



## Removing Rain Component from A Color Image

Priti S.Gokhale, Prof P.S. Malge  
Department of Electronics Engineering,  
WIT,Solapur,India

### ABSTRACT

It is well-known that a bad weather, e.g. haze, rain, or snow affects severely the quality of the captured images or videos. Also raindrops adhered to a glass window or camera lens can severely affect the visibility of background scene and degrade the image quality, which consequently degrades the performance of many image processing and computer vision system algorithms. These algorithms are used in various applications such as object detection, tracking, recognition, and surveillance also in navigation. Rain removal from a video or a single image has been an active research topic over the past decade. Today, it continues to draw attentions in outdoor vision systems (e.g. surveillance) where the ultimate goal is to produce a clear and clean image or video. The most critical task here is to separate rain component from the other part. For that purpose, we are proposing an efficient algorithm to remove rain from a color image.

**Key words:** generative and discriminative networks (GAN), LSTM

### I. INTRODUCTION

An image captured in the rainy time, so it is mainly or lightly covered with bright streaks. Also raindrops attached to a glass window, windscreen or lens can hamper the visibility of background scene and degrades an image. The impact of rain streaks/drops on images and video is often undesirable. The effect of rain can also severely affect the performance of outdoor vision systems. Effective methods for removing rain component are needed for a wide range of practical applications. However when an object structure and orientation is similar to that of rain component i.e. rain streaks/drops .It is hard to simultaneously remove rain and preserve the structure.

First, rain is semi-transparent. Because of this, the objects will not be occluded completely but some blurring may appear. Second, pixels with different intensities will be affected by rain differently. When the pixel's primary intensity is relatively low, rain will enhance its intensity. When a high intensity pixel is affected by rain, its intensity will become lower. This is to say that rain-affected pixels tend to have the same intensity because the reflection of rain is dominating under this scenario. Images which are taken from an outdoor condition, bad weather like rain confuse

human viewers also brings difficulty to image processing and also the performance of computer vision algorithms decreases.

For removal of rain component from an image the Generative Adversarial Network (GAN) can be used. It is a model that takes the noise and creates an image. The generative network attempts to produce an image which is free from raindrops. The generator Adversarial network consists of generator network which is having two sub-networks in it one is attentive recurrent network and the other is contextual auto encoder. The purpose of the attentive recurrent network is to find regions in the input image that need to get attention. These regions are mainly the raindrop regions and their surrounding structures that are necessary for the contextual auto encoder to focus on, so that it can generate better local image restoration. The purpose of our Contextual Auto-encoder is to generate an image that is free from raindrops.

### II .LITERATURE REVIEW

Rui Qian1[1] proposed an efficient algorithm to remove rain component from a color image. Here they have used an Attentive Generative Adversarial Network with adversarial training.Two important steps in generator network followed are:1)Creation of attention map using ResNet and LSTM 2)Contextual Autoencoder. They have used the attentive recurrent network is to find regions in the input image that need to get attention. These regions are mainly the raindrop regions and their surrounding structures that are necessary for the contextual auto encoder to focus on, so that it can generate better local image restoration

Xueyang et al [2] proposed new deep network architecture for removing rain streaks from individual images based on the deep convolutional neural network (CNN).They proposed a deep detail network to directly reduce the mapping range from input to output. To further improve the de-rained result, they used a priori image domain knowledge by focusing on high frequency detail during training, which removes background interference and focuses the model on the structure of rain in images.

Wang et al [3],Archana H[8] and Arti R. Waghchaure[9] proposed an efficient algorithm to remove rain or snow from a single color image using image decomposition and dictionary learning. At first, a combination of rain/snow

detection and a guided filter is used to decompose the input image into a complementary pair: 1) the low-frequency part which is free from rain/snow and 2) the high-frequency part that contains details of the image details with rain/snow component. Then, they focus on the extraction of image's details from the high-frequency part designing a 3-layer hierarchical scheme to remove rain component properly.

He zhang [4] proposed an efficient algorithm for image de-raining using a conditional generative adversarial network. They investigated the use of generative modeling for synthesizing de-rained image from given input rainy image for improved stability in training and reducing artifacts introduced by GANs in the output images they proposed the use of new refined loss function in the GAN optimization framework. In addition a multiscale discriminator is proposed the new refined loss function in the GAN optimization framework a multiscale discriminator is proposed a leverage features from different scales to determine whether the de-rained image is real or fake. They conducted extensive experiments on synthetic and real world data set to evaluate the performance of the proposed method. Furthermore experimental results are evaluated on object detection using faster RCNN demonstrated significant improvements in detection performance when IDCGAN method is used as a preprocessing step.

Xianguo Li [5] proposed a novel Attentive Generative Adversarial Network for Waterdrop detection and removal of Rubber Conveyor Belt Image. Here they have increased the number of visual attentive-recurrent network layers, skip connections are added on the automatic encoder.

HuangxingLin[6] proposed a adjacent aggregation network for single image raindrop removal. This network employs an encoder-decoder structure with a new adjacent aggregation operation to effectively extract informative features.

UlkuUzun [7] proposed a Cycle-Spinning GAN for Raindrop Removal from Images. Here Cycle spinning has been shown to increase the perceptual quality in terms of full-reference metrics. It also increases the object detection performance, justifying its usefulness when used as a pre-processing operation under adverse weather conditions in intelligent transportation systems and outdoor surveillance systems.

### III.METHODOLOGY

The pipeline of the proposed rain removal algorithm can be as shown in Fig. 3.1. Principally, the degradation occurs because raindrop regions contain different imageries from those without raindrops. Unlike non-raindrop regions, raindrop regions are formed by rays of reflected light from a wider environment, due to the shape of raindrops, which is similar to that of a fish-eye lens. Moreover, in most cases, the focus of the camera is on the background scene, making the appearance of raindrops blur. In this algorithm, visibility

degradation problem can be addressed. Given an image impaired by raindrops, the goal is to remove the raindrops and produce a clean and clear image. The method is fully automatic. We consider that it will benefit image processing and computer vision applications, particularly for those suffering from raindrops, dirt, or similar artifacts.

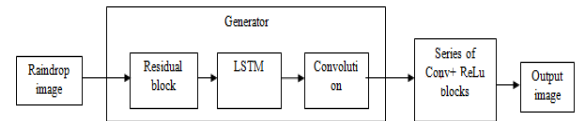


Fig.1

Architecture of Proposed System

The raindrop-removal problem is intractable, since first the regions which are occluded by raindrops are not given. Second, the information about the background scene of the occluded regions is completely lost for most part. The problem gets worse when the raindrops are relatively large and distributed densely across the input image. To resolve the problem, we can use a generative adversarial network, which generates the outputs. To deal with the complexity of the problem, generative network first attempts to produce an attention map. This attention map is the most critical part of the network, since it will guide the next process in the generative network to focus on raindrop regions. This map is produced by a recurrent network consisting of deep residual networks (ResNets) combined with a convolutional LSTM and a few standard convolutional layers. This is attentive-recurrent network.

The second part of this algorithm is generative network is an autoencoder, which takes both the input image and the attention map as the input. To obtain wider contextual information, in the decoder side of the autoencoder, we can apply multi-scale losses. Each of these losses compares the difference between the output of the convolutional layers and the corresponding ground truth that has been downscaled accordingly. The input of the convolutional layers is the features from a decoder layer. Besides these losses, for the final output of the autoencoder, we can apply a perceptual loss to obtain a more global similarity to the ground truth. This final output is also the output of generative network.

Overall, besides introducing a novel method of raindrop removal, other main contribution is the injection of the attention map into generative networks, which works effectively in removing raindrops. Fig.1 shows the overall architecture of proposed network. Following the idea of generative adversarial networks, Given an input image degraded by raindrops, the generative network attempts to produce an image and free from raindrops.

As shown in Fig.1, generative network consists of two sub-networks: an attentive-recurrent network and a contextual auto encoder. The purpose of the attentiverecurrent network is to find regions in the input

image that need to get attention. These regions are mainly the raindrop regions and their surrounding structures that are necessary for the contextual auto encoder to focus on, so that it can generate better local image restoration, Attentive-Recurrent Network: Visual attention models have been applied to localizing targeted regions in an image to capture features of the regions. The idea has been utilized for visual recognition and classification. In a similar way, we consider visual attention to be important for generating raindrop-free background images, since it allows the network to know where the removal/restoration should be focused on. As shown in architecture in Fig.1, we can employ a recurrent network to generate visual attention. Each block (of each time step) in recurrent network comprises of five layers of ResNet that help extract features from the input image and the mask of the previous block, a convolutional LSTM unit and convolutional layers for generating the 2D attention maps.

#### IV ALGORITHM DETAILS

##### 4.1 Convolutional LSTM

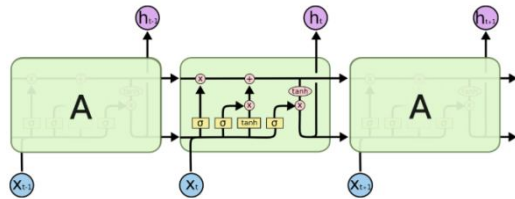


Fig.2 A LSTM cell

Data collected over successive periods of time are characterized as a Time Series. In such cases, an interesting approach is to use a model based on **LSTM** (Long Short Term Memory), a Recurrent Neural Network architecture. In this kind of architecture, the model passes the previous hidden state to the next step of the sequence. Therefore holding information on previous data the network has seen before and using it to make decisions. In other words, the data order is extremely important

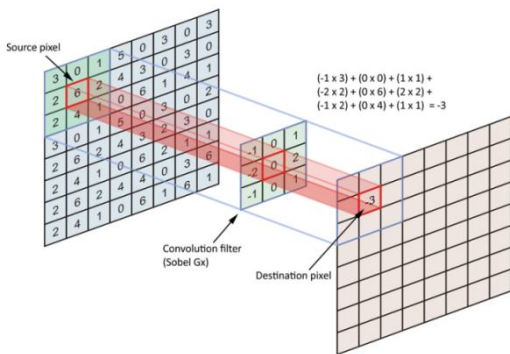


Fig.3 Convolution of an image with one filter

In our case, sequential images, one approach is using **ConvLSTM** layers. It is a Recurrent layer, just like the

LSTM, but internal matrix multiplications are exchanged with convolution operations. As a result, the data that flows through the ConvLSTM cells keeps the input dimension (3D in our case) instead of being just a 1D vector with features

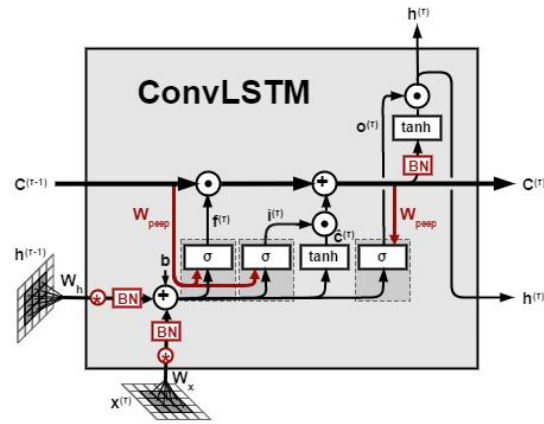


Fig.4. A ConvLSTM cell

Here image passes through the convolutions layers and its result is a set flattened to a 1D array with the obtained features. When repeating this process to all images in the time set, the result is a set of features over time, and this is the LSTM layer input.

#### Architecture

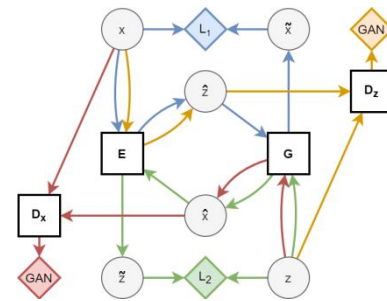


Fig.5 The high-level AEGAN architecture. Networks are shown as boxes, values as circles, and losses as diamonds. Colours represent combined networks, where red is a regular image-generating GAN, yellow is a GAN for producing latent vectors, blue is an image autoencoder, and green is a latent vector autoencoder..

AEGAN is a four-network model comprising of two GANs and two autoencoders, illustrated in Figure 1, and is a generalization of the CycleGAN technique for unpaired image-to-image translation where one of the image domains is replaced with random noise. In short, we train two networks to translate between sample space X and latent space Z, and we train another two networks to discriminate between real and fake samples and latent vectors.

### Networks (Squares):

- $G$  is the generator network. It takes a latent vector  $z$  as input and returns an image  $x$  as output.
- $E$  is the encoder network. It takes an image  $x$  as input and returns a latent vector  $z$  as output.
- $D_x$  is the image discriminator network. It takes an image  $x$  as input and returns the probability that  $x$  was drawn from the original dataset as output.
- $D_z$  is the latent discriminator network. It takes a latent vector  $z$  as input and returns the probability that  $z$  was drawn from the latent distribution as output.

### Values (Circles):

- $x$ : genuine samples from the original dataset. This is a bit ambiguous, because in some places I use  $x$  to mean any value in the domain  $X$ . Sorry about that.
- $z$ : genuine samples from the latent-generating distribution (random noise).
- $x_{hat}$ : samples produced by  $G$  given a real random vector, i.e.  $x_{hat}=G(z)$ .
- $z_{hat}$ : vectors produced by  $E$  given a real sample, i.e.  $z_{hat}=E(x)$ .

### Training

The AEGAN is trained in the same way as a GAN, alternately updating the generators ( $G$  and  $E$ ) and the discriminators ( $D_x$  and  $D_z$ ). The AEGAN loss function is slightly more complex than the typical GAN loss, however. It consists of four adversarial components

$$L_{GAN_{\hat{x}}}(G, D_x) = E_{x \sim p_{data}}[\log D_x(x)] + E_{z \sim p(z)}[\log(1 - D_x(G(z)))]$$

$$L_{GAN_{\hat{x}}}(G, E, D_x) = E_{x \sim p_{data}}[\log D_x(x)] + E_{z \sim p_{data}}[\log(1 - D_x(G(E(x))))]$$

$$L_{GAN_{\hat{x}}}(E, D_z) = E_{x \sim p_{data}}[\log D_z(z)] + E_{z \sim p(z)}[\log(1 - D_z(E(x)))]$$

$$L_{GAN_{\hat{x}}}(G, E, D_z) = E_{x \sim p(z)}[\log D_z(z)] + E_{z \sim p(z)}[\log(1 - D_z(E(G(z))))]$$

The adversarial components of the AEGAN loss.

$$L_{\tau}(G, E) = \lambda_{\tau x} E_{x \sim p_{data}}[\|G(E(x)) - x\|_1] + \lambda_{\tau z} E_{z \sim p(z)}[\|E(G(z)) - z\|_2]$$

The reconstruction components of the AEGAN loss. The  $\lambda$ s are hyperparameters that control the relative weights of the reconstruction components. Which, all summed, form the AEGAN loss.  $E$  and  $G$  try to minimize this loss while  $D_x$  and  $D_z$  try to maximize it. If you don't care for the math, the intuition is simple:

1.  $G$  tries to trick  $D_x$  into believing the generated samples  $x_{hat}$  and the autoencoded samples  $x_{tilde}$  are real, while  $D_x$  tries to distinguish those from the real samples  $x$ .
2.  $E$  tries to trick  $D_z$  into believing the generated samples  $z_{hat}$  and the autoencoded samples  $z_{tilde}$  are real, while  $D_z$  tries to distinguish those from the real samples  $z$ .
3.  $G$  and  $E$  have to work together so that the autoencoded samples  $G(E(x))=x_{tilde}$  are similar to the original  $x$ , and that the autoencoded samples  $E(G(z))=z_{tilde}$  are similar to the original  $z$ .

## V .RESULTS



Fig 6. Input image with rain component

Fig.7 Output image after applying GAN which is free from rain component

### 5.1 Result Analysis:rrR

#### Result Analysis:

Table 1. Showing quantitative comparison on the basis of PSNR value.

PSNR value in dB	Before processing	After Processing
		31

From table no.1. the psnr values for image before and after processing is shown.After processing we got higher psnr value indicating quality of reconstructed image is better than input image.

## VI.CONCLUSION

Here we are proposing a single-image based rain removal method which will remove rain components from the image and will produce clean and clear image. The method utilizes a generative adversarial network (GAN), where the generative network produces the attention map via an attentive-recurrent network and applies this map along with the input image to generate a rain component-free image through a contextual auto-encoder.

## REFERENCES

- [1] Rui Qian<sup>1</sup> , Robby T. Tan<sup>2,3\*</sup> , Wenhan Yang<sup>1</sup> , Jiajun Su<sup>1</sup> , and Jiaying Liu<sup>1</sup>, “Attentive Generative Adversarial Network for Raindrop Removal from A Single Image.”
- [2] Xueyang Fu<sup>1</sup> Jiabin Huang<sup>1</sup> Delu Zeng<sup>2</sup> Yue Huang<sup>1</sup> Xinghao Ding<sup>1\*</sup> John Paisley, “Removing rain from single images via a deep detail network”
- [3] Wang, Y., Liu, S., Chen, C., & Zeng, B. “A Hierarchical Approach for Rain or Snow Removing in a Single Color Image.” IEEE Transactions on Image Processing, 26(8), 3936–3.

- [4] He Zhang, Vishwanath and Vishal M. Patel, "Image De-raining Using a Conditional Generative Adversarial Network".
- [5] Xianguo Li , Zongpeng Liu , Bin Li,XinxinFeng,XiaoLiu,andDebao Zhou, "A Novel Attentive Generative Adversarial Network for Waterdrop Detection and Removal of Rubber Conveyor Belt Image".
- [6] Huangxing Lin, Changxing Jing, Yue Huang and Xinghao Ding, " A2Net: Adjacent Aggregation Networks for Image Raindrop Removal".
- [7] U. Uzun, A. Temizel, "Cycle-Spinning GAN for Raindrop Removal from Images".
- [8] Archana H., 2 Dr.M. Priya 3 Dr.A. Muthukumaravel, "Rain Or Snow Removing In A Single Color Image- A Novel Algorithm", International Journal of Pure and Applied Mathematics Volume 118 No. 20 2018, 923-933.
- [9] Arti R. Waghchaure1 , Rupali B. Bachhav2 , Shital S Bhalerao, "A Hierarchical approach for rain removing in single color image", Vol-4 Issue-3 2018 IJARIIIE-ISSN(O)-2395-4396.