

Should We Encode Rain Streaks in Video as Deterministic or Stochastic?

Wei Wei¹, Lixuan Yi¹, Qi Xie¹, Qian Zhao^{1,2}, Deyu Meng^{1,2,*}, Zongben Xu^{1,2}

¹School of Mathematics and Statistics, Xi'an Jiaotong University

²Ministry of Education Key Lab of Intelligent Networks and Network Security, Xi'an Jiaotong University

{weiwewe, yilixuan, xq.liwu}@stu.xjtu.edu.cn, {timmy.zhaoqian, dymeng, zbxu}@mail.xjtu.edu.cn

Abstract

Videos taken in the wild sometimes contain unexpected rain streaks, which brings difficulty in subsequent video processing tasks. Rain streak removal in a video (RSRV) is thus an important issue and has been attracting much attention in computer vision. Different from previous RSRV methods formulating rain streaks as a deterministic message, this work first encodes the rains in a stochastic manner, i.e., a patch-based mixture of Gaussians. Such modification makes the proposed model capable of finely adapting a wider range of rain variations instead of certain types of rain configurations as traditional. By integrating with the spatiotemporal smoothness configuration of moving objects and low-rank structure of background scene, we propose a concise model for RSRV, containing one likelihood term imposed on the rain streak layer and two prior terms on the moving object and background scene layers of the video. Experiments implemented on videos with synthetic and real rains verify the superiority of the proposed method, as compared with the state-of-the-art methods, both visually and quantitatively in various performance metrics.

1. Introduction

Videos captured by outdoor surveillance equipments or cameras sometimes contain rain streaks. Such unexpected corruption may degrade performance of subsequent video processing tasks, like feature detection, stereo correspondence [14], object tracking, segmentation and recognition [25]. Rain streak removal from a video (RSRV) has thus become an important task in recent computer vision research.

Since firstly raised by Garg *et al.* [12], multiple methods have been proposed for this task and attained good rain removing performance in videos with different rain circumstances. Most of these methods need to specify certain physical characteristics of rain streaks, *e.g.*, photometric appearance [13], chromatic consistency [20], spatiotemporal

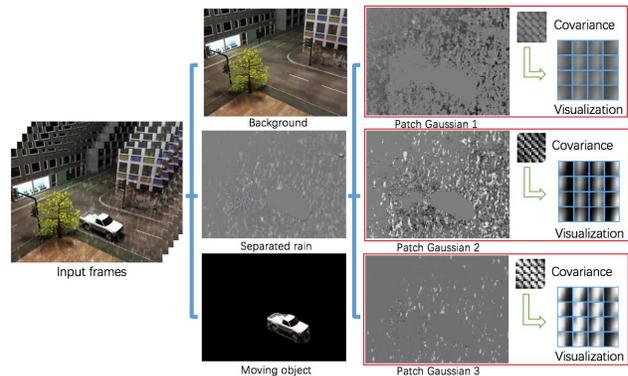


Figure 1. An input video (left) is separated into three layers of background scene, rain streaks and moving object (middle) by the P-MoG method. The rain streak layer contains three 4×4 patch-based Gaussian components (right), whose covariance matrices (16×16) are shown correspondingly in the righthand. Alongside each covariance matrix, we show 16 4×4 patches wrapped by all columns of it. Each patch visualizes the correlation map between one patch position to all others. One can see evident correlations among patch pixels in the lower component, complying with rain drops with block shapes, less while evident oriented correlations in the middle component, corresponding to rain streaks with thinner line shapes, and very weak correlations among patch pixels in the upper component, representing scattered light rain grains and random camera noise.

configurations [29] and local structure correlations [8], and design certain techniques for quantitatively encoding these prior rain knowledge to facilitate a proper separation of rain streaks from the video background. Some latest method along this line achieved state-of-the-art performance, by pre-training a classifier with some auxiliary annotated samples with/without rains and extracting discriminative structures for differentiating rain parts from no-rain ones [17].

However, there are still limitations existing in current RSRV methods. Firstly, most current methods assume rain streaks as deterministic information with specific structures or discriminative features. In practice, however, the rain streaks are always with a large variation across different videos taken under diverse scenarios and circumstances,

*Deyu Meng is the corresponding author.

which makes it hardly be generally represented by specific prior structures manually designed by intuition or fixed detectors trained on subjectively corrected pre-annotated samples. Secondly, while a video with rain streaks can be approximately decomposed into three layers: rain streaks, moving objects and background scene, as depicted in Fig. 1, current methods have not fully taken advantage of the useful prior structures of the latter two layers (*e.g.*, spatiotemporal smoothness for moving objects and low-rankness for background scene) to facilitate a complementary function on rain layer extraction. Thirdly, albeit achieving state-of-the-art performance, the RSRV methods by using discriminative features underlying rain streaks require to pre-collect a set of annotated samples. This on one hand takes extra human labor cost, and on the other hand might possibly bring bias to the obtained rain streak removal classifier, which tends to overfit to certain rain types contained in the pre-trained set while not performing well on more diverse rain shapes in practical videos.

To alleviate these issues, we propose a new RSRV method in this work. Different from current methods assuming rain streaks as deterministic knowledge, the proposed method formulates rain as a stochastic one, *i.e.*, a patch-based mixture of Gaussians (P-MoG) distribution. Albeit simple, this formulation can appropriately represent a general peculiarity of rain streaks. As depicted in Fig. 1, the extracted three Gaussians from the video with rain streaks on 4×4 patches finely comply with rain drops with block shapes (with evident correlations among patch pixels, see right lower panel of Fig. 1), rain streaks with thinner line shapes, and scattered light rain grains (with weak correlations among patch pixels, see right upper panel of Fig. 1). This formulation forms the likelihood/loss term in the presented model. Besides, to fully employ the helpful structures of moving objects and background scene in a video, we encode their priors as 3DTV and low-rankness forms, respectively, which are formulated as the prior/regularization terms in our model. In this manner, the finally designed model is with a surprisingly concise form (Eq. (8)), while can perform better beyond state-of-the-art.

Specifically, the main contributions of this work are:

- Instead of deterministic as traditional, this work firstly assumes rain streaks in a video as stochastic, specifically, distributed as a patch-based mixture of Gaussians. Such easy expression can always finely represent diverse rain streak configurations in a video, and thus is of a good generalization capability.
- This work firstly fully encodes different characteristics of three layers, including rain streaks (one likelihood term), moving objects and background scene (two regularization terms), in a video, and integrates them into one concise model for the RSRV task. Through solv-

ing this model, different layers can be well complemented between each other to simultaneously get a fine output of rain streak extraction and rain-free video recovery, as clearly depicted in Fig. 6 and 7.

- We design an EM algorithm to solve the proposed model. All involved parameters can be readily solved in closed-form or by directly employing off-the-shelf efficient toolkits. Experiments implemented on a series of synthetic and real videos with rain streaks verify the superiority of the proposed method beyond state-of-the-art, without need to use any extra information other than the input video.

The remainder of the paper is organized as follows. Section 2 discusses the related work. Section 3 presents the P-MoG model and related EM algorithm. Section 4 shows the experimental results and finally we make a conclusion in Section 5.

2. Related work

2.1. Methods on rain removal in a video

Garg and Nayar first studied the photometric appearance of rain drops [12] and developed a comprehensive rain detection method for videos with respect to the dynamic motion of rain drops with irradiance constraint. The method assumes linear space-time correlation implying that rain follows straight route in a steady rate. The method is theoretically sound but might yield poor results when rain drops are of different scales and layered due to their different distances to the camera. Against camera-taken rainy images/videos, Garg and Nayar [13, 14] further presented a method to reduce or enhance rain drop before camera shots by altering camera settings such as field depth and exposure time. However, this algorithm runs at the expense of image quality deduction and has trouble handling heavy rain since rain drop sizes may be with large diversity.

Afterwards, more physical intrinsic properties of rain streaks have been explored and formulated in algorithm designing. For example, Zhang *et al.* [36] constructed a rain streak removal method by exploiting temporal and chromatic properties of rain and utilizing k -means clustering to differentiate rain and background, and combined chromatic constraint to exclude moving objects. Later, Liu *et al.* [20] further developed a theory of chromatic property of rain. Barnum *et al.* [1] made use of the fact that regular visual effects of rain streaks become united in the Fourier domain to detect and remove rain. Recently, Santhaseelan *et al.* [26] used phase congruency features to detect rain and applied chromatic constrains to excluding false candidates.

Another difficulty in rain streak detection in a video is observed: the rain and moving objects usually share resemblance in edge information and are hard to be separated.

Against this issue, Brewer and Liu [5] estimated aspect ratio range of rain streaks based on camera settings to distinguish rain streaks. A similar idea was forwarded by Bossu *et al.* [2], which applied size selection to segmenting rain and moving objects. This method employed histogram feature and raised that the orientation of rain streaks can be well represented by Gaussian mixture model. Besides, Chen *et al.* [8] discovered the spatio-temporal correlation among local patches with rain streaks and used low-rank term to encode such structure knowledge to help extract rain streaks from other parts of a video. Moreover, Chen *et al.* [7] raised a method to remove rain in the video with highly dynamic moving objects based on motion segmentation by GMM.

The current state-of-the-art for video rain streak removal can be represented by the method raised by Kim *et al.* [17], which needs to use some extra supervised knowledge (images/videos with/without rain streaks) to help train a rain classifier (by SVM), and then gradually ameliorates this classifier on a coarsely obtained rain map of the input video to extract the final detected rain parts. A similar strategy was utilized in a previous method presented by Tripathi *et al.* [28, 29], which also needs to use some extra information to pre-train a rain detector and then ameliorates it (by Naive Bayes) using discriminative knowledge of rain streaks (spatial and temporal features).

Differing from the current methods, which mainly assume rain streaks as deterministic information with specific photometric appearance, chromatic consistency, spatiotemporal configurations, or discriminative structures, the proposed method formulates rain streaks as stochastic information, distributed as a patch-based mixture of Gaussians. As shown in Fig. 1, this assumption is intuitively rational and can be easily understood. By further imposing complementary spatiotemporal smoothness prior term on moving objects and low-rank prior term on background scenes, our model is with a surprisingly simple and concise form (Eq. (8)), while can evidently outperform the state-of-the-art without needing any extra knowledge for pre-training.

2.2. Methods on rain removal in a single image

For comprehensiveness, we also briefly review the rain streak removal methods in a single image.

Kang *et al.* [16] 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.* [21] also relied on discriminative sparse codes, but built upon a nonlinear screen blend model to remove rain in a single image. Just in 2016, Li *et al.* [19] 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. The state-

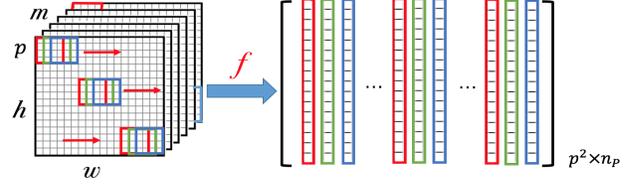


Figure 2. Graphical illustration of the map f as defined in Eq. (2). (left) Input tensor. (right) Output patch matrix.

of-the-art rain removal strategy is presented very recently by Fu *et al.* [10, 11], which developed a deep CNN (called DerainNet) model to extract discriminative features of rains in high frequency layer of an image. Similarly, this method also needs to collect a set of labeled images (with/without rain streaks) to train the CNN parameters.

This study puts emphasis on the rain streak removal issue in video. Note that most of these image-based methods also formulated rain streaks in a deterministic manner, *e.g.*, with sparse representation under a specific pre-specified or adaptively trained dictionary or of discriminative features from those no-rain images, and always needed extra annotated images for pre-training. In this sense, the methodology of this method is also novel.

3. The P-MoG model

3.1. Problem formulation

An input video is represented by a tensor $\mathcal{D} \in \mathbb{R}^{h \times w \times m}$, where h, w, m represent the height, width and frame number of the video, respectively. As aforementioned, the video can be decomposed into three layers of rain streaks, foreground moving objects and background scene, represented by \mathcal{R}, \mathcal{F} and \mathcal{B} (all belonging to $\mathbb{R}^{h \times w \times m}$), respectively. For convenience, we denote italic upper letter of a tensor (denoted by calligraphic upper letter), as its mode-3 unfolding matrix, *e.g.*, $B \in \mathbb{R}^{hw \times m}$ is denoted to be unfolded from $\mathcal{B} \in \mathbb{R}^{h \times w \times m}$ along its 3rd mode.

We then introduce how to model each layer in P-MoG.

Modeling rain streak layer. As analyzed in the introduction, we model this layer with a P-MoG distribution. Here we introduce a map f which is defined as:

$$f : \mathbb{R}^{h \times w \times m} \longrightarrow \mathbb{R}^{p^2 \times n_p}, \quad (1)$$

where p denotes the patch size and n_p is the total patch number of the entire video. The meaning of this map can be easily understood by Fig. 2. Denote $f(\mathcal{R})_n$ ($n = 1, \dots, n_p$) as the n^{th} column of $f(\mathcal{R})$, and then P-MoG assumption means that:

$$f(\mathcal{R})_n \sim \sum_{k=1}^K \pi_k \mathcal{N}(f(\mathcal{R})_n | 0, \Sigma_k), \quad (2)$$

where $\mathcal{N}(\cdot | \mu, \Sigma)$ represents a Gaussian distribution with

mean μ and covariance matrix $\Sigma \in \mathbb{R}^{p^2 \times p^2}$. $\pi_k \geq 0$ is the mixing coefficient with $\sum_{k=1}^K \pi_k = 1$.

Modeling background layer. For a video captured under static camera, background scene generally keeps steady over the frames except from the variation of illumination and interference of moving objects. Therefore, similar as many other background subtraction methods [6, 22, 34, 37, 38], this layer can be rationally encoded as a low-rank prior structure. This can be easily formulated as the following low-rank matrix factorization expression:

$$B = UV^T, \quad (3)$$

where $U \in \mathbb{R}^{d \times r}$, $V \in \mathbb{R}^{n \times r}$, and $r < \min(d, n)$ implying the low-rankness of the reconstruction UV^T .

Modeling moving object layer. Besides background and rain, the foreground moving objects in the dynamic rain video is a non-negligible segment. We extract the moving objects by introducing a binary tensor $\mathcal{H} \in \mathbb{R}^{h \times w \times m}$ denoting moving object support:

$$\mathcal{H}_{ijk} = \begin{cases} 1, & \text{location } ijk \text{ is moving object,} \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

Here we assume rain is attached only in background scenes (the rain stuck on the moving objects will be removed by a post-processing step, described in Section 3.3). Hence for background part, we have:

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

For foreground moving objects part, we have:

$$\mathcal{H} \circ \mathcal{D} = \mathcal{H} \circ \mathcal{F}. \quad (6)$$

Summing up Eq. (5) and (6), we then obtain our decomposition model involving the moving object support as:

$$\mathcal{D} = \mathcal{H}^\perp \circ \mathcal{B} + \mathcal{H} \circ \mathcal{F} + \mathcal{H}^\perp \circ \mathcal{R}. \quad (7)$$

Considering the sparse feature of moving object, we add an l_1 -penalty to regularize moving object support \mathcal{H} [39].



Figure 3. (a) Input frame; (b) Moving object support obtained by 2DTV penalty; (c) Moving object support obtained by weighted 3DTV penalty. Here the weights are chosen to be 5: 5: 1 along video width, height and time. It is easily seen that the support of rain streaks can be finely removed from the support detected from the 3DTV.

Additionally, we have the prior knowledge that foreground moving objects are with continuous shapes along both space and time, and thus we can regularize with weighted 3-dimensional total variation (3DTV) penalty on its support \mathcal{H} . Such regularizer can bring another byproduct to help separating rain streak and moving objects, since the former is always of a small configurations and moves very fast along time, making it with much less temporal continuity. Utilizing weighted 3DTV can thus naturally distinguish the these two layers, as shown in Fig. 3.

P-MoG model: By integrating the three models imposed on rain streak, moving object and background scene layers as aforementioned, we can get the final P-MoG model with parameters $\Theta = \{U, V, \Pi, \Sigma, \mathcal{H}\}$ (Π, Σ denote mixture proportions and Gaussian covariance matrices):

$$\begin{aligned} \min_{\Theta} & - \sum_{n=1}^{n_p} \log \sum_{k=1}^K \pi_k \mathcal{N}(f(\mathcal{H}^\perp \circ \mathcal{R})_n | 0, \Sigma_k) \\ & + \alpha \|\mathcal{H}\|_{3DTV} + \beta \|\mathcal{H}\|_1 \quad (8) \\ \text{s.t.} & \mathcal{H}^\perp \circ \mathcal{R} = \mathcal{H}^\perp \circ (\mathcal{D} - \mathcal{B}), \quad B = UV^T. \end{aligned}$$

3.2. EM algorithm

The EM algorithm [9] can be readily employed to solve the P-MoG model. The algorithm iterates between calculating the responsibility of all patch Gaussian components (E step) and optimizing the parameters in Θ (M step).

E step. Introduce a latent variable z_{nk} , where $z_{nk} \in \{0, 1\}$ and $\sum_{k=1}^K z_{nk} = 1$, representing the assignment of the noise $f(\mathcal{H}^\perp \circ \mathcal{R})_n$ to a specific component of the mixture. The posterior probability of component k given the noise $f(\mathcal{H}^\perp \circ \mathcal{R})_n$ takes the form:

$$\begin{aligned} \gamma_{nk} &= E(z_{nk}) \\ &= \frac{\pi_k \mathcal{N}(f(\mathcal{H}^\perp \circ (\mathcal{D} - \text{fold}(UV^T)))_n | 0, \Sigma_k)}{\sum_k \pi_k \mathcal{N}(f(\mathcal{H}^\perp \circ (\mathcal{D} - \text{fold}(UV^T)))_n | 0, \Sigma_k)}. \quad (9) \end{aligned}$$

M step. This step needs to minimize the following object function with respect to parameters in Θ :

$$\begin{aligned} \min_{\Theta} & \sum_{n=1}^{n_p} \sum_{k=1}^K \gamma_{nk} \left(\frac{1}{2} f(\mathcal{H}^\perp \circ \mathcal{R})_n^T \Sigma_k^{-1} f(\mathcal{H}^\perp \circ \mathcal{R})_n \right. \\ & \left. + \frac{1}{2} \log |\Sigma_k| - \log \pi_k \right) + \alpha \|\mathcal{H}\|_{3DTV} + \beta \|\mathcal{H}\|_1 \quad (10) \\ \text{s.t.} & \mathcal{H}^\perp \circ \mathcal{R} = \mathcal{H}^\perp \circ (\mathcal{D} - \text{fold}(UV^T)). \end{aligned}$$

We can use alternative optimization strategy to iteratively optimize each variable in the model.

Update Π, Σ . The closed-form updating equation for both parameters can be easily deduced as:

$$N_k = \sum_{n=1}^{n_p} \gamma_{nk}, \quad \pi_k = \frac{N_k}{N}, \quad (11)$$

$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{n_p} \gamma_{nk} f(\mathcal{H}^\perp \circ \mathcal{R})_n f(\mathcal{H}^\perp \circ \mathcal{R})_n^T. \quad (12)$$

Update U, V . We apply alternating direction method of multipliers (ADMM) [3] algorithm to solve the subproblem.

First, we introduce $L \in \mathbb{R}^{p^2 \times n_p}$, and Eq. (10) turns into:

$$\begin{aligned} \min_{U, V, L} & \sum_{n=1}^{n_p} \sum_{k=1}^K \gamma_{nk} \frac{(f(\mathcal{D}) - L)_n^T \Sigma_k^{-1} (f(\mathcal{D}) - L)_n}{2} \\ & + \alpha \|\mathcal{H}\|_{3DTV} + \beta \|\mathcal{H}\|_1 \\ \text{s.t. } & L = f(\mathcal{H} \circ \mathcal{D} + \mathcal{H}^\perp \circ \text{fold}(UV^T)), \end{aligned} \quad (13)$$

and its Lagrangian form is:

$$\begin{aligned} \min_{U, V, L} & \sum_{n=1}^{n_p} \sum_{k=1}^K \gamma_{nk} \frac{(f(\mathcal{D}) - L)_n^T \Sigma_k^{-1} (f(\mathcal{D}) - L)_n}{2} \\ & + \alpha \|\mathcal{H}\|_{3DTV} + \beta \|\mathcal{H}\|_1 + \frac{\mu}{2} \|L + \mu^{-1} \Lambda \\ & - f(\mathcal{H} \circ \mathcal{D} + \mathcal{H}^\perp \circ \text{fold}(UV^T))\|_F^2. \end{aligned} \quad (14)$$

We then need to iteratively updating L and U, V as follows: by fixing the other variables except L , the optimization of L becomes divisible by each column. We then have for $n = 1, 2, \dots, n_p$, that

$$L_n = \left(\sum_{k=1}^K \gamma_{nk} \Sigma_k^{-1} + \mu I \right)^{-1} \left(\sum_{k=1}^K \gamma_{nk} \Sigma_k^{-1} f(\mathcal{D}) + \mu f(\mathcal{D} - \mathcal{H}^\perp \circ \mathcal{R}) - \Lambda \right). \quad (15)$$

With L attained, the optimization of U, V becomes:

$$\min_{U, V} \|L + \mu^{-1} \Lambda - f(\mathcal{H} \circ \mathcal{D} + \mathcal{H}^\perp \circ \text{fold}(UV^T))\|_F^2, \quad (16)$$

which is equivalent to:

$$\min_{U, V} \sum_{ij(m, n) \in \Omega_{ij}} (L_{mn} + \mu^{-1} \Lambda_{mn} - H_{ij} \circ D_{ij} - H_{ij}^\perp \circ (U_i V_j^T))^2. \quad (17)$$

Here, Ω_{ij} represents the set of indices in patch matrix $f(\mathcal{X})$ corresponding to the unfolded matrix element X_{ij} . Since there are repeated elements conducted by the overlapped patches, we define matrix $W \in \mathbb{R}^{hw \times m}$, where W_{ij} represents the number of repeated times for the single pixel. Hence Eq. (17) can be written as:

$$\min_{U, V} \sum_{ij} W_{ij} \left(\tilde{L}_{ij} - H_{ij} \circ (D_{ij} - U_i V_j^T) - U_i V_j^T \right)^2, \quad (18)$$

where

$$\tilde{L}_{ij} = \frac{\sum_{(m, n) \in \Omega_{ij}} (L_{mn} + \mu^{-1} \Lambda_{mn})}{W_{ij}}. \quad (19)$$

Updating U, V then yields:

$$\begin{aligned} V^{(t+1)} &= \arg \min_V \|\sqrt{W} \circ (\tilde{L} - H \circ (D - U^{(t)} V^{(t)T}) - U^{(t)} V^T)\|_F^2, \\ U^{(t+1)} &= \arg \min_U \|\sqrt{W} \circ (\tilde{L} - H \circ (D - U^{(t)} V^{(t+1)T}) - U V^{(t+1)T})\|_F^2, \end{aligned} \quad (20)$$

which can be solved by off-the-shelf weighted l_2 -norm LRMF method [24].

Update \mathcal{H} . The subproblem of (10) with respect to \mathcal{H} is:

$$\begin{aligned} \min_{\mathcal{H}} & \frac{\mu}{2} \|L - f(\mathcal{H} \circ \mathcal{D} + \mathcal{H}^\perp \circ \text{fold}(UV^T)) + \mu^{-1} \Lambda\|_F^2 \\ & + \alpha \|\mathcal{H}\|_{3DTV} + \beta \|\mathcal{H}\|_1. \end{aligned} \quad (21)$$

Algorithm 1 P-MoG Model for RSRV

Input: video $\mathcal{D} \in \mathbb{R}^{h \times w \times m}$; subspace rank: r ; the number of Gaussians: K ; patch size: p .

Initialization: Randomly initialize U, V , patch MoG parameters $\Pi, \Sigma, \mathcal{H} = 0$.

- 1: **while** not converge **do**
- 2: (E step) Evaluate γ_{nk} by Eq. (9).
- 3: (M step) Evaluate patch based MoG parameters Π, Σ by Eq. (11) and (12).
- 4: Update variable L by Eq. (15).
- 5: Update U, V by Eq. (20).
- 6: Update \mathcal{H} by graph cut algorithm.
- 7: **end while**
- 8: Obtain \mathcal{B} by Eq. (3).
- 9: Post-possess \mathcal{F} by Eq. (22).

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

Considering the entries of \mathcal{H} are binary, *i.e.*, either 1 or 0, it is intrinsically an energy minimization problem of a Markov Random Field (MRF) and can be readily solved by graph cut optimization algorithm [4, 18].

3.3. Post-processing

The proposed P-MoG method helps separate the background, rain streaks and moving objects from the input video with rains. Hence we could combine background and moving object together to obtain a derained video. Before that, we can use an easy post-processing step to further ameliorate the rain streak removal effect by enforcing continuity on the moving object layer. A simple model can be used for this task:

$$\mathcal{F} = \arg \min_{\mathcal{F}} \sum_{n=1}^{n_p} \log \sum_{k=1}^K \pi_k \mathcal{N}((\mathcal{H} \circ \mathcal{D} - \mathcal{F})_n | 0, \Sigma_k) + \lambda \|\mathcal{F}\|_{TV}, \quad (22)$$

which can be readily solved by TV regularization algorithm [30, 33]. We summarize our algorithm as Algorithm 1.

4. Experimental Results

We evaluate the performance of the proposed algorithm on synthetic and natural videos with rain streaks. Throughout our experiments, we set the patch size of the video as 2 for efficiency of our algorithm. The Gaussian component number in our method is set as 3.

4.1. On videos with synthetic rain streaks

In this section we show experiments on videos added with various types of rain streaks. Since the ground truth videos without rain are known, we can compare all competing methods both in quantity and in visualization. We utilize four videos from CDNET database [15]¹ which vary largely

¹<http://www.changedetection.net>

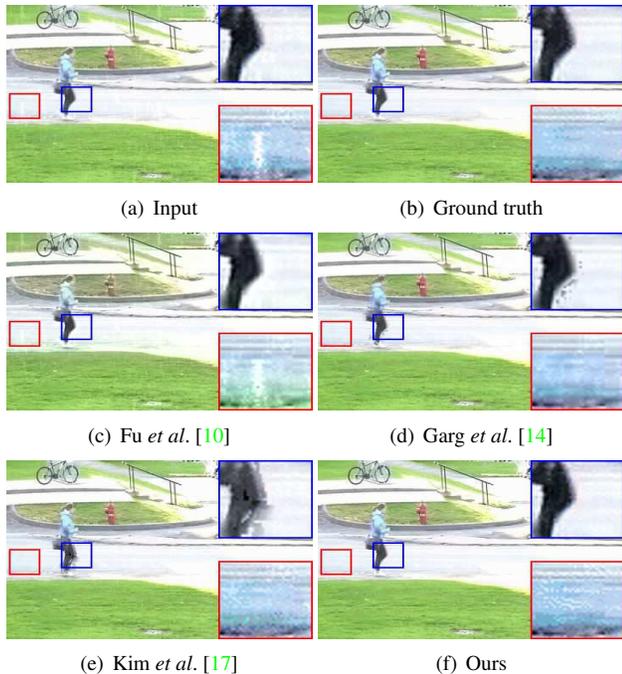


Figure 4. Visual comparison of rain removal effects obtained by different methods on synthetic experiment. Two demarcated areas in each image are amplified at a 3 time larger scale and with the same degree of contrast for easy observation of details.

in moving object numbers and background scenes. We add different kinds of rain streaks taken by photographers under black background², varied from tiny drizzling to heavy rain storm, as can be clearly observed in Figs. 4-7. The compared methods include Fu *et al.* [10], the state-of-the-art method for rain removal in a single image, Garg *et al.* [14], and Kim *et al.* [17], the state-of-the-art method for RSRV. For fair comparison with image based method [10], we employ the same network structure, while add a new temporal dimension (with size 3) for each input, as well as the filters. Each input is then a 3D patch and encodes the temporal information among 3 adjacent frames. Besides, we generate training samples from 15k adjacent frame sequences w/o rains (rain styles are similar as aforementioned) from videos whose background is fixed, w/o moving objects, like the settings in our experiments.

Fig. 4 shows a light rain scene with a girl passing by the static camera. It is easy to observe that clear rain streaks have been left in the rain removal maps obtained by Fu *et al.*'s algorithm. Garg *et al.*'s and Kim *et al.*'s methods have a better rain removal effects, while at the expense of degrading the visual performance of the moving object. In comparison, the proposed P-MoG method can not only more comprehensively remove rain streaks in the video, but also

²http://www.2gei.com/video/effect/1_rain/

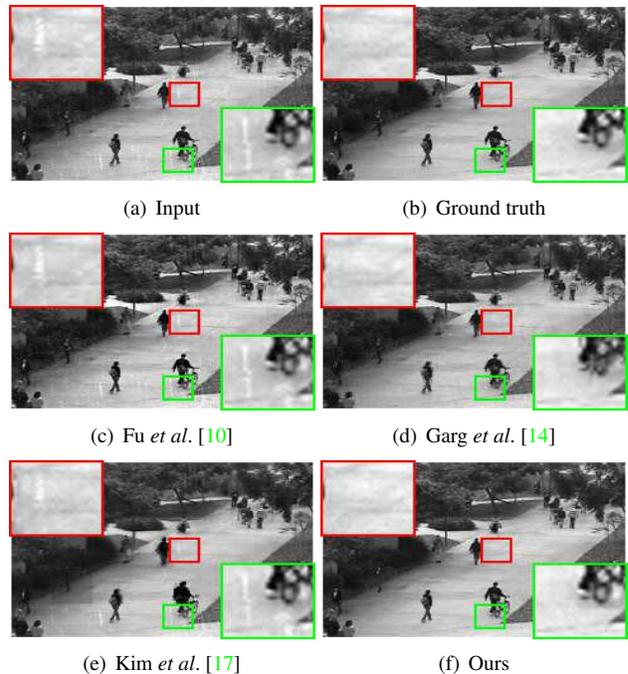


Figure 5. Visual comparison of rain removal effects obtained by different methods on synthetic experiment.

best keep the shape and texture details³.

Fig. 5 shows the scenario of light rain but with complex moving objects. This video contains only 24 frames in total, thus offering less temporal information. In this video, Fu *et al.*'s method can work well on some frames of the video, but rain streaks can still be clearly seen in multiple frames. Garg *et al.*'s and Kim *et al.*'s methods also perform not very well since evident moving objects in videos bring blurring effect and less frames limit its capability to train a discriminative rain map. Comparatively, the proposed method still attains promising visual effect in both rain removal and detail preservation from the ground truth video.

Figs. 6 and 7 illustrate the performance of all competing methods on videos with relatively heavy rains. Rain streaks in Fig. 6 are thick whereas that in Fig. 7 are more like rain storm with tons of rain streaks blown by wind. All compared methods fail to remove all rain streaks, while our P-MoG method performs consistently well in these hard scenarios, successfully detecting most rain streaks of video in the rain streak layer, while not involving much texture or edge information belonging to moving object or background layers, which can be also observed in Fig. 1.

Quantitative comparisons are listed in Tables 1 and 2. Here we use five performance metrics, PSNR, VIF [27], SSIM [32], FSIM [35] and UQI [31]. From the tables, it is seen that our method attains evidently better results in terms

³The results of all comparison methods on entire videos can be seen in supplementary material

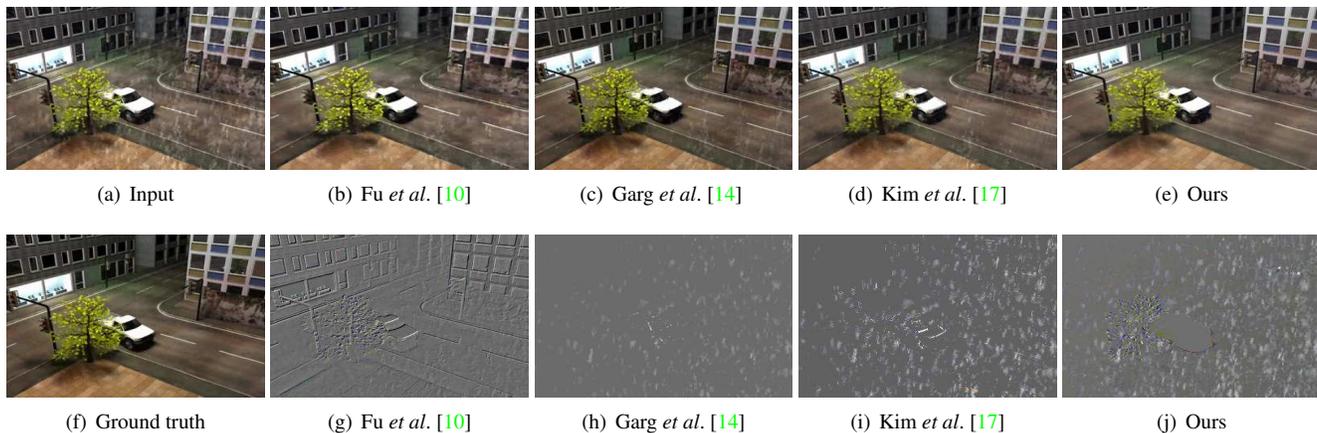


Figure 6. Rain removal results and corresponding rain streaks extracted by different methods on a video with heavy rains.

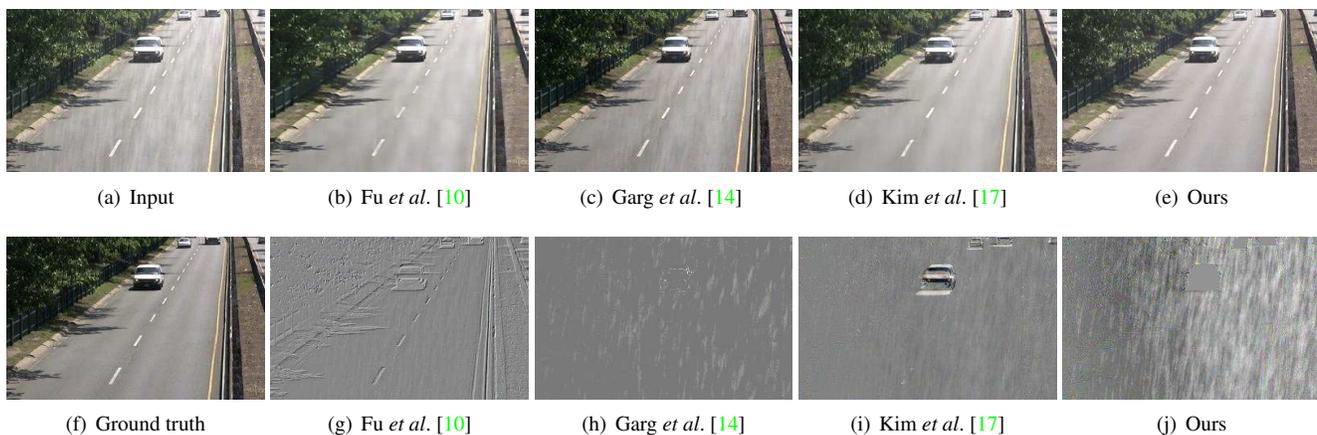


Figure 7. Rain removal results and corresponding rain streaks extracted by different methods on a video with heavy rains.

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

Data set	Fig. 4				Fig. 5				Fig. 6				Fig. 7			
	VIF	SSIM	FSIM	UQI												
Input	0.846	0.981	0.991	0.934	0.731	0.950	0.975	0.927	0.591	0.877	0.935	0.816	0.717	0.917	0.970	0.763
Fu [10]	0.696	0.956	0.968	0.847	0.673	0.948	0.971	0.923	0.530	0.887	0.933	0.812	0.670	0.935	0.967	0.808
Garg [14]	0.862	0.984	0.990	0.949	0.745	0.961	0.979	0.944	0.712	0.935	0.969	0.887	0.707	0.920	0.972	0.772
Kim [17]	0.810	0.981	0.987	0.941	0.642	0.949	0.968	0.933	0.666	0.943	0.967	0.907	0.589	0.912	0.960	0.758
Ours	0.904	0.993	0.993	0.969	0.786	0.977	0.985	0.968	0.757	0.960	0.980	0.952	0.768	0.949	0.981	0.891

Table 2. Performance comparison of different methods on synthetic rain videos in items of PSNR.

	Fig. 4	Fig. 5	Fig. 6	Fig. 7
Input	39.366	33.704	28.151	23.819
Fu [10]	27.628	32.008	27.927	23.624
Garg [14]	37.853	33.464	32.397	24.641
Kim [17]	36.474	31.347	32.532	25.101
Ours	41.201	34.822	30.471	24.503

of VIF, SSIM, FSIM and UQI in all cases. Since these measures mainly focus on image structure and are more consistent with human’s visual perception, the superiority of the proposed method can be substantiated. As for PSNR, our method performs not the best in two heavy rain cases. This can be explained by the fact that our method removes not only rain streaks in the video, but also the possible camera

noise as well. Therefore if the ground truth is with evident camera noise, the PSNR value of our method inclines to not very faithfully represent the practical quality of video recovery. Actually, from the visual effects, the superiority of our method can still be clearly observed.

4.2. On real videos with rain streaks

Fig. 8 shows the results of different methods on the ‘wall’ sequence⁴. The rain streaks in the original video are dense. The comparison methods include: Garg *et al.* [14]⁴, Tripathy *et al.* [28]⁵, Zhang *et al.* [36] (code is written by

⁴http://www.cs.columbia.edu/CAVE/projects/camera_rain/

⁵http://www.ecdept.iitkgp.ernet.in/web/faculty/smkho/docs/rain_removal/rain_removal.html

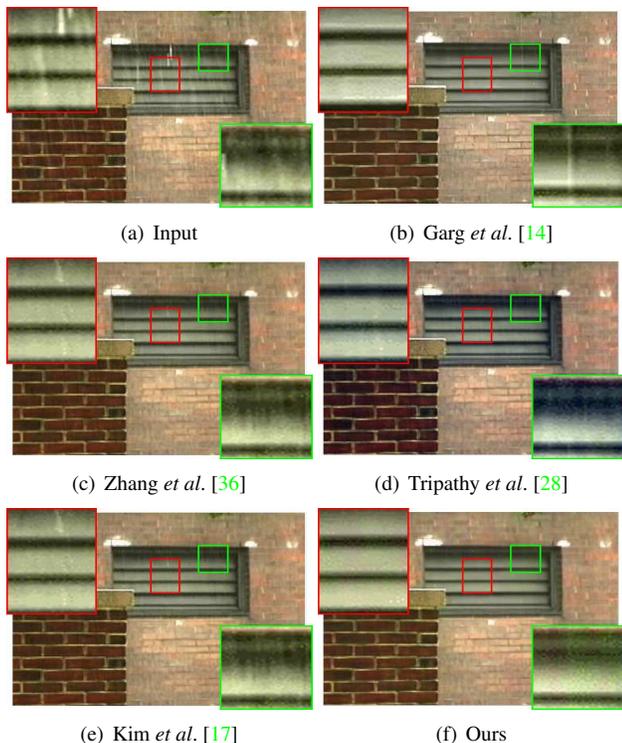


Figure 8. Rain streak removal performance of different methods on a real heavy rain video without moving object.

ourselves), and Kim *et al.* [17]⁶. The results of the former two are directly downloaded from the corresponding websites, while those of the latter two are obtained by implementing codes on the video. The superiority of the proposed method is easy to be observed in both more complete rain removal effect and better texture/edge recovery.

In Fig. 9, we present the results of the same frame selected from the sequence ‘traffic’⁴. The original video consists of walking pedestrian and moving vehicles. The RSRV results obtained by Garg *et al.* [14], Zhang *et al.* [36], Liu *et al.* [20] and Tripathy [28, 29] are directly downloaded⁷, while Kim *et al.* [17]’s code is available⁶ and we run it in this video. Similar to the last experiment, the proposed P-MoG method also attains superior performance in both rain streak removal and detail recovery from video. In this experiments, it is further advantageous in moving object recovery. As a comparison, some of the other competing methods evidently fail in this task.

5. Conclusion

In this paper, we have introduced a simple and concise model for rain streak removal from a video. Differing from

⁶<http://mcl.korea.ac.kr/deraining/>

⁷http://www.ecdept.iitkgp.ernet.in/web/faculty/smukho/docs/spatiotemporal/rain_rain_spatiotemporal.html

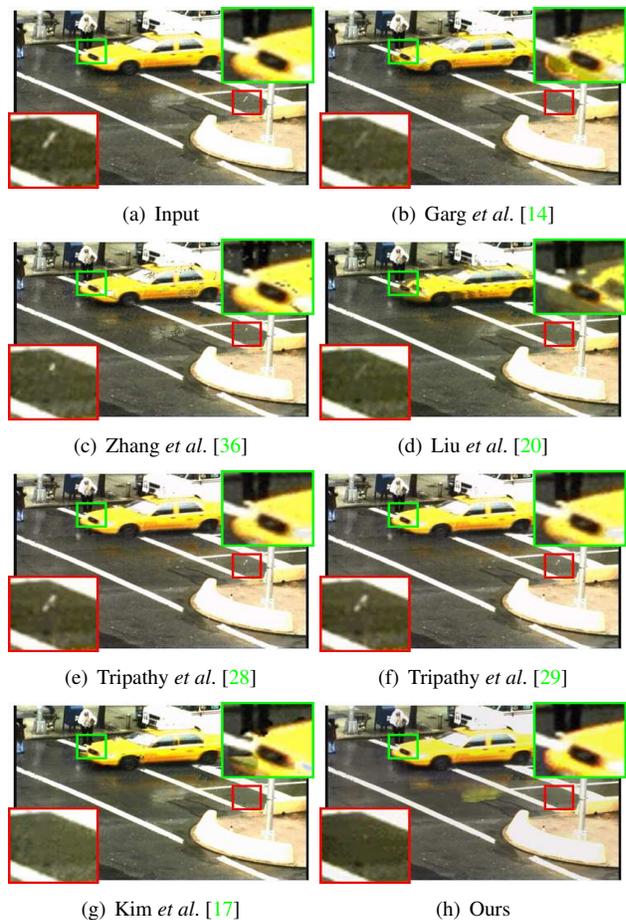


Figure 9. Rain streak removal performance of different methods on a real dynamic rain video with moving objects.

the existing methods which mainly consider rain streaks as deterministic knowledge, the proposed method encodes this information as stochastic knowledge and use patch-based mixture of Gaussians to formulate it. Together with the priors imposed on moving objects and background scene, this concise model shows surprisingly good performance in the RSRV task on various videos with large variations of rain types. The rationality of taking rain streaks as stochastic is thus substantiated. For future work, we consider to solve the rain removal problem with a moving camera. Also, we intend to combine online strategies to further improve its efficiency and realize the requirement of real-time rain removal task.

Acknowledgement

This research was supported by the China NSFC projects under contract 61373114, 61661166011, 11690011, 61603292, and the 973 Program of China under contract 2013CB329404.

References

- [1] P. C. Barnum, S. Narasimhan, and T. Kanade. Analysis of rain and snow in frequency space. *International Journal of Computer Vision*, 86(2):256, 2010. **2**
- [2] J. Bossu, N. Hautire, and J. P. Tarel. Rain or snow detection in image sequences through use of a histogram of orientation of streaks. *International Journal of Computer Vision*, 93(3):348–367, 2011. **3**
- [3] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011. **4**
- [4] 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. **5**
- [5] N. Brewer and N. Liu. Using the shape characteristics of rain to identify and remove rain from video. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 451–458. Springer, 2008. **3**
- [6] 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*, 25(10):4677–4690, 2016. **4**
- [7] J. Chen and L.-P. Chau. A rain pixel recovery algorithm for videos with highly dynamic scenes. *IEEE Transactions on Image Processing*, 23(3):1097–1104, 2014. **3**
- [8] 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. **1, 3**
- [9] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977. **4**
- [10] 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. **3, 6, 7**
- [11] X. Fu, H. J. D. Zeng, Y. Huang, X. Ding, and J. Paisley. Removing rain from single images via a deep detail network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3855–3863, 2017. **3**
- [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. **1, 2**
- [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. **1, 2**
- [14] K. Garg and S. K. Nayar. Vision and rain. *International Journal of Computer Vision*, 75(1):3–27, 2007. **1, 2, 6, 7, 8**
- [15] 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. **5**
- [16] 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. **3**
- [17] 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. **1, 3, 6, 7, 8**
- [18] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159, 2004. **5**
- [19] 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. **3**
- [20] 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. **1, 2, 8**
- [21] 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. **3**
- [22] 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. **4**
- [23] D. Meng, Q. Zhao, and Z. Xu. Improve robustness of sparse pca by l 1-norm maximization. *Pattern Recognition*, 45(1):487–497, 2012.
- [24] K. Mitra, S. Sheorey, and R. Chellappa. Large-scale matrix factorization with missing data under additional constraints. In *Advances in Neural Information Processing Systems*, pages 1651–1659, 2010. **5**
- [25] 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. **1**
- [26] 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. **2**
- [27] H. R. Sheikh and A. C. Bovik. Image information and visual quality. *IEEE Transactions on Image Processing*, 15(2):430–444, 2006. **6**
- [28] 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. **3, 7, 8**
- [29] A. K. Tripathi and S. Mukhopadhyay. Video post processing: low-latency spatiotemporal approach for detection and removal of rain. *IET Image Processing*, 6(2):181–196, 2012. **1, 3, 8**
- [30] 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 International Conference on Machine Learning*, pages 235–243, 2014. **5**
- [31] Z. Wang and A. C. Bovik. A universal image quality index. *IEEE Signal Processing Letters*, 9(3):81–84, 2002. **6**

- [32] 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. 6
- [33] 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. 5
- [34] H. Yong, D. Meng, W. Zuo, and L. Zhang. Robust online matrix factorization for dynamic background subtraction. *arXiv preprint arXiv:1705.10000*, 2017. 4
- [35] 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. 6
- [36] 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. 2, 7, 8
- [37] Q. Zhao, D. Meng, Z. Xu, W. Zuo, and Y. Yan. l_1 -norm low-rank matrix factorization by variational bayesian method. *IEEE transactions on neural networks and learning systems*, 26(4):825–839, 2015. 4
- [38] 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. 4
- [39] X. Zhou, C. Yang, and W. Yu. Moving object detection by detecting contiguous outliers in the low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(3):597–610, 2013. 4