

Automatic Single-Image-Based Rain Streaks Removal via Image Decomposition

Li-Wei Kang, *Member, IEEE*, Chia-Wen Lin, *Senior Member, IEEE*, and Yu-Hsiang Fu

Abstract—Rain removal from a video is a challenging problem and has been recently investigated extensively. Nevertheless, the problem of rain removal from a single image was rarely studied in the literature, where no temporal information among successive images can be exploited, making the problem very challenging. In this paper, we propose a single-image-based rain removal framework via properly formulating rain removal as an image decomposition problem based on morphological component analysis (MCA). Instead of directly applying conventional image decomposition technique, we first decompose an image into the low-frequency and high-frequency parts using a bilateral filter. The high-frequency part is then decomposed into “rain component” and “non-rain component” by performing dictionary learning and sparse coding. As a result, the rain component can be successfully removed from the image while preserving most original image details. Experimental results demonstrate the efficacy of the proposed algorithm.

Index Terms—Rain removal, sparse representation, dictionary learning, image decomposition, morphological component analysis (MCA).

I. INTRODUCTION

DIFFERENT weather conditions such as rain, snow, haze, or fog will cause complex visual effects of spatial or temporal domains in images or videos [1]–[10]. Such effects may significantly degrade the performances of outdoor vision systems relying on image/video feature extraction [11]–[16] or visual attention model [17], such as image registration [9], event detection [8], object detection [14]–[16], tracking, and recognition, scene analysis [17] and classification, image indexing and retrieval [11], and image copy/near-duplicate detection. A comprehensive survey of detection approaches for outdoor environmental factors, such as rain and snow, to enhance the accuracy of video-based automatic incident detection systems can be found in [8].

A. Vision-based Rain Removal

Removal of rain streaks has recently received much attention.

Manuscript received March 19, 2011. This work was supported in part by the National Science Council, Taiwan, under Grants NSC98-2221-E-007-080-MY3, NSC99-2218-E-001-010 and NSC99-2811-E-001-006.

Li-Wei Kang is with the Institute of Information Science, Academia Sinica, Taipei, Taiwan.

Chia-Wen Lin is with the Department of Electrical Engineering, National Tsing Hua University, Hsinchu, Taiwan (phone: +886-3-573-1152; fax: +886-3-571-5971; e-mail: cwlin@ee.nthu.edu.tw).

Yu-Hsiang Fu is with the MStar Semiconductor Inc., Hsinchu, Taiwan.

A pioneering work on detecting and removing rain streaks in a video was proposed in [2], where the authors developed a correlation model capturing the dynamics of rain and a physics-based motion blur model characterizing the photometry of rain. It was subsequently shown in [3] that some camera parameters, such as exposure time and depth of field can be selected to mitigate the effects of rain without altering the appearance of the scene. Moreover, an improved video rain streak removal algorithm incorporating both temporal and chromatic properties was proposed in [6]. In addition, the utilization of shape characteristics of rain streak for identifying and removing rain streaks from videos was proposed in [7]. Furthermore, a model of the shape and appearance of a single rain or snow streak in the image space was developed in [1] to detect rain or snow streaks. Then, the amount of rain or snow in the video can be reduced or increased.

Moreover, some research works [9], [10] focus on raindrop detection in images or videos (usually on car windshields) which is different from the detection of rain streaks. A video-based raindrop detection method for improving the accuracy of image registration was proposed in [9], where a photometric raindrop model was utilized to perform monocular raindrop detection in video frames. In addition, a detection method for detecting raindrops on car windshields using geometric-photometric environment construction and intensity-based correlation was proposed in [10], which can be applied to vision-based driver assistance systems.

B. Motivations of Single-Image-Based Rain Streak Removal

So far, the research works on rain streak removal found in the literature have been mainly focused on video-based approaches that exploit temporal correlation in multiple successive frames. Nevertheless, when only a single image is available, such as an image captured from a digital camera/camera-phone or downloaded from the Internet, a single-image based rain streak removal approach is required, which was rarely investigated before. In addition, some video rain removal approaches [3] based on adjusting camera parameters may not be suitable to consumer camcorders [6] and cannot be applied to existing acquired image/video data. Furthermore, for removing rain streaks from videos acquired from a moving camera, the performances of existing video-based approaches may be significantly degraded. The reason is that, since these video-based approaches usually perform rain streak detection, followed by interpolating the detected pixels affected by rain streaks in each frame, the non-stationary background due to camera motions and inaccurate motion estimation caused by

the interference of rain streaks would degrade the accuracy of video-based rain streak detection and pixel interpolation. Even though some camera motion estimation techniques can be applied first to compensate for the camera motions [6], its performance may also be degraded by rain streaks or large moving activity. Moreover, for the case of steady effects of rain, i.e., pixels may be affected by rain across multiple consecutive frames, it is hard to detect these pixels or find reliable information from neighboring frames to recover them [2].

Moreover, many image-based applications such as mobile visual search [11], object detection/recognition, image registration, image stitching, and salient region detection heavily rely on extraction of gradient-based features that are rotation- and scale-invariant. Some widely-used features (descriptors) such as SIFT (scale-invariant feature transform) [12], SURF (speeded up robust features) [13], and HOG (histogram of oriented gradients) [14]–[16] are mainly based on computation of image gradients. The performances of these gradient-based feature extraction schemes, however, can be significantly degraded by rain streaks appearing in an image since the rain streaks introduce additional time-varying gradients in similar directions. For example, as illustrated in Fig. 1, the additional unreliable interesting points caused by rain streaks degrade the invariant properties of SIFT/SURF and lead to potentially erroneous image matching in related applications. As an example shown in Fig. 2, we applied the HOG-based pedestrian detector released from [16] to the rain image shown in Fig. 2(a) and its rain-removed version (obtained by the proposed method presented in Sec. III) shown in Fig. 2(b), respectively. It can be found that the detection accuracy for the rain-removed version is better. In addition, visual attention models [17] compute a saliency map topographically encoding for saliency at each location in the visual input that simulates which elements of a visual scene are likely to attract the attention of human observers. Nevertheless, the performances of the model for related applications may also be degraded if rain streaks directly interact with the interested target in an image. Therefore, single-frame-based rain streak removal is desirable.

C. Contribution of Proposed Method

It should be noted that separating and removing rain streaks from the non-rain part in a single frame is not a trivial work as rain streaks are usually highly mixed with the non-rain part making the decomposition of non-rain part very challenging. In this paper, we propose a single-image-based rain streak removal framework by formulating rain streak removal as an image decomposition problem based on MCA [19]–[23]. In our method, an image is first decomposed into the low-frequency and high-frequency parts using a bilateral filter. The high-frequency part is then decomposed into “rain component” and “non-rain component” by performing dictionary learning and sparse coding based on MCA. The major contribution of this paper is three-fold: (i) to the best of our knowledge, our method is among the first to achieve rain streak removal while preserving geometrical details in a single frame, where no temporal or motion information among successive images is

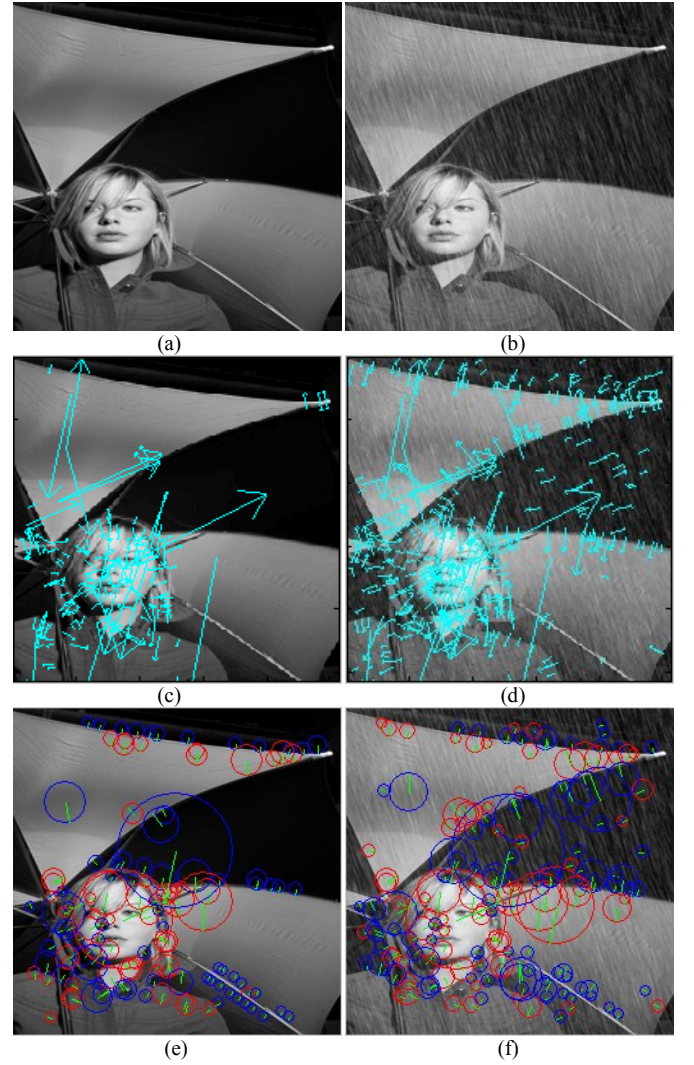


Fig. 1. Examples of interesting point detection: (a) the original non-rain image; (b) the rain image of (a); (c) SIFT interesting point detection for (a) (169 points); (d) SIFT interesting point detection for (b) (421 points); (e) SURF interesting point detection for (a) (131 points); and (f) SURF interesting point detection for (b) (173 points).

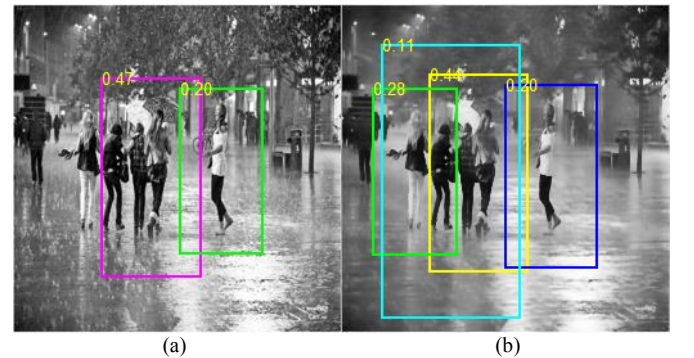


Fig. 2. Applying the HOG-based pedestrian detector released from [16] to: (a) the original rain image (4 pedestrians detected); and (b) the rain-removed version (obtained by the proposed method) of (a) (5 pedestrians detected).

required; (ii) we propose the first automatic MCA-based image decomposition framework for rain streak removal; and (iii) the learning of the dictionary for decomposing rain streaks from an image is fully automatic and self-contained, where no extra training samples are required in the dictionary learning stage.

TABLE I
NOTATION

Symbols	Meanings
I	Input rain image
I^{Non_Rain}	Rain-removed version of I
N	Number of pixels in I
S	Number of layers (or morphological components) in I
I_s	The s -th component decomposed from I
b_s^k	The k -th image patch extracted from I_s
θ_s, θ_s^k	Sparse coefficients of I_s and b_s^k
D_s	Dictionary for sparsely representing I_s
λ, τ	Regularization parameters
M_s, m_s	Numbers of elements of θ_s and θ_s^k
y^k, θ^k	The k -th patch training exemplar and its sparse coefficients
I_{LF}, I_{HF}	Low-frequency and high-frequency parts of I
I_{HF}^G, I_{HF}^R	Geometric component and rain component of I_{HF}
D_{HF}	Dictionary learned from the training exemplars extracted from I_{HF}
D_{HF_G}, D_{HF_R}	Sub-dictionaries divided from for sparsely representing I_{HF}^G and I_{HF}^R
D_{EG}	Extended global dictionary
b_{HF}^k	The k -th patch extracted from I_{HF}
θ_{HF}^k	Sparse coefficients of b_{HF}^k
$\tilde{\theta}_{HF}^k$	Reconstructed sparse coefficients of b_{HF}^k
$\tilde{\theta}_{HF_G}^k, \tilde{\theta}_{HF_R}^k$	Sparse coefficients in $\tilde{\theta}_{HF}^k$ corresponding to D_{HF_G} and D_{HF_R}
$\tilde{b}_{HF_G}^k, \tilde{b}_{HF_R}^k$	The k -th reconstructed geometric patch and rain patch
$\mu(D_1, D_2)$	Mutual coherence between dictionaries D_1 and D_2

The rest of this paper is organized as follows. In Sec. II, we briefly review the concepts of MCA-based image decomposition, sparse coding, and dictionary learning techniques. Sec. III presents the proposed single-image-based rain streak removal framework. In Sec. IV, experimental results are demonstrated. Finally, Sec. V concludes this paper.

II. MCA-BASED IMAGE DECOMPOSITION, SPARSE CODING, AND DICTIONARY LEARNING

The key idea of MCA is to utilize the morphological diversity of different features contained in the data to be decomposed and to associate each morphological component to a dictionary of atoms. In this section, the conventional MCA-based image decomposition approaches [19]–[23], sparse coding [24], and dictionary learning [25], [26] techniques are briefly introduced. The symbols used in this paper are listed in TABLE I.

A. MCA-based Image Decomposition

Suppose that an image I of N pixels is a superposition of S layers (called morphological components), denoted by $I = \sum_{s=1}^S I_s$, where I_s denotes the s -th component, such as the geometric or textural component of I . To decompose the image I into $\{I_s\}_{s=1}^S$, the MCA algorithms [19]–[23] iteratively minimize the following energy function:

$$E(\{I_s\}_{s=1}^S, \{\theta_s\}_{s=1}^S) = \frac{1}{2} \left\| I - \sum_{s=1}^S I_s \right\|_2^2 + \tau \sum_{s=1}^S E_s(I_s, \theta_s), \quad (1)$$

where $\theta \in \mathbb{R}^{M_s}$ denotes the sparse coefficients corresponding to I_s with respect to dictionary D_s , τ is a regularization parameter, and E_s is the energy defined according to the type of

D_s (global or local). For a global dictionary $D_s \in \mathbb{R}^{N \times M_s}$, $N \leq M_s$, the energy function E_s is defined as

$$E_s(I_s, \theta_s) = \frac{1}{2} \|I_s - D_s \theta_s\|_2^2 + \lambda \|\theta_s\|_1, \quad (2)$$

where λ is a regularization parameter. Usually, to decompose an image into its geometric and textural components, traditional basis functions, such as wavelets or curvelets, are used as the dictionary for representing the geometric component whereas global DCT (discrete cosine transform) basis functions are used as the dictionary for representing the textural component of the image [19]–[23].

With respect to a local dictionary $D_s \in \mathbb{R}^{n \times m_s}$, $n \leq m_s$, $\theta_s^k \in \mathbb{R}^{m_s}$ represents the sparse coefficients of patch $b_s^k \in \mathbb{R}^n$, $k = 1, 2, \dots, N$, extracted from I_s . Each patch b_s^k can be extracted centralized with a pixel of I_s and overlapped with adjacent patches. The energy function E_s for the local dictionary can be defined as

$$E_s(I_s, \theta_s) = \sum_{k=1}^N \left(\frac{1}{2} \|b_s^k - D_s \theta_s^k\|_2^2 + \lambda \|\theta_s^k\|_1 \right), \quad (3)$$

Usually, a local dictionary for representing the textural component of an image is either composed of traditional basis functions, such as local DCT [19]–[21], [23], or constructed from the dictionary learning procedure [22] described in Sec. II-B.

The MCA algorithms solve (1) by iteratively performing for each component I_s the following two steps: (i) update of the sparse coefficients: this step performs sparse coding to solve θ_s or $\{\theta_s^k\}_{k=1}^N$ to minimize $E_s(I_s, \theta_s)$ while fixing I_s ; and (ii) update of the components: this step updates I_s or $\{b_s^k\}_{k=1}^N$ while fixing θ_s or $\{\theta_s^k\}_{k=1}^N$.

More specifically, in the case of decomposing I into two components I_s , $s = 1, 2$, a key step of MCA is to properly select a dictionary built by combining two sub-dictionaries D_s , $s = 1, 2$, D_1 and D_2 can be either global or local dictionaries and should be mutually incoherent, that is, D_1 can provide sparse representation for I_1 , but not for I_2 , and vice versa. To decompose I into geometric (I_1) and textural (I_2) components, global wavelet or global curvelet is used as D_1 , whereas global DCT or local DCT is used as D_2 in [19]–[21], [23]. A comprehensive description of dictionary selections and related parameter settings for different kinds of image decomposition can be found in Table 2 of [20]. On the other hand, in [22], a global wavelet/curvelet basis is also used as D_1 , whereas D_2 is constructed through a local dictionary learning process described below.

B. Sparse Coding and Dictionary Learning

Sparse coding [24] is the technique of finding a sparse representation for a signal with a small number of nonzero or significant coefficients corresponding to the atoms in a dictionary [25], [26]. As mentioned previously, it is required to construct a dictionary D_s containing the local structures of textures for sparsely representing each patch b_s^k extracted from the textural component I_s of image I . In some applications, we

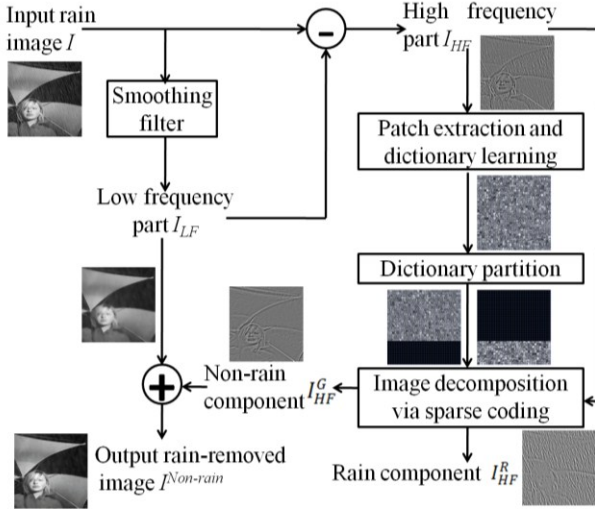


Fig. 3. Block diagram of the proposed rain streak removal method.

may use a set of available training exemplars (similar to the patches extracted from the component we want to decompose) $y^k \in R^n, k = 1, 2, \dots, P$ to learn a dictionary D_s sparsifying y^k by solving the following optimization problem:

$$\min_{D_s \in R^{n \times m_s}, \theta^k \in R^{m_s}} \sum_{k=1}^P \left(\frac{1}{2} \|y^k - D_s \theta^k\|_2^2 + \lambda \|\theta^k\|_1 \right), \quad (4)$$

where θ^k denotes the sparse coefficients of y^k with respect to D_s and λ is a regularization parameter. Equation (4) can be efficiently solved by performing a dictionary learning algorithm, such as K-SVD [25] or online dictionary learning [26] algorithms, where the sparse coding step is usually achieved via OMP (orthogonal matching pursuit) [24]. Finally, the image decomposition is achieved by iteratively performing the MCA algorithm to solve I_s (while fixing D_s) described in Sec. II-A and the dictionary learning algorithm to learn D_s (while fixing I_s) until convergence. The convergence of the MCA image decomposition algorithms has been proven in [22].

III. PROPOSED RAIN STREAK REMOVAL FRAMEWORK

Fig. 3 shows the proposed single-image-based rain streak removal framework, in which rain streak-removal is formulated as an image decomposition problem. In our method, the input rain image is first roughly decomposed into the low-frequency (LF) part and the high-frequency (HF) part using the bilateral filter [27], [28], where the most basic information will be retained in the LF part while the rain streaks and the other edge/texture information will be included in the HF part of the image as illustrated in Figs. 4(a) and 4(b). Then, we perform the proposed MCA-based image decomposition to the HF part that can be further decomposed into the rain component [see Fig. 4(c)] and the geometric (non-rain) component [see Fig. 4(d)]. In the image decomposition step, a dictionary learned from the training exemplars extracted from the HF part of the image itself can be divided into two sub-dictionaries by performing HOG [14] feature-based dictionary atom clustering. Then, we perform sparse coding [24] based on the two sub-dictionaries to

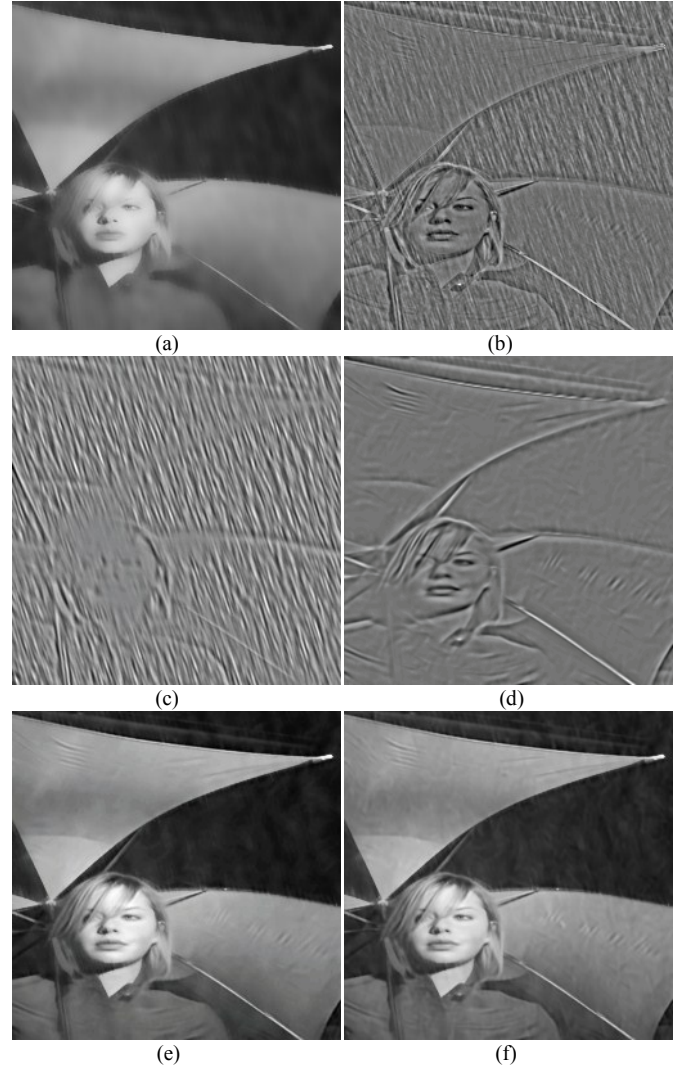


Fig. 4. Step-by-step results of the proposed rain streak removal process: (a) the low-frequency (LF) part of the rain image in Fig. 1(b) decomposed using the bilateral filter; (b) the high-frequency (HF) part; (c) the rain component; and (d) geometric component. Combining (d) and the LF part shown in (a) to obtain: (e) the rain-removed version for the rain image shown in Fig. 1(b) (VIF = 0.50, $\mu = 0.6970$); (f) the rain-removed version for the rain image shown in Fig. 1(b) with D_{EG} (VIF = 0.52).

achieve MCA-based image decomposition, where the geometric component in the HF part can be obtained, followed by integrating with the LF part of the image to obtain the rain-removed version of this image as illustrated in Figs. 4(e) and 4(f). The detailed method shall be elaborated below.

A. Major Differences between Proposed Method and Traditional MCA-based Approaches

As mentioned in Sec. II, traditional MCA algorithms usually use a fixed global dictionary based on wavelets/curvelets to represent the geometric component of an image. To represent the textural component of an image, either a fixed global (global DCT) or local (local DCT) dictionary is used. In addition, a learned dictionary may also be used to represent the textural component. Nevertheless, to decompose an image into the geometric and textural components, the selection of dictionaries and related parameter tuning seems to be heavily empirical, as the examples shown in Table 2 of [20]. Based on

our experience, it is not easy to select a proper fixed dictionary to represent rain streaks due to its variety.

In addition, learning a dictionary for representing textural component usually assumes that a set of exemplar patches for the texture to be represented can be either known in advance or extracted from an image to be decomposed itself. Nevertheless, in practice, it is usually not easy to select correct rain patches in a single rain image automatically. It is also not easy to directly extract pure rain patches for dictionary learning from a rain image due to that rain streaks usually cover most regions in a rain image. That is, the geometric and rain components are usually largely mixed. Moreover, even though a traditional fixed global dictionary based on wavelets/curvelets can well sparsely represent the geometric component of an image, using a learned dictionary based on the exemplar patches extracted from the component itself would be much better [29].

Therefore, rather than using a fixed dictionary, assuming prior training exemplar patches available, or resorting to tuning parameters for the used dictionary, our method extracts a set of selected patches from the HF (high-frequency) part of a rain image itself to learn a dictionary. Then, based on the features extracted from individual atoms, we classify the atoms constituting the dictionary into two clusters to form two sub-dictionaries for representing the geometric and rain components of the image, respectively. The dictionary learning process in the proposed method is elaborated in Sec III-C.

Traditional MCA algorithms are all directly performed on the pixel domain of an image. However, it is typically not easy to directly decompose an image into its geometric and rain components in the pixel domain, because the geometric and rain components are usually largely mixed in a rain image, making the dictionary learning process difficult to clearly identify the “geometric (non-rain) atoms” and “rain atoms” from the training patches directly extracted from the pixel domain. This may lead to removing too many image contents that belong to the geometric component but are erroneously classified to the rain component.

Therefore, we propose to first roughly decompose a rain image into the low-frequency (LF) part and the HF part. Obviously, the most basic information of the image is retained in the LF part whereas the rain component and the other edge/texture information are mainly included in the HF part. The decomposition problem can be therefore converted to decomposing the HF part into the rain and other textural components. Such decomposition aids in the dictionary learning process as it is easier to classify in the HF part “rain atoms” and “non-rain atoms” into two clusters based on some specific characteristics of rain streaks.

Furthermore, traditional MCA-based image decomposition approaches are all achieved by iteratively performing the MCA algorithm and the dictionary learning algorithm until convergence. In contrast, the proposed method is non-iterative except for that the utilized dictionary learning, clustering, and sparse coding tools are essentially iterative, as will be explained below.

B. Preprocessing and Problem Formulation

For an input rain image I , in the preprocessing step, we apply a bilateral filter [27] to roughly decompose I into the LF part (I_{LF}) and HF part (I_{HF}), i.e., $I = I_{LF} + I_{HF}$. The bilateral filter can smooth an image while preserving edges, by means of a nonlinear combination of nearby image values. In this step, we adjust the strength of smoothness of the bilateral filter to remove all of the rain streaks from I , as an illustrative example shown in Figs. 4(a) and 4(b). Then, our method learns a dictionary D_{HF} based on the training exemplar patches extracted from I_{HF} to further decompose I_{HF} , where D_{HF} can be further divided into two sub-dictionaries, D_{HF_G} and D_{HF_R} ($D_{HF} = [D_{HF_G} | D_{HF_R}]$), for representing the geometric and rain components of I_{HF} , respectively. As a result, we formulate the problem of rain streak removal for image I as a sparse coding-based image decomposition problem as follows:

$$\min_{\theta_{HF}^k \in \mathbb{R}^m} \|b_{HF}^k - D_{HF}\theta_{HF}^k\|_2^2 \text{ s.t. } \|\theta_{HF}^k\|_0 \leq L, \quad (5)$$

where $b_{HF}^k \in \mathbb{R}^n$ represents the k -th patch extracted from I_{HF} , $k = 1, 2, \dots, P$. $\theta_{HF}^k \in \mathbb{R}^m$ are the sparse coefficients of b_{HF}^k with respect to $D_{HF} \in \mathbb{R}^{n \times m}$, $n \leq m$, and L denotes the sparsity or maximum number of nonzero coefficients of θ_{HF}^k . Each patch b_{HF}^k can be reconstructed and used to recover either the geometric or rain component of I_{HF} depending on the corresponding nonzero coefficients in θ_{HF}^k , i.e., the used atoms from D_{HF_G} or D_{HF_R} .

C. Dictionary Learning and Partition

1) *Dictionary Learning*: In this step, we extract from I_{HF} a set of overlapping patches as the training exemplars y^k for learning dictionary D_{HF} . We formulate the dictionary learning problem as [25], [26]

$$\min_{D_{HF} \in \mathbb{R}^{n \times m}, \theta^k \in \mathbb{R}^m} \frac{1}{P} \sum_{k=1}^P \left(\frac{1}{2} \|y^k - D_{HF}\theta^k\|_2^2 + \lambda \|\theta^k\|_1 \right), \quad (6)$$

where θ^k denotes the sparse coefficients of y^k with respect to D_{HF} and λ is a regularization parameter. In this work, we apply an efficient online dictionary learning algorithm proposed in [26] to solve (6) to obtain D_{HF} , as illustrated in Fig. 5.

2) *Dictionary Partition and Identification*: We find that the atoms constituting D_{HF} can be roughly divided into two clusters (sub-dictionaries) for representing the geometric and rain components of I_{HF} , respectively. Intuitively, the most significant feature for a rain atom can be extracted via “image gradient.” In the proposed method, we utilize the HOG descriptor [14] to describe each atom in D_{HF} . We then apply the K-means algorithm to classify all of the atoms in D_{HF} into two clusters D_1 and D_2 based on their HOG feature descriptors. The following procedure is to identify which cluster consisting of rain atoms and which cluster consisting of geometric or non-rain atoms. First, we calculate the variance of gradient direction for each atom d_{ij} , $j = 1, 2, \dots, N$, in cluster D_i , as VG_{ij} , where N_i denotes the number of atoms in D_i , $i = 1, 2$. Then, we calculate the mean of VG_{ij} for each cluster D_i as

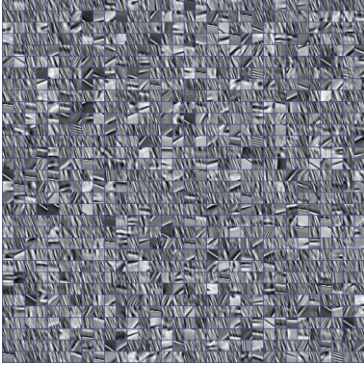


Fig. 5. Dictionary learned from the patches extracted from the HF part shown in Fig. 4(b) via the online dictionary learning for sparse coding algorithm [26], where each atom is of size 16×16 .

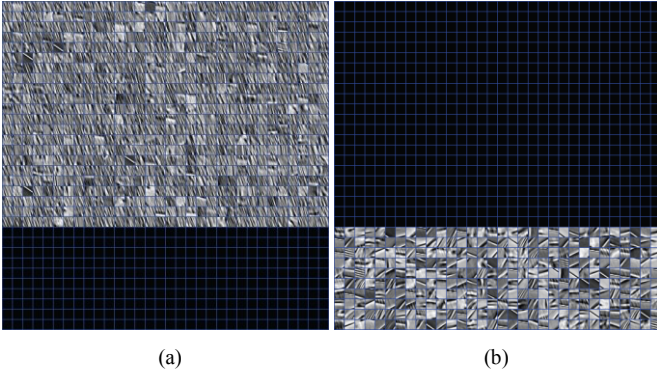


Fig. 6. Dictionary partition for the dictionary shown in Fig. 5: (a) rain sub-dictionary; and (b) geometric or non-rain sub-dictionary.

MVG_i . Based on the fact that the edge directions of rain streaks in an atom are usually consistent, *i.e.*, the variance of gradient direction for a rain atom should be small, we identify the cluster with the smaller MVG_i as rain sub-dictionary D_{HF_R} , and the other one as geometric (or non-rain) sub-dictionary D_{HF_G} , as depicted in Fig. 6.

On the other hand, although the dictionary learning step in the proposed method can be fully self-contained, where no extra training samples are required, the decomposition performance can be further improved by collecting a set of exemplar patches from the HF parts of some training non-rain images to learn an extended global dictionary D_{EG} to enrich the dictionary. Fig. 7 illustrates an example of D_{EG} . Then, we integrate D_{EG} with D_{HF_G} of each image to form the final geometric sub-dictionary of the image.

3) *Diversities of Two Sub-dictionaries*: The MCA algorithms distinguish between the morphological components by taking advantage of the diversities of two dictionaries D_1 and D_2 , which can be measured by the mutual incoherence of them [21]. The mutual coherence $\mu(D_1, D_2)$ between D_1 and D_2 can be defined as

$$\mu(D_1, D_2) = \max_{d_{1i} \in D_1, d_{2j} \in D_2} |\langle d_{1i}, d_{2j} \rangle|, \quad (7)$$

where d_{1i} and d_{2j} stand for the i -th and j -th atoms (rearranged as a column vector) in D_1 and D_2 , respectively, and $\langle d_{1i}, d_{2j} \rangle$ denotes the inner product of d_{1i} and d_{2j} . When each atom is

normalized to have a unit l_2 -norm, the range of $\mu(D_1, D_2)$ is $[0, 1]$. As a result, the mutual incoherence is $1 - \mu(D_1, D_2)$. The smaller the mutual coherence is, the larger the diversities of the two sub-dictionaries will be, and thus the better the decomposition performance based on the two dictionaries will be. The experimental evaluations of the mutual incoherence of rain sub-dictionary D_{HF_R} and geometric sub-dictionary D_{HF_G} for decomposing a rain image in the proposed method are presented in Sec. IV.

D. Removal of Rain Streaks

Based on the two dictionaries D_{HF_R} and D_{HF_G} , we perform sparse coding by applying the OMP (orthogonal matching pursuit) algorithm [24] for each patch b_{HF}^k extracted from I_{HF} via minimization of (5) to find its sparse coefficients $\tilde{\theta}_{HF}^k$. Different from traditional MCA algorithms, where the sparse coding and dictionary learning should be iteratively performed, we perform sparse coding only once for each patch b_{HF}^k with respect to $D_{HF} = [D_{HF_G} | D_{HF_R}]$.

Then, each reconstructed patch b_{HF}^k can be used to recover either geometric component I_{HF}^G or rain component I_{HF}^R of I_{HF} based on the sparse coefficients $\tilde{\theta}_{HF}^k$ as follows. We set the coefficients corresponding to D_{HF_G} in $\tilde{\theta}_{HF}^k$ to zeros to obtain $\tilde{\theta}_{HF_R}^k$, while the coefficients corresponding to D_{HF_R} in $\tilde{\theta}_{HF}^k$ to zeros to obtain $\tilde{\theta}_{HF_G}^k$. Therefore, each patch b_{HF}^k can be re-expressed as either $\tilde{b}_{HF_G}^k = D_{HF_G} \times \tilde{\theta}_{HF_G}^k$ or $\tilde{b}_{HF_R}^k = D_{HF_R} \times \tilde{\theta}_{HF_R}^k$, which can be used to recover I_{HF}^G or I_{HF}^R , respectively, by averaging the pixel values in overlapping regions. Finally, the rain-removed version of the image I can be obtained via $I^{Non_Rain} = I_{LF} + I_{HF}^G$, as illustrated in Fig. 4(e). In summary, the proposed single-image-based rain streak

TABLE II
SINGLE-IMAGE-BASED RAIN STREAK REMOVAL ALGORITHM

Input: a rain image I .

Output: the rain-removed version I^{Non_Rain} of I .

1. Apply the bilateral filter to obtain the LF part I_{LF} and HF part I_{HF} , such that $I = I_{LF} + I_{HF}$.
2. Extract a set of image patches $y^k \in \mathbb{R}^n, k = 1, 2, \dots, P$, from I_{HF} . Apply the online dictionary learning for sparse coding algorithm to solve

$$\min_{D_{HF} \in \mathbb{R}^{n \times m}, \theta^k \in \mathbb{R}^m} \frac{1}{P} \sum_{k=1}^P \left(\frac{1}{2} \|y^k - D_{HF} \theta^k\|_2^2 + \lambda \|\theta^k\|_1 \right)$$

to obtain the dictionary D_{HF} consisting of the atoms that can sparsely represent $y^k, k = 1, 2, \dots, P$.

3. Extract HOG feature descriptor for each atom in D_{HF} . Apply K -means algorithm to classify all of the atoms into two clusters based on their HOG feature descriptors.
4. Identify one of the two clusters as “rain sub-dictionary,” D_{HF_R} and the other one as “geometric sub-dictionary,” D_{HF_G} .
5. Apply MCA by performing OMP to solve

$$\min_{\theta_{HF}^k \in \mathbb{R}^m} \|b_{HF}^k - D_{HF} \theta_{HF}^k\|_2^2 \text{ s.t. } \|\theta_{HF}^k\|_0 \leq L$$

for each patch $b_{HF}^k \in \mathbb{R}^n, k = 1, 2, \dots, P$, in I_{HF} with respect to $D_{HF} = [D_{HF_G} | D_{HF_R}]$.

6. Reconstruct each patch b_{HF}^k to recover either geometric component I_{HF}^G or rain component I_{HF}^R of I_{HF} based on the corresponding sparse coefficients obtained from Step 5.
7. Return the rain-removed version of I via $I^{Non_Rain} = I_{LF} + I_{HF}^G$.

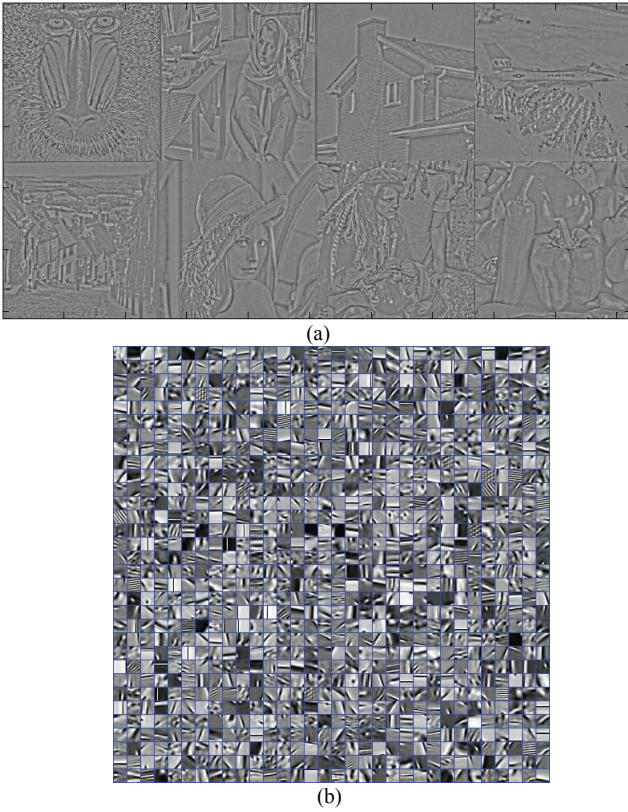


Fig. 7. Extended global dictionary (D_{EG}) learning: (a) the HF parts of the 8 training non-rain images; and (b) the learned extended global dictionary.

removal method is summarized in TABLE II.

IV. EXPERIMENTS AND DISCUSSION

A. Performance Evaluation

Because we cannot find any other single-frame-based approach, to evaluate the performance of the proposed algorithm, we first compare the proposed method with a low-pass filtering method called the bilateral filter proposed in [27], which has been extensively applied and investigated recently for image processing, such as image denoising [28]. We collected several natural or synthesized rain images from the Internet with ground-truth images (non-rain versions) for a few of them. To evaluate the quality of a rain-removed image with a ground-truth, we used the visual information fidelity (VIF) metric [30] which has been shown to outperform PSNR (peak signal-to-noise ratio) metric. More test results can be found in our project website [31].

Besides, we also compare our method with a video-based rain removal method based on adjusting camera parameters proposed in [3] (denoted by “video-based camera see”), which should outperform most of other video-based techniques without adjusting cameras. We captured some single frames from the videos released from [3] and compared our results with the ones of [3] from the same videos. For each video released from [3], the preceding frames are rain frames, followed by succeeding rain-removed frames in the same scene. We pick a single rain frame from the preceding frames for rain removal and compared our results with the

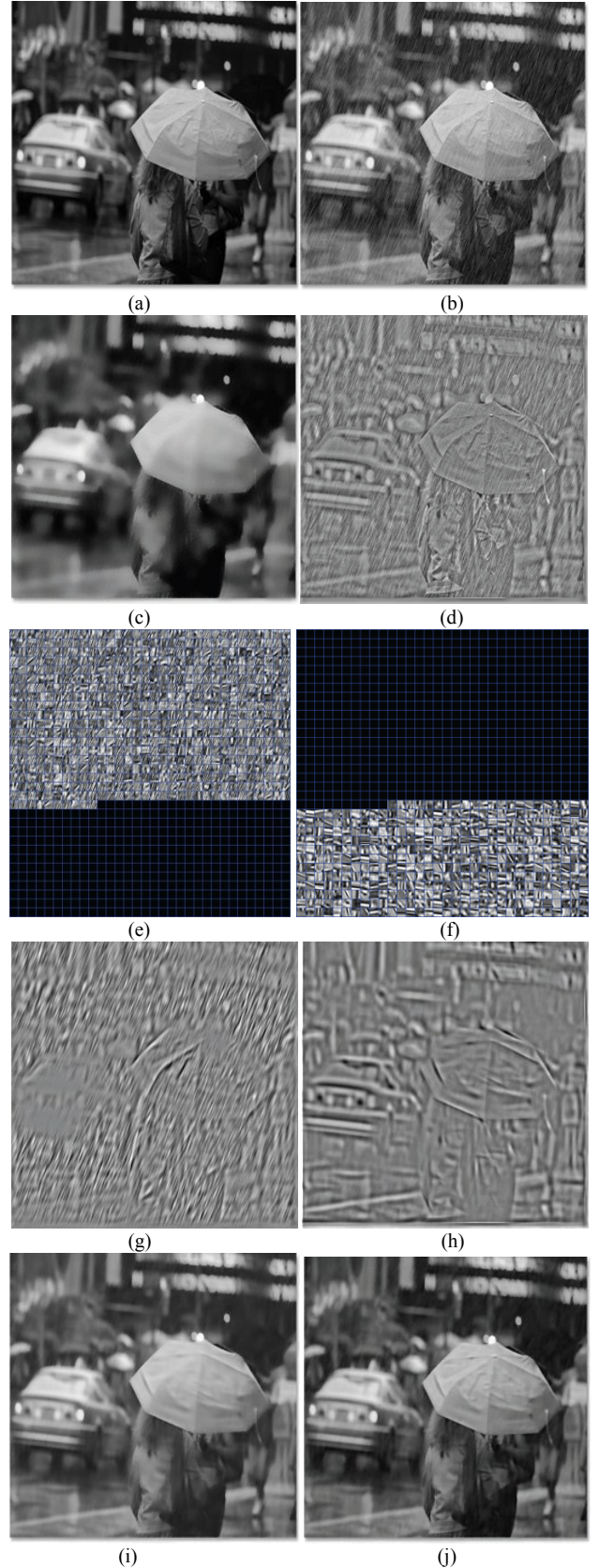


Fig. 8. Rain removal results: (a) the original non-rain image (ground-truth); (b) the rain image of (a); (c) the rain-removed version of (b) via the bilateral filter (VIF = 0.31); (d) the HF part of (b); (e) the rain sub-dictionary for (d); (f) the geometric sub-dictionary for (d); (g) the rain component of (d); (h) the geometric component of (d); (i) the rain-removed version of (b) via the proposed method (VIF = 0.53, $\mu = 0.7618$); and (j) the rain-removed version of (b) via the proposed method with D_{EG} (VIF = 0.57).

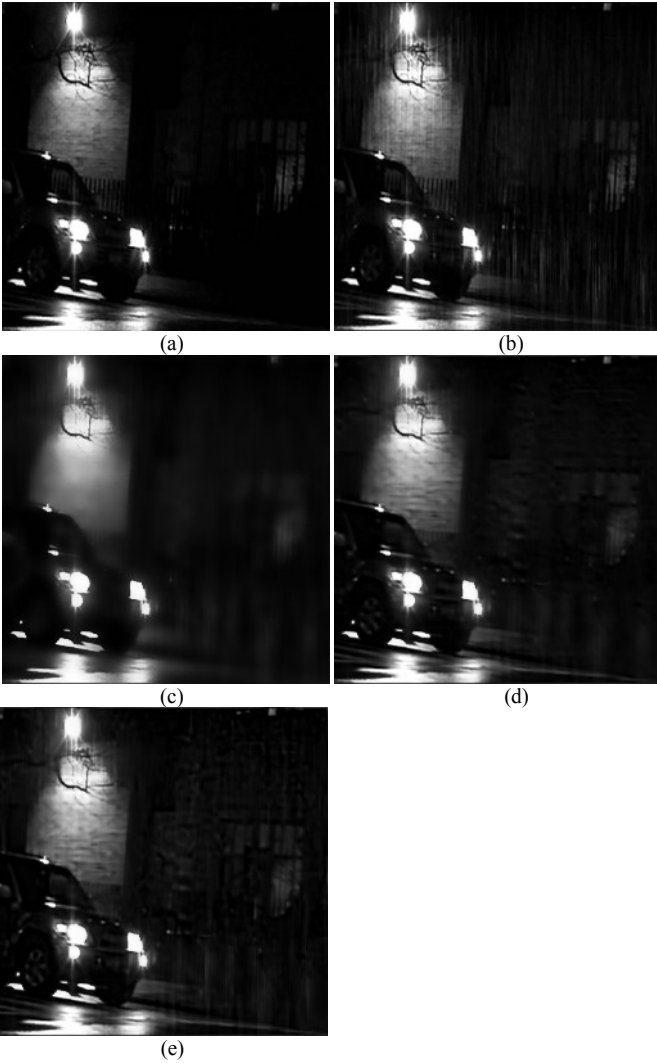


Fig. 9. Rain removal results: (a) the original non-rain image; (b) the rain image of (a); the rain-removed versions of (b) via the: (c) bilateral filter (VIF = 0.21); (d) proposed method (VIF = 0.36, $\mu = 0.7467$); and (e) proposed method with D_{EG} (VIF = 0.38).

rain-removed one [3] of a similar frame from the succeeding frames in the same video (no exactly the same frame is available for comparison).

The related parameter settings of the proposed method are described as follows. The implementation of the bilateral filter is provided by [32], where we set the spatial-domain and intensity-domain standard deviations to 6 and 0.2, respectively, to ensure that most rain streaks in a rain image can be removed. In the dictionary learning step, we used the efficient implementation provided by [26] with default regularization parameter (λ) set to 0.15. For each test gray-scale image of size $N_1 \times N_2$ ($= 256 \times 256$ in our experiments), the patch size, number of training patches, dictionary size, and the number of training iterations are set to $N = 16 \times 16$, $P = (N_1 - \sqrt{n} + 1) \times (N_2 - \sqrt{n} + 1)$, $m = 1024$, and 100, respectively. We also used the efficient OMP implementation provided by [26] with a default number of nonzero coefficients set to at most 10 ($L = 10$). The used HOG implementation is provided by [15]

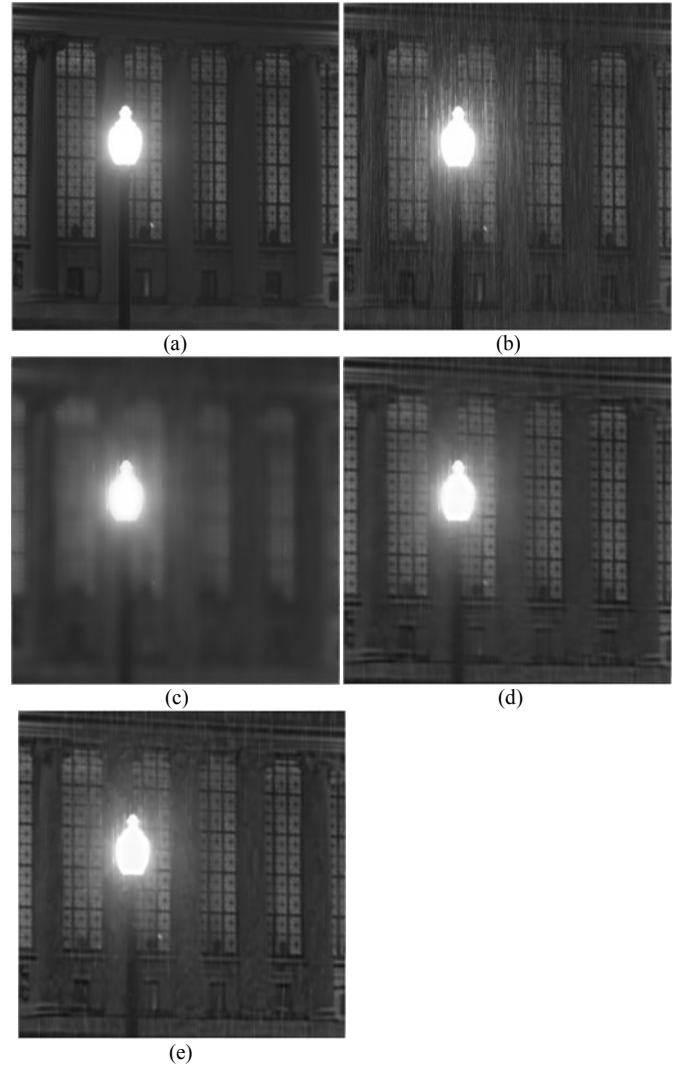


Fig. 10. Rain removal results: (a) the original non-rain image; (b) the rain image of (a); the rain-removed versions of (b) via the: (c) bilateral filter (VIF = 0.09); (d) proposed method (VIF = 0.20, $\mu = 0.8081$); and (e) proposed method with D_{EG} (VIF = 0.20).

with the dimension of each feature descriptor set to 81. The number of iterations for K-means clustering is 100.

We also evaluate the performance of the proposed method with extended global dictionary D_{EG} to be integrated with the respective geometric sub-dictionary for each test image. We collected several training patches extracted from the HF parts of eight widely-used non-rain images, including *Baboon*, *Barbara*, *F-16*, *Goldhill*, *House*, *Lena*, *Man*, and *Pepper* images. The patch size, dictionary size, and number of training iterations are set to 16×16 , 1024, and 200, respectively. The D_{EG} learning process is offline performed only once. The eight training images and D_{EG} are shown in Fig. 7.

The rain removal results obtained from the bilateral filter [27], the proposed method with two sub-dictionaries, and the proposed method with D_{EG} are shown in Figs. 4 and 8–14, where the test images in Figs. 9–11 are rendered rain images provided in [5]. The simulation results demonstrate that although the bilateral filter can remove most rain streaks, it simultaneously removes much image detail as well. The

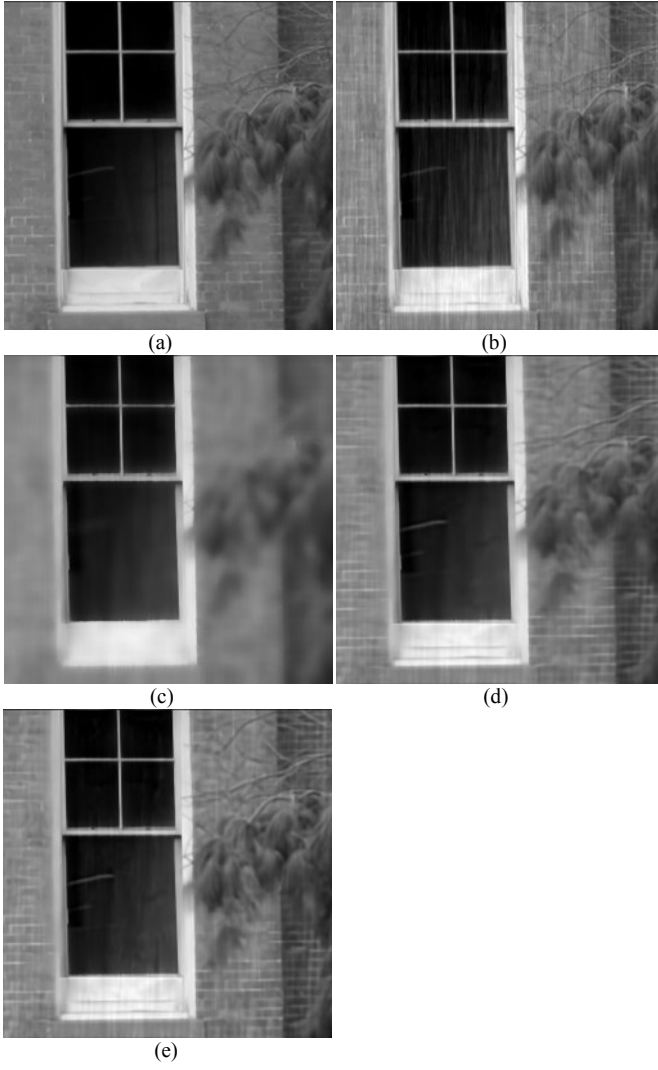


Fig. 11. Rain removal results: (a) the original non-rain image; (b) the rain image of (a); the rain-removed versions of (b) via the: (c) bilateral filter (VIF = 0.29); (d) proposed method (VIF = 0.56, $\mu = 0.7210$); and (e) proposed method with D_{EG} (VIF = 0.60).

proposed method successfully removes most rain streaks while preserving most non-rain image details in most cases. Moreover, the results obtained from the “video-based camera see” method [3] and proposed methods are shown in Figs. 15 and 16. The simulation results demonstrate that the performance of the proposed methods is comparable with the “video-based camera see” method when rain streaks are obviously visible in a single frame.

It can be observed from Figs. 4 and 8–16 that, compared to the proposed methods without and with D_{EG} , integrating D_{EG} with the respective geometric sub-dictionary for each test image leads to slightly better visual quality while significantly increasing the computational complexity (see the run-time analysis in Table III shown below) of sparse coding due to the much larger size of D_{EG} . The reason why sparse coding with D_{EG} usually achieves slightly better visual quality than that without D_{EG} is that D_{EG} provides more non-rain atoms for sparse coding to recover rain-removed version with more image details. Note, the values of mutual coherence (μ)

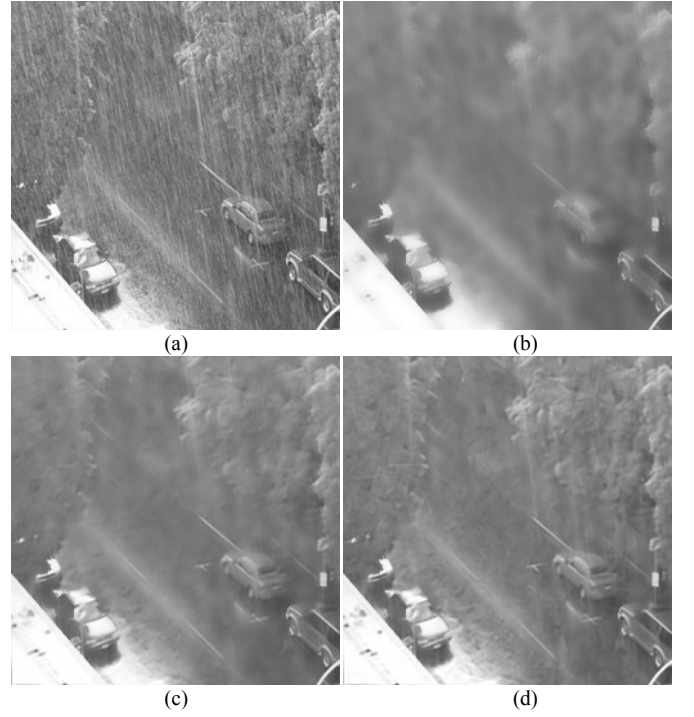


Fig. 12. Rain removal results: (a) the original rain image; and the rain-removed versions of (a) via the: (b) bilateral filter; (c) proposed method ($\mu = 0.7099$); and (d) proposed method with D_{EG} .

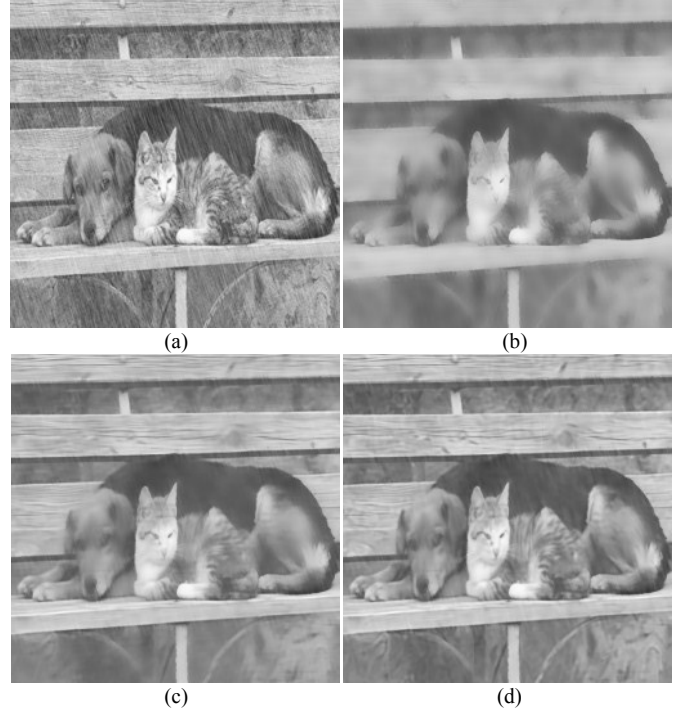


Fig. 13. Rain removal results: (a) the original rain image; and the rain-removed versions of (a) via the: (b) bilateral filter; (c) proposed method ($\mu = 0.7593$); and (d) proposed method with D_{EG} .

between the two sub-dictionaries usually fall in the range of [0.6, 0.9], which is not very close to zero. The main reason is that the two sub-dictionaries used in the proposed method are generated from a single learned dictionary based on a single feature (HOG) based clustering. It is unavoidable that the two dictionaries may have few somewhat coherent atoms, which

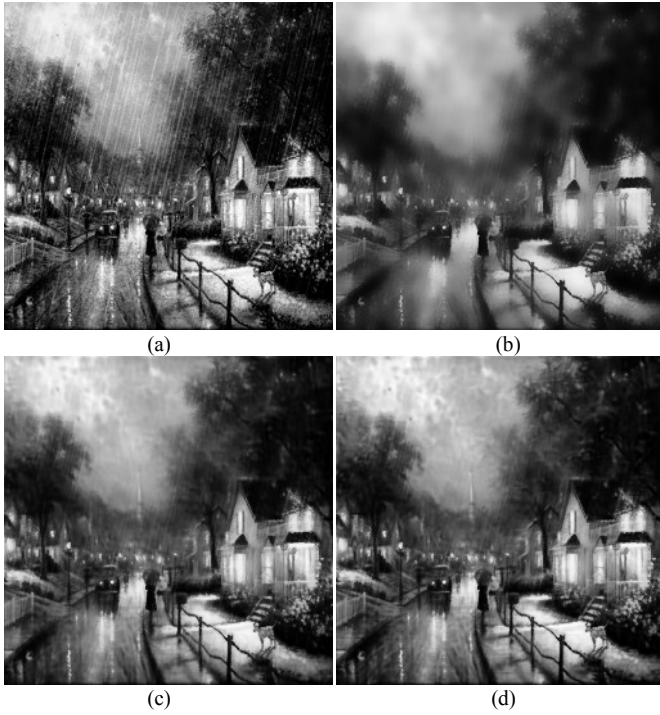


Fig. 14. Rain removal results: (a) the original rain image; and the rain-removed versions of (a) via the: (b) bilateral filter; (c) proposed method ($\mu = 0.6758$); and (d) proposed method with a global dictionary.

will dominate the μ value. In the literature reporting μ values, minimization of the μ between a sensing matrix and a fixed dictionary for learning an optimal sensing matrix was mentioned in [33]. In [33], some t -averaged μ values (approaching μ when t grows) between two matrices were reported to be in a range of $[0.4, 0.6]$, where one matrix is randomly initialized. Hence, based on the obtained rain removal results of our method and the comparison of the ranges of μ between our method and [33], the μ values of our method are usually small enough.

The proposed method was implemented in MATLAB[®] on a personal computer equipped with Intel[®] Core[™] i5-460M processor and 4 GB memory. The run-time of each key step, including the bilateral filtering, dictionary learning, dictionary partition, and sparse coding (without and with D_{EG}), for each test image (Figs. 8–12) is listed in TABLE III. It can be found that the run-time of the dictionary learning step dominates the total run time, which may be further decreased for future work.

TABLE III
RUM-TIME (IN SECONDS) ANALYSIS OF KEY OPERATIONS IN THE PROPOSED METHOD

	Bilateral filtering	Dictionary learning	Dictionary partition	Sparse coding		Total time	
				w/o	with	w/o	with
Fig. 8	1.51	72.99	2.18	8.58	24.07	87.51	101.61
Fig. 9	1.48	66.30	2.17	7.84	22.82	80.21	94.39
Fig. 10	1.48	63.91	2.22	7.50	22.51	77.46	94.91
Fig. 11	1.50	66.21	2.28	7.00	21.45	79.17	94.06
Fig. 12	1.47	78.21	2.22	9.19	24.87	93.34	108.77

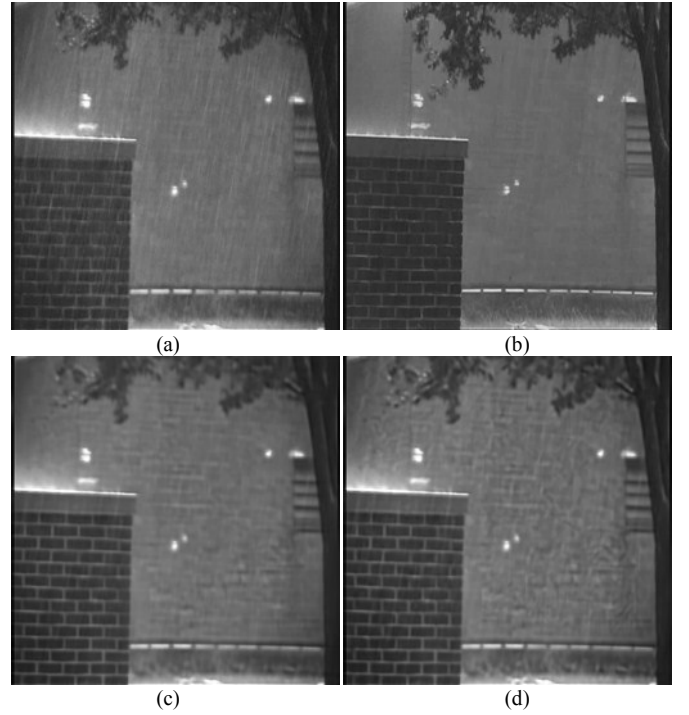


Fig. 15. Rain removal results: (a) the original rain image; and the rain-removed versions of (a) via the: (b) video-based camera see method [3]; (c) proposed method ($\mu = 0.8740$); and (d) proposed method with an extended global dictionary.



Fig. 16. Rain removal results: (a) the original rain image; and the rain-removed versions of (a) via the: (b) video-based camera see method [3]; (c) proposed method ($\mu = 0.8191$); and (d) proposed method with D_{EG} .

B. Discussion

Besides collecting training exemplar patches from some training non-rain images for learning D_{EG} , we may extract

training patches from the same or neighboring camera(s) when extending the proposed method to video rain removal. That is, we may extract exemplar patches from the neighboring rain-removed frames captured by intra/inter cameras in the same scene. Then, we can integrate the geometric sub-dictionary obtained from the HF part itself and the extended global dictionary learned from the pre-collected training patches to form the final geometric sub-dictionary.

On the other hand, we currently use the K -means algorithm (a hard clustering algorithm) to divide a learned dictionary for the HF part of a rain image. Actually, we may use some soft clustering algorithm, such as the fuzzy C -means (FCM), i.e., soft K -means algorithm, to adaptively classify each atom to the cluster it is most suitable to. Moreover, it may be also beneficial to remove some unreliable atoms. Nevertheless, we have not decided any guideline to guarantee the performance induced by using FCM algorithm yet.

V. CONCLUSION

In this paper, we have proposed a single-image-based rain streak removal framework by formulating rain removal as an MCA-based image decomposition problem solved by performing sparse coding and dictionary learning algorithms. Our experimental results show that the proposed method can effectively remove rain streaks without significantly blurring the original image details. For future work, the performance may be further improved by enhancing the sparse coding, dictionary learning, and partition of dictionary steps. For example, when performing sparse coding, some locality constraint may be imposed to guarantee that similar patches should have similar sparse codes/coefficients [34]. Moreover, the proposed method may be extended to remove rain streaks from videos or other kinds of repeated textures.

REFERENCES

- [1] P. C. Barnum, S. Narasimhan, and T. Kanade, "Analysis of rain and snow in frequency space," *Int. J. Comput. Vis.*, vol. 86, no. 2–3, pp. 256–274, 2010.
- [2] K. Garg and S. K. Nayar, "Detection and removal of rain from videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, June 2004, vol. 1, pp. 528–535.
- [3] K. Garg and S. K. Nayar, "When does a camera see rain?" in *Proc. of IEEE Int. Conf. Comput. Vis.*, Oct. 2005, vol. 2, pp. 1067–1074.
- [4] K. Garg and S. K. Nayar, "Vision and rain," *Int. J. Comput. Vis.*, vol. 75, no. 1, pp. 3–27, 2007.
- [5] K. Garg and S. K. Nayar, "Photorealistic rendering of rain streaks," *ACM Trans. on Graphics*, vol. 25, no. 3, pp. 996–1002, July 2006.
- [6] X. Zhang, H. Li, Y. Qi, W. K. Leow, and T. K. Ng, "Rain removal in video by combining temporal and chromatic properties," in *Proc. IEEE Int. Conf. Multimedia Expo*, Toronto, Ont. Canada, July 2006, pp. 461–464.
- [7] N. Brewer and N. Liu, "Using the shape characteristics of rain to identify and remove rain from video," *Lecture Notes in Computer Science*, vol. 5342/2008, pp. 451–458, 2008.
- [8] M. S. Shehata, J. Cai, W. M. Badawy, T. W. Burr, M. S. Pervez, R. J. Johannesson, and A. Radmanesh, "Video-based automatic incident detection for smart roads: the outdoor environmental challenges regarding false alarms," *IEEE Trans. Intell. Transportation Syst.*, vol. 9, no. 2, pp. 349–360, June 2008.
- [9] M. Roser and A. Geiger, "Video-based raindrop detection for improved image registration," *IEEE Int. Conf. Comput. Vis. Workshops*, Kyoto, Sept. 2009, pp. 570–577.
- [10] J. C. Halimeh and M. Roser, "Raindrop detection on car windshields using geometric-photometric environment construction and intensity-based correlation," in *Proc. of IEEE Intell. Vehicles Symp.*, Xi'an, China, June 2009, pp. 610–615.
- [11] <http://www.google.com/mobile/goggles/>.
- [12] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
- [13] H. Baya, A. Essa, T. Tuytelaars, and L. V. Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understanding*, vol. 110, no. 3, pp. 346–359, June 2008.
- [14] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, San Diego, CA, USA, June 2005, vol. 1, pp. 886–893.
- [15] O. Ludwig, D. Delgado, V. Goncalves, and U. Nunes, "Trainable classifier-fusion schemes: an application to pedestrian detection," in *Proc. IEEE Int. Conf. Intell. Transportation Syst.*, St. Louis, MO, USA, Oct. 2009, pp. 1–6.
- [16] S. Maji, A. C. Berg, and J. Malik, "Classification using intersection kernel support vector machines is efficient," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Anchorage, Alaska, USA, June 2008, pp. 1–8.
- [17] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov 1998.
- [18] Y. H. Fu, L. W. Kang, C. W. Lin, and C. T. Hsu, "Single-frame-based rain removal via image decomposition," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Process.*, May 2011, Prague, Czech Republic.
- [19] J. M. Fadili, J. L. Starck, J. Bobin, and Y. Moudden, "Image decomposition and separation using sparse representations: an overview," *Proc. IEEE*, vol. 98, no. 6, pp. 983–994, June 2010.
- [20] J. M. Fadili, J. L. Starck, M. Elad, and D. L. Donoho, "MCALab: reproducible research in signal and image decomposition and inpainting," *IEEE Computing in Science & Engineering*, vol. 12, no. 1, pp. 44–63, 2010.
- [21] J. Bobin, J. L. Starck, J. M. Fadili, Y. Moudden, and D. L. Donoho, "Morphological component analysis: an adaptive thresholding strategy," *IEEE Trans. Image Process.*, vol. 16, no. 11, pp. 2675–2681, Nov. 2007.
- [22] G. Peyré, J. Fadili, and J. L. Starck, "Learning adapted dictionaries for geometry and texture separation," in *Proc. SPIE*, vol. 6701, 2007.
- [23] J. L. Starck, M. Elad, and D. L. Donoho, "Image decomposition via the combination of sparse representations and a variational approach," *IEEE Trans. Image Process.*, vol. 14, no. 10, pp. 1570–1582, Oct. 2005.
- [24] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.
- [25] M. Aharon, M. Elad, and A. M. Bruckstein, "The K-SVD: an algorithm for designing of overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
- [26] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, vol. 11, pp. 19–60, 2010.
- [27] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. IEEE Int. Conf. Comput. Vis.*, Bombay, India, Jan. 1998, pp. 839–846.
- [28] M. Zhang and B. K. Gunturk, "Multiresolution bilateral filtering for image denoising," *IEEE Trans. Image Process.*, vol. 17, no. 12, pp. 2324–2333, Dec. 2008.
- [29] H. W. Chen, L. W. Kang, and C. S. Lu, "Dictionary learning-based distributed compressive video sensing," in *Proc. Picture Coding Symp.*, Nagoya, Japan, Dec. 2010.
- [30] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 430–444, Feb. 2006.
- [31] NTHU Rain Removal project. [Online]. Available: http://www.ee.nthu.edu.tw/cwlin/Rain_Removal/Rain_Removal.htm.
- [32] <http://mesh.brown.edu/dlanman/courses.html>.
- [33] J. M. Duarte-Carvajalino and G. Sapiro, "Learning to sense sparse signals: simultaneous sensing matrix and sparsifying dictionary optimization," *IEEE Trans. Image Process.*, vol. 18, no. 7, pp. 1395–1408, July 2009.
- [34] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, San Francisco, CA, USA, June 2010, pp. 3360–3367.