}
$$

Therefore, $\mathcal{Y}$, $\mathcal{S}$, $\mathcal{X}$, and $\mathcal{T}$ can be updated as:

$$
\begin{aligned}
\mathcal{Y}^{(t+1)} &= \mathcal{S}\text{hrink}_{\frac{\alpha_1}{\beta_1}}\left(\nabla_y(\mathcal{R}^{(t)}) + \frac{\mathbf{\Lambda}_1^{(t)}}{\beta_1}\right), \\
\mathcal{S}^{(t+1)} &= \mathcal{S}\text{hrink}_{\frac{\alpha_2}{\beta_2}}\left(\mathcal{R}^{(t)} + \frac{\mathbf{\Lambda}_2^{(t)}}{\beta_2}\right), \\
\mathcal{X}^{(t+1)} &= \mathcal{S}\text{hrink}_{\frac{\alpha_3}{\beta_3}}\left(\nabla_x(\mathcal{O} - \mathcal{R}^{(t)}) + \frac{\mathbf{\Lambda}_3^{(t)}}{\beta_3}\right), \\
\mathcal{T}^{(t+1)} &= \mathcal{S}\text{hrink}_{\frac{\alpha_4}{\beta_4}}\left(\nabla_t(\mathcal{O} - \mathcal{R}^{(t)}) + \frac{\mathbf{\Lambda}_4^{(t)}}{\beta_4}\right).
\end{aligned} \tag{4}
$$

with other parameters fixed, respectively. The time complexity of the each sub-problem above is $O(mnt)$.

Algorithm 1 Algorithm for video rain streaks removal

Input: The rainy video $\mathcal{O}$;

Initialization: $\mathcal{B}^{(0)} = \mathcal{O}$, $\mathcal{R}^{(0)} =$ zeros$(m \times n \times t)$

while not converged **do**

  Update $\mathcal{Y}$, $\mathcal{S}$, $\mathcal{X}$, and $\mathcal{T}$ via (4);

  Update $\mathcal{L}$ via (5);

  Update $\mathcal{B}$ and $\mathcal{R}$ via (6);

  Update the multipiers via (7);

end while

Output: The estimation of rain-free video $\mathcal{X}$ and rain streaks $\mathcal{R}$;

$\mathcal{L}$-subproblem The $\mathcal{L}$-subproblem is:

$$
\mathcal{L}^+ = \arg\min_{\mathcal{L}} \quad \|\mathcal{L}\|_* + \frac{\beta_3}{2}\|(\mathcal{O} - \mathcal{R}) - \mathcal{L} + \frac{\mathbf{\Lambda}_3}{\beta_3}\|_F^2.
$$

Since we adopt the tensor nuclear norm definition as $\|\mathcal{X}\|_* = \sum_{i=1}^n \|\boldsymbol{X}_i\|_*$, where $\boldsymbol{X}_i = \text{Unfold}_i(\mathcal{X})$, then $\mathcal{L}$ can be updated as:

$$
\mathcal{L}^{(t+1)} = \sum_{i=1}^{3} \frac{1}{3}\text{Fold}_i(\boldsymbol{L}_i^{(t+1)}), \tag{5}
$$

where $\boldsymbol{L}_{(i)}^{(t+1)} = \mathcal{D}_{\frac{1}{\beta_3}}\left(\boldsymbol{B}_{(i)}^{(t)} + \frac{\mathbf{\Lambda}_{3}{}_{(i)}^{(t)}}{\beta_3}\right) \quad (i = 1, 2, 3)$ and $\mathcal{D}_{\frac{1}{\beta_3}}(\boldsymbol{X})$ in indicates doing soft-thresholding to the singular values of $\boldsymbol{X}$.

$\mathcal{R}$-subproblem The $\mathcal{R}$ sub-problem is a least squares problem:

$$
\begin{aligned}
\mathcal{R}^+ = \arg\min_{\mathcal{R}} \quad & \frac{\beta_1}{2}\|\nabla_y(\mathcal{R}) - \mathcal{Y} + \frac{\mathbf{\Lambda}_1}{\beta_1}\|_F^2 \\
& + \frac{\beta_2}{2}\|\mathcal{R} - \mathcal{S} + \frac{\mathbf{\Lambda}_2}{\beta_2}\|_F^2 \\
& + \frac{\beta_3}{2}\|\nabla_x(\mathcal{O} - \mathcal{R}) - \mathcal{X} + \frac{\mathbf{\Lambda}_3}{\beta_3}\|_F^2 \\
& + \frac{\beta_4}{2}\|\nabla_t(\mathcal{O} - \mathcal{R}) - \mathcal{T} + \frac{\mathbf{\Lambda}_4}{\beta_4}\|_F^2 \\
& + \frac{\beta_5}{2}\|\mathcal{O} - \mathcal{R} - \mathcal{L} + \frac{\mathbf{\Lambda}_5}{\beta_5}\|_F^2,
\end{aligned}
$$

which has the following closed-form solution:

$$
\mathcal{R}^{(t+1)} = \mathcal{F}^{-1}\left(\frac{\mathcal{F}(\mathcal{K}_1)}{\mathcal{F}(\mathcal{K}_2)}\right), \tag{6}
$$

where, $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the fast Fourier transform (FFT) and its inverse, $\mathcal{K}_1 = \nabla_y^{\mathrm{T}}(\beta_1\mathcal{Y}^{(t+1)} - \mathbf{\Lambda}_1^{(t)}) + \beta_2\mathcal{S}^{(t+1)} - \mathbf{\Lambda}_2^{(t)} + \nabla_x^{\mathrm{T}}(\beta_3\nabla_x\mathcal{O} - \beta_3\mathcal{X}^{(t+1)} + \mathbf{\Lambda}_3^{(t)}) + \nabla_t^{\mathrm{T}}(\beta_4\nabla_t\mathcal{O}^{(t+1)} - \beta_4\mathcal{T}^{(t+1)} + \mathbf{\Lambda}_4^{(t)})) + \beta_5(\mathcal{O} - \mathcal{L}^{(t+1)}) + \Lambda_5^{(t)}$ and $\mathcal{K}_2 =$

$\beta_1 \nabla_y^{\mathrm{T}} \nabla_y + \beta_2 \mathcal{I} + \beta_3 \nabla_x^{\mathrm{T}} \nabla_x + \beta_4 \nabla_t^{\mathrm{T}} \nabla_t + \mathcal{I}$. Elements in $\boldsymbol{\mathcal{R}}$, which are smller than 0 or bigger than the same elements in $\boldsymbol{\mathcal{O}}$ would be shrank. The time complexity of updating $\boldsymbol{\mathcal{R}}$ is $O(mnt \cdot \log(mnt))$

Multipliers update Following the framework of the ADMM, the Lagrange multipliers $\boldsymbol{\Lambda} = [\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \cdots, \boldsymbol{\Lambda}_5]$ can be updated as:

$$\begin{cases} \boldsymbol{\Lambda}_1^{(t+1)} = \boldsymbol{\Lambda}_1^{(t)} + \beta_1(\nabla_y(\boldsymbol{\mathcal{O}} - \boldsymbol{\mathcal{R}}^{(t+1)}) - \boldsymbol{\mathcal{Y}}^{(t+1)}), \\ \boldsymbol{\Lambda}_2^{(t+1)} = \boldsymbol{\Lambda}_2^{(t)} + \beta_2(\boldsymbol{\mathcal{O}} - \boldsymbol{\mathcal{R}}^{(t+1)} - \boldsymbol{\mathcal{S}}^{(t+1)}), \\ \boldsymbol{\Lambda}_3^{(t+1)} = \boldsymbol{\Lambda}_3^{(t)} + \beta_3(\nabla_x \boldsymbol{\mathcal{R}}^{(t+1)} - \boldsymbol{\mathcal{X}}^{(t+1)}), \\ \boldsymbol{\Lambda}_4^{(t+1)} = \boldsymbol{\Lambda}_4^{(t)} + \beta_4(\nabla_t \boldsymbol{\mathcal{R}}^{(t+1)} - \boldsymbol{\mathcal{T}}^{(t+1)}), \\ \boldsymbol{\Lambda}_5^{(t+1)} = \boldsymbol{\Lambda}_5^{(t)} + \beta_5(\boldsymbol{\mathcal{R}}^{(t+1)} - \boldsymbol{\mathcal{L}}^{(t+1)}). \end{cases} \tag{7}$$

The proposed algorithm for video rain streaks removal can be summarized in Algorithm 1. In fact, the objective function in (3) can be divided into two blocks. One is the nuclear norm term, while another block contains the other four l_1 norm terms. Hence, our algorithm fits the typical ADMM framework, and its convergency is theoretically guaranteed (see more details in the supplementary materials).

5. Experimental results

To validate the effectiveness and efficiency of the proposed method, we compare our method with recent state-of-the-art methods, including the method using temporally correlation and low-rankness [4][1] (denoted as 15'TIP), sparse coding based dictionary learning method [5][2] (denoted as 15'ICCV) and the method using layer priors [24] (denoted as 16'CVPR). Actually the 15'ICCV and 16'CVPR are single image based derain methods, but their performances sometimes surpass the video based methods. Moreover, only some frames of the experimental results using the real videos are able to be illustrated in this paper. Hence, the comparisons with these two single image based methods are reasonable and challenging. Additionally, in the following experiments, the parameters $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$ are selected from $\{10^1, 10^2, 10^3\}$ and β_i $(i = 1, 2, \cdots, 5)$ are set 50.

5.1. Synthetic data

For synthetic data, since the ground truth clean video is available, three evaluation measures are employed, including peak signal-to-noise ratio (PSNR), structure similarity (SSIM) [25] and the residual error (RES)[3]. Six videos, named as "carphone", "container", "coastguard", "bridgefar", "highway" and "foreman"[4], are selected as the ground truth videos.

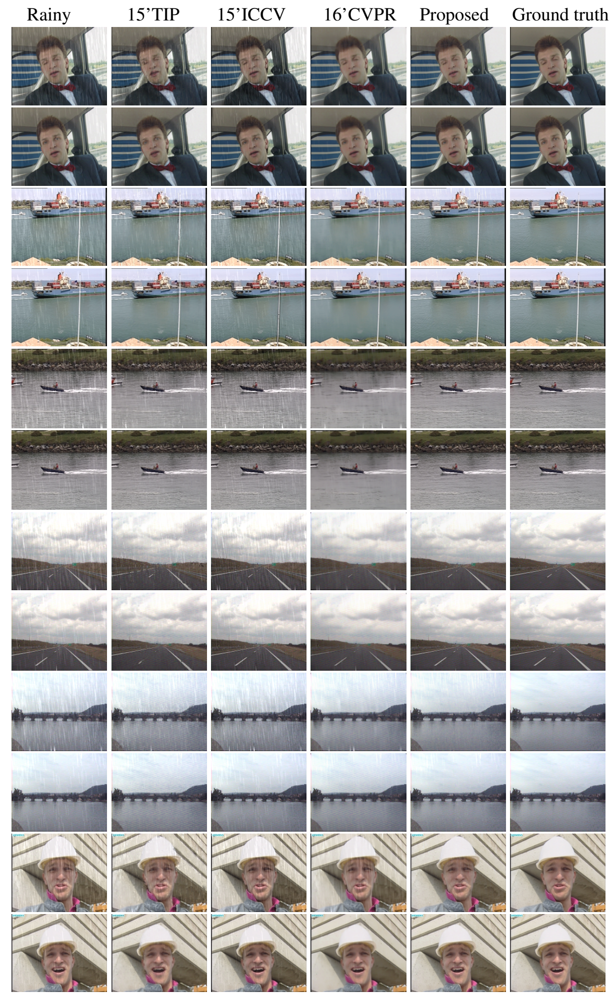

Figure 2. From left to right: the rainy frames, results of 15'TIP, 15'ICCV, 16'CVPR, the proposed method, and the ground truth frame. From top to bottom: the "carphone", "container", "coastguard", "bridgefar", "highway" and "foreman" videos with heavy and light synthetic rain, respectively.

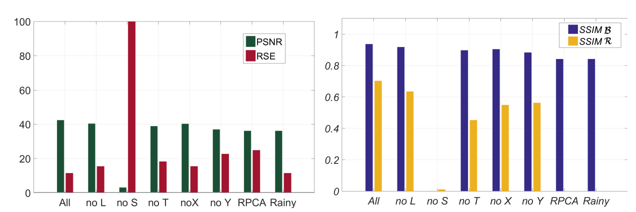

Figure 3. The performance of the proposed method and its degraded methods, which respectively leave one regularizer out.

[1]Code available on http://www.math.nus.edu.sg/∼matjh/research/ research.htm.

[2]Code available on http://mcl.korea.ac.kr/ jhkim/deraining/.

[3]Defined as RSE= $\|\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{Y}}\|_F$, where $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$ denote the estimated clean videos and the ground truth, respectively.

[4]http://trace.eas.asu.edu/yuv/.

Table 3. Quantitative comparisons of rain streaks removal results on the selected 6 synthetic videos.

	rain type	Heavy									Light								
		Whole				Average				Time (s)	Whole				Average				Time (s)
video	Method	PSNR	SSIM ($\mathcal{B}$)	SSIM ($\mathcal{R}$)	RSE	PSNR	SSIM ($\boldsymbol{B}$)	SSIM ($\boldsymbol{R}$)	RSE		PSNR	SSIM ($\mathcal{B}$)	SSIM ($\mathcal{R}$)	RSE	PSNR	SSIM ($\boldsymbol{B}$)	SSIM ($\boldsymbol{R}$)	RSE	
carphone	Rainy	26.830	0.579	—	69.176	26.843	0.614	—	7.246	—	35.256	0.771	—	26.221	35.319	0.832	—	2.739	—
	15'TIP	29.028	0.619	0.523	53.712	29.078	0.645	0.401	5.614	2029.673	34.852	0.890	0.628	27.470	35.024	0.892	0.368	2.851	1211.811
	15'ICCV	27.478	0.590	0.138	64.205	27.496	0.618	0.054	6.723	1558.478	31.280	0.777	0.111	41.446	31.336	0.827	0.046	4.331	1593.010
	16'CVPR	32.396	0.713	0.706	36.777	32.339	0.768	0.688	3.850	7582.206	34.086	0.813	0.444	31.083	33.787	0.840	0.309	3.257	7300.395
	Proposed	**33.597**	**0.820**	**0.790**	**31.741**	**33.632**	**0.819**	**0.721**	**3.320**	**11.377**	**40.104**	**0.926**	**0.732**	**15.006**	**40.532**	**0.927**	**0.431**	**1.532**	**11.230**
container	Rainy	27.634	0.558	—	63.063	27.640	0.608	—	6.608	—	36.151	0.757	0.000	23.655	36.185	0.832	—	2.475	—
	15'TIP	29.994	0.606	0.573	48.058	30.021	0.647	0.441	5.029	1750.081	35.428	0.900	0.631	25.707	35.484	0.906	0.376	2.686	1200.039
	15'ICCV	29.031	0.570	0.127	53.690	29.052	0.616	0.061	5.621	1591.627	31.082	0.763	0.090	42.398	31.106	0.829	0.040	4.439	1614.712
	16'CVPR	32.659	0.643	0.649	35.820	32.555	0.716	0.626	3.753	4497.388	33.478	0.694	0.334	33.436	33.147	0.733	0.218	3.505	5476.641
	Proposed	**37.975**	**0.910**	**0.920**	**19.174**	**37.985**	**0.913**	**0.877**	**2.008**	**11.351**	**46.730**	**0.963**	**0.814**	**6.998**	**46.771**	**0.966**	**0.489**	**0.732**	**11.447**
coastguard	Rainy	27.716	0.769	—	69.487	26.726	0.807	—	7.280	—	35.061	0.929	—	26.587	35.113	0.945	—	2.779	—
	15'TIP	33.347	0.926	**0.846**	32.385	33.599	0.924	**0.772**	3.341	2467.202	33.279	0.917	0.429	32.641	33.515	0.915	0.241	3.372	1875.336
	15'ICCV	28.531	0.790	0.176	56.389	28.595	0.819	0.093	5.889	1528.879	32.161	0.932	0.165	37.126	32.941	0.944	0.075	3.713	1737.723
	16'CVPR	30.585	0.727	0.592	46.843	30.154	0.742	0.526	4.907	4551.357	29.683	0.734	0.134	51.784	29.281	0.725	0.112	5.425	5144.503
	Proposed	**34.039**	**0.947**	0.793	**29.905**	**34.203**	**0.949**	0.724	**3.104**	**11.736**	**39.573**	**0.981**	**0.701**	**15.815**	**39.805**	**0.982**	**0.431**	**1.636**	**11.927**
highway	Rainy	27.789	0.571	—	61.947	27.801	0.623	—	6.489	—	36.208	0.841	—	23.500	36.270	0.876	—	2.455	—
	15'TIP	31.720	0.622	0.650	39.395	31.762	0.646	0.537	4.119	1681.520	35.587	0.807	0.491	25.242	35.668	0.814	0.299	2.633	1141.919
	15'ICCV	29.841	0.596	0.124	48.911	29.856	0.639	0.060	5.122	1644.783	36.639	0.855	0.111	22.361	36.690	0.883	0.049	2.337	1598.950
	16'CVPR	32.244	0.565	0.627	38.768	31.867	0.610	0.590	4.062	10327.949	32.054	0.636	0.323	40.171	31.554	0.648	0.228	4.210	4874.038
	Proposed	**36.743**	**0.831**	**0.773**	**22.096**	**36.761**	**0.840**	**0.719**	**2.313**	**11.682**	**42.457**	**0.936**	**0.702**	**11.445**	**42.552**	**0.939**	**0.429**	**1.193**	**11.707**
bridgefar	Rainy	28.128	0.584	—	59.576	28.141	0.623	—	6.240	—	36.310	0.837	—	23.224	36.381	0.858	—	2.425	—
	15'TIP	32.245	0.557	0.548	37.086	35.257	0.573	0.411	3.885	1574.131	37.469	0.781	0.488	20.323	37.492	0.781	0.254	2.128	1117.143
	15'ICCV	29.960	0.601	0.084	48.427	29.973	0.632	0.029	5.053	1638.194	34.895	0.843	0.056	27.334	34.936	0.860	0.024	2.859	1663.539
	16'CVPR	31.736	0.482	0.387	39.699	31.667	0.519	0.359	4.158	5017.966	33.527	0.516	0.180	34.718	32.820	0.525	0.133	3.639	4928.519
	Proposed	**36.342**	**0.807**	**0.696**	**23.139**	**36.352**	**0.808**	**0.640**	**2.424**	**11.353**	**42.361**	**0.925**	**0.642**	**11.571**	**42.393**	**0.920**	**0.363**	**1.211**	**11.252**
foreman	Rainy	27.128	0.682	—	66.839	27.135	0.695	—	7.004	—	35.626	0.850	0.000	25.128	35.665	0.879	0.000	2.628	—
	15'TIP	28.684	0.708	0.471	55.881	28.750	0.713	0.356	5.835	2020.531	34.443	0.927	0.563	28.794	34.959	0.923	0.298	2.923	1380.608
	15'ICCV	28.570	0.687	0.039	56.621	28.577	0.698	0.013	5.932	1608.919	33.262	0.853	0.039	32.989	33.282	0.879	0.001	3.454	1583.973
	16'CVPR	32.416	0.791	0.700	36.640	32.362	0.816	0.678	3.838	5077.714	33.645	0.854	0.375	32.900	33.290	0.862	0.276	3.448	5417.467
	Proposed	**34.324**	**0.896**	**0.825**	**29.193**	**34.525**	**0.889**	**0.756**	**3.022**	**11.196**	**39.591**	**0.956**	**0.675**	**15.919**	**40.104**	**0.952**	**0.365**	**1.618**	**11.070**

Figure 4. Results on our the Matrix.

Rain streaks generation We generate the rain by the following steps. Firstly, a salt and pepper noise is added to a zero tensor with the same size as the ground truth videos. The denser the noise is, the heavier the synthetic rain will be. Then, a motion blur is added to the noisy zero tensor, and a small angle (5 degree) exists between the motion direction and the y-axis. Finally, the blurred noisy zero tensor is linearly superposed to the ground truth videos, and the intensities of pixels, which are greater than 1, are set as 1.

Discussion of each component There are five components in our model (2). To make their effects clear, we test our method by leaving each component out, respectively. Additionally, when only containing the sparse and low-rank terms, our model degrades to a robust principle components analysis model, which is similar to the method in [3]. We

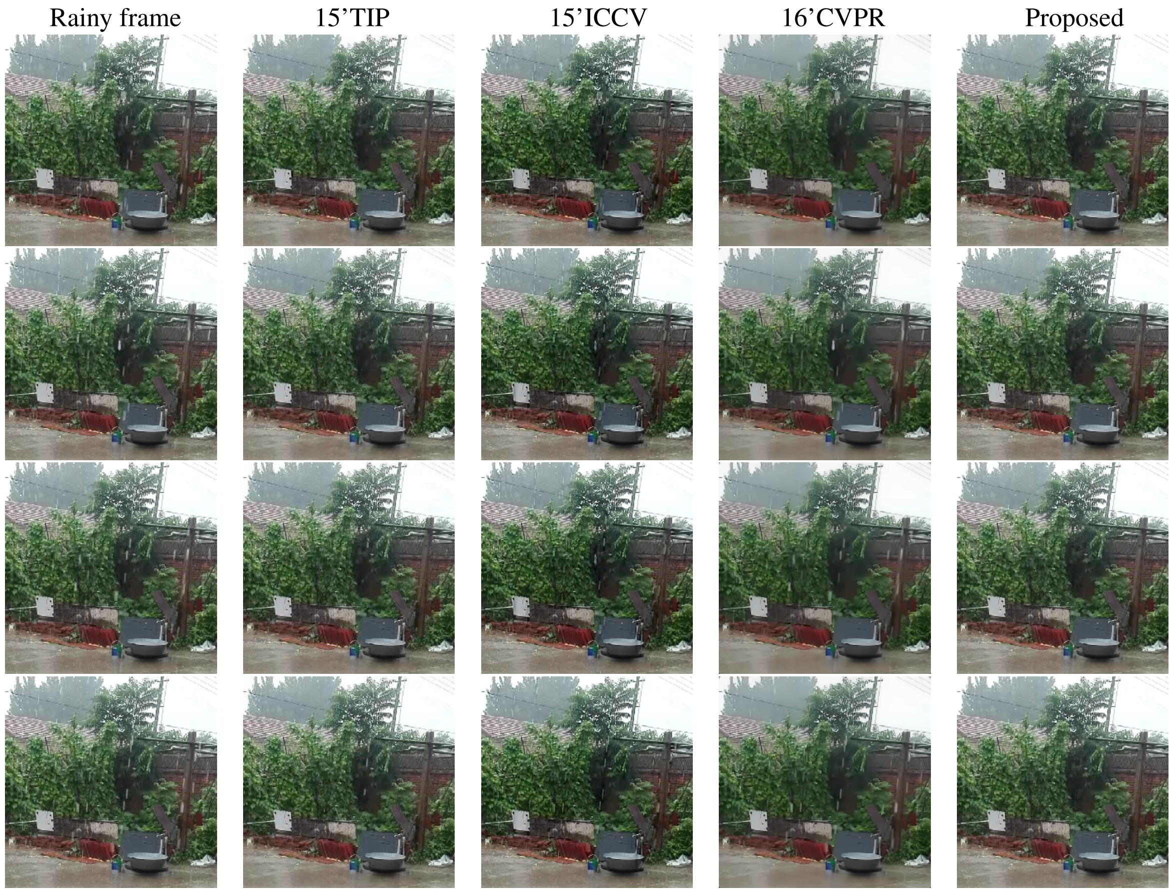

Figure 5. Results on our video.

show the performances of the proposed method and its degraded versions in Fig. 3. We can conclude that each component contributes to the separation of rain streaks.

Performance comparisons Fig. 2 shows one frame of the results of 15’TIP, 15’ICCV, 16’CVPR and the proposed method, while the corresponding quantitative comparisons are presented in Table 3. As observed, our method considerably outperforms the other three methods in terms of both visual quality on the selected three evaluation measures. With reference to the ground truth (the right most column in Fig. 2), our method removes almost all rain streaks and maintains details, while many rain streaks still exist in the results of 15’TIP and 15’ICCV. Although the 16’CVPR method removes more rain streaks than 15’TIP and 15’ICCV, spatial details are erased. For instance, in the “coastguard” video (the 5th and 6th row in Fig. 2), water waves are smoothed by 16’CVPR, while well preserved by our method. Furthermore, it is inspiring that our method takes significantly less time than other three methods.

5.2. Real data

Fig. 4 and Fig. 5 show four adjacent frames of the results. The first real video is clipped from the well-known movie “the Matrix”, and the second one is recorded by one of the authors in a rainy day. Qualitatively, our method provides the best results both on removing rain streaks and retaining spatial details. We can see that there are still many rain streaks on the results of 15’TIP and 15’ICCV, while 16’CVPR erases some spatial details, for instance, the nose of Agent Smith in the 2nd frame and the leaves in Fig. 5. Besides, when the camera is dynamic, the rapid changing between two adjacent frames seriously effects the performance of 15’TIP. More experimental results of real data, including rotation case, will be presented in the supplementary materials.

6. Conclusion

We have proposed a novel tensor based approach to remove the video rain streaks. Actually, it is a bit counter-intuitive to see the derivation of total-variation, cooperated with low-rankness, beats the derivation of sparse dictionary learning and patch prior, because the latter two significantly outperformed total-variation in image denoising. Apart from that the video based methods can utilize more information than image based approaches, we attribute the outperformance of our method to our intensive analysis on the priors of rainy videos, clean videos and rain streaks. As a matter of fact, the priors, taken into consideration by us, varied from spatial to temporal, from local to global. Hence, it is reasonable to achieve such performance.

Our method is not without limitations. If the rainy direction is far away from the y-axis, we can handle it with image rotation, but for the digital data, the rotation inevitably causes distortion. In addition, how to handle the remaining rain artifacts is still an open problem. These issues are targeted for future work.

Acknowledgment

The authors would like to express their thanks to Dr. Yu Li for sharing the codes of the 16'CVPR [24]. This research is supported by 973 Program (2013CB329404), the National Science Foundation of China (61370147, 61402082), the Fundamental Research Funds for the Central Universities (ZYGX2013Z005).

References

[1] 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*, vol. 21, no. 4, pp. 1742–1755, 2012.

[2] 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*, 2013, pp. 1968–1975.

[3] A. E. Abdel-Hakim, "A novel approach for rain removal from videos using low-rank recovery," in *2014 5th IEEE International Conference on Intelligent Systems, Modelling and Simulation*, 2014, pp. 351–356.

[4] J.-H. Kim, J.-Y. Sim, and C.-S. Kim, "Video deraining and desnowing using temporal correlation and low-rank matrix completion," *IEEE Transactions on Image Processing*, vol. 24, no. 9, pp. 2658–2670, 2015.

[5] 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*, 2015, pp. 3397–3405.

[6] 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*, 2016, pp. 2736–2744.

[7] M. S. Shehata, J. Cai, W. M. Badawy, T. W. Burr, M. S. Pervez, R. J. Johannesson, and A. Radmanesh, "Video-based automatic incident detection for smart roads: the outdoor environmental challenges regarding false alarms," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 2, pp. 349–360, 2008.

[8] S. Maji, A. C. Berg, and J. Malik, "Classification using intersection kernel support vector machines is efficient," in *IEEE Conference on Computer Vision and Pattern Recognition, 2008.* IEEE, 2008, pp. 1–8.

[9] O. L. Junior, D. Delgado, V. Gonçalves, and U. Nunes, "Trainable classifier-fusion schemes: an application to pedestrian detection," in *Intelligent Transportation Systems*, vol. 2, 2009.

[10] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 25, no. 5, pp. 564–577, 2003.

[11] K. Garg and S. K. Nayar, "Vision and rain," *International Journal of Computer Vision*, vol. 75, no. 1, pp. 3–27, 2007.

[12] L. Itti, C. Koch, E. Niebur *et al.*, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[13] M. Bouali and S. Ladjal, "Toward optimal destriping of modis data using a unidirectional variational model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 8, pp. 2924–2935, 2011.

[14] Y. Chang, L. Yan, H. Fang, and C. Luo, "Anisotropic spectral-spatial total variation model for multispectral remote sensing image destriping," *IEEE Transactions on Image Processing*, vol. 24, no. 6, pp. 1852–1866, 2015.

[15] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[16] S.-H. Sun, S.-P. Fan, and Y.-C. F. Wang, "Exploiting image structural similarity for single image rain removal," in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 4482–4486.

[17] A. K. Tripathi and S. Mukhopadhyay, "Removal of rain from videos: a review," *Signal, Image and Video Processing*, vol. 8, no. 8, pp. 1421–1430, 2014.

[18] W. Cao, Y. Wang, J. Sun, D. Meng, C. Yang, A. Cichocki, and Z. Xu, "Total variation regularized tensor rpca for background subtraction from compressive measurements," *IEEE Transactions on Image Processing*, vol. 25, no. 9, pp. 4075–4090, 2016.

[19] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," 2013, pp. 2114–2121.

[20] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations & Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[21] X.-L. Zhao, F. Wang, T.-Z. Huang, M. K. Ng, and R. J. Plemmons, "Deblurring and sparse unmixing for hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 7, pp. 4045–4058, 2013.

[22] A. C. Sauve, A. Hero, W. L. Rogers, S. Wilderman, and N. Clinthorne, "3D image reconstruction for a Compton SPECT camera model," *IEEE Transactions on Nuclear Science*, vol. 46, no. 6, pp. 2075–2084, 1999.

[23] Y. Xu, W. Yin, Z. Wen, and Y. Zhang, "An alternating direction algorithm for matrix completion with nonnegative factors," *Frontiers of Mathematics in China*, vol. 7, no. 2, pp. 365–384, 2012.

[24] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, "Rain streak removal using layer priors," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[25] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.