A Generalized Low-Rank Appearance Model for Spatio-Temporally Correlated Rain Streaks

Yi-Lei Chen Chiou-Ting Hsu
Department of Computer Science,
National Tsing Hua University, Taiwan
fallcolor@gmail.com cthsu@cs.nthu.edu.tw

Abstract

In this paper, we propose a novel low-rank appearance model for removing rain streaks. Different from previous work, our method needs neither rain pixel detection nor time-consuming dictionary learning stage. Instead, as rain streaks usually reveal similar and repeated patterns on imaging scene, we propose and generalize a low-rank model from matrix to tensor structure in order to capture the spatio-temporally correlated rain streaks. With the appearance model, we thus remove rain streaks from image/video (and also other high-order image structure) in a unified way. Our experimental results demonstrate competitive (or even better) visual quality and efficient run-time in comparison with state of the art.

1. Introduction

Dynamic weather, such as rain, snow, and haze, usually brings unpleasant visual artifacts in outdoor vision system and would deteriorate the performance of subsequent vision tasks. In comparison with other deweathering methods (e.g. dehazing), rain removal might be the most challenging one because raindrops usually lead to dense streaks with unpredictable behaviors (e.g., direction, rainfall intensity) over imaging plane. Since these dense streaks introduce additional gradients and would hinder the detection of reliable features, rain removal is indeed crucial and indispensible for in-depth image analysis.

Garg et.al [1-4] first devoted to build the appearance model of rain streak and propose to detect rain pixels in video. As pointed out in [1], falling raindrops produce rain streaks with properties in both rain appearance and dynamic motion. The chromaticity [5] and shape [6] of rain streaks were also considered in other research. Recently, Barnum et.al [7] adopted frequency domain analysis to model rain and snow, and Bossu et.al [8] utilized histogram of orientation to detect rain or snow streaks. All of these methods [1-8] work on video inputs and rely on the correct detection of corrupted pixels. However, once the temporal information becomes unreliable (e.g., unstablized video) or unavailable (e.g., single frame), their performance and applicability would be severely degraded.

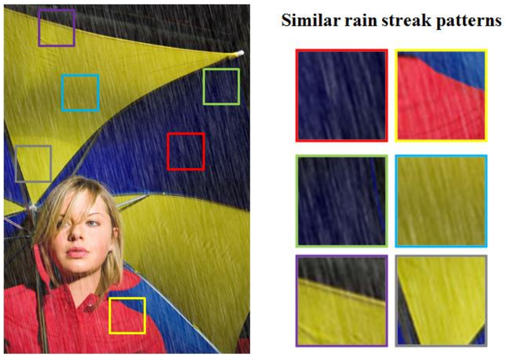


Figure 1: An observation of similar patterns in a raining image.

Considering rain removal on one single frame, Kang et.al [9] proposed an image decomposition-based method to remove rain streaks without referring to temporal features. They first removed low-frequency content of raining images, and then self-learned two dictionaries followed by sparse coding to decompose one image patch into rain streak and high-frequency content. Such decomposition-based method shows good discriminability without using detection stage. However, the time-consuming dictionary learning may limit their practicality. Recently, a guided-image-filter (GIF) based method [10] was proposed for rain removal on one single color image. The authors utilized the color channel difference as guidance, which serves as non-rain prior, to conduct guided image filtering. Since the proposed non-rain prior only roughly captures the image content, removal of large raindrops would also over-smooth the edges and textures. Thus the results usually look unrealistic and of poor quality.

Different from existing methods, in this paper, we propose a novel low-rank rain appearance model on 2-D images, and generalize our model to high-order image structure (e.g., color image, video). Therefore, the proposed model is not limited to any particular source input. Based on this model, we then propose to decompose the corrupted input into *rain streak component*, *rain-free component*, and *imaging noise*. Note that, our method needs no pre-processing (e.g., rain detection, subtraction of low frequency content) which may causes other side effects, and also needs no dictionary learning because the low-rank model inherently captures the subspace spanned by rain streak patterns. Inspired by [10], we also propose a GIF-based detail enhancement by reusing the estimated imaging noise. The experiments of image/video rain removal will show both the effectiveness and efficiency of our approach.

1.1. Motivation and related work

We use Fig. 1 to explain our motivation: *a rainy scene usually contains similar patterns of rain streak in different local patches*. With this observation, our goal is to model the patch dependency in an elegant way. In this paper, we assume this dependency is linear and propose to use low-rank model to characterize the appearance of rain streaks. Although the success of low-rank property has been shown in modeling the image background [11] and noise-free [12] patches, no investigation for rain streak pattern has been studied. One idea similar to ours is [13], where the authors rearranged non-overlapping patches into a matrix and proposed to minimize the matrix rank. Such low-rank model separated the texture part from cartoon part for image decomposition. In comparison with [13], we propose a more general, higher-order low-rank model (i.e., tensor object) with non-convex rank penalty. We will show that our generalization improves both the performance and applicability in image/video rain removal.

1.2. Notations and tensor basics

We summarize the notations used in this paper: lower case letters ($x, y, \ldots$) denote scalars, bold lower case letters ($\mathbf{x}, \mathbf{y}, \ldots$) denote vectors, bold upper case letters ($\mathbf{X}, \mathbf{Y}, \ldots$) are matrices, and calligraphic upper case letters ($\mathcal{X}, \mathcal{Y}, \ldots$) are higher order tensors. $(\mathbf{X})_{i,j}$ denotes the $(i,j)^{\text{th}}$ element of $\mathbf{X}$, and is similarly defined in tensor. $\sigma_r(\mathbf{X})$ is the r^{th} largest singular value of matrix $\mathbf{X}$. $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote fast Fourier transform (FFT) operator and the inverse FFT, respectively. $\overline{\mathbf{X}}$ is the complex conjugate of $\mathbf{X}$. As to tensor operation, the mode-k unfolding of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ is defined by a matrix $\mathbf{X}^{(k)} \in \mathbb{R}^{I_k \times \prod_{k' \neq k} I_{k'}}$, and the mode-$k$ folding indicates the inverse process.

2. Appearance model of rain streaks

We consider an image $\mathbf{I}$ captured in a rainy scene as the addition of rain streaks $\mathbf{I}_{\mathrm{R}}$ over a rain-free image $\mathbf{I}_{\mathrm{S}}$:

$$\mathbf{I} = \mathbf{I}_{\mathrm{R}} + \mathbf{I}_{\mathrm{S}} \,, \tag{1}$$

In equation (1), the key to successful factorization is to effectively characterize the rain streaks $\mathbf{I}_{\mathrm{R}}$. Instead of modeling the appearance of each raindrop, here we first give two observations on common rain streaks:

i Rain streaks usually have similar directions in the same imaging environment.

ii Raindrops, which fall at a nearly constant speed, reveal similar rain streaks for a period of time.

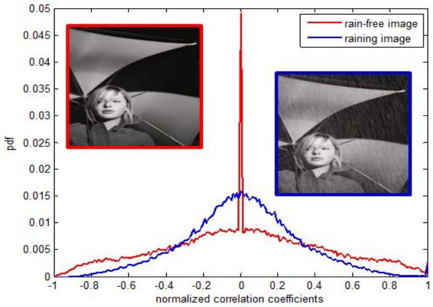


Figure 2: The NCC distribution of raining/rain-free images.

The first observation indicates the similarity between rains streaks at different spatial locations, and the second implies the repeatability of rain streaks along time axis. Both observations suggest high spatio-temporal correlation between rain streaks in a rainy scene. To verify our observations, in Fig. 2, we measure and show the distributions of normalized correlation coefficients (NCC) between different patches on raining/rain-free images. The NCCs of rain-free image are close to zero and show that these patches are mostly uncorrelated. On the other hand, once the image is corrupted by rain streaks, their patch dependency would obviously increase because of those superposed rain streak patterns. This strong evidence verifies our observations in patch level.

Therefore, we define a rain streak pattern as one local image patch and assume a bag of patches collected from image/video are linearly dependent. In this paper, we modify the *patch map* process [13] to capture the patch dependency. In [13], the authors divided an image $\mathbf{I} \in \mathbb{R}^{m \times n}$ into mn/r^2 non-overlapping patches by the mapping $\mathrm{P}: \mathbb{R}^{m \times n} \to \mathbb{R}^{r^2 \times (mn/r^2)}$, and defined the inverse mapping P^{-1} by $\mathbb{R}^{r^2 \times (mn/r^2)} \to \mathbb{R}^{m \times n}$. Different from [13], we further consider the patch structure across two non-overlapping patches and generalize this mapping to high order image structure:

i $\mathrm{P}: \mathbb{R}^{m \times n} \to \mathbb{R}^{r^2 \times k}$ and $k \geq mn/r^2$.

ii $\mathrm{P}: \mathbb{R}^{m \times n \times D_3 \times \cdots \times D_N} \to \mathbb{R}^{r^2 \times k \times D_3 \times \cdots \times D_N}$

Note that, our *patch map* function using overlapping patches may bring additive bias to the objective costs. However, we will show in the next section such bias could be neglected by proper normalization in our algorithm.

Using the proposed *patch map* function, now we define the appearance model of rain streaks by

$$\operatorname{rank}(\mathrm{P}(\mathbf{I}_{\mathrm{R}})) \,, \tag{2}$$

if the input $\mathbf{I}$ is a two-dimensional image; otherwise, we define the model in high-order image structure by

$$\operatorname{rank}(\mathrm{P}(\mathcal{I}_{\mathrm{R}})) \,. \tag{3}$$

3. Image/video rain removal

Using the low-rank rain appearance model defined in section 2, we propose a generalized rain removal method.

Very different from existing methods working on one specific source input (e.g., video [1-8], single frame [9], color image [10]), our proposed method is feasible to every source input. In section 3.1, we first describe how we estimate $\mathbf{I}_\mathrm{R}$ and $\mathbf{I}_\mathrm{S}$ in equation (1) when the input is a 2-D image. Then we extend our algorithm to high-order image structure in section 3.2. Finally, we propose a simple but effective rain removal method incorporating with an efficient detail enhancement in section 3.3.

3.1. Rain streak estimation on 2-D images

In section 2, we have proposed a low-rank model for the common appearance of rain streak pattern. To characterize the rain-free image $\mathbf{I}_\mathrm{S}$, we use the isotropic total variation (TV) as the regularization term of $\mathbf{I}_\mathrm{S}$:

$$\|\mathbf{I}_\mathrm{S}\|_\mathrm{TV} = \sum_{i,j}\sqrt{(g_x * \mathbf{I}_\mathrm{S})_{i,j}^2 + (g_y * \mathbf{I}_\mathrm{S})_{i,j}^2}\ , \qquad (4)$$

where g_x and g_y are the gradient operators along x and y axes, respectively, and $*$ denotes the convolution operation. Note that, although TV has been used to generate piecewise-smooth "cartoon" layer [14] of images, it tends to characterize edge-aware structure regardless of small-scale features (e.g., texture). Therefore, we do not claim the TV term could completely restore natural images. Instead, the role of TV regularization is to discriminate most of natural image content from highly-patterned rain streaks and thus facilitate the ill-posed factorization in equation (1).

By further considering the imaging noise $\mathbf{I}-\mathbf{I}_\mathrm{R}-\mathbf{I}_\mathrm{S}$, we formulate our rain streak estimation as

$$\hat{\mathbf{I}}_\mathrm{R}, \hat{\mathbf{I}}_\mathrm{S} = \operatorname{argmin} \alpha \operatorname{rank}\big(\mathrm{P}(\mathbf{I}_\mathrm{R})\big) + \beta\|\mathbf{I}_\mathrm{S}\|_\mathrm{TV} + \tfrac{1}{2}\|\mathbf{I}-\mathbf{I}_\mathrm{R}-\mathbf{I}_\mathrm{S}\|_\mathrm{F}^2\ . \qquad (5)$$

Equation (5) is an NP-hard rank minimization problem because the matrix rank is difficult to approximate. To make it tractable, most existing methods [11-13] used the tightest convex envelope of matrix rank - *Schatten 1-norm*, also called *nuclear norm*, to replace the rank function. Equation (5) is accordingly modified as

$$\hat{\mathbf{I}}_\mathrm{R}, \hat{\mathbf{I}}_\mathrm{S} = \operatorname{argmin} \alpha \sum_r\big|\sigma_r\big(\mathrm{P}(\mathbf{I}_\mathrm{R})\big)\big|^p + \beta\|\mathbf{I}_\mathrm{S}\|_\mathrm{TV} + \tfrac{1}{2}\|\mathbf{I}-\mathbf{I}_\mathrm{R}-\mathbf{I}_\mathrm{S}\|_\mathrm{F}^2\ . \qquad (6)$$

Although [11-13] all suggested $p = 1$ could guarantee global optimum for convex problems, a recent study in [15] pointed out that the non-convex penalty ($0 < p < 1$) usually leads to superior performance by their generalized shrinkage operation. In our method, we suggest using the non-convex rank penalty. A thorough comparison will be given in experiment sections.

Another difficulty in equation (6) mainly lies in the nonlinearity and non-differentiability of TV term. Thanks to the success of split Bregman iteration [16], we use a similar variable splitting technique and obtain

$$\hat{\mathbf{I}}_\mathrm{R}, \hat{\mathbf{I}}_\mathrm{S} = \operatorname{argmin} \alpha \sum_r\big|\sigma_r\big(\mathrm{P}(\mathbf{I}_\mathrm{R})\big)\big|^p + \beta\sum_{i,j}\Big\|\big[(\mathcal{I}_\mathrm{g})_{1,i,j}, (\mathcal{I}_\mathrm{g})_{2,i,j}\big]^\mathrm{T}\Big\|_{2,1} + \frac{1}{2}\|\mathbf{I}-\mathbf{I}_\mathrm{R}-\mathbf{I}_\mathrm{S}\|_\mathrm{F}^2$$

$$\text{s.t.}\ (\mathcal{I}_\mathrm{g})_{1,i,j} = (g_x * \mathbf{I}_\mathrm{S})_{i,j} \text{ and } (\mathcal{I}_\mathrm{g})_{2,i,j} = (g_y * \mathbf{I}_\mathrm{S})_{i,j}\ . \qquad (7)$$

In equation (7), $\mathcal{I}_\mathrm{g} \in \mathbb{R}^{2\times m\times n}$ is an auxiliary variable for the gradient vector. Now every gradient vector could be pixel-wisely optimized by *l_{21}- norm* [17].

To solve equation (7) efficiently, we adopt the inexact Augmented Lagrange Multiplier (IALM) method [18], whose convergence property has been well-studied in [19]. We first introduce the augmented Lagrange function L_1:

$$\mathrm{L}_1\big(\mathbf{I}_\mathrm{R}, \mathbf{I}_\mathrm{S}, \mathcal{I}_\mathrm{g}, \mathcal{Y}, \mu\big) = \alpha \sum_r\big|\sigma_r\big(\mathrm{P}(\mathbf{I}_\mathrm{R})\big)\big|^p + \beta\sum_{i,j}\Big\|\big[(\mathcal{I}_\mathrm{g})_{1,i,j}, (\mathcal{I}_\mathrm{g})_{2,i,j}\big]^\mathrm{T}\Big\|_{2,1} + \tfrac{1}{2}\|\mathbf{I}-\mathbf{I}_\mathrm{R}-\mathbf{I}_\mathrm{S}\|_\mathrm{F}^2 + \langle \mathcal{Y}, \mathcal{I}_\mathrm{g}-\mathcal{I}_\mathrm{h}\rangle + \tfrac{\mu}{2}\big\|\mathcal{I}_\mathrm{g}-\mathcal{I}_\mathrm{h}\big\|_\mathrm{F}^2\ , \qquad (8)$$

where $\mathcal{I}_\mathrm{h} \in \mathbb{R}^{2\times m\times n}$ denotes the stack of $g_x * \mathbf{I}_\mathrm{S}$ and $g_y * \mathbf{I}_\mathrm{S}$ in order to simplify our notations. Following the update rule of IALM, we could obtain at least one global or local optimum of equation (7) by

$$\begin{gathered}
{\mathbf{I}_\mathrm{R}}^{t+1} = \operatorname{argmin} \mathrm{L}_1\big({\mathbf{I}_\mathrm{R}}^{t}, {\mathbf{I}_\mathrm{S}}^{t}, {\mathcal{I}_\mathrm{g}}^{t}, \mathcal{Y}^t, \mu^t\big)\ ,\\
{\mathbf{I}_\mathrm{S}}^{t+1} = \operatorname{argmin} \mathrm{L}_1\big({\mathbf{I}_\mathrm{R}}^{t+1}, {\mathbf{I}_\mathrm{S}}^{t}, {\mathcal{I}_\mathrm{g}}^{t}, \mathcal{Y}^t, \mu^t\big)\ ,\\
{\mathcal{I}_\mathrm{g}}^{t+1} = \operatorname{argmin} \mathrm{L}_1\big({\mathbf{I}_\mathrm{R}}^{t+1}, {\mathbf{I}_\mathrm{S}}^{t+1}, {\mathcal{I}_\mathrm{g}}^{t}, \mathcal{Y}^t, \mu^t\big)\ ,\\
\mathcal{Y}^{t+1} = \mathcal{Y}^t + \mu^t\big({\mathcal{I}_\mathrm{g}}^{t+1} - {\mathcal{I}_\mathrm{h}}^{t+1}\big)\ \text{, and}\\
\mu^{t+1} = \rho\mu^t\ ,
\end{gathered} \qquad (9)$$

where t is the iteration index and ρ is a penalty parameter (>1). Below we detail how to update $\mathbf{I}_\mathrm{R}, \mathbf{I}_\mathrm{S}, \mathcal{I}_\mathrm{g}$, respectively.

3.1.1 Optimization of $\mathbf{I}_R$

By ignoring those terms independent of $\mathbf{I}_\mathrm{R}$, we obtain the sub-Lagrange function:

$$\alpha \sum_r\big|\sigma_r\big(\mathrm{P}(\mathbf{I}_\mathrm{R})\big)\big|^p + \tfrac{1}{2}\|\mathbf{I}_\mathrm{R} - (\mathbf{I}-\mathbf{I}_\mathrm{S})\|_\mathrm{F}^2\ . \qquad (10)$$

To make equation (10) tractable, we could rewrite the noise term $\|\mathbf{I}_\mathrm{R} - (\mathbf{I}-\mathbf{I}_\mathrm{S})\|_\mathrm{F}^2$ into $\|\mathrm{P}(\mathbf{I}_\mathrm{R}) - \mathrm{P}(\mathbf{I}-\mathbf{I}_\mathrm{S})\|_\mathrm{F}^2$. Recall that our proposed *patch map* function allows overlapping patches and hence there is a bias between the two objective costs. Therefore, we multiply a variable λ and obtain

$$\alpha \sum_r\big|\sigma_r\big(\mathrm{P}(\mathbf{I}_\mathrm{R})\big)\big|^p + \tfrac{\lambda}{2}\|\mathrm{P}(\mathbf{I}_\mathrm{R}) - \mathrm{P}(\mathbf{I}-\mathbf{I}_\mathrm{S})\|_\mathrm{F}^2\ , \qquad (11)$$

where the normalization term λ is given by $\|\mathbf{I}\|_\mathrm{F}^2/\|\mathrm{P}(\mathbf{I})\|_\mathrm{F}^2$ at first and updated by $\big\|{\mathbf{I}_\mathrm{R}}^t - (\mathbf{I}-{\mathbf{I}_\mathrm{S}}^t)\big\|_\mathrm{F}^2/\big\|\mathrm{P}({\mathbf{I}_\mathrm{R}}^t) - \mathrm{P}(\mathbf{I}-{\mathbf{I}_\mathrm{S}}^t)\big\|_\mathrm{F}^2$ in the $(t+1)^\mathrm{th}$ iteration.

Equation (11) is a typical low-rank matrix approximation which has a closed-form optimum. Using the generalized shrinkage operation [15], we have

$$\hat{\mathbf{I}}_\mathrm{R} = \mathrm{P}^{-1}\big(\mathbf{U}\boldsymbol{\Sigma}_{\alpha/\lambda}\mathbf{V}^\mathrm{T}\big)\ , \qquad (12)$$

where $\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\mathrm{T}$ is the singular value decomposition (SVD) of matrix $\mathrm{P}(\mathbf{I}-\mathbf{I}_\mathrm{S})$ and $\boldsymbol{\Sigma}_{\alpha/\lambda}$ denotes the diagonal matrix with all shrunk singular values of $\mathrm{P}(\mathbf{I}-\mathbf{I}_\mathrm{S})$, i.e.,

$$\hat{\sigma}_r = \max(|v| - (\alpha/\lambda)|v|^{p-1}, 0)\frac{v}{|v|}\ , \quad \text{and } v = \sigma_r\big(\mathrm{P}(\mathbf{I}-\mathbf{I}_\mathrm{S})\big)\ . \qquad (13)$$

3.1.2 Optimization of $\mathbf{I}_S$

The sub-Lagrange function w.r.t $\mathbf{I}_\mathrm{S}$ is

$$\frac{1}{2}\|\mathbf{I}_{\mathrm{S}}-(\mathbf{I}-\mathbf{I}_{\mathrm{R}})\|_{\mathrm{F}}^{2}+ \frac{\mu}{2}\left(\|g_x*\mathbf{I}_{\mathrm{S}}-\mathbf{I}_{\mathrm{x}}\|_{\mathrm{F}}^{2}+\left\|g_y*\mathbf{I}_{\mathrm{S}}-\mathbf{I}_{\mathrm{y}}\right\|_{\mathrm{F}}^{2}\right), \quad (14)$$

where $(\mathbf{I}_{\mathrm{x}})_{i,j}=\left(\mathcal{I}_{\mathrm{g}}+\mathcal{Y}/\mu\right)_{1,i,j}$, $\left(\mathbf{I}_{\mathrm{y}}\right)_{i,j}=\left(\mathcal{I}_{\mathrm{g}}+\mathcal{Y}/\mu\right)_{2,i,j}$. According to Plancherel's theorem [20], we derive the closed-form optimum by FFT:

$$\hat{\mathbf{I}}_{\mathrm{S}}=\mathcal{F}^{-1}\left(\frac{\mathcal{F}(\mathbf{I}-\mathbf{I}_{\mathrm{R}})+\mu(\overline{\mathcal{F}(g_x)}\circ\mathcal{F}(\mathbf{I}_{\mathrm{x}})+\overline{\mathcal{F}(g_y)}\circ\mathcal{F}(\mathbf{I}_{\mathrm{y}}))}{1+\mu(\overline{\mathcal{F}(g_x)}\circ\mathcal{F}(g_x)+\overline{\mathcal{F}(g_y)}\circ\mathcal{F}(g_y))}\right). \quad (15)$$

Using FFT in equation (15), where all the operations are element-wise, we need not to solve a large linear system.

3.1.3 Optimization of $\mathcal{I}_g$

Since every gradient vector could be pixel-wisely optimized, we update $\mathcal{I}_{\mathrm{g}}$ by the shrinkage operation [17]. Let $\phi=\left\|[(\mathcal{I}_{\mathrm{h}}-\mathcal{Y}/\mu)_{1,i,j},(\mathcal{I}_{\mathrm{h}}-\mathcal{Y}/\mu)_{2,i,j}]^{\mathrm{T}}\right\|$, we have

$$\left(\mathcal{I}_{\mathrm{g}}\right)_{l,i,j}=\frac{\phi-\beta/\mu}{\phi}(\mathcal{I}_{\mathrm{h}}-\mathcal{Y}/\mu)_{l,i,j}\,,\ l=1,2$$

if $\phi>\beta/\mu$; otherwise, we have

$$\left(\mathcal{I}_{\mathrm{g}}\right)_{l,i,j}=0\,,\ l=1,2\,. \quad (16)$$

To demonstrate the effectiveness of our model, we give the estimated rain streaks of some 2-D raining images (released by [9]) in Fig. 3. The results show that our method accurately captures flexible rain streaks from heavy/light rains or with different directions. In addition, almost no image content is mistakenly estimated as rain streaks.

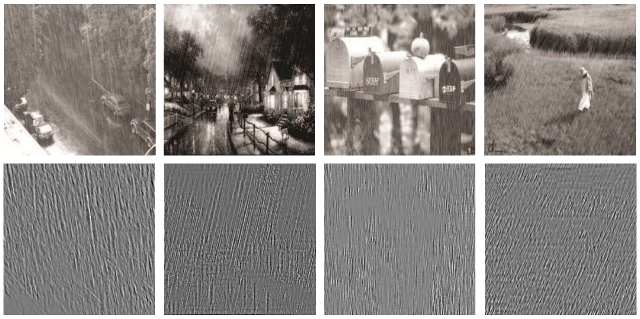


Figure 3: Some examples (released by [9]) of rain streak estimation. The 1st row shows the original raining images and the 2nd row shows their rain streaks estimated by our method.

3.2. Rain streaks estimation on high-order images

We generalize equation (5) into high-order images by replacing rank(P($\mathbf{I}_{\mathrm{R}}$)) with rank(P($\mathcal{I}_{\mathrm{R}}$)). Considering the temporal coherency, we also modify $\|\mathcal{I}_{\mathrm{S}}\|_{\mathrm{TV}}$ by including the gradients along t axis:

$$\sum_{i_1,\ldots,i_N}\sqrt{(g_x*\mathcal{I}_{\mathrm{S}})_{i_1,\ldots,i_N}^{2}+(g_y*\mathcal{I}_{\mathrm{S}})_{i_1,\ldots,i_N}^{2}+(\omega g_t*\mathcal{I}_{\mathrm{S}})_{i_1,\ldots,i_N}^{2}}\,, \quad (17)$$

where ω is set as 0 for color images and is 1 for videos. Thanks to the success of tensor trace norm recently developed in [21], we propose to extend the *Schatten p-norm* from matrix to tensor in our model. In [21], the authors defined the trace norm of $\mathcal{I}\in\mathbb{R}^{D_1\times D_2\times D_3\times\cdots\times D_N}$ by $\|\mathcal{I}\|_*=\frac{1}{N}\sum_{d=1}^{N}\left\|\mathbf{I}^{(d)}\right\|_*=\frac{1}{N}\sum_{d=1}^{N}\sum_{r_d}\left|\sigma_{r_d}\left(\mathbf{I}^{(d)}\right)\right|$. Therefore, we accordingly propose to define the *Schatten p-norm* of $\mathcal{I}$ by $\frac{1}{N}\sum_{d=1}^{N}\sum_{r_d}\left|\sigma_{r_d}\left(\mathbf{I}^{(d)}\right)\right|^p$ and have

$$\hat{\mathcal{I}}_{\mathrm{R}},\hat{\mathcal{I}}_{\mathrm{S}}=\operatorname{argmin}\frac{\alpha}{N}\sum_{d=1}^{N}\sum_{r_d}\left|\sigma_{r_d}(\mathrm{P}(\mathbf{I}_{\mathrm{R}}^{(d)}))\right|^p+ \beta\|\mathcal{I}_{\mathrm{S}}\|_{\mathrm{TV}}+\frac{1}{2}\|\mathcal{I}-\mathcal{I}_{\mathrm{R}}-\mathcal{I}_{\mathrm{S}}\|_{\mathrm{F}}^{2}\,. \quad (18)$$

To relax the interdependency of N *Schatten p-norm*, we first rewrite equation (18) into

$$\hat{\mathcal{I}}_{\mathrm{R}},\hat{\mathcal{I}}_{\mathrm{S}}=\operatorname{argmin}\frac{\alpha}{N}\sum_{d=1}^{N}\sum_{r_d}\left|\sigma_{r_d}(\mathbf{W}_d)\right|^p+ \beta\|\mathcal{I}_{\mathrm{S}}\|_{\mathrm{TV}}+\frac{1}{2}\|\mathcal{I}-\mathcal{I}_{\mathrm{R}}-\mathcal{I}_{\mathrm{S}}\|_{\mathrm{F}}^{2}\quad \text{s.t. } \mathbf{W}_d=\mathrm{P}\left(\mathbf{I}_{\mathrm{R}}^{(d)}\right). \quad (19)$$

Similar to equation (7), we include an additional equality constraint $\left(\mathcal{I}_{\mathrm{g}}\right)_{3,i_1,\ldots,i_N}=(\omega g_t*\mathcal{I}_{\mathrm{S}})_{i_1,\ldots,i_N}$, and use IALM to solve equation (19) by introducing L_2:

$$\begin{aligned}\mathrm{L}_2\left(\mathcal{I}_{\mathrm{R}},\mathcal{I}_{\mathrm{S}},\mathcal{I}_{\mathrm{g}},\mathcal{Y},\mathbf{Y}_1,\ldots,\mathbf{Y}_N,\mu\right)&=\frac{\alpha}{N}\sum_{d=1}^{N}\sum_{r_d}\left|\sigma_{r_d}(\mathbf{W}_d)\right|^p\\ &+\beta\sum_{i_1,\ldots,i_N}\left\|\left[\left(\mathcal{I}_{\mathrm{g}}\right)_{1,i_1,\ldots,i_N},\left(\mathcal{I}_{\mathrm{g}}\right)_{2,i_1,\ldots,i_N},\left(\mathcal{I}_{\mathrm{g}}\right)_{3,i_1,\ldots,i_N}\right]^{\mathrm{T}}\right\|_{2,1}\\ &+\frac{1}{2}\|\mathcal{I}-\mathcal{I}_{\mathrm{R}}-\mathcal{I}_{\mathrm{S}}\|_{\mathrm{F}}^{2}+\langle\mathcal{Y},\mathcal{I}_{\mathrm{g}}-\mathcal{I}_{\mathrm{h}}\rangle+\frac{\mu}{2}\left\|\mathcal{I}_{\mathrm{g}}-\mathcal{I}_{\mathrm{h}}\right\|_{\mathrm{F}}^{2}\\ &+\sum_{d=1}^{N}\left(\left\langle\mathbf{Y}_d,\mathbf{W}_d-\mathrm{P}\left(\mathbf{I}_{\mathrm{R}}^{(d)}\right)\right\rangle+\frac{\mu}{2}\left\|\mathbf{W}_d-\mathrm{P}\left(\mathbf{I}_{\mathrm{R}}^{(d)}\right)\right\|_{\mathrm{F}}^{2}\right). \quad (20)\end{aligned}$$

Our algorithm is built on the one in section 3.1. To update $\mathcal{I}_{\mathrm{S}}$ and $\mathcal{I}_{\mathrm{g}}$, we replace the 2-D FFT operator in equation (15) by N-D FFT and also accordingly modify ϕ in equation (16). The major difference lies in the estimation of $\mathcal{I}_{\mathrm{R}}$. Below we detail how to update the prima variable $\mathcal{I}_{\mathrm{R}}$ and the auxiliary variables $\mathbf{W}_1,\ldots,\mathbf{W}_N$.

3.2.1 Optimization of $\boldsymbol{W}_d$

The sub-Lagrange function w.r.t $\mathbf{W}_d$ is

$$\frac{\alpha}{N}\sum_r|\sigma_r(\mathbf{W}_d)|^p+\frac{\mu}{2}\left\|\mathbf{W}_d-\left(\mathrm{P}\left(\mathbf{I}_{\mathrm{R}}^{(d)}\right)-\mathbf{Y}_{\mathrm{d}}/\mu\right)\right\|_{\mathrm{F}}^{2}. \quad (21)$$

By generalized shrinkage operation [15], we have

$$\mathbf{W}_d=\mathbf{U}\mathbf{\Sigma}_{\alpha/\mu N}\mathbf{V}^{\mathrm{T}}, \quad (22)$$

where $\mathbf{U\Sigma V}^{\mathrm{T}}$ is the SVD of matrix $\mathrm{P}\left(\mathbf{I}_{\mathrm{R}}^{(d)}\right)-\mathbf{Y}_{\mathrm{d}}/\mu$.

3.2.2 Optimization of $\mathcal{I}_R$

The sub-Lagrange function w.r.t $\mathcal{I}_{\mathrm{R}}$ is

$$\frac{1}{2}\|\mathcal{I}_{\mathrm{R}}-\mathcal{I}+\mathcal{I}_{\mathrm{S}}\|_{\mathrm{F}}^{2}+\frac{\mu}{2}\sum_{d=1}^{N}\left\|\mathrm{P}\left(\mathbf{I}_{\mathrm{R}}^{(d)}\right)-(\mathbf{W}_d+\mathbf{Y}_d/\mu)\right\|_{\mathrm{F}}^{2}. \quad (23)$$

Similar to section 3.1.1, we introduce the normalization factor λ and obtain

$$\frac{1}{2}\|\mathcal{I}_{\mathrm{R}}-\mathcal{I}+\mathcal{I}_{\mathrm{S}}\|_{\mathrm{F}}^{2}+\frac{\mu}{2\lambda}\sum_{d=1}^{N}\left\|\mathbf{I}_{\mathrm{R}}^{(d)}-\mathrm{P}^{-1}(\mathbf{W}_d+\mathbf{Y}_d/\mu)\right\|_{\mathrm{F}}^{2}. \quad (24)$$

Let $\mathcal{Z}_d$ be the mode-d folding of matrix $\mathrm{P}^{-1}(\mathbf{W}_d+\mathbf{Y}_d/\mu)$. The optimum is derived as

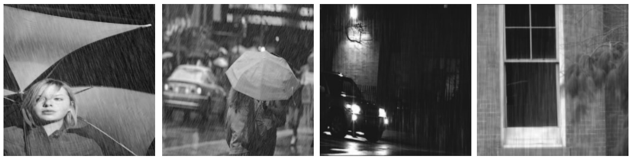
$$\hat{\mathcal{I}}_{\mathrm{R}}=\frac{\lambda(\mathcal{I}-\mathcal{I}_{\mathrm{S}})+\mu\sum_{d=1}^{N}\mathcal{Z}_d}{\lambda+\mu N}\,. \quad (25)$$

Note that, although our proposed model inherently characterize the appearance of rain streaks with various image dimensions, similar image patches along time axis (especially captured by a static camera) may also exhibit low-rank structure and severely degrades the performance.

To avoid $\mathcal{I}_{\mathrm{R}}$ capturing image content, after the *patch map* process, we conduct random patch permutation on each 2-D image to alleviate the temporal consistency.

3.3. Rain streaks removal

When using the rain-free image $\mathbf{I}_{\mathrm{S}}$ as our rain removal result, we observe that the noise layer also contains little image content or details. To fully utilize such information, we conduct guided image filtering (GIF) [23] to enhance the image contrast. The basic idea of GIF is to generate an output image which is not only similar to the input image but also exhibits the structure of the guidance image. Since most rain streaks are characterized by $\mathbf{I}_{\mathrm{R}}$(or $\mathcal{I}_{\mathrm{R}}$) and leave little trace on the residual, we use the noise layer as the input and $\mathbf{I}-\mathbf{I}_{\mathrm{R}}$ (or $\mathcal{I}-\mathcal{I}_{\mathrm{R}}$) as our guidance. The filtered result, denoted by $\mathbf{I}_{\mathrm{D}}$ (or $\mathcal{I}_{\mathrm{D}}$), then serves as the detail layer to improve the visual quality. Finally, we obtain the rain-removal result as $\mathbf{I}_{\mathrm{S}}+\tau\mathbf{I}_{\mathrm{D}}$ or $\mathcal{I}_{\mathrm{S}}+\tau\mathcal{I}_{\mathrm{D}}$.



(a) (b) (c) (d)

Figure 4: Four synthetic raining images.

4. Experimental results

We first detail our experimental settings. All images and video frames are resized into 256x256, and we empirically determine the patch size *r* as16 and the patch offset as 12 to balance the tradeoff between visual quality and efficiency. As to the parameters, we fix μ and ρ as 0.1 and 1.1 in all cases, and adaptively determine α as the k^{th} largest singular value of P($\mathbf{I}$), where $k=\lfloor 0.95*N_r \rfloor$ and N_r is the number of nonzero singular values. Only one parameter β needs to be fine-tuned. In general, smaller β leads to cleaner but fewer rain streaks. In our experiments, we test β from 0.001 to 0.01 to get the most visually-pleasant result.

4.1. Rain removal on monochromatic images

We use four synthetic raining images (see Fig. 4) released by [9] to evaluate our method. Because the ground truth images are also available, we could compare our results with [9] under the same criterion VIF [24]. Due to the limited pages, we only show one example in Fig. 5 for subjective assessment and give all quantitative results in Table 1. Two cases of rank penalty (*p*=1, 0.5) in our method are first compared. We fine-tune β to guarantee the estimated $\mathbf{I}_{\mathrm{R}}$ have similar average of magnitudes. From Fig.5 (a) and (e), the case of *p*=0.5 removes more rain streaks and achieves better visual quality. The reason can be explained by Fig. 6, where smaller *p* more significantly reduces the patch rank under similar average of magnitudes. Since smaller *p* means a better approximation of matrix rank, it is superior to capture the underlying rain streak patterns and is also less sensitive to image content.

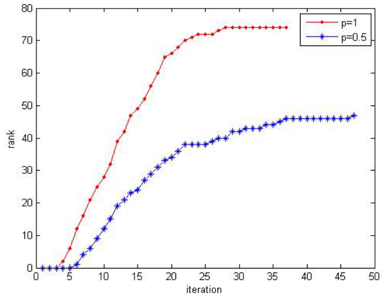


Figure 6: The estimated rank of rain streak patterns using *p*=1 and *p*=0.5.

To compare with the high-frequency (HF) part estimated in [9] (see Fig. 5 (k)), we also follow the same procedure to subtract the low-frequency part (bilateral filtering results of raining image) from our two rain-removed images. From Fig. 5, our method with *p*=0.5 achieves very competitive result with [9] in terms of visual quality. When comparing the rain streaks and HF content, we observe that [9] could preserve small-scale features very well but tends to lose edge structure similar to rain streak pattern (see the red rectangles). The reason is because their method mainly relies on the discriminability of dictionary bases on individual patch. In contrast, our method could preserve global image structure which benefits from the TV term, and thus retain most HF content. Although our method may lose small-scale features, we could still use the proposed detail enhancement to compensate such information.

Because of the superiority of using a smaller *p*, we fix *p*=0.5 in the other three images and also the following experiments. In Table 1, the proposed method outperforms [9] by VIF in all images. These results again verify our model in the ability of single-frame-based rain removal.

TABLE 1. EVALUATION OF IMAGE RAIN REMOVAL BY VIF METRIC [24]. THE RESULTS OF [9] ARE REPORTED IN [9].

		Fig. 4(a)	*Fig. 4(b)*	*Fig. 4(c)*	*Fig. 4(d)*
[9]		0.5	0.53	0.36	0.56
[9] (+ extended dictionary)		0.52	0.57	0.38	0.60
Ours	$\tau=0$	**0.5275**	0.5909	0.4037	0.6290
	$\tau=3$	0.5204	**0.5957**	0.4148	0.6446
	$\tau=5$	0.5120	0.5950	**0.4195**	**0.6518**

4.2. Rain removal on color images and videos

We use three video sequences, including the “heavy rain” released by [2], the “mag” released by [3], and the “forrest” released by [7], to evaluate our method. As to the case of single color image, we pick one frame from “heavy rain” to conduct rain removal, and also implement the GIF-based

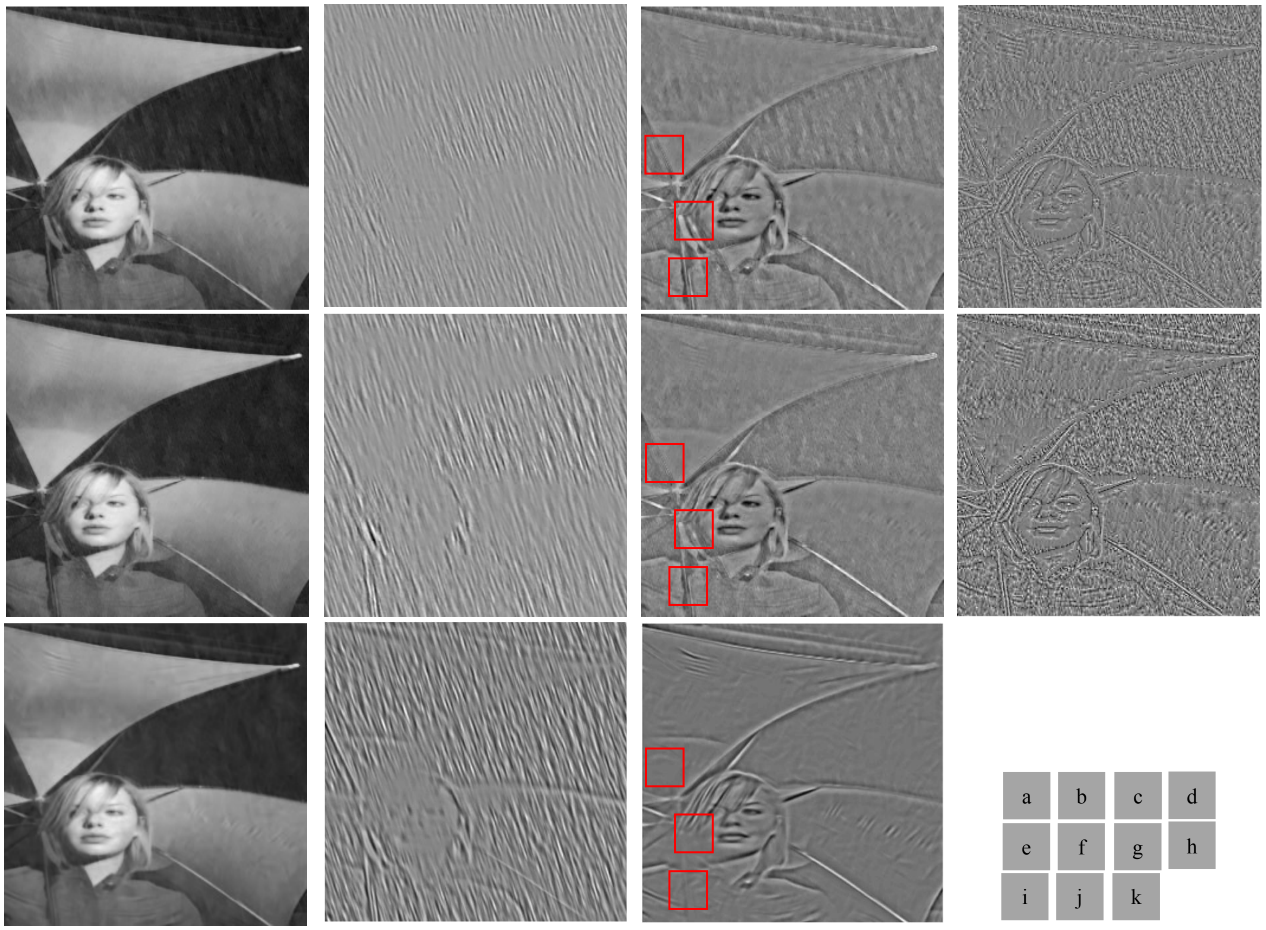


Figure 5: Results of gray-level image rain removal (the 1st column) and their estimated rain streaks (the 2nd column), high frequency content (the 3rd column), and imaging noise (the 4th column). The images in (a)-(d) and (e)-(h) are obtained by our method using p=1 and p=0.5, respectively. The results in (i)-(k) are obtained by [9]. (The noise term is not provided in [9])

method [10] for comparison. We show the rain-removed results in Fig. 7 (b)-(c) and their estimated rain streaks in Fig. 7 (f)-(g). Although we have fine-tuned the GIF parameters, [10] still obtains over-smooth result with inaccurate rain streaks because their proposed guidance image only estimates rough image content. In contrast, our method removes rain streaks much accurately.

As to the case of video rain removal, all of the results could be found in our supplementary files. Here we also show the same frame in the image-based case and its corresponding rain streaks in Fig. 7 for comparison. Because now we further suppress the temporal gradients by equation (17), the rain-free image (Fig. 7(d)) contains almost no rain streaks and achieves better visual quality in comparison with our image-based result (Fig. 7(c)). In addition, the estimated rain streaks (Fig. 7(h)) contain less content structure than the image-based one (Fig. 7(g)). We also show the detail-enhanced image in Fig. 7(e), and zoom-in their texture parts in Fig. 7(i)-(j). One could observe that the local contrast becomes clearer and more visually-pleasant due to the proposed detail enhancement.

Fig. 8-9 show our rain-removed frames of the other two sequences in comparison with the results released by [3] and [7], respectively. Note that, [3] and [7] only work on video input and need a “rain detection” stage either in pixel or frequency domain. The two-stage rain removal usually leads to flickering artifacts if rain pixels are not completely detected (please refer to our video results for comparison). In contrast, our method tends to obtain spatio-temporally consistent results (see the zoomed-in areas in Fig. 8-9). One may notice that, in Fig. 9, although our method removes more accurate “rain streak”, some of the large raindrops still remain. However, this result is expected because our goal is to remove dense rain streaks instead of raindrops. Finally, as pointed out in [5], raindrops bring similar changes for three color channels at different pixels. We find that our rain streak layer is nearly monochromatic when conducting our method on the whole color image/video (see the video result “heavy rain”); i.e., our high-order model inherently characterizes this chromatic property.

4.3. Discussion

We implement our method in MATLAB on the PC

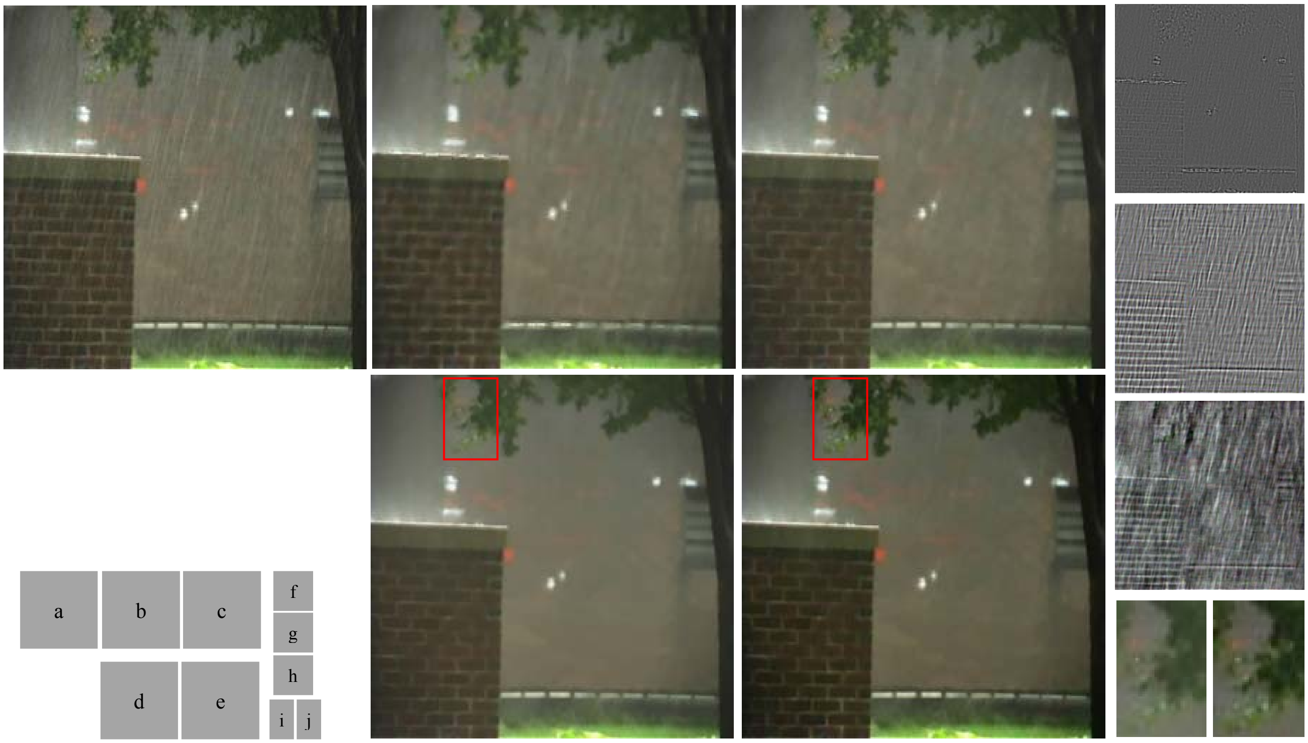


Figure 7: Results of color frame rain removal. (a) The 1st frame of sequence "heavy rain"; (b)-(d) and (f)-(h) are the rain-removed images and removed components obtained by [10], our image-based method, and our video-based method; (e) is the detail-enhanced result of (d) with $\boldsymbol{\tau} = \mathbf{5}$; and (i)-(j) are the enlarged red rectangles in (d)-(e), respectively.

equipped with Intel Core i7-2600 processor and 4-GB memory. The run time of bilateral filtering and our rain removal on each image is about 1.05 and 3.2 seconds, while the report in [9] is about 1.5 and 8 to 22 seconds but they need more than one minute for dictionary learning. This result shows that our method is much more efficient than [9] in the single-frame-based rain removal and also achieves competitive or even better quality. One may notice, in Fig. 7, the brick wall is also captured by our model because of its strongly-correlated pattern. This is indeed the limitation of our low-rank model. However, we could easily tackle this problem via a "rain dictionary" $\mathbf{D}$; i.e., in equation (5), we replace $\mathrm{rank}\big(\mathrm{P}(\mathbf{I}_{\mathrm{R}})\big)$ by $\mathrm{rank}(\mathbf{Z})$ and further include $\mathrm{P}(\mathbf{I}_{\mathrm{R}}) = \mathbf{DZ}$. Fig. 10 shows an example, where we use a very simple method (rather than KSVD learning in [9]) to obtain $\mathbf{D}$: first subtracting the temporally mean-filtered result, and extracting HOG features from overlapped patches to conduct 2-means clustering. The larger cluster is then selected as our dictionary. This strategy is quite efficient and, as shown in Fig. 10 (c), indeed improves the estimated rain streak layer substantially.

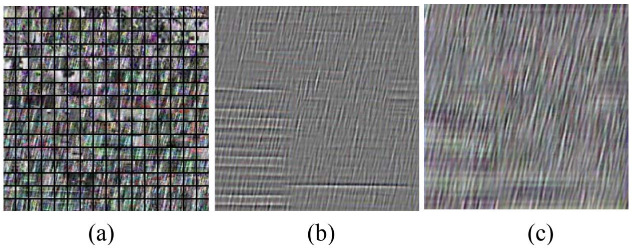

(a) (b) (c)

Figure 10: (a) The rain dictionary $\mathbf{D}$ learned from "heavy rain", and the estimated rain streaks (b) without $\mathbf{D}$ and (c) with $\mathbf{D}$.

5. Conclusion

In this paper, we propose a generalized model, which utilizes the nice properties of low-rank pattern, for common rain streak appearance. The proposed model characterizes the spatio-temporally correlated rain streaks, and thus could tackle both image and video rain removal. In comparison with existing methods, our major contribution is threefold: 1) the proposed model is feasible to different source inputs without any pre-processing; 2) our method inherently characterizes the dictionary spanned by low-rank rain streak patterns without learning; and 3) different from pixel/patch-based methods, both global structure (image total variation) and local appearance (low-rank patch) are leveraged in our method. Finally, the experimental results demonstrate our superiority in both subjective/objective evaluation and time complexity.

References

[1] K. Garg and S. K. Nayar. Vision and rain. *Int. J. Comput. Vis.*, 75(1):3–27, 2007.

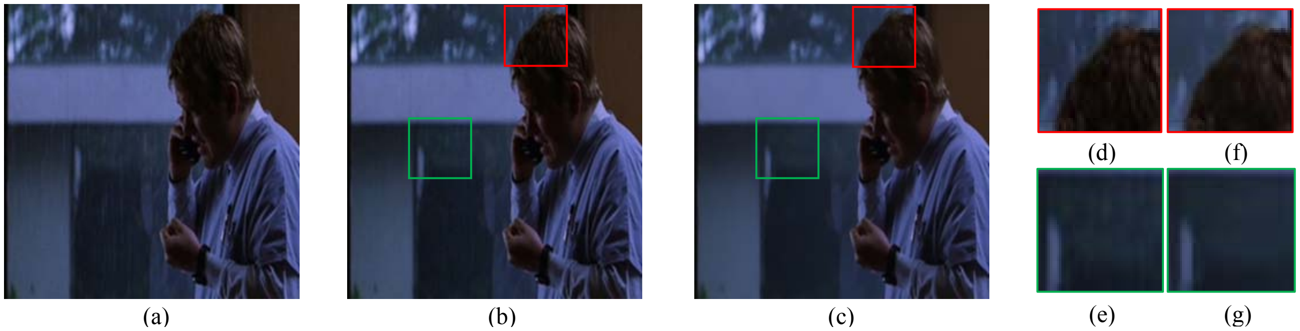


Figure 8: (a) A selected frame from video “mag”; and the rain-removed results obtained by (b) [3]; and (c) our method. (d)-(e) and (f)-(g) show the enlarged red/green rectangles in (b)-(c), respectively.

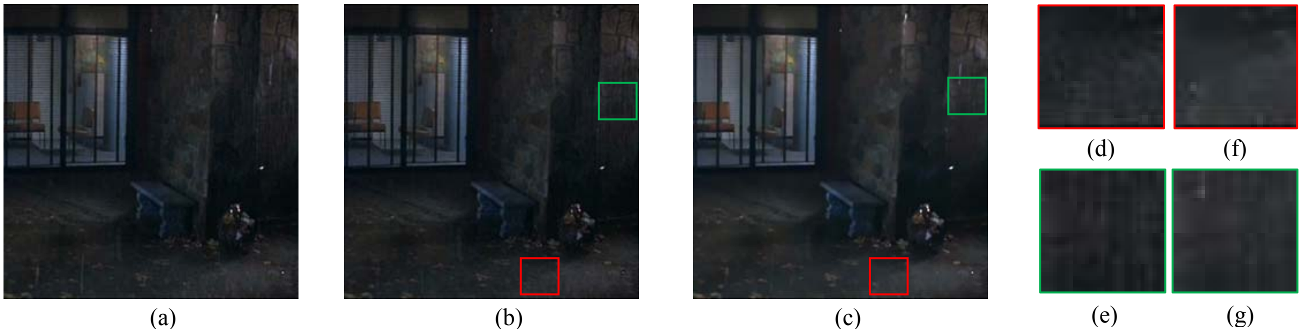


Figure 9: (a) A selected frame from video “forrest”; and the rain-removed results obtained by (b) [7]; and (c) our method. (d)-(e) and (f)-(g) show the enlarged red/green rectangles in (b)-(c), respectively.

[2] K. Garg and S. K. Nayar. When does a camera see rain? In *Proc. ICCV*, 2005.

[3] K. Garg and S. K. Nayar. Detection and removal of rain from videos. In *Proc. CVPR*, 2004.

[4] K. Garg and S. K. Nayar. Probabilistic rendering of rain streaks. *ACM Trans. Graph*., 25(3):996-1002, 2006.

[5] X. Zhang, H. Li, Y. Qi, W. K. Leow, and T. K. Ng. Rain removal in videos by combining temporal and chromatic properties. In *Proc. ICME*, 2006.

[6] N. Brewer and N. Liu. Using the shape characteristics of rain to identify and remove rain from video. In *Int. Conf. Workshop on Structural, Syntactic and Statistical Pattern Recognition*, 2008.

[7] P. C. Barnum, S. Narasimhan, and T. Kanade. Analysis of rain and snow in frequency space. *Int. J. Comput. Vis*., 86(2-3):256–274, 2010.

[8] J. Bossu, N. Hautiere, and J. P. Tarel. Rain and snow detection in image sequences through use of histogram of orientation of streaks. *Int. J. Comput. Vis*., 93(3):348–367, 2011.

[9] L.-W. Kang, C.-W. Lin, and Y.-H. Fu. Automatic single-image-based rain streaks removal via image decomposition. *TIP*, 21(4):1742-1755, 2012.

[10] J. Xu, W. Zhao, P. Liu, and X. Tang. Removing rain and snow in as single image using guided filter. In *Int. Conf. Computer Science and Automation Engineering*, 2012.

[11] C. Lang, G. Liu, J. Yu, and S. Yan. Saliency detection by multitask sparsity pursuit. *TIP*, 21(3):1327-1338, 2012.

[12] H. Ji, C. Liu, Z. Shen, and Y. Xu. Robust video denoising using low rank mj. atrix completion. In *Proc. CVPR*, 2010.

[13] H. Schaeffer and S. Osher. A low patch-rank interpretation of texture. CAM Report, 2011.

[14] 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.

[15] R. Chartrand. Nonconvex splitting for regularized low-rank+sparse decomposition. *TSP*, 60(11):5810-5819, 2012.

[16] T. Goldstein and S. Osher. The split bregman method for L1-regularized problems. *SIAM Journal on Imaging Science*, 2(2):323, 2009.

[17] G. Liu, Z. Lin, and Y. Yu. Robust subspace segmentation by low-rank representation. In *Proc. ICML*, 2010.

[18] Z. Lin, M. Chen, L. Wu, and Y. Ma. The augmented lagrange multiplier method for exact recover of corrupted low-rank matrices. UIUC Technical Report UILU-ENG-09-2215, 2009.

[19] B. He, M. Tao, and X. Yuan. Alternating direction method with gaussian back substitution for separable convex programming. *SIAM J. Optim.*, 22(2):313-340, 2012.

[20] R. N. Bracewell. *The Fourier transform and its applications*. McGtaw-Hill Science, 3 Edition, 1999.

[21] J. Liu, P. Wonka, and J. Ye. Tensor completion for estimating missing values in visual data. In *Proc. ICCV*, 2009.

[22] M. Signoretto, L .D. Lathauwer, and J. A. K. Suykens. Nuclear norms for tensors and their use for convex multilinear estimation. Internal Report 10-186, ESAT-SISTA, K. U. Leuven, 2010.

[23] K. He, J. Sun, and X. Tang. Guided image filtering. In *Proc. ECCV*, 2010.

[24] H. R. Sheikh and A. C. Bovik. Image information and visual quality. *TIP*, 15(2):430-444, 2006.