Using Deep Learning to improve the quality of rain-accumulated images

by

P Madhu Chandra

Roll. No. 2018IMT-049

विश्वजीवनामृतं ज्ञानम्

ABV–INDIAN INSTITUTE OF INFORMATION TECHNOLOGY AND MANAGEMENT GWALIOR (M.P.), INDIA

Introduction

- Rain is one of the most common weather and phenomena that we see in our daily life.
- Heavy rainfall usually cause atmospheric vielings like smog or mist type layers.These atmospheric vielings with overlapping rain streaks in several different directions usually obstruct the background image.
- During these heavy rains our visual capacity usually degrades and we won't be able to see background images clearly as they were obstructed and blurred.

Introduction

- Now a days computer vision based applications are used everywhere.
- Heavy rainfall can also affect the various computer vision based systems.
- Due to the rain streaks and atmospheric vielings in heavy rains these computer vision based systems wont be able to capture images clearly and will reduce the vision of such systems up to a certain extent.
- Thus, various computer vision based systems like Security surveillance,Traffic Systems,Object detection and automated driving cars will usually like to fail in such heavy rains.

Introduction

Figure 1: Image effected by Rain.

Figure 2: Image without Rain

Rain Removal

- During weather conditions like rainfall and smog the cameras used in these computer vision based systems wont be able to capture images and videos properly. Due to this, the vision of those usually degrades in such atmospheric conditions and it have adverse effects on industries and fields where these computer vision based applications are being used.
- This removal process is usually classified into two approaches one approach is based on model and other approach is based on data. We can also call these approaches as Data oriented and Model ori- ented approaches. Traditional methods normally use model oriented techniques. Model oriented techniques usually depend on the proper utilization of available information like spatial information and properties of background and characteristics of rain streak etc,.

Rain Removal in videos

- Rain removal in videos is very much different when compared to single images. Because in videos we will try to make use of the information available in whole video.
- In videos, we will try to establish the relationship between sequences of frames and make use of that information while removing the rain from videos.
- In the video based techniques most traditional methods use the information available in time domain, frequency domain and some methods also use low rank and sparse based techniques.

Rain Removal in Images

- Rain removal in single images is far more different from videos and is difficult also.
- In single images we don't have temporal information available. Due to this it is very difficult to remove rain in single images.
- Rain removal in single image methods have also been divided like model oriented and data oriented techniques. Most of the traditional methods are model oriented because in early days there is a lot of difficulty to have large datasets and it is further more difficult to have datasets of rain images.
- Most traditional methods are often model driven and they tried to used methods like dictionary learning,sparse representational and filter based methods. Most these methods have developed a model where they will remove rain streaks in single stage. But usually these rain streaks are of different directions and usually overlap on one another creating adverse effects on captured Images.

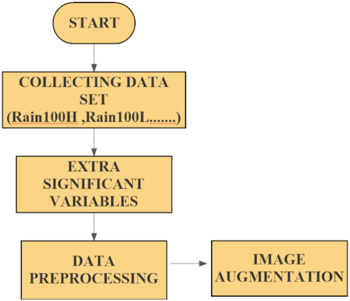
Objectives

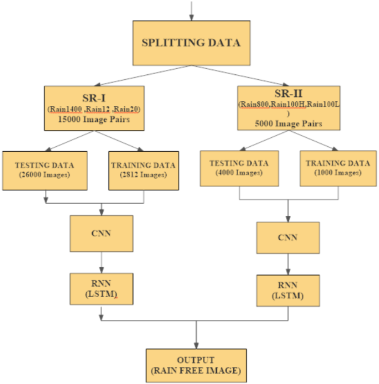
- To develop a deep learning model for improving the quality of rain affected Images.
- To develop a model which will remove rain streaks in several stages and make use of the available contextual information in later stages.
- To develop a model which can recover Images affected by complex rain streaks even in heavy rainfalls by making use of Recurrent Neural Network architecture.
- To make comparison of the proposed model and state of art methods by using performance metrics used in previous papers.

Dataset

- One of the main problem m of rain removal is proper availability of datasets. So we have taken available synthesized datatsets Rain12, Rain20, Rain100H, Rain100L Rain800 and Rain1400 .Using these datasets we have proposed to new datasets for our model called Synthesized Rain Image I (SRI I) dataset and Synthesized Rain Image II (SRI II) dataset.
- For Synthesized rain image I(SRI I) dataset we have used Rain1400, Rain12 and Rain20 datasets.Similarly Synthesized Rain Image II (SRI II) dataset was created by using Rain800, Rain100H and Rain100L datasets

SYSTEM ARCHITECTURE





- The most commonly and widely used rain model for removal of rain streak layers in a Image is:

$$I = B + R \tag{1}$$

where I represents our captured Image, R represents the rain streak layers formed on our captured Image due to rainfall and atmospheric vielings, B represents the background scene of Image without rain streaks and atmospheric vielings

- This Eq(1) model usually assumes that Rain Streak layer R as a single layer.But in reality,it is very different because during heavy rains the rain drops are usually of several different shapes falling in different directions which results in creation of several rain streak layers which are overlapping on each other.
- So, considering these all different rain streak layers we can change enhance rain model to

$$I = B + \sum_{i=1}^{n} R^i \tag{2}$$

Methodology

- During heavy rains another problem is that due to accumulation of several rains streaks in different directions in air will cause attenuation and scattering.Due to this it will lead to increase in differences of brightness in rain streak layers and makes images hazed and blurry. So considering this we can further generalise our rain model to:

$$I = B + \sum_{i=1}^{n} \alpha_i R^i \tag{3}$$

Methodology

- In Eq.(3) R^i represents different rain streak layers where R^i stands for **i-th** rain streak layer, α_i represents rain streak layer brightness and **n** represents the maximum number of such different rain streaks.
- So,by using the Eq.(3) I am proposing to build a deep learning model which can detect the rain streak layers R_i and remove them in several different stages.
- Our Deep learning model mainly consists of two important parts.They are
  - Dilated Convolutions and SE Blocks.
  - Recurrent Structure.

- In our model we want to use more contextualised information so that we can use that information in further stages.

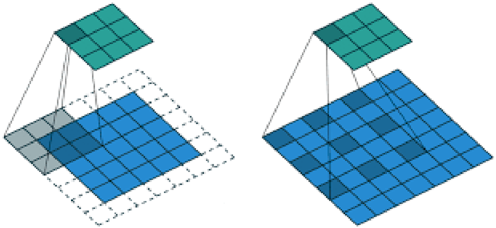


Figure 3: Standard Convolution vs Dilated Convolution

- So we have used dilated convolution which will help us to have more contextual information with larger receptive field and lesser parameters, thus it also helps in increasing performance of our model.

Methodology

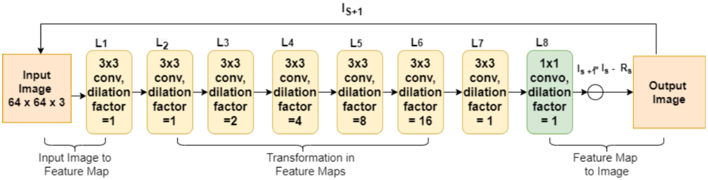


Figure 4: Model Architecture

- In above model fig, we have 8 layers and it will take images of size $64 \times 64 \times 3$ as its input, where at first layer L1 we will give our image as input and we perform 3×3 convolution to convert our image space into feature space.
- From Layer L2 to L7 we have used 3×3 convolution. Instead of using standard convolution we will use dilated convolutions here use to acquire larger receptive field. So, to incorporate that dilation operation we will use dilation factor of 1,2,4,8,.. from Layer L2 to Layer L6
- In the layers L2 to L7 we will transform our feature map. At layer L7 and L8 we will perform standard convolution with 3×3 and 1×1 convolutions respectively. Here at layer L7 and L8 we will transform our feature space to Image Space.

- At each layer we will get a feature map after performing the convolution, the channels of each feature maps can be considered as embedding of each streak layer R^i. In our base model Eq-(3) we have used different α_i values for different rain streaks.
- After every layer we have pooling layer where we used average pooling. Similarly we have ReLu layer after every layer except last which will add non-linearity. At last layer we will get rain streak and we will separate this rain streak from our image to get background without rainfall. We will compare that obtained background image with original background image and try to optimise our model by optimising loss function.

- At each layer we will get a feature map after performing the convolution, the channels of each feature maps can be considered as embedding of each streak layer R^{i}. In our base model Eq-(3) we have used different α_i values for different rain streaks.
- We want to remove rain streak layers in several different stages instead of removing it in single stage we want to remove different rain streak layer at each stage and for that purpose we have used recurrent neural network in our model.

- We have used the following process:

$$I_1 = I, \tag{4}$$

$$R_i = f_c(I_s), \tag{5}$$

$$B_i = I_i - R_i, \tag{6}$$

$$I_{i+1} = I_i - R_i, \tag{7}$$

- where i stands for current state, R_i stands for rain streak layer of current state, B_i stands for background image and I_{i+1} stands for output background image of the i-th stage, which we will use as input for next stage.

- Here at the end of every stage we will get background image as output by removing rain streak layer.We will use that output as the input to the next stage in our recurrent network. Similarly, we will use this for every stage and these all stages will work together as we are giving the output of one stage as input to its next stage. Thus, these separate stages will make use of the available contextual information in future stages because rain streak layers will usually decrease after every stage.
- In order to make proper use of available previous information we will incorporate our recurrent neural network with different memory unit architectures like LSTM, MultiplicativeLSTM and GRU.

- The loss function used for our model is standard squared error function and described in the equation 8.

$$L(R_i,R) = \sum_{i=1}^{n} \left|\left| \sum_{k}^{n} R_i - R \right|\right|^2 \tag{8}$$

Here R_i represents all the rain streak layers produced at different stages after rain removal and R is original rain streaks.

Methodology

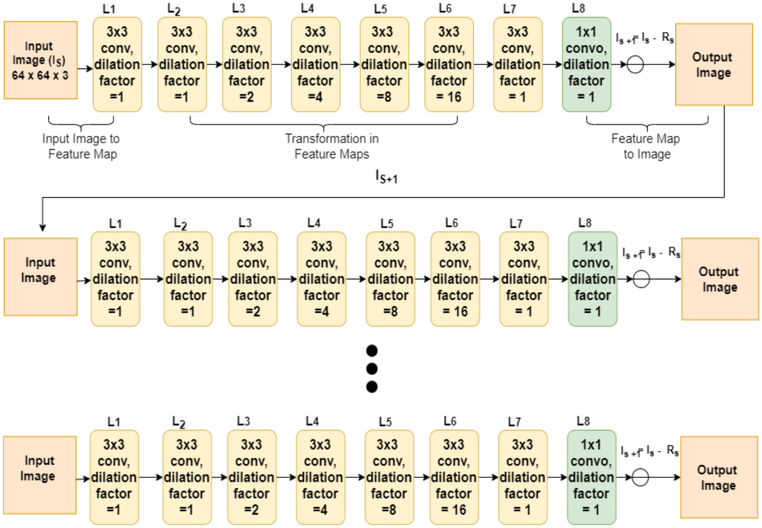


Figure 5: Recurrent Model Architecture

- **Training Settings** In the training process, we used our training dataset prepared from Synthesized Rain Image I (SRI I) dataset and Synthesized Rain Image II (SRI II) dataset.Our Model has benn trained on Google collab, we have used Nvidia GTX 1060 GPU. We have taken a batch size of 128. For optimization purposes, we have taken adam algorithm and used a initial learning of rate 0.003.

Results

- Structure similarity index (SSIM) is calculated by using pytorch builtin ssim function. To this builtin ssim function we will provide original background image O and our output background image at i-th stage O_i. Similarly like psnr we will take average ssim to model by computing ssim at every stage.

$$SSIM = ssim\left(O_i - O\right) \tag{9}$$

Results

- Below table shows results of our proposed deep learning model and other standard methods on the SRI I and SRI II datasets.

Model	SRI I		SRI II	
	PSNR	SSIM	PSNR	SSIM
DDN	32.394	0.905	26.017	0.864
LP	33.213	0.911	27.432	0.871
JORDER	**33.615**	0.919	**28.142**	0.881
PreNet	33.393	**0.921**	27.987	**0.898**
Model with RNN	32.261	0.893	26.014	0.861
Model with GRU	**33.582**	0.918	**28.053**	0.879
Model with LSTM	33.375	**0.923**	27.865	**0.902**
Model with Multiplicative LSTM	**33.645**	**0.922**	**28.217**	**0.898**

Table 1: Comparison Table

Results

- Below we have shown the results of our model and compared them with initial input images and original background images so that we can see how our model performed.

Figure 6: Rained Image

Figure 7: Predicted Image

Figure 8: Background Image

Results

Figure 9: Rained Image

Figure 10: Predicted Image

Figure 11: Background Image

Results

Figure 12: Rained Image

Figure 13: Predicted Image

Figure 14: Background Image

Contributions

- We proposed two new datasets SRI I and SRI II which contains synthesized rain image pairs.
- We proposed a deep learning model which can remove rain streaks from single image.
- We proposed a model which removes rain in multiple stages rather than treating it as single step process.
- We proposed a model which makes propose use of contextual information at different stages and also establishes relationship between those stages by using recurrent neural network

Concluding remarks

- To make proper use of available contextual information during feature extraction process we have used dilated convolutions which not only increases receptive field size but also decreases number of parameters.
- We also used recurrent architectures with memory units like LSTM, GRU and Multiplicative LSTM which establishes relation between different stages and increases performance.
- Instead of using standard batch normalization for reducing internal covariate shift during training we have used squeeze and excitation blocks which improved performance and decreased required computation memory as we dont have to store copy of feature maps while training.
- Results show that our model with multplicative LSTM has performed very well for complex rain images and have better PSNR and SSIM value when compared to existing methods standard methods.

Future Works

- Considerable space for change exists. The first thing is obvious: that we have used synthetic rain dataset for training and testing purposes. Compared to synthetic dataset if we have real world rain dataset it will be very useful and greatly increase the performance because real world rain dataset will have atmospheric vielings and effects which cant be replicated properly in synthesized datasets

Nathan Brewer and Nianjun Liu.
Using the shape characteristics of rain to identify and remove rain from video.
In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 451–458. Springer, 2008.

Wan-Joo Park and Kwae-Hi Lee.
Rain removal using kalman filter in video.
In *2008 International Conference on Smart Manufacturing Application*, pages 494–497. IEEE, 2008.

David Eigen, Dilip Krishnan, and Rob Fergus.
Restoring an image taken through a window covered with dirt or rain.
In *Proceedings of the IEEE international conference on computer vision*, pages 633–640, 2013.

Jie Chen, Cheen-Hau Tan, Junhui Hou, Lap-Pui Chau, and He Li.
Robust video content alignment and compensation for rain removal in a cnn framework.
In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6286–6295, 2018.

Rui Qian, Robby T Tan, Wenhan Yang, Jiajun Su, and Jiaying Liu.
Attentive generative adversarial network for raindrop removal from a single image.
In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2482–2491, 2018.

Dongwei Ren, Wangmeng Zuo, Qinghua Hu, Pengfei Zhu, and Deyu Meng.
Progressive image deraining networks: a better and simpler baseline.
In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3937–3946, 2019.

Thank You