© 2018 IJSRST | Volume 4 | Issue 2 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X
Themed Section: Science and Technology

Gradient-Based Photorealistic Rendering of Rain Streaks for Image Processing Applications

V. Santhiya, R. Sangeetha Priya, Brenda Mohan
Department of Electronics and Communication Engineering Jeppiaar SRR Engineering College, I.T Highway, Padur, Chennai, India

ABSTRACT

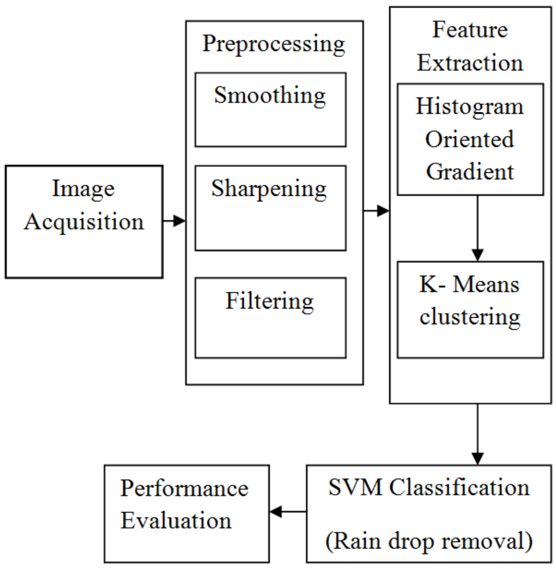
To record criminal activities who loot company properties in various places such as parking lots, outside the work place, banks, industries etc., surveillance systems are installed. Under aggressive and bad weather conditions, the images recorded contains rain streaks and are difficult to conceptualize. In these situations, images will help the police in investigation. Removal of rain droplets from still images has been an effective research topic for various image processing applications. In this paper, to the best of our knowledge, we use Histogram Oriented Gradient (HOG) to extract rain droplets from an image. After extraction, K-means clustering and SVM classifiers removes the common rain patterns present in the image. Finally, the performance is evaluated by calculating the PSNR value for original image and comparing the values with the PSNR of rain removed image, which are more or less found to be equal.
Keywords : Image Acquisition, Preprocessing, Histogram Oriented Gradient, K-Means Clustering, SVM Classifiers, Performance Evaluation.

I. INTRODUCTION

Generally in smart cars, which is fully self- governing with maximal communication potential. It requires no human assistance and can independently go anywhere at any time in any conditions. This system can manage with all circumstances automatically within defined use but it may not be able to survive with all weather conditions such as heavy rain, snow etc. In these situations, the rain drops must be removed with advanced rain removal algorithm which helps the automated smart car to drive safely. Many algorithms have been proposed [1-5] to remove rain streaks in the images. [1] Demonstrated a method for globally detecting rain and snow by a physical and statistical model to highlight the spatio-temporal frequencies. But frequency-based detection has errors when the frequencies corresponding to rain and snow are too cluttered. [2] Uses an algorithm of adaptive nonlocal means filter. By scrutinizing the shape and orientation of the elliptical kernel at each pixel location, they identified the rain streaks regions. But this method is not efficient in removing rain streaks present in video sequences, by employing the temporal information. [3] Proposed a method of histogram of orientation of streaks. Data of this histogram are then modeled as a Gaussian uniform mixture. But in the presence of light rain, mixture of Gaussian is no longer relevant. [4] Developed a comprehensive model for the visual appearance of rain. But this method does not remove acutely defocused rain streaks. [5] Demonstrated a framework that consists of two parts. (1) Framework for detection based on phase congruency features. (2) Reconstruction of scene using optical flow estimation from local phase information.

Recently, several methods have been proposed [6-10]. In [6], the input image is first decomposed into base layer and detail layer. To detect and to remove rain streaks from detail layer, sparse coding dictionary learning is used. Finally, by combining base layer and rain removed detail layer, output is obtained. Similarly in [7], by using hybrid feature set removal of rain streaks and restoration of non rain component is achieved. [8] Introduced self learning based image decomposition to distinguish rain streaks from detail layer automatically. To recover original clean image from rainy image [9] uses discriminative sparse coding. [10] Uses convolutional neural network on high frequency detail content to remove rain streaks. [6-10] shows different methods based on dictionary learning. In this work, we propose a simple but efficient method for removing rain streaks by using Histogram Oriented Gradient (HOG) and Support Vector Machine (SVM). HOG is nothing but a feature descriptor. There are four steps to calculate histogram of gradients. (1) To convert color image into gray scale image. (2) To calculate luminance gradient for each pixel. (3) To determine the orientations for each pixel. (4) To calculate luminance and orientations for descriptor blocks. SVM is one among the machine learning techniques used to train a model to classify if an image contains rain or not. Finally the performance is evaluated by calculating peak signal to noise ratio for rain removed image and it is compared with the PSNR value of original image.

II. BLOCK DIAGRAM



BLOCK DIAGRAM DESCRIPTION

IMAGE ACQUISITION

Image acquisition is the first stage of any vision system. To perform many different vision tasks, various methods of processing techniques can be applied to the image after the image has been obtained. However, if the image has not been acquired acceptably then the intended tasks may not be achieved.

PRE PROCESSING

To enhance the image features which are important for further processing or to suppress the unwanted distortions/noise present in an image, preprocessing is done.

1. SMOOTHING

Smoothing is the process of reducing sharp transitions in the gray level of an image. It is used to reduce noise present in an image and to de-blur the image. Averaging filter or box filter can be described as

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

It replaces each pixel value with the average of its neighborhood value. Since all weights are equal, it is called as box filter. Sketch of explanation

$$A = \frac{1}{m^2}\sum_{i=1}^{m^2} I_m$$

$I_m = s_m + n_m$ with n being i.i.d. G(0,σ^2)

$$E(A) = \frac{1}{m^2}\sum s_m$$

$$\text{var}(A) = [(A-E(A))^2] = \frac{\sigma^2}{m}$$

2. SHARPENING

Sharpening is quite opposite to smoothing process. It is used to highlight fine details and to enhance the detail which is blurred by error or by some natural effect in an image. In principle, a signal that is proportional to a high-pass filtered version of original image is added to the original image. In detail, the original image is first filtered by a high-pass filter. It extracts the high frequency components, and the original image is added to the scaled version of the high-pass filter output, thus sharpened image is produced. The sharpening operation can be represented by

$$S_{i,j} = x_{i,j} + \lambda F(x_{i,j})$$

Where $x_{i,j}$ is the original pixel value at the coordinate (i,j), $F(x_{i,j})$ is the high pass filter, λ is a tuning parameter and if increased it produces more sharpened image.

3. FILTERING

Here median filtering is used in this step to enhance the results of later processing. The pixel intensity is replaced with the median of pixel intensities within a window. Median filtering is a nonlinear operation often used to minimize impulsive or salt and pepper noise that occurs due to random bit error. It is also used to conserve edges in an image.

HISTOGRAM ORIENTED GRADIENT

Histogram of oriented gradient is one among many feature descriptor intended for detecting objects (rain). Rain detection chain is obtained by using the window classifier based on conventional SVM. There are many benefits in representation of HOG. It captures the structure or edge of gradient which is very characteristic of local structure and it does so in a local representation with an uncomplicated controllable degree of invariance to local geometric and photometric transformations: translations or rotations make small variations if they are quite smaller than the local spatial or orientation bin size.

From local regions with 16×16 pixels, the HOG features are separated. And from each of 4×4 local cells, the 8 orientation histograms of edge gradients are calculated. By applying the SOBEL filters the edge gradients and orientations are obtained. Therefore the total amount of HOG features developed into 128 = 8 × (4×4).

In this paper, we derive HOG features from 16×16 local regions. The calculation of orientations and edge gradient are performed at each pixel in this local region. The edge gradients and orientations are obtained by using DOBEL filter. The SOBEL filters are used to compute the orientation θ(x, y) and gradient m(x, y) which are calculated using the y-direction and x-directional gradients dy(x, y) and dx(x, y).

$$m(x,y) = \sqrt{dx(x,y)^2 + dy(x,y)^2}$$

$$\theta(x,y) = \begin{cases} \tan^{-1}\left(\dfrac{dy(x,y)}{dx(x,y)}\right) - \pi \\ \text{if } \mathrm{dx}(x,y) < 0 \; and \; dy(x,y) < 0 \\ \tan^{-1}\left(\dfrac{dy(x,y)}{dx(x,y)}\right) + \pi \\ \text{if } \mathrm{dx}(x,y) < 0 \; and \; dy(x,y) > 0 \\ \tan^{-1}\left(\dfrac{dy(x,y)}{dx(x,y)}\right) otherwise \end{cases}$$

K MEANS CLUSTERING

An efficient algorithm for image clustering is K-Means algorithm. This algorithm has different approach from other form of numerical schemes. K-Means algorithm mainly used to find out the rain droplets using clustering and shape modeling of rain droplets. It is fast when compared to different droplet identification schemes.

For classifying the new data into the existing clusters one can concern the one-nearest neighbor classifier in the centre of the cluster which is obtained by k-means. Hence it is also known as nearest centroid classifier. [11] Proposed, In High Frequency (HF) part, the representation of rainy and non-rainy component, the atoms which consists of dictionary of High Frequency part is divided into two sub-dictionaries. i.e. rainy and non-rainy sub dictionaries. Image gradient is used for removing the most feature of rain atom. In DHF each atom is characterized by the HOG (Histogram of Oriented Gradient) feature descriptor. After removing the each atom by HOG feature, then K-means algorithm is applied for separating all of the atoms in

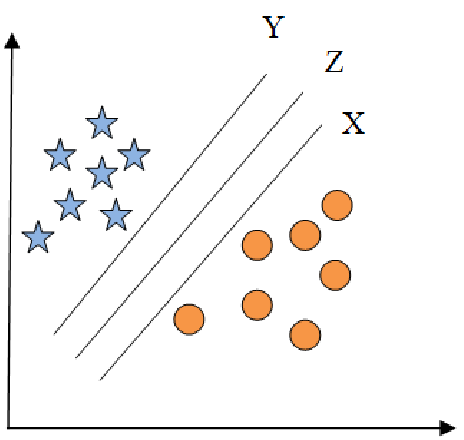
DHF into two sub dictionaries D-1 and D-2 based on Histogram of orientation Gradient feature descriptors for determining which cluster consists of rain atoms and which cluster consists of non-rain atoms.

SVM CLASSIFICATION

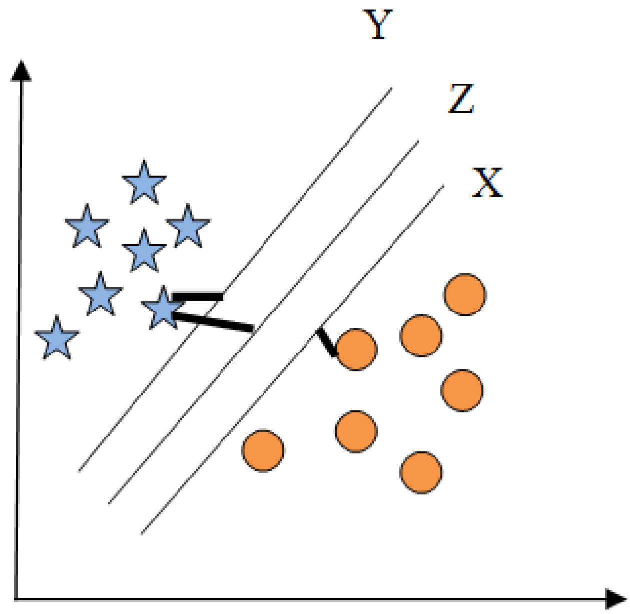
There are numerous classification techniques like k-nearest-neighbor, decision tress, neural networks and Support Vector Machines (SVM). Among all methods, SVM is the fast, simplest and powerful technique. Support Vector Machine is a superintend machine learning algorithm which can be used for both regression and classification task. It separate classes in all types of weather conditions. In this algorithm, we mark each data item as a point in n-dimensional space (where n = number of features) with each feature's value being the value of a specific coordinate. Then, we perform classification by identifying the hyper-plane that distinguishes the two classes correctly. In this principle, a hyper plane is provoked by a linear SVM in the descriptor space D and distinguishes the descriptors by evaluating where the evaluation depends on which side of the hyper plane the descriptor vector (point) lies.

Determine the right hyper-plane:

In this scenario, there are three hyper planes X, Y and Z. We have to locate the right hyper-plane for the classification of star and circle.



Maximizing the distances between nearest neighbor and hyper-plane gives the correct hyper-plane. This is called as Margin and it is explained in the below figure.



Here the distance (margin) for hyper plane Z is high compared to X and Y. Hyper plane with low margin if selected, there is a high chance of miss classification.

PERFORMANCE EVALUATION

The quality of image compression is mainly compared by the two error metrics. They are the Mean Square Error (MSE) and the Peak Signal to Noise Ratio (PSNR). The MSE describes the cumulative squared error between the original and compressed image, whereas measure of the peak error is represented by a PSNR. When the value of MSE decreases, the error rate also decreases and vice versa.

The quality of reconstruction of lossy and lossless compression (e.g., for image compression) is measured by the PSNR value. Here the original data is the signal and the noise is the error established by compression. PSNR is an approximation to human perception of reconstruction quality when compared with compression codes. Even though a higher PSNR actually indicates that the reconstruction is of higher quality level, in some circumstances it may not. PSNR value is most easily determined through the mean squared error. $MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - k(i,j)]^2$

The PSNR is defined as

$$PSNR = 10\log_{10}\left(\frac{MAX^2{}_I}{MSE}\right)$$
$$= 20\log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right)$$

Where,

MAX_I = maximum possible pixel value of the image.

MSE = mean square error.

III. RESULTS

We evaluate the performance of the proposed on rainy image. It effectively removes the rain streaks present in an image because this proposed algorithm uses histogram oriented rain streaks and SVM classifiers and performance is assessed by calculating peak signal to noise ratio values which is found to be moreover same with PSNR value of original value. This proposed algorithm removes most rain streaks in all images originally and at the same time it maintains original textures.

Figure 1. Original input image

Figure 2. Rain removed image

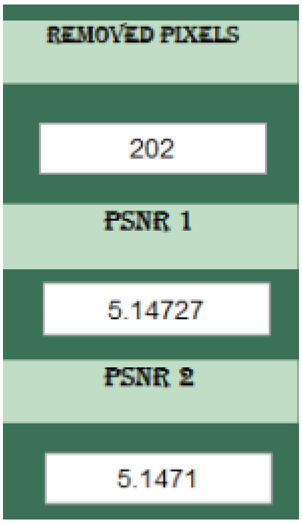


Figure 3. PSNR 1 represents peak signal to noise ratio for input image and PSNR 2 represents peak signal to noise ratio for rain removed image (obtained by the proposed method).

IV. CONCLUSION

In this paper, we proposed an efficient algorithm for single image rain removal using HOG and SVM classifiers. We first improve the results of image for later processing in pre processing step. Then using histogram of oriented gradient, we identified the common rain streaks patterns present in the image. Further using SVM, we separate the classes into rainy part and non rainy part and the rainy part region is removed. Finally, performance is assessed for original image and rain removed image which is found to be moreover equal. A feature research issue is to expand the proposed algorithm to remove rain droplets in video sequences.

V. REFERENCES

[1]. P. C. Barnum, S. Narasimhan, and T. Kanade, "Analysis of rain and snow in frequency space," Int. J. Comput. Vis., vol. 86, nos. 2-3, pp. 256-274, Jan. 2010.

[2]. J. -H. Kim, C. Le, J.-Y. Sim, and C.-S. Kim, "Single-image deraining using an adaptive

nonlocal means filter," in Proc. IEEE Int. Conf. Image Process. (ICIP), pp. 914-917, Sep. 2013 .
[3]. J. Bossu, N. Hautiere, and J.P. Tarel, "Rain or snow detection in image sequences through use of a histogram of orientation of streaks," Int. J. Comput. Vis., vol. 93, no. 3, pp. 348-367, Jul.2011.
[4]. K. Garg and S. K. Nayar, "Detection and removal of rain from videos," in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.(CVPR), pp. I-528-I-535, Jun./Jul. 2004.
[5]. V.Santhaseelam and V. K. Asari, "Utilizing local phase information to remove rain from video," Int. J. Comput. Vis., vol. 112, no. 1, pp. 71-89, Mar. 2015.
[6]. L. W. Kang, C. W. Lin, and Y. H. Fu, " Automatic single-image-based rain streaks removal via image decomposition," IEEE Trans. Image Process., vol. 21, no. 4, pp.1742- 1755, Apr.2012.
[7]. D.-Y. Chen, C.-C. Chen, and L.-W. Kang, "Visual depth guided color image rain streaks removal using sparse coding," IEEE Trans. Circuits Syst. Video Technol., vol.24, no. 8. pp. 1430-1455, Aug. 2014.
[8]. D.-A.Huang, L.-W.Kang, Y.-C.F. Wang, and C.-W.Lin, "Self-learning based image decomposition with applications to single image denoising," IEEE Trans. Multimedia, vol.16, no. 1, pp. 83-93, Jan. 2014.
[9]. Y. Luo, Y. Xu, and H. Ji, "Removing rain from a single image via discriminative sparse coding," in Proc. IEEE Int. Conf. Comput. Vis.(ICCV), , pp.3397-3405, Dec.2015.
[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 Trans. Image processing, vol. 26, no. 6, June 2017.
[11]. N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., San Diego, CA, vol. 1, pp. 886-893, Jun. 2005.