Frequency Domain Directional Filtering Based Rain Streaks Removal from a Single Color Image

Changbo Liu[1], Yanwei Pang[1], Jian Wang[1], Aiping Yang[1], and Jing Pan[1,2]

[1] School of Electronic Information Engineering, Tianjin University, Tianjin 300072, China
[2] School of Electronic Engineering, Tianjin University of Technology and Education, Tianjin 300222, China

Abstract. Bad weather conditions, such as rain or snow, degrade outdoor vision system performance. Rain removal from a single image has been investigated extensively. However, existing built rain streak models are greatly influenced by inaccurate parameter estimation or non-stationary background due to camera motions. In this paper, we propose a rain streak removal algorithm from a single color image based on frequency-domain directional filter. Experiment results demonstrate that the proposed algorithm can deal with rain steaks of any directions, and maintains more details than existing algorithms.

Keywords: rain removal, directional filter, image enhancement.

1 Introduction

Adverse weather condition, such as rain or snow, severely degrades outdoor vision system's performance. Vision system can't offer reliable target detection [1], object recognition and tracing, feature extraction [2] results due to image degradation by rain. As a result, to improve the robustness of outdoor surveillance system, we have to restore the corrupted image and remove the bad effects of bad weather. Meanwhile, in the field of computer vision, adverse weather condition will increase the noise of the image, reduce the robustness of feature extraction and the generalization ability of classifier, degrade the performance of image segmentation.

Removal of rain streaks from images or images or videos has recently received much attention in the past few years. Many algorithms have been proposed for the removal of rain streaks. These algorithms can be classified into three categories [3], spatial-temporal domain based, frequency domain based and matrix factorization based. Traditional methods on detecting and removing rain streaks usually focus on the temporal domain. Garg and Nayar[4] proposed a method on detecting and removing rain streaks in a video. They developed a correlation model demonstrating the dynamics of rain and a motion blur model characterizing the photometry of rain. Subsequently, they found that some camera parameters can be adjusted to reduce the effects of rain without altering the appearance of the scene [5]. Moreover, Zhang [6] proposed an improved video rain streak removal algorithm incorporating both temporal and chromatic properties. Brewer [7] further utilized the shape characteristics of

D.-S. Huang et al. (Eds.): ICIC 2014, LNCS 8588, pp. 415–424, 2014.

rain streaks for identifying and removing rain streaks from videos. Furthermore, for the frequency based circumstances, Barnum [8] built a model of the shape and appearance of a single rain or snow streak in the image space, and remove rain or snow streaks with its frequency spectrum. Bossu [9] proposed selection rules based on photometry and size to select the potential rain streaks in a video. They computed a histogram of orientations of rain streaks estimated with geometric moments. Moreover, Kang [10] used the method of dictionary leaning. A rain image was first separated into geometry layer and texture layer by bilateral filtering. Then they divided the dictionary online-trained by texture layer into rain-dictionary and no-rain dictionary, and restored the rain-removed image with the no-rain dictionary only.

By analyzing existing algorithms, we found that the existing built rain streak models are greatly influenced by inaccurate parameter estimation or non-stationary background due to camera motions. Consequently, their performances are badly degraded. In this paper, we propose a framework for a single color image rain streaks removal, which utilizes both the uniform spread feature of rain streaks in spatial domain and directional feature in frequency domain. The major contribution of this paper is twofold: 1) we propose a new rain streak direction determination approach, which utilizes the uniform spread feature in spatial domain and directional feature in frequency domain, to build a block-wise HOE feature to determine the direction of rain streaks. 2) our method is a new rain streaks removal method using directional filter in frequency domain. On the basis of rain streaks' direction, we design a corresponding directional filter and remove rain streaks in frequency domain.

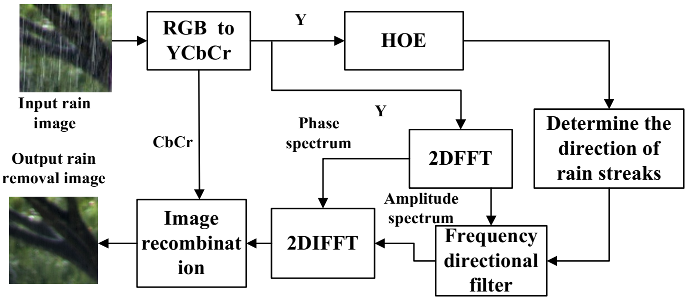

Fig. 1. Block diagram of proposed algorithm

2 Proposed Algorithm

The block diagram in Fig.1 shows the proposed rain streak removal framework, which includes 3 major steps, pre-processing, determination of the direction of rain steaks, frequency domain directional filtering based rain streaks removal. First, the input color image is converted into YCbCr color space, and the Y-component image is divided into blocks. Then we compute the dominant direction of the global edge based on the local Histogram of Oriented Edge to determine the direction of the rain streaks. After that, we design a corresponding 2D frequency domain directional filter, and then utilize the filter to remove rain streaks in frequency domain of Y-component.

At last, we reconstruct the color image by combining the filtered Y-component image with original Cb and Cr component image.

2.1 Pre-processing

Fig. 2 shows an input color image and its corresponding RGB channel images. It is observed that rain steaks are obvious in all 3 channels. However, rain steaks can only influence the Y-channel image [3], and hardly influence the CbCr color channels.

The above observations are fit for most rain images. Considering the reduction of computation cost and making the algorithm more robust, the raised algorithm convert the input color image into YCbCr color space, and then deals with the Y-component image only. The RGB-to-YCbCr relation is shown in (1).

$$\begin{aligned} Y &= 16 + 0.257 \times R + 0.564 \times G + 0.098 \times B \\ C_b &= 128 - 0.148 \times R - 0.291 \times G + 0.439 \times B \\ C_r &= 128 + 0.439 \times R - 0.368 \times G - 0.071 \times B \end{aligned} \tag{1}$$

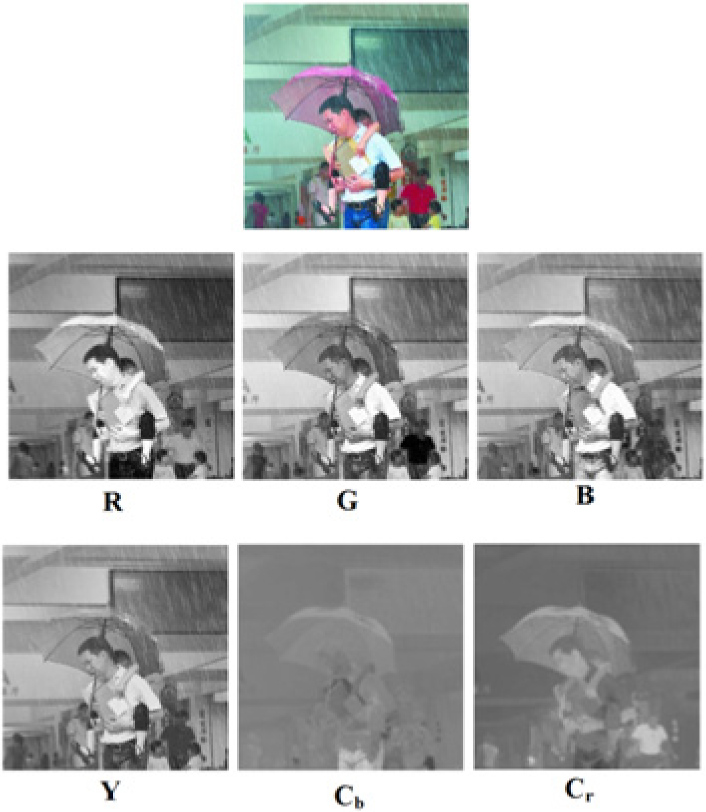

Fig. 2. Color rain image RGB image and YCbCr image

The proposed algorithm determines the rain steaks' direction based on the local Histogram of Oriented Edge. Considering that many rain images are in poor light conditions, rain steaks are not so distinct from background. Classic Canny operator is used to detect the Y-component edges. Comparing with other edge detection methods, Canny operator is exactly fit for detecting the weak edges. There is better connectivity between the detected edges.

2.2 Determination of the Direction of Rain Steaks

The directions of rain steaks are almost the same in the original image and the edge image. Based on the former edge binary image, we compute the horizontal gradient amplitude G_x, vertical gradient amplitude G_y and the gradient orientation ϕ for each edge point in Y-component image. G_x, G_y and ϕ are defined as (2)~(4),

$$\begin{aligned} G_x(x, y) = {} & L(x+1, y-1) + L(x+1, y) + L(x+1, y+1) \\ & - L(x-1, y-1) - L(x-1, y) - L(x-1, y+1) \end{aligned} \tag{2}$$

$$\begin{aligned} G_x(x, y) = {} & L(x-1, y+1) + L(x, y+1) + L(x+1, y+1) \\ & - L(x-1, y-1) - L(x, y-1) - L(x+1, y-1) \end{aligned} \tag{3}$$

$$\phi(x, y) = \arctan(G_y / G_x) \tag{4}$$

where $-90° \le \phi \le 90°$. Considering the edge orientation θ is perpendicular to ϕ, we have the relations $\theta = \phi + 90°$ and $0° \le \theta \le 180°$.

Table 1. Algorithm to determine the direction of rain streaks

1. Divide the edge image into 4*4=16 blocks
2. Compute the amount of edge points within each block. Take the block as clean(no rain steaks) if the proportion of edge points in the sum of points in each block is too low.
3. For the rest blocks, compute the histogram of oriented edge in each block based on the inner edge information within each block. Note the histogram as $HIST_b(k)$(k=1,2,...,8; b=1,2,..., 16), k is the interval of direction, b is the sequence number of blocks. Take the top 3 maximal value's corresponding interval, noted as $ANG_b(1)$, $ANG_b(2)$ and $ANG_b(3)$.
4. For all the 16 blocks, we get 16*3 = 48 interval numbers. Take all the numbers and we get a new histogram of oriented edge called global salient histogram of oriented edge, noted as GHIST(k).
5. With GHIST(k), we choose the maximal value as the direction of rain streaks.

We divide 180° into 8 intervals, noted as interval 1: [0, 22.5°), interval 2:[22.5°, 45°), interval 3:[45°, 67.5°), interval 4:[67.5°, 90°), interval 5:[90°, 112.5°), interval 6:[112.5°, 135°), interval 7:[135°, 157.5°), interval 8:[157.5°, 180°). Considering the background of image corresponds to edge point in all directions, it tends to be impossible to detect the direction of rain streaks with the global histogram of oriented edge. As a result, we propose to deal with the edge in blocks.

The global histogram of oriented edge shows the distribution of edge information in the image. Since the rain streaks are uniform spread within the image, and the edge of the background is usually intensive, the uniform spread of the edge of rain streaks can be well shown with the use of blocks. Also, the local background edge information can be suppressed effectively.

2.3 Frequency Domain Directional Filtering Based Rain Streaks Removal

Most algorithms remove the rain streaks with the spatial domain, which is easily performed. However, there tends to be obvious profile between rain streaks and the background, which causes bad visual effect. To overcome that, we propose a new frequency domain directional filtering based rain streaks removal method.

A frequency wedge directional filter is needed for the removal. Wedge directional filter is a wedge-pass 2D digital filter in frequency domain. There are 3 ways to design a wedge filter, direct optimization method, down-sample method and transform method. Direct optimization is complicated and the filtering structure can't be well designed. The down-sample method designs the 2D filter with 2D down-sample. Horizontal and vertical wedge directional filter can be obtained by shift and down-sample. However, the filter is restricted by sample-factor and only fits for the condition of a single direction filtering. The McClellan-transform [11] based method is most widely used to design a 2D directional filter and works best. Shyu [12] extended the McClellan transform and applied it to all kinds of 2D FIR filters. The pass band can be designed by adjusting the parameters. The proposed method designs the filter with Shyu's method as below.

1D FIR filter impulse response $h(n)$ can be noted as

$$H(\omega) = \sum_{n=0}^{N} a_p(n) \cos(\omega n) \tag{5}$$

The coefficient $a_{\mathrm{p}}(n)$ can be represented by a p-polynomial,

$$a_p(n) = \sum_{m=0}^{M} a(n,m) p^m \tag{6}$$

Use Chebyshev polynomial substituting the $\cos(n\omega)$,

$$H(\omega) = \sum_{n=0}^{N} a_p(n) T_n\left(\cos(\omega)\right) \tag{7}$$

Table 2. Procedure of designing an 8-direction frequency filter

1. Design a 1D prototype filter. Using the Kaiser Window to design a 1D low-pass filter with pass frequency $w_p = 0.45\pi$, stop frequency $w_s = 0.55\pi$, pass band amplitude is 1, peak passband ripple $\alpha_p = 0.5\text{dB}$, minimum stopband attenuation $\alpha_s = 40\text{dB}$.
2. With the transform function (9) and different value of {A, B, C, D, E}, we can get different 90° wedge filters and parallelogram filters. The wedge filters can be used to design four 8-directional sub-band filters by multiplying the 4-directional filters and the parallelogram filters. Then with the 4-directional filters subtracting the four 8-directional filters, we get the other four 8-directional sub-band filters. Subtracting by 1, we can get the all eight 8-directional masking filters.

Bringing in the transform function $\Phi(\omega_1, \omega_2)$, the transform is realized by tiny impulse transform. According to McClellan transform, design the 2D filter with 1D ones, the frequency response is

$$H(\omega_1, \omega_2) = \sum_{n=0}^{N} a(n) T_n\left(\Phi(\omega_1, \omega_2)\right) \tag{8}$$

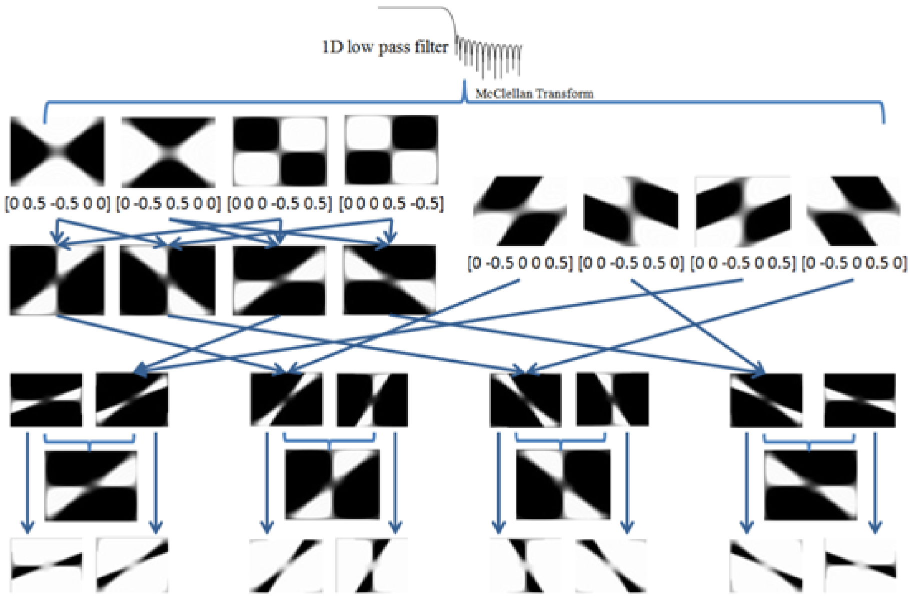

Fig. 3. 8-directional filters design based on McClellan Transform

The transform function is

$$\begin{aligned}\Phi(\omega_1,\omega_2) = {} & A + B\cos(\omega_1) + C\cos(\omega_2) \\ & + D\cos(\omega_1 - \omega_2) + E\cos(\omega_1 + \omega_2)\end{aligned} \tag{9}$$

In the function, the choice of {A, B, C, D, E} depends on the shape of transfer function.

Table 3. Algorithm procedure of frequency domain directional filter based rain removal

1. Note 2D-FFT of Y-component as **I**, amplitude spectrum as **F**, phase spectrum as **P**.
2. Choose a corresponding directional filter from the eight pre-designed 8-directional filters based on the rain streaks direction we determined before. We denote the **F**-filtering result as **F′**. Then we combine **F′** with the phase spectrum **P** to conduct inverse-FFT to return to the spatial domain. The result is denoted as **I′**.
3. With the original Cb-component, Cr-component images and **I′**, we convert the three components from YCbCr to RGB image, and then reconstruct the rain-removed color image.

3 Experiments and Results

Our experiments were implemented in MATLAB 2013b on an Intel I7 CPU (Core 3.4 GHz), 16G RAM PC. To testify the validity of our method, we chose 20 images to do the experiment, among which 8 images come from [10], the others were from the internet or taken by us in real life.

Fig. 4 shows part of the results. The images in the left column are original rain color images, the images in the left column are rain streaks removal result images. Though the proposed algorithm is not very complicated, the results in the first three rows are satisfying.

The last 2 rows of Fig. 4 show the circumstance of unsatisfying results. In the second row from bottom, even the direction of rain streaks can be correctly determined, since the rain streaks have a large influence on the background, the rain removal results are not so well as the previous ones. The image in the last row is in a complicated background. There are a lot of bush in the image, which have rich edge information. Though the rain streaks are obvious, the background has a bad influence on the edges, which makes the determination of the direction of rain streaks impossible. For the first circumstance, the bilateral technique can be applied first to divide the image into geometric and texture layers. Deal with the texture layer only. For the latter circumstance, it's suggested that the image content be analyzed as the image being divided into blocks. If the image block contains rich texture information, its edge direction information could be ignored while analyzing the rain streaks direction.

Fig. 5 shows the comparison results between our algorithm and Kang's method. It is obvious that in the result of our method, the texture is better preserved, especially the parts of face and hand. The reason is that, since Kang first separate the image into geometric layer and texture layer, most high-frequency details are separated into the high-frequency layer, which is used to train the dictionary. After that, using HOG feature and K-means to classify the atoms will misclassify the high frequency texture atoms into the rain dictionary, which will result in the loss of original texture information.

Fig. 4. Rain removal results: Left: original image. Right: rain-removed image.

Besides that, in our algorithm, the look-up-table could be used to accelerate the processing speed. Specifically, before (4), we set up the relations among G_x, G_y and ϕ, stored that as a file. During the processing, after we get G_x and G_y, the look-up-table file is loaded into RAM. By look up the table, we can easily get ϕ of each edge pixel. Using the optimized algorithm, it costs only 280ms to process a color image of size 512*512 while Kang's algorithm needs to train dictionary online, which would cost more than 10s to process a color image of the same size.

(a)

(b)

Fig. 5. (a) Comparison between our method and Kang's method: Left top: original image. Right top: Kang's result. Bottom: our result. (b) Left: Kang's result. Right: our result.

4 Conclusion

Considering the characteristics of rain in spatial domain and frequency domain, we propose a frequency domain directional filtering based rain streaks removal algorithm for a single color image. Combining with the thought of dividing image into blocks, we propose a salient global histogram of oriented edge feature to determine the direction of rain streaks. Based on that, we choose the corresponding directional filter to remove the rain in frequency domain. The experiments show that comparing with the existing algorithms, the proposed algorithm can correctly determine the direction of rain streaks, and provide significantly improvement on the artifacts brought by removal process. For the future work, the performance of our work may be further improved by combining features in the time domain, spatial domain, and frequency domain and further realize the time-spatial-frequency three-domain-joint rain streaks removal.

Acknowledgement. This work was supported in part by the National Basic Research Program of China 973 Program (Grant No. 2014CB340400), the National Natural Science Foundation of China (Grant Nos. 61372145, 61172121, 61271412, and 61222109), and the Open Funding Project of State Key Laboratory of Virtual Reality Technology and Systems, Beihang University (Grant No. BUAA-VR-13KF).

References

1. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: Proceedings of International Conference on Computer Vision and Pattern Recognition, San Diego, CA, vol. 1, pp. 886–893 (2005)
2. Lowe, D.G.: Distinctive Image Features from Scale-invariant Keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
3. Tripathi, A.K., Mukhopadhyay, S.: Removal of Rain from Videos: a Review. Signal, Image and Video Processing (2012)
4. Garg, K., Nayar, S.K.: Detection and Removal of Rain from Videos. In: Proceedings of International Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 528–535 (2004)
5. Garg, K., Nayar, S.K.: When does a camera see rain? International Journal of Computer Vision 2, 1067–1074 (2005)
6. Zhang, X., Li, H., Qi, Y., Leow, W.K., Ng, T.K.: Rain Removal in Video by Combining Temporal and Chromatic Properties. In: Proceedings of International Conference on Multimedia & Expo, Toronto, ON, Canada, pp. 461–464 (2006)
7. Brewer, N., Liu, N.: Using the Shape Characteristics of Rain to Identify and Remove Rain from Video. In: da Vitoria Lobo, N., Kasparis, T., Roli, F., Kwok, J.T., Georgiopoulos, M., Anagnostopoulos, G.C., Loog, M. (eds.) S+SSPR 2008. LNCS, vol. 5342, pp. 451–458. Springer, Heidelberg (2008)
8. Barnum, P.C., Narasimhan, S., Kanade, T.: Analysis of rain and snow in frequency space. International Journal of Computer Vision 86(2/3), 256–274 (2010)
9. Bossu, J., Hautière, N., Tarel, J.P.: 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)
10. Kang, L., Lin, C., Fu, Y.: Automatic Single-Image-Based Rain Streaks Removal via Image Decomposition. IEEE Transactions on Image Processing 21(4), 1743–1755 (2012)
11. McClellan, J.M.: The Design of 2-D Digital Filters by Transformations. In: Proceedings of 7th Princeton Conference on Information and Systems, pp. 247–251 (1973)
12. Shyu, J.J., Pei, S.C., Huang, Y.D.: Design of Variable Two-dimensional FIR Digital Filters by McClellan Transformation. IEEE Transactions on Circuits and Systems 56(3), 574–582 (2009)