Video Rain Streak Removal By Multiscale Convolutional Sparse Coding

Minghan Li[1], Qi Xie[1], Qian Zhao[1], Wei Wei[1], Shuhang Gu[2], Jing Tao[1], Deyu Meng[1] *
[1]National Engineering Laboratory for Algorithm and Analysis Technologiy on Big Data and Ministry of Education Key Lab of Intelligent Networks and Network Security, Xian Jiaotong University
[2]computer vision lab, eth, zurich
{liminghan, xq.liwu}@stu.xjtu.edu.cn, timmy.zhaoqian@gmail.com,
weiweiwe@stu.xjtu.edu.cn, shuhanggu@gmail.com,{jtao, dymeng}@mail.xjtu.edu.cn

Abstract

Videos captured by outdoor surveillance equipments sometimes contain unexpected rain streaks, which brings difficulty in subsequent video processing tasks. Rain streak removal from a video is thus an important topic in recent computer vision research. In this paper, we raise two intrinsic characteristics specifically possessed by rain streaks. Firstly, the rain streaks in a video contain repetitive local patterns sparsely scattered over different positions of the video. Secondly, the rain streaks are with multiscale configurations due to their occurrence on positions with different distances to the cameras. Based on such understanding, we specifically formulate both characteristics into a multiscale convolutional sparse coding (MS-CSC) model for the video rain streak removal task. Specifically, we use multiple convolutional filters convolved on the sparse feature maps to deliver the former characteristic, and further use multiscale filters to represent different scales of rain streaks. Such a new encoding manner makes the proposed method capable of properly extracting rain streaks from videos, thus getting fine video deraining effects. Experiments implemented on synthetic and real videos verify the superiority of the proposed method, as compared with the state-of-the-art ones along this research line, both visually and quantitatively.

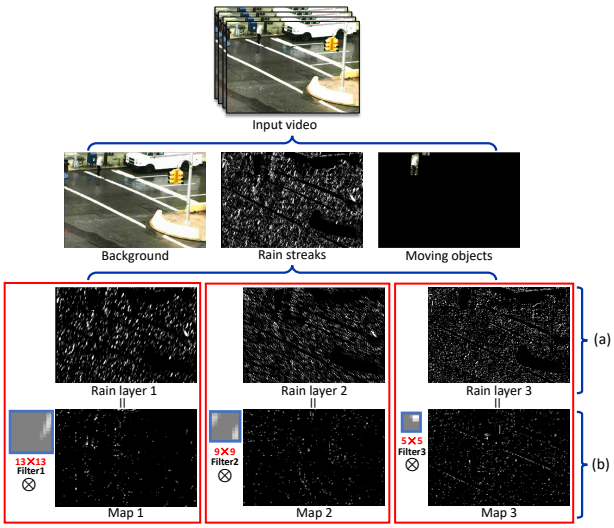


Figure 1. An natural rainy video (upper) is separated into three layers (middle) of background scene, rain streaks and moving objects by the proposed multiscale convolutional sparse coding (MS-CSC) model. The rain streaks can be decomposed into diverse rain structures (lower row (a)), corresponding to different scales of rain appearance. All those decompositions are attained through three scale filters convolved on sparse feature maps (lower row (b)).

1. Introduction

Rainy videos captured by outdoor surveillance equipments may degenerate the performance of subsequent video processing tasks, like human detection [8], person re-identification [10], stereo correspondence [14], object tracking and recognition [29], and scene analysis [19]. Thus, removing rain streaks from a video is an important issue and has attracted much attention in computer vision.

Since first raised by Garg and Nayar [12] in 2004, many methods have been proposed for this task and attained good performance under different rain circumstances. Many of these methods implement the task by carefully formulating certain physical characteristics of rain steaks, *e.g.* photometric appearance [13], geometrical features [30], chromatic consistency [25], spatio-temporal configurations [34], local structure correlations [7], and design certain techniques for quantitatively formulating these prior rain knowledge to facilitate a proper separation of rain streaks from the video background [20]. Some recent methods along this line achieve decent performance, by pre-training a discriminator with some pre-annotated sample pairs, with or without

*Deyu Meng is the corresponding author.

rains, and extracting discriminative features for distinguishing rain parts from no-rain ones [22]. Different from current methods which mainly treat rain streaks as deterministic knowledge, Wei *et al*. [38] formulated rain streaks as a stochastic distribution, *i.e*., patch-based mixture of Gaussians, and achieved satisfactory performance for the task in a purely unsupervised manner.

However, some insightful characteristics possessed by rain streaks in a video still have not been fully explored and employed in the previous methods. On one hand, the rain streaks in a video are always with apparent local patterns, which repetitively exist while be sparsely scattered over different locations of the video. This can be evidently seen in Fig. 1(a), which shows that the rains contained in a video can be separated into several layers, each containing many local patterns with similar direction, thickness, and shapes. On the other hand, since rain streaks are captured from different distances by the surveillance camera, they are always depicted as multiscale forms. This phenomenon can also be obviously seen in Fig. 1(b), which depicts that multiple scales of rain shapes in different layers, like thick rain chunks, thin rain lines, and fine-grained rain drops. The main aim of this study is to fully explore and characterize such intrinsic prior structures possessed by rain streaks to further enhance the rain removal capability of current state-of-the-art approaches.

To this aim, we design a new rain streak removal model by fully considering the aforementioned prior structures possessed by rain streaks. Specifically, we use multiple convolutional filters imposed on sparse feature maps (*i.e*., convolutional sparse coding) to deliver repetitive shape characteristic of rain streaks, and further use multiscale filters to represent different scales of rain streaks. The mechanism of such modeling way can be easily understood by seeing the illustration in Fig. 1(b). Such a new encoding manner makes the proposed method capable of more faithfully reflecting the intrinsic structures of rain streaks. Thus the new method is expected to more properly extract rain streaks from videos while finely recover the structure of the video without rains.

Note that different from current methods which represents the rain streaks as either deterministic (*e.g*., spatial-temporal smoothness [20]) or stochastic (e.g., patch-based MoG [38]) knowledge, the proposed method integratively consider both types of knowledge under a unified framework. The multiscale convolutional filters deliver deterministic local structures of rain streak, while the feature map with sparsity conveys stochastic information of rain streak distribution. The proposed regime is thus expected to attain a fine compromise between two types of rain modeling manners and make the capabilities of both manners compensate between each other to make the method available in general rain scenarios.

In summary, this paper makes the following three-fold contributions:

1. Two intrinsic characteristics possessed by rain streaks in a video are fully explored and utilized in this work, including the multiscale shapes and the local repetitive patterns sparsely scattered over the video.
2. A concise multiscale convolutional sparse coding (MS-CSC) model is designed to faithfully represent the two explored prior configurations of rain streaks.
3. An alternative optimization algorithm is readily designed to solve the proposed model, where all the parameters involved in the model can be efficiently solved in an iterative manner. Experiments implemented on a series of synthetic and real videos containing rain streaks verify the superiority of the proposed method, both visually and quantitatively.

The paper is organized as follows, Section 2 introduces related works for rain streak removal techniques on videos, as well as on images. Section 3 presents the MS-CSC model and the iterative algorithm for solving the model. Experiments are shown in Section 4, and the paper is briefly concluded in Section 5.

2. Related work

Numerous methods have been proposed to improve the visibility of images/videos captured with rain streak interference. In the following, we present a short review for the developments on this task.

2.1. Video rain removal methods

Garg and Nayar [12] first studied the photometric appearance of rain drops and developed a comprehensive rain detection method for videos, which utilizes a linear space-time correlation model to detect the dynamics of raindrops and a physics-based motion blur model to explains the photometry of rain. Against camera-taken rainy images/videos, Garg and Nayar [13][14] further proposed a method to reduce the effects of rain before camera shots by adjusting the camera parameters such as field depth and exposure time.

In the past years, more physical intrinsic properties of rain streaks have been explored and formulated in algorithm designing. For example, Zhang *et al*. [48] incorporated both chromatic and temporal properties to utilize K-means clustering for distinguishing background and rain streaks from videos. To enhance the robustness of rain removal, Barnum *et al*. [2] employed the regular visual effects of rain in global frequency information to approximate rain streaks as a motion-blurred Gaussian. Later, Santhaseelan *et al*. [31] used local phase congruency to detect rain and applied chromatic constrains to excluding false candidates. Chen *et al*. [7] proposed a spatio-temporal correlation among local patches with rain streaks and used low-rank term to help extract rain streaks from a video.

To improve the capability of rain removal, Kim *et al*. [22] proposed a method based on temporal correlation and low-rank matrix completion. The method needs to use some extra supervised knowledge (images/videos with/without rain streaks) to help training a rain classifier. Recently, Jiang *et al*. [20] proposed a tensor-based video rain streak removal approach by considering the sparsity of rain streaks, smoothness along the raindrops and the rain-perpendicular direction, and global and local correlation along time direction. Meanwhile, to deal with heavy rain in dynamic scenes, Ren *et al*. [30] divided rain into two categories: sparse ones and dense ones, which slightly relieves the matter. Wei *et al*. [38] first encodes rain streaks as a patch-based mixture of Gaussians and proposed 3DTV to constrain moving objects. Such stochastic manner for encoding rain streaks could make the method deliver a wider range of rain information.

All of the current methods have not fully made use of the two intrinsic characteristics, *i.e*., repetitive local patterns scattered over different positions of the video and multiscale shapes. While as shown in Fig. 1, such rain structures are evident and intuitively exist for general rain streaks. We thus expect to enhance the performance of state-of-the-arts along this line by exploring these two useful rain properties.

2.2. Single image rain removal methods

For comprehensiveness, we also briefly review the rain streak removal methods on a single image. Kang *et al*. [21] firstly proposed a method formulating rain removal as an image decomposition problem based on morphological component analysis. They achieved rain component from the high frequency part of an image by using dictionary learning and sparse coding. Luo *et al*. [26] also relied on discriminative sparse codes, but built upon a nonlinear screen blend model to remove rain in a single image. In 2016, Li *et al*. [24] utilized patch-based GMM priors to distinguish and remove rains from background in a single image, which needs to pre-train a GMM with a set of pre-collected natural images without rain streaks. Different from previous methods, Zhang *et al*. [46] introduced a new refined loss function into GAN framework and proposed a derain network called Image De-raining Conditional General Adversarial Network (ID-CGAN).

The state-of-the-art rain removal strategies are presented very recently by Fu *et al*. [11] and Yang *et al*. [41], respectively. Fu *et al*. [11] firstly developed a deep CNN (called DerainNet) model to extract discriminative features of rains in high frequency layer of an image. Yang *et al*. [41] designed a multi-task deep learning architecture that learns the binary rain streak map, the appearance of rain streaks, and the clean background. Both of the networks need to pre-collect a set of labeled images (with/without rain streaks) to learn its network parameters. Recently, Gu *et al*. [17] jointly analyzed sparse representation and synthesis sparse representation to encode background scene and rain streaks on an image. Zhang *et al*. [45] learned a set of generic sparsity-based and low-rank representation-based convolutional filters for efficiently representing background and rain streaks on an image, respectively.

This study puts emphasis on the rain streak removal issue in video. Though these image-based methods can deal with rain removal in a video via a frame-by-frame manner, the extra utilization of temporal information always makes the video-based methods work better than image-based ones.

2.3. Multiscale approach

For decades, the multiscale strategy has been applied to almost every branch of computer vision, especially in image and video processing. In image segmentation, Baatz *et al*. [1] firstly used a general segmentation algorithm based on homogeneity definitions to free adaptable to the scale of interest. In image quality assessment, Wang *et al*. [37] proposed a multiscale structural similarity method and developed an image synthesis method to calibrate the parameters that define the relative importance of different scales. To improve the invariance of CNN activations, Gong *et al*. [15] firstly presented a simple but effective scheme called multiscale orderless pooling. For dense prediction, Yu *et al*. [43] developed a new convolutional network module using dilated convolutions to systematically aggregate multi-scale contextual information without losing resolution.

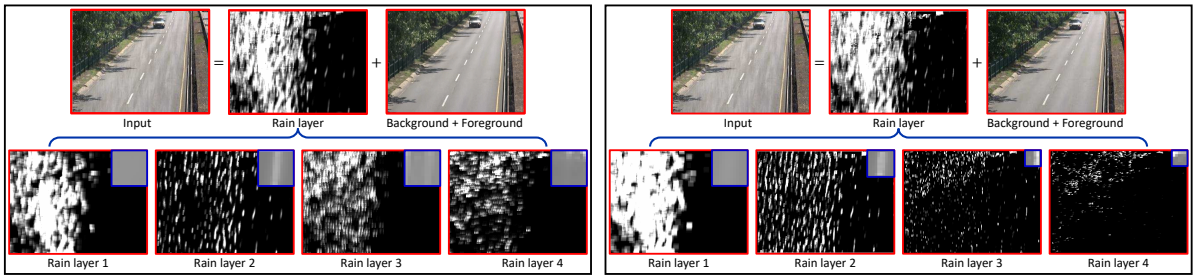
3. MS-CSC Model for Video Rain Removal

An input video is represented by $\mathcal{X} \in R^{h\times w\times n}$, where h, w, and n represent the height, width, and frame number of the video, respectively. As analyzed in the introduction, a video can be decomposed as $\mathcal{X} = \mathcal{B} + \mathcal{R} + \mathcal{F}$, where $\mathcal{B}, \mathcal{R}$, and $\mathcal{F} \in R^{h\times w\times n}$ represent the background scene, rain layer, and moving objects of the video, respectively. In this paper, we use $\mathcal{X}, X$, x, and x to denote tensor, matrix, vector, and scalar, respectively.

3.1. Problem formulation

Modeling rain layers by MS-CSC. Convolutional sparse coding [44], which has been used to emphasize local texture patterns in an image, can be properly utilized to describe the local repetitive patterns of rain streaks. Traditional CSC [17, 45, 18] usually sets filters with same size. However, as we have analyzed in Introduction, rain streaks in a video are generally with multiscale shapes since they are pictured from different distances by cameras. Thus, the MS-CSC model is naturally constructed to depict intricate structures of rain streaks as follows:

$$\mathcal{R} = \sum_{k=1}^{K}\sum_{s=1}^{n_k} D_{ks} \otimes \mathcal{M}_{ks}. \tag{1}$$



(a) CSC model (b) MS-CSC model

Figure 2. Upper: decomposition of a video into a rain layer and that without rains. Lower: different scales of separated rain layers as well as the corresponding filters. (a) The results obtained by CSC model with single-scale filters. (b) The results obtained by MS-CSC model with multi-scale filters.

where $\mathcal{M} = \{\mathcal{M}_{ks}\}_{k,s=1}^{K,n_k} \subset R^{h\times w\times n}$ is a set of feature maps that approximate the rain streak positions, and $D = \{D_{ks}\}_{k,s=1}^{K,n_k} \subset R^{p_k\times p_k}$ denotes the filters representing the repetitive local patterns of rain streaks. K and n_k denote the numbers of entire filters and filters at the k-th scale, respectively. Considering the sparsity of feature maps, this work employs L_1-penalty [28] to regularize the feature maps $\mathcal{M}$. We expect that such a reconstructed-$\mathcal{R}$ can finely extract the rain streaks from the input video. The mechanism of the proposed MS-CSC can be easily understood by Fig. 2. It is observed that the CSC model with single-scale filters fails to decompose the rain streaks layers in physical-meaning-interpretable manner. By contrast, the proposed MS-CSC model can reasonably divide rain streaks into multiple scales and structures, where each layer can be easily explained and finely comply with the instinctive rain separation by human vision system.

Modeling background with low-rank term. For a video captured by surveillance cameras, the background scene keeps steady over the frames except from the variation of illumination and interference of moving objects. Therefore, the similar background layer can be formulated as recovering a low-dimensional subspace [27] [49] [50][42][5][6]. The standard approaches to subspace learning is the following low-rank matrix factorization (LRMF):

$$\mathcal{B} = \text{Fold}(UV^T), \tag{2}$$

where $U \in R^{d\times r}, V \in R^{n\times r}$, $d = hw$, $r < \min(d, n)$, and the operation of 'Fold' refers to fold up each column of a matrix into the corresponding frame matrix of a tensor.

Modeling moving objects with Markov random field. Moving objects in a rain scene are difficult to handle. To a certain extent, the exact locations of moving objects can avoid deformations and artifacts. Therefore, inspired by Wei. *et al*. [38], this work explicitly detects the moving objects with Markov random fields (MRF). Let $\mathcal{H} \in R^{h\times w\times n}$ be a binary tensor denoting the moving object support:

$$\mathcal{H}_{ijn} = \begin{cases} 1, & \text{location } ijn \text{ is moving objects,} \\ 0, & \text{location } ijn \text{ is background.} \end{cases} \tag{3}$$

Let $\mathcal{H}^\perp$ is complementary with $\mathcal{H}$ satisfied $\mathcal{H} + \mathcal{H}^\perp = 1$. Therefore, moving objects part of the video satisfies the following equation:

$$\mathcal{H} \circ \mathcal{X} = \mathcal{H} \circ (\mathcal{F} + \mathcal{R}). \tag{4}$$

Moving objects layer $\mathcal{F}$, relative to rain streak, is smooth, so this work imposes total variation (TV) penalty to regularize it. In a similar way, background part of the video can be expressed as:

$$\mathcal{H}^\perp \circ \mathcal{X} = \mathcal{H}^\perp \circ (\mathcal{B} + \mathcal{R}). \tag{5}$$

Considering the sparse feature and continuous shapes along both space and time of moving object, this work imposes L_1-penalty [28] and weighted 3-dimensional total variation (3DTV) penalty to regularize the moving objects support $\mathcal{H}$.

By integrating the aforementioned three models, the proposed MS-CSC model with parameters $\Theta = \{D, \mathcal{M}, \mathcal{H}, F, U, V, \mathcal{R}\}$ can be constructed as follows:

$$\begin{aligned} \min_{\Theta} \mathcal{L}(\Theta) =& \| \mathcal{H}^\perp \circ (\mathcal{X} - \text{Fold}(UV^T) - \mathcal{R}) \|_F^2 \\ &+ \| \mathcal{H} \circ (\mathcal{X} - \mathcal{F} - \mathcal{R}) \|_F^2 + \lambda \| \mathcal{F} \|_{TV} \\ &+ \alpha \| \mathcal{H} \|_{3DTV} + \beta \| \mathcal{H} \|_1 + b \sum_{k=1}^{K} \sum_{s=1}^{n_k} \| \mathcal{M}_{ks} \|_1 \\ s.t. \quad & \mathcal{R} = \sum_{k=1}^{K} \sum_{s=1}^{n_k} D_{ks} \otimes \mathcal{M}_{ks}, \quad \| D_{ks} \|_F^2 \leq 1. \end{aligned} \tag{6}$$

3.2. Alternative optimization algorithm

Due to the non-convexity of the objective function, the proposed model is difficult to obtain the solution in one step. Hence, we adopt an alternating search algorithm to iteratively optimize each variable involved in the energy minimization over Θ. Its corresponding augmented Lagrangian function can be written as follows:

$$\begin{aligned} \mathcal{L}_\rho(\Theta, \mathcal{T}) =& \| \mathcal{H}^\perp \circ (\mathcal{X} - \text{Fold}(UV^T) - \mathcal{R}) \|_F^2 \\ &+ \| \mathcal{H} \circ (\mathcal{X} - \mathcal{F} - \mathcal{R}) \|_F^2 + \lambda \| \mathcal{F} \|_{TV} \\ &+ \alpha \| \mathcal{H} \|_{3DTV} + \beta \| \mathcal{H} \|_1 + b \sum_{k=1}^{K} \sum_{s=1}^{n_k} \| \mathcal{M}_{ks} \|_1 \\ &+ \frac{\rho}{2} \| \sum_{k=1}^{K} \sum_{s=1}^{n_k} D_{ks} \otimes \mathcal{M}_{ks} - \mathcal{R} + \mathcal{T} \|_F^2, \end{aligned} \tag{7}$$

where $\mathcal{T}$ and ρ are the Lagrange variable and penalty parameter, respectively.

Updating $\mathcal{H}$: The subproblem with respect to $\mathcal{H}$ is

$$\min_{\mathcal{H}} \| \mathcal{H}^{\perp} \circ (\mathcal{X} - \text{Fold}(UV^T) - \mathcal{R}) \|_F^2 + \| \mathcal{H} \circ (\mathcal{X} - \mathcal{F} - \mathcal{R}) \|_F^2 + \alpha \| \mathcal{H} \|_{3DTV} + \beta \| \mathcal{H} \|_1 . \quad (8)$$

This is a standard energy minimization problem of MRF, which can be readily solved by graph cut optimization algorithm [3][23].

Updating $\mathcal{F}$: The subproblem with respect to $\mathcal{F}$ is

$$\min_{\mathcal{F}} \| \mathcal{H} \circ (\mathcal{X} - \mathcal{F} - \mathcal{R}) \|_F^2 + \lambda \| \mathcal{F} \|_{TV}, \quad (9)$$

which can be easily solved by the TV regularization algorithm [35][40].

Updating U, V: The components of Eq.(7) related to U and V can be rewritten as a matrix form:

$$\min_{U,V} \| H^{\perp} \circ (X - UV^T - R) \|_F^2, \quad (10)$$

where X and R denote the unfolding matrix forms of $\mathcal{X}$ and $\mathcal{R}$, respectively. Each column of $X = [\mathrm{x}_1, \cdots, \mathrm{x}_n] \in R^{d \times n}$ represents the corresponding frame. The subproblem Eq.(10) is exactly equivalent to the weighted L_2 LRMF problem and can use any off-the-shelf algorithms to update U and V, such as the Alternated Least Squares (ALS) [9], WLRA [33] and DN [4]. We adopted the WLRA method in experments due to its simplicity of implementation and good performance.

Updating $\mathcal{M}$: Fixing $\mathcal{R}$ and the filters D, we solve the following subproblem to obtain $\mathcal{M}$:

$$\min_{\mathcal{M}_{ks}} \frac{1}{2} \| \sum_{k=1}^{K} \sum_{s=1}^{n_k} D_{ks} \otimes \mathcal{M}_{ks} - \mathcal{R} + \mathcal{T} \|_F^2 + \frac{b}{\rho} \sum_{k=1}^{K} \sum_{s=1}^{n_k} \| \mathcal{M}_{ks} \|_1 . \quad (11)$$

It is a standard CSC problem and can be readily solved by [39]. The algorithm adopts the ADMM scheme and exploits the FFT to improve computation efficiency.

Updating D: The subproblem with respect to D is:

$$\min_{D_{ks}} \frac{1}{2} \| \sum_{k=1}^{K} \sum_{s=1}^{n_k} D_{ks} \otimes \mathcal{M}_{ks} - \mathcal{R} + \mathcal{T} \|_F^2, \text{ s.t. } \| D_{ks} \|_F^2 \leq 1. \quad (12)$$

To update the filters dictionary, let the linear operator M_{ks} satisfy $M_{ks}\mathrm{d}_{ks} = D_{ks} \otimes \mathcal{M}_{ks}$, where $\mathrm{d}_{ks} = \text{vec}(D_{ks})$. The objective function can be rewritten as follows:

$$\min_{\mathrm{d}} \frac{1}{2} \| M\mathrm{d} - \mathrm{r} + \mathrm{t} \|^2, \quad (13)$$

where $M = [M_{11}, \cdots, M_{1n_1}, \cdots, M_{K1}, \cdots, M_{Kn_K}], \mathrm{d} = [\mathrm{d}_{11}^T, \cdots, \mathrm{d}_{1n_1}^T, \cdots, \mathrm{d}_{K1}^T, \cdots, \mathrm{d}_{Kn_K}^T]^T, \mathrm{r} - \mathrm{t} = \text{vec}(\mathcal{R} - \mathcal{T})$ are the block matrices/vectors. We utilize a proximal gradient descent method to solve Eq.(13):

$$\begin{cases} \mathrm{d}^{t+0.5} & = \mathrm{d}^t - \tau M^T (M\mathrm{d} - \mathrm{r} + \mathrm{t}) \\ \mathrm{d}^{t+1} & = \text{Prox}_{\|\cdot\| \leq 1}(\mathrm{d}^{t+0.5}). \end{cases} \quad (14)$$

In (14), τ is the step length of the gradient descent step, and $\text{Prox}_{\|\cdot\| \leq 1}(*)$ is the L_2-ball proximal operator, which makes each filter satisfy the constraint $\| D_{kt} \|_F^2 \leq 1$.

Updating $\mathcal{R}$: The subproblem with respect to $\mathcal{R}$ is

$$\min_{\mathcal{R}} \| \mathcal{H}^{\perp} \circ (\mathcal{X} - \text{Fold}(UV^T) - \mathcal{R}) \|_F^2 + \| \mathcal{H} \circ (\mathcal{X} - \mathcal{F} - \mathcal{R}) \|_F^2 + \frac{\rho}{2} \| \sum_{k=1}^{K} \sum_{t=1}^{n_k} D_{kt} \otimes \mathcal{M}_{kt} - \mathcal{R} + \mathcal{T} \|_F^2 . \quad (15)$$

The closed-form solution is

$$\mathcal{R}_{ijn} = \begin{cases} \{\rho \mathcal{Q}_{ijn} + 2(\mathcal{X} - \mathcal{F})_{ijn}\}/(\rho + 2), & \forall ijn \in \Omega; \\ \{\rho \mathcal{Q}_{ijn} + 2(\mathcal{X} - \text{Fold}(UV^T))_{ijn}\}/(\rho + 2), & \forall ijn \in \Omega^{\perp}; \end{cases} \quad (16)$$

where we set $\mathcal{Q} = \sum_{k,s} D_{ks} \otimes \mathcal{M}_{ks} + \mathcal{T}$ for simple expression, and $\Omega = \{(i, j, n) | \mathcal{H}_{ijn} = 1\}$ represents all background pixels.

Updating $\mathcal{T}$: Under the general ADMM settings, $\mathcal{T}$ can be updated as follows:

$$\mathcal{T}^{t+1} = \mathcal{T}^t + \sum_{k,s} D_{ks} \otimes \mathcal{M}_{ks} - \mathcal{R}. \quad (17)$$

Algorithm 1 MS-CSC Model

Input: video $\mathcal{X} \in \mathbb{R}^{h \times w \times n}$, subspace rank: r, different scales of filters: $p = [p_1, \cdots, p_K]$.

Initialization: Initialize $U, V, \mathcal{D}$; support $\mathcal{H} = 0$.

1: **while** not converge **do**
2: Update $\mathcal{H}, \mathcal{F}$ by Eq.(8), (9), respectively.
3: Update U, V by Eq.(10).
4: Update $\mathcal{M}, \mathcal{D}$ by Eq.(11), (14), respectively.
5: Update $\mathcal{R}, \mathcal{T}$ by Eq.(16), (17), respectively.
6: **end while**
7: Obtain background $\mathcal{B} = \text{Fold}(U^T V)$.

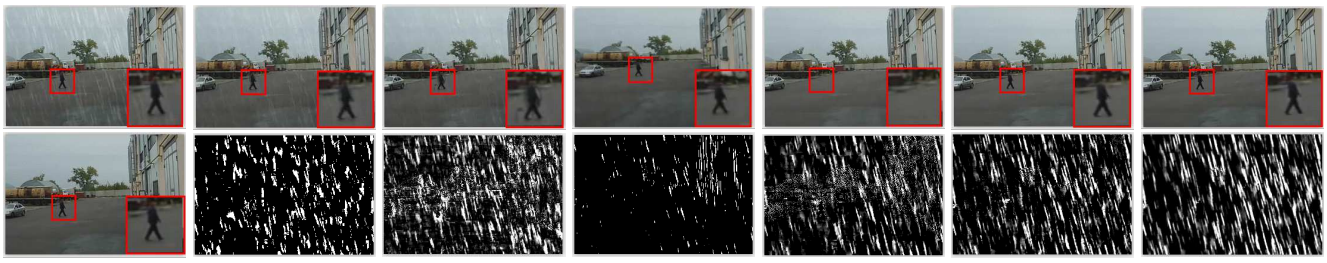
Output: derained video = $\mathcal{H}^{\perp} \circ \mathcal{B} + \mathcal{H} \circ \mathcal{F}$.

4. Experimental results

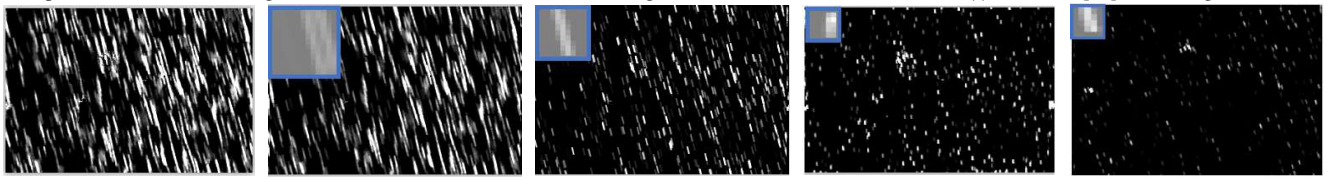
To evaluate the performance of the proposed algorithm, synthetic and real videos with various rain conditions are employed, including the wind direction, moving objects, etc. In our experiments, we utilize six videos including three ones captured by real world monitoring system and three synthetic ones. We set the number and the size of filters in our method as 3 and 13, 9, 5 in the videos with relatively light rain (Fig. 5 and 7), respectively. For the rest of videos with heavy rain, we set the number and the size of filters as 4 and 13, 9, 5, 5, respectively. More comprehensive performance comparisons for all competing methods are provided as demo videos in supplementary material.[1] All experiments were implemented on a PC with i7/CPU, 32G/RAM and MATLAB 2017 compiler. [2].

[1]The results of our method as well as the comparative methods on the entire videos are provided in the supplementary material.

[2]The code of our method is released in the website: http://gr.xjtu.edu.cn/web/dymeng/2

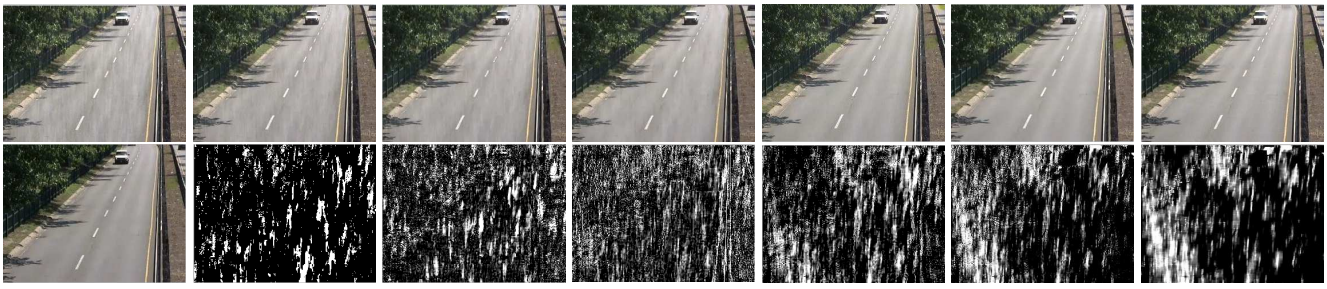


(a) Input/Groundtruth (b) Garg *et al*. [14] (c) Kim *et al*. [22] (d) Jiang *et al*. [20] (e) Ren *et al*. [30] (f) Wei *et al*. [38] (g) Ours



(h) Whole rain layer (i) Rain layer 1 (j) Rain layer 2 (k) Rain layer 31 (l) Rain layer 32

Figure 3. The first two rows: rain removal results and corresponding rain layers extracted by different methods on a video with usual rain. The bottom row: multiscale filters and corresponding rain structures extracted form the MS-CSC method.



(a) Input/Groundtruth (b) Garg *et al*. [14] (c) Kim *et al*. [34] (d) Jiang *et al*. [22] (e) Ren *et al*. [30] (f) Wei *et al*. [38] (g) Ours

Figure 4. Rain streak removal performance of different methods on a rainy video with heavy rain.

Table 1. Performance comparison of all competing methods on synthetic rain videos in items of PSNR, VIF, FSIM, UQI and SSIM.

Dataset	Fig. 3					Fig. 4					Fig. 5				
Metrics	PSNR	VIF	FSIM	UQI	SSIM	PSNR	VIF	FSIM	UQI	SSIM	PSNR	VIF	FSIM	UQI	SSIM
Input	28.22	0.637	0.935	0.9938	0.927	23.82	0.766	0.970	0.9404	0.929	36.03	0.910	0.979	0.9986	0.974
Garg [14]	29.83	0.661	0.955	0.9957	0.946	24.15	0.611	0.960	0.9482	0.911	30.78	0.672	0.974	0.9912	0.955
Kim [22]	30.44	0.602	0.958	0.9971	0.952	22.39	0.526	0.932	0.9462	0.886	32.01	0.650	0.970	0.9961	0.955
Jiang[20]	31.93	0.745	0.971	0.9977	0.974	24.32	0.713	0.966	0.9523	0.938	37.61	0.876	0.991	0.9995	0.992
Ren [30]	28.26	0.685	0.970	0.9932	0.962	23.52	0.681	0.966	0.9408	0.927	30.17	0.640	0.961	0.9986	0.938
Wei [38]	29.76	0.830	**0.992**	0.9943	0.988	24.47	0.779	**0.980**	0.9454	0.951	37.36	0.805	0.988	**0.9999**	0.982
Ours	**33.89**	**0.865**	**0.992**	**0.9980**	**0.992**	**25.37**	**0.790**	**0.980**	**0.9530**	**0.957**	**42.92**	**0.943**	**0.996**	**0.9999**	**0.994**

4.1. Synthetic rain streak removal experiments

In this section, we show experiments on videos added with various types of synthetic rain streaks. Three videos from CDNET database [16][3] largely vary in moving objects and background scenes. We add different types of rain streaks taken by photographers under black background on these videos, varing from tiny drizzling to heavy rain storm and vertical rain to slash line. Since the ground truth videos without rain are available, we can compare all competing methods both in quantity and in visualization. To validate the accuracy of the proposed method, we compare our method with representative state-of-the-art methods, including Garg *et al*. [14][4], Kim *et al*. [22][5], Jiang *et al*. [20][6], Ren *et al*. [30][7], and Wei *et al*. [38][8].

Fig. 3 illustrates the performance of all compared methods on videos with usual rain. The rain removal results displayed in the first row indicate that Garg *et al*.'s, Kim *et al*.'s, and Jiang *et al*.'s methods do not work well in rain streak detection, and Ren *et al*.'s method improperly removes moving objects as rain streaks. The middle row shows that all compared methods mix different degrees of background information into rain layer. In comparison, the proposed MS-CSC method can not only more comprehensively remove rain streaks in the video, but also best keep the shape and texture details. An interesting observation is that the multiscale rain structures extracted by the proposed MS-CSC method finely accords with human visual system, including long rain blocks

[3]http://www.changedetection.net

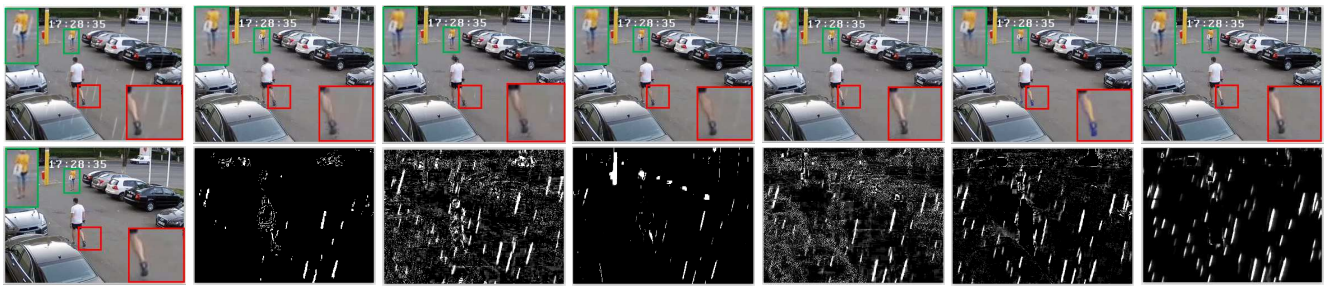
[4]http://www.cs.columbia.edu/CAVE/projects/camera rain/

[5]http://mcl.korea.ac.kr/deraining/

[6]the authors has directly provided us the code

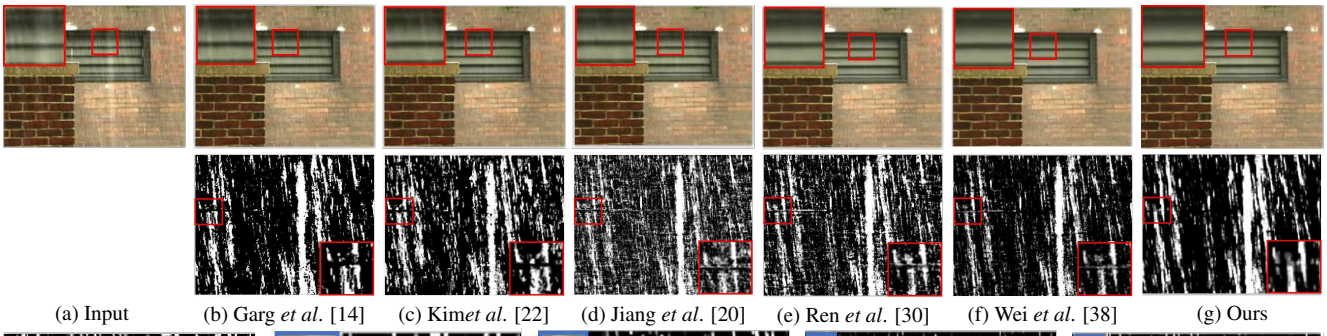
[7]http://vision.sia.cn/our%20team/RenWeihong-homepage/vision-renweihong%28English%29.html

[8]http://gr.xjtu.edu.cn/web/dymeng/2

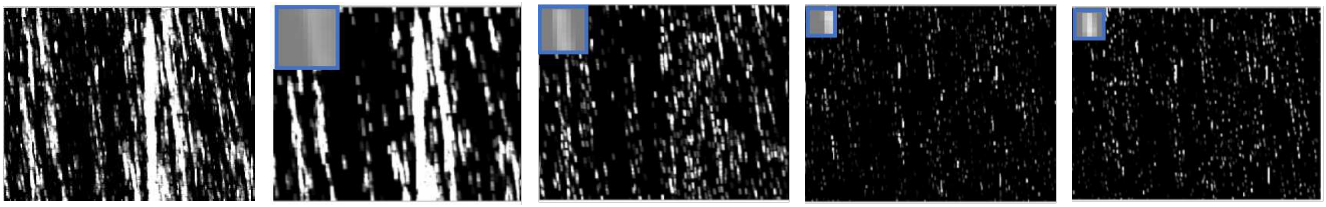


(a) Input/Groundtruth (b) Garg *et al*. [14] (c) Kim *et al*. [22] (d) Jiang *et al*. [20] (e) Ren *et al*. [30] (f) Wei *et al*. [38] (g) Ours

Figure 5. Rain removal results and corresponding rain layers extracted by different methods on a video with light rain.



(a) Input (b) Garg *et al*. [14] (c) Kim*et al*. [22] (d) Jiang *et al*. [20] (e) Ren *et al*. [30] (f) Wei *et al*. [38] (g) Ours



(l) Whole rain layer (h) Rain layer 1 (i) Rain layer 2 (j) Rain layer 31 (k) Rain layer 32

Figure 6. The first two rows: rain removal results and corresponding rain layer extracted by different methods on a video with heavy rain. The bottom row: multiscale filters and corresponding diverse rain structures learned form our method.

(rain layer 1), medium thin rain lines (rain layer 2), light rain lines (rain layer 31), and scattered small light rain grains (rain layer 32), shown in the bottom row of the figure. Further more, multiscale filters and corresponding rain layers evidently verify the discovered two intrinsic characteristics possessed by rain streaks: multiscale structures and repetitive local patterns sparsely scattered over different positions of the video.

Fig. 4 shows a video with heavy rain. The rain removal results of Garg *et al*.'s, Kim *et al*.'s, and Jiang *et al*.'s methods indicate that heavy rain reduces their capability of rain detection. Besides, Ren *et al*.'s method fails to perfectly deal with the moving objets, which inclines to degrade the performance of subsequent video processing. Wei *et al*.'s method can work well on this video, but it tends to view rain streaks as aggregation of noises rather than natural streamline. Comparatively, the proposed method still attains promising visual effect in both rain removal and detail preservation from the video with heavy rain.

Fig. 5 shows a light rain scene with a man passing by the surveillance camera. Similar to above experiments, all competing methods fails to remove relatively light rain lines or destroy the edge information of moving objects, while the proposed MS-CSC method still clearly removes all rain streaks on the video without extra information left behind in the separated rain layer.

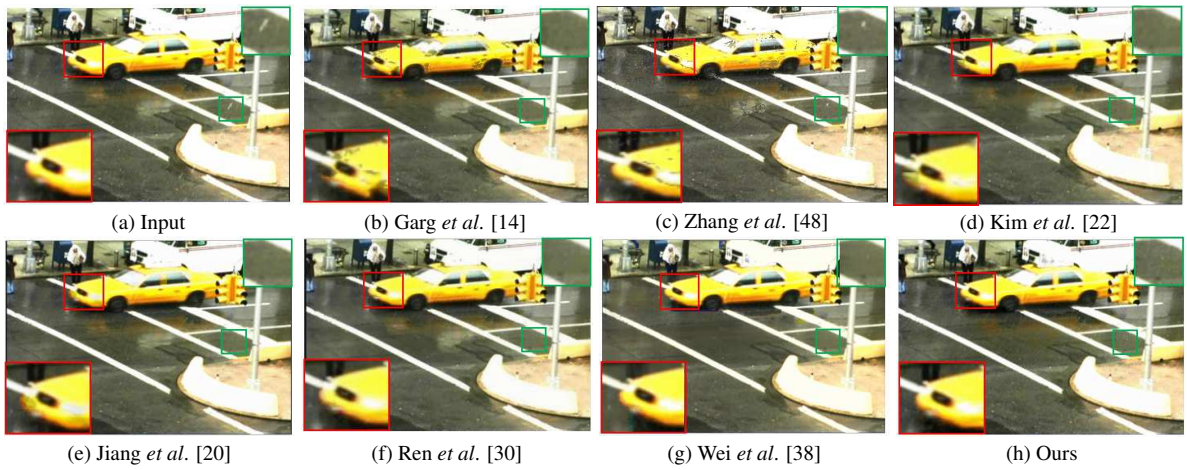
Quantitative comparisons are listed in Table 1. Here we use five image quality assessment (IQA) metrics, PSNR, VIF [32], FSIM [47], UQI [36] and SSIM [37] to verify the performance of synthetic videos. From the table, it is seen that our method attains evidently better results in terms of all those measures in all cases. Since these measures mainly focus on image structure and are more consistent with humans visual perception, the superiority of the proposed method can thus be substantiated.

4.2. Rain streak removal experiments on real videos

In this section, we show the results of proposed MS-CSC method on real world rain scenarios.

Fig. 6 shows the results of different methods on the real world video with heavy rains. The results displayed in the first two rows show that all compared methods still contain a few rain streaks in the results and involve edges information from background in the extracted rain layer. By contrast, our method still clearly removes all rain streaks on the video without extra information left behind in the rain layer.

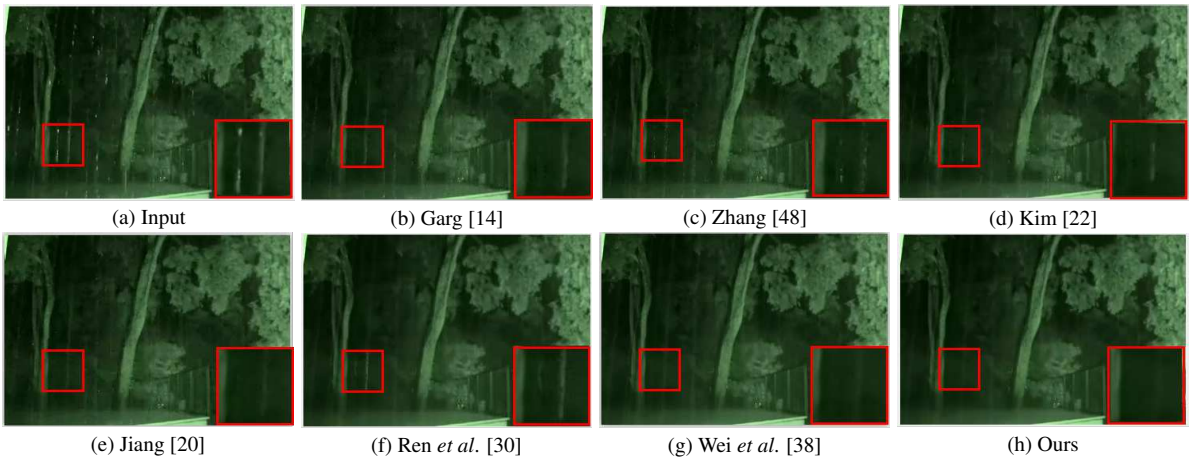
In Fig. 7, we present the rain removal results on the video with complex moving objects, consisting of walking pedestrian and moving vehicles. The results of Zhang [48] are obtained by code written by ourself. The proposed MS-CSC method works consistently well in this relative complex task. Comparatively,



(a) Input (b) Garg *et al*. [14] (c) Zhang *et al*. [48] (d) Kim *et al*. [22]

(e) Jiang *et al*. [20] (f) Ren *et al*. [30] (g) Wei *et al*. [38] (h) Ours

Figure 7. Rain streak removal performance of different methods on a real rainy video with complex moving objects.



(a) Input (b) Garg [14] (c) Zhang [48] (d) Kim [22]

(e) Jiang [20] (f) Ren *et al*. [30] (g) Wei *et al*. [38] (h) Ours

Figure 8. Rain streak removal performance of different methods on a real video captured by surveillance equipments at night.

most other competing methods either miss detecting a few rain streaks. Fig. 8 further presents the rain removal results on a real world video captured by surveillance equipments at night, where code of Zhang *et al*. [48] is written by ourself. Similar to above experiments, the proposed MS-CSC method is still capable of attaining superior performance on the challenging video. More supplementary experiments are showed in here[9].

We list the running times of all competing methods in table 2, which shows that our method is slower than two methods while faster than other competing ones. Considering the evident superiority of the proposed MS-CSC method both in quantity and visualization, it should be rational to say that it is relatively efficient.

Table 2. Processing time comparison of all competing methods on three videos, respectively with 150, 100,120 frames.

Dataset	Fig. 3	Fig. 6	fig. 7
Frame Size	210×310	288×368	240×360
Garg [14]	1.9	6.7	2.3
Kim [22]	2020.1	2740.8	1929.1
Jiang [20]	43.4	64.9	53.5
Ren [30]	526.9	540.5	798.8
Wei [38]	441.4	622.3	755.4
Ours	297.4	337.6	222.6

[9]https://sites.google.com/view/cvpr-anonymity

5. Conclusion

In this paper, we extracted two intrinsic characteristics of rain streaks, repetitive local patterns sparsely scattered over different positions of the video and multiscale structures, and formulated these two intrinsic characteristics as multiscale convolutional sparse coding. The MS-CSC model can usually decomposes the rain layer into different levels of rain streaks with physical meanings, like long rain blocks, thin rain lines, scattered light rain grains, and small rain blocks. Together with the priors imposed on moving objects and background scene, the proposed MS-CSC model shows surprisingly good performance on synthetic and real videos with various types of rains. The experiments on real and synthetic video sequences demonstrate that our method performs better than other state-of-the-art methods. For future work, we will attempt to combine online strategies to further improve its efficiency and meet the requirement of real-time rain removal task.

Acknowledgement

This research was supported by the China NSFC projects under contracts 616611 66011, 1169 0011, 6160 3292, 6172 1002, Macau STDF 003/2016/AFJ and 973 Program of China 2013CB329404

References

[1] BaatzM and A. Schape. An optimization approach for high quality multi-scale image segmentation. *Beitrage Zum Agit-symposium*, pages 12–23, 2000.

[2] P. C. Barnum, S. Narasimhan, and T. Kanade. Analysis of rain and snow in frequency space. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 86(2):256, 2010.

[3] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[4] A. Buchanan and A. Fitzgibbon. Damped newton algorithms for matrix factorization with missing data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[5] X. Cao, Q. Zhao, D. Meng, Y. Chen, and Z. Xu. Robust low-rank matrix factorization under general mixture noise distributions. *IEEE Transactions on Image Processing*, 2016.

[6] Y. Chen, X. Cao, Q. Zhao, D. Meng, and Z. Xu. Denoising hyperspectral image with non-i.i.d. noise structure. *IEEE Transactions on Cybernetics*, 2017.

[7] 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.

[8] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *IEEE Computer Society Conference on Computer Vision*, 1(12):886–893, 2005.

[9] F. De la Torre and M. J. Black. A framework for robust subspace learning. *International Journal of Computer Vision.*, 54:117–142, 2003.

[10] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani. Person re-identification by symmetry-driven accumulation of local features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[11] 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.

[12] K. Garg and S. K. Nayar. Detection and removal of rain from videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages I–I. IEEE, 2004.

[13] K. Garg and S. K. Nayar. When does a camera see rain? In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1067–1074 Vol. 2, 2005.

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

[15] Y. Gong, WangL, GuoR, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. *Springer International Publishing*.

[16] N. Goyette, P.-M. Jodoin, F. Porikli, J. Konrad, and P. Ishwar. Changedetection. net: A new change detection benchmark dataset. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 1–8. IEEE, 2012.

[17] S. Gu, D. Meng, W. Zuo, and L. Zhang. Joint convolutional analysis and synthesis sparse representation for single image layer separation. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

[18] S. Gu, W. Zuo, X. Xie, D. Meng, X. Feng, and L. Zhang. Convolutional sparse coding for image super-resolution. In *International Vonference on Computer Vision*, 2015.

[19] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *A model of saliency-based visual attention for rapid scene analysis.*, 20(11):1254–1259, 1998.

[20] T.-X. Jiang, T.-Z. Huang, X.-L. Zhao, L.-J. Deng, and Y. Wang. A novel tensor-based video rain streaks removal approach via utilizing discriminatively intrinsic priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[21] 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, 2012.

[22] 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*, 24(9):2658–70, 2015.

[23] V. Kolmogorov and R. Zabin. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159, 2004.

[24] 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.

[25] P. Liu, J. Xu, J. Liu, and X. Tang. Pixel based temporal analysis using chromatic property for removing rain from videos. *Computer & Information Science*, 2(1), 2009.

[26] 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, 2015.

[27] D. Meng and F. De La Torre. Robust matrix factorization with unknown noise. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1337–1344, 2013.

[28] D. Meng, Q. Zhao, and Z. Xu. Improve robustness of sparse pca by l 1-norm maximization. In *Pattern Recognition*, pages 487–497, 2012.

[29] S. Mukhopadhyay and A. K. Tripathi. Combating bad weather part i: Rain removal from video. *Synthesis Lectures on Image, Video, and Multimedia Processing*, 7(2):1–93, 2014.

[30] W. Ren, J. Tian, Z. Han, A. Chan, and Y. Tang. Video desnowing and deraining based on matrix decomposition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[31] V. Santhaseelan and V. K. Asari. Utilizing local phase information to remove rain from video. *International Journal of Computer Vision*, 112(1):71–89, 2015.

[32] H. R. Sheikh and A. C. Bovik. Image information and visual quality. *IEEE Transactions on Image Processing*, 15(2):430–444, 2006.

[33] N. Srebro and T. Jaakkola. Weighted low-rank approximations. In *Proceedings of International Conference on Machine Learning*, 2003.

[34] A. K. Tripathi and S. Mukhopadhyay. A probabilistic approach for detection and removal of rain from videos. *Iete Journal of Research*, 57(1):82, 2011.

[35] J. Wang, Q. Li, S. Yang, W. Fan, P. Wonka, and J. Ye. A highly scalable parallel algorithm for isotropic total variation models. In *Proceedings of The 31st International Conference on Machine Learning*, pages 235–243, 2014.

[36] Z. Wang and A. C. Bovik. A universal image quality index. *IEEE Signal Processing Letters*, 9(3):81–84, 2002.

[37] 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*, 13(4):600–612, 2004.

[38] W. Wei, L. Yi, Q. Xie, Q. Zhao, D. Meng, and Z. Xu. Should we encode rain streaks in video as deterministic or stochastic? In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

[39] B. Wohlberg. Efficient convolutional sparse coding. In *In IEEE ICASSP*, 2014.

[40] S. Yang, J. Wang, W. Fan, X. Zhang, P. Wonka, and J. Ye. An efficient admm algorithm for multidimensional anisotropic total variation regularization problems. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 641–649. ACM, 2013.

[41] W. Yang, R. T. Tan, J. Feng, J. Liu, Z. Guo, and S. Yan. Deep joint rain detection and removal from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[42] H. Yong, D. Meng, W. Zuo, and L. . Zhang. Robust online matrix factorization for dynamic background subtraction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[43] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *Computer Vision and Pattern Recognition*, 2016.

[44] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus. Deconvolutional networks. In *IEEE Winter Conference on Applications of Computer Vision*.

[45] H. Zhang and V. M. Patel. Convolutional sparse and low-rank coding-based rain streak removal. In *IEEE Winter Conference on Applications of Computer Vision*, 2017.

[46] H. Zhang and V. M. Sindagi, V an Patel. Image de-raining using a conditional generative adversarial network. In *arXiv:1701.05957v3*, pages 111–122, 2017.

[47] L. Zhang, L. Zhang, X. Mou, and D. Zhang. Fsim: A feature similarity index for image quality assessment. *IEEE Transactions on Image Processing*, 20(8):2378–2386, 2011.

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

[49] Q. Zhao, D. Meng, Z. Xu, W. Zuo, and Y. Yan. l1-norm low-rank matrix factorization by variational bayesian-method. *IEEE Transactions on Neural Networks and Learning Systems*, 26(4):829–839, 2015.

[50] Q. Zhao, D. Meng, Z. Xu, W. Zuo, and L. Zhang. Robust principal component analysis with complex noise. In *International Conference on Machine Learning*, pages 55–63, 2014.