Detection and removal of rain streaks or snowflakes in a single color image

G.Pavani[1], K.Sravan Kumar[2]
[1]PG Student, Dept. of Electronics and Communication Engineering, JNTUACEA, AP, India.
[2]Assistant Professor (Adhoc), Dept. of Electronics and Communication Engineering, JNTUACEA, AP, India.

ABSTRACT

Detection of rain streaks and snow flakes is removing of rain drops and ice crystals present in the single color image. We introduce one new algorithm based on median guided filter, in this proposed method first detection of rain streaks or snow flakes will take place to locate the course pixels. To decompose the image further we are using median guided filter is very effective than the normal guided filter It will smoothens the image so that noise was reduced. Further to retain the image details from high frequency part sparse coding based on online dictionary learning to extract the image details. It is more efficient and takes less time for processing. To detect the people we use histogram of oriented gradient (HOG) with a combination of state vector machine (SVM) in proposed method. To enhance the visual quality of an image histogram stretching is used to reduce the complexity of an existing method. Finally we measure the PSNR and SSIM values to show the difference between existing and the proposed method.

Index terms :- rain removal, median guided image filter, online dictionary learning with sparse coding.

1.INTRODUCTION

Vision systems can be used for tracking, recognition, navigation etc. adverse weather such as rain or snow will effect the quality of the captured images and
removal of these rain is a complex and challenging task.

Previously many algorithms is based on rain removal in videos and recently many are using in still images. In still images they first applied photometric and chromatic constraints for rain detection and then removal filters are used both in spatial and temporal information are exploiting during rain pixel, and a pixel in a fixed location is unlikely covered by rain throughout the entire video and the changes of red(R),green(G),blue(B) values are same as an effected rain pixel proposed in [2]. Based on these characteristics we can extract the image details in our proposed algorithm.

In our work we consider the rain or snow removal from a single color image, we have utilized some common similarities in rain or snow components in sparse coding and in online dictionary learning [4].

With these we can remove rain drops in heavy rain conditions also. In these to enhance the visual quality we use very simple technique that is histogram stretching will reduce the complexity of the existing algorithm.

In our proposed algorithm [1] we use a median guided image filter it will decompose more image details than the normal guided filter [3],so that it is easy to extract with the dictionary learning. previously we have many dictionary learning methods like Prof.M.ELAD [5] that uses K-SVD and it is not so adaptive and also slower in speed. And another we have used HOG with SVM these will detect the people and make us to remove rain from people and get clear image .to increase the performance of an existing system and to decrease the noise and increase the similarity between input and out images we use proposed method detection and removal of rain streaks and snow flakes in a single color image by median guided image filter.

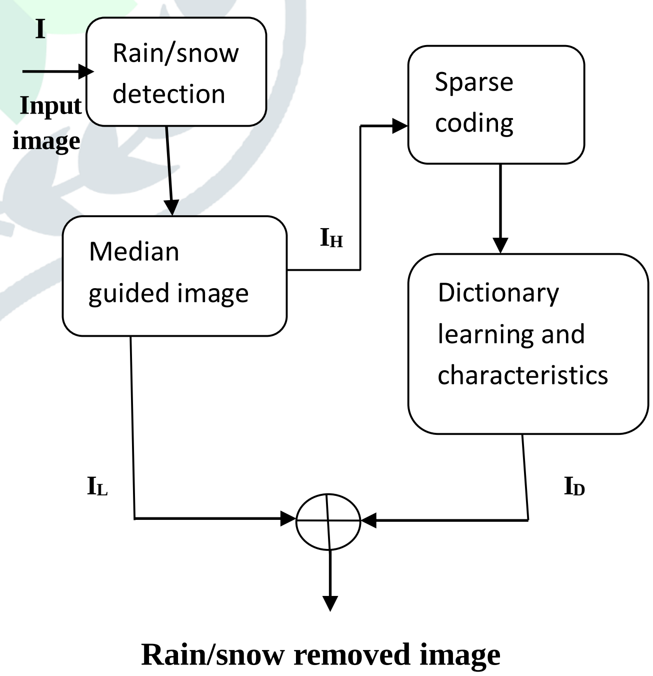


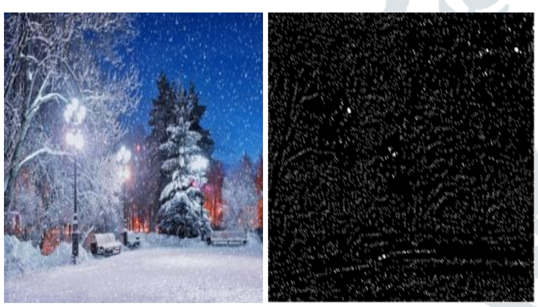
Fig. 1: Flow chart of the proposed method

2.Rain streaks/Snow flakes detection

for input rain /snow image I we can apply directly some low pass filters to decompose the image but it hardly filter out the rain drops or snow flakes so that we use these concept first detection and a median guided filter is used to decompose the image .

so that here we are detection as simple as possible by taking the first similarity by [2] . the rain/snow pixel consists of a large pixel mean value by taking these as consideration .we calculate 4 mean values I_K k=(1,2,3,4) in 7*7 window $w^{(k)}$ with each pixel in image I(i,j) located at every where that is in the middle bottom left, top left,bottom right ,top right and here $I_{M\ (i,j)}$ is located to be binary location mapping . after calculating the pixel mean values .then after locating the rain/snow image I_M is enforced to be zero .otherwise I_M is enforced to be 1. In these high intensities is located as a rain/snow flakes where as low intensities part will be neglected.

Fig(a) Fig(b)



Fig(c) **fig(d)**

Fig 2: the detection results of rain or snow image (a),(c) are the input images of rain or snow and (b) and (d) are detection of rain streaks and snowflakes in a image.

In the next step we need to fill the holes produced by the course locations of I_M and we need to decompose the image as low frequency and high frequency components by filling the holes with non dynamic components mean value then the image was decomposed by using the median guided filter. For the input image I we compute the Hadamard result of I and the parallel area framework I_M as given below by using that product

$$M_I = I \circ I_M$$

Since I_M is the binary holes exists in an image M_I is the location of all detected rain streaks or snow flakes.

3.Median guided image filter

By using the mean values of non rain or snow components we can fill the holes and further we filter out we use these median guided filter . In these first we are taking input image I and we apply the weighted median filter [6] that filter output we are taking as guided image in the guided median filter. these input image is a local linear model between guide image and input image which has to be smoothed. Both GIF and WGIF are the image filters which are used for edge preserving and smoothing of an image. Both image filters can transfer structures from guide image to filtered output. Here λ value is taken as 0.2^2 and radius =0.4

Algorithm 1: Guided Image Filter

Input : The input image I, guide image g, radius r, regularization parameter λ.

Output : filtered output o.

1. $\mathrm{mean_g} = \mathrm{f_{mean}}(\mathrm{g})$
   $\mathrm{mean_I} = \mathrm{f_{mean}}(\mathrm{I})$
   $\mathrm{corr_g} = \mathrm{f_{mean}}(\mathrm{g}.*\ \mathrm{g})$
   $\mathrm{corr_{gI}} = \mathrm{f_{mean}}(\mathrm{g}.*\ \mathrm{I})$
2. $\mathrm{var_g} = \mathrm{corr_g} - \mathrm{mean_g}.*\ \mathrm{mean_g}$
   $\mathrm{cov_{gi}} = \mathrm{corr_{gi}}\mathrm{mean_g}.*\ \mathrm{mean_p}$
3. $\mathrm{a} = \mathrm{cov_{gI}}./(\mathrm{var_g} + \mathrm{eps})$
   $\mathrm{b} = \mathrm{mean_I} - \mathrm{a}.*\ \mathrm{mean_g}$
4. $\mathrm{mean_a} = \mathrm{f_{mean}}(\mathrm{a})$
   $\mathrm{mean_b} = \mathrm{f_{mean}}(\mathrm{b})$
5. $\mathrm{o} = \mathrm{mean_a}.*\ \mathrm{g} + \mathrm{mean_b}$

where f_{mean} is a mean filter.

Here output image O is low frequency part of an image here all most rain streaks or snow flakes are removed but details of the image are lost so we need to retain the image details from high frequency part .in these output image O we subtract from input image then we get the high frequency part of an image.

$$I_H = I - O$$

Here O is low frequency image and I is the output image.

With these high frequency part of an image we need to extract image details by different methods and finally add to the low frequency part of an image. For that we are using sparse coding with dictionary learning [4]

4. Dictionary learning for sparse coding

In these normal methods are not enough to extract the image details we require very quick and fast one that is dictionary learning in an online fashion that was designed by mairal et al.,[4] with these finally we can separate dynamic components and non dynamic components based on common characteristics of rain or snow in a single color image.

Let us consider Y in R^P it forms a sparse coding over a dictionary D in R^{P*q} with q columns here we can assign as atoms, and γ be the regularization parameter it doesn't have any relation between α. $\|\alpha\|_1$ is the sparsity value and it is calculated based on l^p norm these shows the number of non zero values in a matrix, and γ value is taken as 0.15 and sparse coding is performed based on LARS(least angle regression) [7]. And dictionary is computed based on these sparse coding .

Algorithm 2: online dictionary learning
Require: y € R^P , D € $R^{P\wedge q}$, γ

1 Initialization: $D0 \in R^{p\wedge q}$; $A0$=0; $B0$=0;

2 For t=1,…,T

3 Draw Yt from the p(x), compute sparse coding using least angle regression [7]

$$\propto t = \underset{\propto \in R^q}{\operatorname{argmin}} \frac{1}{2} ||Y_t - D_{t-1}\alpha||_2^2 + \gamma||\alpha||_1^1$$

$A_t = A_{t-1} + \boldsymbol{\gamma_t \gamma_t T}$, $B_t = B_{t-1} + \boldsymbol{Y_t \gamma_t T}$

4 Dictionary update

$$Dt = \underset{D \in R^{P*q}}{\operatorname{argmin}} \frac{1}{2} ||Y_i - D \propto_I||_2^2 + \gamma||\alpha|$$

5 end for

From the dictionary got above, powerful segments what's more, non-dynamic parts can be isolated by dictionary particles. To be specific, some dictionary as represent dynamic segments and others for non-dynamic segments. Based on classification proposed by zhang et al [2]. I_H^{ND} is the non rain or snow component in the high frequency part

dictionary particles representing dynamic parts will have a littler total of pixel variance channel fluctuation. In this manner, we figure the aggregate of pixel variance channel fluctuation the sum of pixel color channel variance (N_K) of each dictionary atom or Colum D_t (t=1,2,3…1024) as

$$N_K = \sum_{(i,j) \in D_K} var(q(i,j))$$

where q(i; j) is a shading vector of pixel in Dt. As indicated by these outcomes, the principal characterization is executed as pursues: we pick an edge T1 to distinguish dynamic segments from the other piece of picture. In the event that Sk < T1, Dt represents dynamic segments. When characterized, a sparse coding is connected to get the coefficients y_t of every word reference particle Dt, t =1,2,3….1024;

Then, all decomposed components x_t can be obtained as follows:

$$X_k = D_k * y_k; \quad k = 1,2,\ldots 1024.,$$

Histogram of oriented gradient(HOG)

In these histogram of oriented gradient a feature descriptor used in computer vision and image processing for the purpose of object detection. This technique is like that of edge direction histograms, scale-invariant element change descriptors, and shape settings, yet varies in that it is processed on a thick lattice of consistently dispersed cells and utilizations covering nearby complexity standardization for improved precision.

The initial step of computation in many component finders in image pre-preparing is to guarantee standardized shading and gamma esteems. be that as it may, this progression can be excluded in HOG descriptor calculation, as the following descriptor standardization basically accomplishes a similar outcome. Image pre-handling in this manner gives little effect on execution. Rather, the initial step of count is the calculation of the inclination esteems.

The most widely recognized technique is to apply the 1-D focused, point discrete subsidiary veil in either of the level and vertical headings. In particular, this strategy requires sifting the shading or force information of the image with the accompanying channel portions: They likewise explored different avenues regarding Gaussian smoothing before applying the subordinate cover, yet comparatively found that oversight of any smoothing performed better by

$$[-1\ 0\ 1] \text{ and } [-1\ 0\ 1]^T$$

In these away we are performing the HOG and with a combination of SVM it produces better results of detecting people and these can be used in vector machine we can find more number of people

We need to extract more image details in the high frequency part so that we are using the third characteristics proposed by zhang et al [2].

The variance of the color pixel corresponding to the rain or snow is very small compared to non rain or snow component by taking these as consideration. In these we first calculate the pixel at the location (i,j) in a given image I it is formed as

$$I(i,j) = [R(i,j), G(i,j), B(i,j)]^T$$

Like these we first calculate the mean vector B(i,j) of an image patch centered at (i,j) to form matrix B.

$$B(i,j) = \frac{1}{|W|} \sum_{(m,n)\in W(i,j)} I(m,n)$$

where W is a window focused at (i, j) and |W| principles for the window size. On a basic level, the window size ought to be bigger than the width of a downpour streak or snowflake to guarantee all downpour/snow can be distinguished. In many pictures tried in our work, we found that the width of larger part of downpour/snow is around 3 to 5 pixels.

Therefore, we fix the window size to be 7*7. In request to expel solitary qualities, a middle separating whose size is likewise 7 * 7 is connected on grid B to acquire B^. At that point, for every component in B^, we compute the difference over its three shading channels.

$$V(i,j)=var(B(i,j))$$

To enhance the visual quality of the output image we use histogram stretching by contrast enhancement it is performed by setting the intensity values that represent the bottom 1% (0.01) and the top 1%(0.99) of the range as the adjustment limits and by linear mapping λ is equal to 1

5.Simulation Results

The result is obtained based upon the input and the output images to show how many components of rain streaks or snow flakes removed in an output image

(a) (b)

Fig (3); (a) input rainy image 1 (b) output for the rain removed image1

(c) (d)

Fig(4);(c) shows the input rainy image 2 (d) shows the output of the rain removed image2

(e) (f)

Fig 5 :(e) input snow image (f) output snow removed image

Comparison of PSNR and SSIM with the existing and the proposed methods

	Image 1 Rain image	Image 2 Rain image	Image 3 Snow image
Ying long wang [1] results(existing)	30.0689 0.6543	34.8679 0.7564	23.6980 0.5432
By using Gaussian filter results	50.321 0.1321	49.325 0.2467	42.6754 0.1567
By using median guided filter results(proposed)	45.5946 0.7011	47.0126 0.8654	34.5432 0.6542

6.Conclusion

This proposed method has attempted to solve the rain streaks or snow flakes removing problem from a single color image by utilizing the common similarities of rain or snow we these we can enhanced the visual quality and structural similarity compare to existing method its performance is speed and reduced the complexity of the algorithm. Before we have psnr value for image1 is 30.0689 after it was increased to 45.5946 and ssim also increases from 0.6543 to 0.7011.

6. References

[1] Yinglong Wang, Student Member, IEEE, Shuaicheng Liu, Member, IEEE, Chen Student Member, IEEE, and Bing Zeng, Fellow "A Hierarchical Approach for Rain or Snow Removing in A Single Color Image", DOI 10.1109/TIP.2017.2708502, IEEE Transactions on Image Processing

[2] X. Zhang, H. Li, Y. Qi, W. K. Leow, and T. K. Ng, "Rain removal in video by combining temporal and chromatic properties," IEEE International Conference on Multimedia

and Expo (ICME-2006), pp. 461-464,Toronto, Ontario, Canada, July 9-12, 2006.

[3] K. He, J. Sun, and X. Tang, “Guided image filtering,” IEEE Transaction son Pattern Analysis and Machine Intelligence, vol. 35, no. 6, pp. 1397-1409, June 2013.

[4] J. Mairal, F. Bach, J. Ponce and G. Sapiro, “Online learning for matrix factorization and sparse coding,” Journal of Machine Learning Research, vol. 11, pp. 19-60, Jan. 2010.

[5] M. Aharon, M. Elad and A. M. Bruckstein, “The K-SVD: An algorithm for designing of overcomplete dictionaries for sparse representation,”IEEE Transactions on Signal Processing, vol. 54, no. 11, pp. 4311-4322,Nov. 2006.

[6] Z. G. Li, J. H. Zheng, Z. J. Zhu, W. Yao, and S. Q. Wu, “Weighted guided image filtering,” IEEE Trans. On Image Processing, vil. 24, no. 1, pp. 120-129, jan. 2015.

[7] B. Efron, T. Hastie, I. Johnstone and R. Tibshirani, “Least angle regression,” Annals of Statistics, vol. 32, no. 2, pp. 407-499, 2004.