Image Restoration of Night Time Hazy Images

Rishabh Chaurasia

M. Tech Computer Science

Supervisor: Dr. Sarbani Palit
CVPRU
Indian Statistical Institute
Kolkata

This project is submitted as a dissertation for the requirements of the Master of Technology programme in Computer Science at Indian Statistical Institute.

Acknowledgement

I would like to thank my dissertation supervisor Dr. Sarbani Palit for agreeing to guide me and for helping me to undertake work in the topic. Without his continuous guide and support this wouldn't have been possible.

I am also very much thankful to Harsh Da, Shashank B. and all my lab mates and seniors of the CVPR unit of Indian Statistical Institute , Kolkata for helping me though out the project with their valuable time and suggestions.

Rishabh Chaurasia

MTech(CS), 2nd year

Roll No.- CS1822

ISI, Kolkata

September, 2020

Abstract

Image restoration is the operation of taking a corrupt/noisy image and estimating the clean, original image. There are various source of noise in image like fog, haze, glow, scattering of light, motion blur and camera mis-focus.Image restoration involve various methods to restore images to original form like image dehazing, image denoising, image super-resolution. Images acquired by a visual system are seriously degraded under hazy and foggy weather, which will affect the detection, tracking, and recognition of targets. Thus, restoring the true scene from such a foggy image is of significance.But we focus on Nighttime scenes,which however commonly include visible lights sources with varying colors. These light sources also often introduce noticeable amounts of glow that is not present in daytime images. So in this work we illustrate you with night and day time image dehazing models and our approach of image denoising which also comes under image restoration. In night time image dehazing we illustrate you by CNN model used in various methods and in Image denoising by Auto-encoder .

Contents

1 Introduction

Night time haze is a traditional atmospheric phenomenon where dust, smoke,fog, glow from various light source and other dry particles obscure the clarity of the atmosphere. Haze causes issues in the area of terrestrial photography, where the light penetration of dense atmosphere may be necessary to image distant subjects and while driving on roads traffic lights with other vehicles light which generate glow cause difficulty in view of driver. This results in the visual effect of a loss of contrast in the subject, due to the effect of light scattering through the haze particles and multiple reflected light rays scatter out to all directions other than the line of sight and attenuate the screen reflection with distance. For these reasons, haze removal is desired in daily life and also computer vision applications.

The night haze removal is a severely ill-posed problem [7] especially due to the presence of various visible light sources with varying colors and non-uniform illumination. Most existing dehazing methods use models that are formulated to describe haze in daytime. Daytime models assume a single uniform light color attributed to a light source not directly visible in the scene. Nighttime scenes, however commonly include visible lights sources with varying colors. These light sources also often introduce noticeable amounts of glow that is not present in daytime haze.

Recent years, numerous daytime haze removal models have been proposed [11] to address the hazy image visibility enhancement. The key to their success mostly relies on the correct estimation of various image priors, and atmospheric light. The standard haze model illustrates the hazing process as a linear combination of airlight and direct transmission. The direct transmission represents the scene reflection whose intensity reduces by the scattering out process. On the other hand, the airlight represents intensity resulted from the scattering in process of the light sources present on the surrounding atmosphere. Hence the transmission light conveys a fraction of the scene reflection and reaches the camera. Again most of the haze models assume that the atmospheric light present in the input image can be estimated by the brightest image region with a strong approximation. After estimating the atmospheric light, the daytime haze methods calculate the transmission light by using various cues such as dark channel, local contrast , image fusion and statistical independence between the albedo and shading. The main implementation differences between these methods are due to various cues incorporated with transmission light estimations.

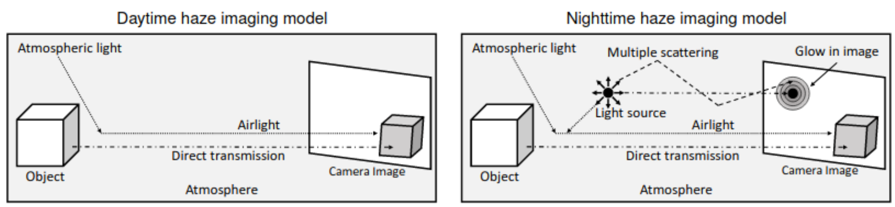

Figure 1: difference in day and night time dehazing model [5]

But the effectiveness of these daytime haze methods is not well demonstrated to correct the night time scenes. The main reason might be that the daytime haze model priors do not hold well for most nighttime scenes. The night scenes commonly have numerous and diverse colored light sources, e.g. building, vehicle, street lights etc. which results in non-uniform illumination. These illuminations not only make ambient light estimation inaccurate but also cause some image priors to become invalid. Besides that, night sources introduce more brightness to existing atmospheric light, boost intensity unrealistically and cause the prominent glow to the scene (Fig. 1).The atmospheric light on night scenes are not assumed to be globally uniform and cannot be calculated from the brightest region of night image. As a result, the atmospheric light approximation can differ significantly from that of the brightest intensity in a scene. Consequently, if we normalize the input image w.r.t brightest region intensity then it would cause a prominent color shift in the input image.

2 Related Work

In this section we will tell you about how single image dehazing in day time works and their proposed deep neural network solution. Then night time single image dehazing, how it is different from day time image dehazing. And then night time image dehazing deep neural network solution with contributing idea.

2.1 Atmospheric scattering model (day time)

To describe the formation of a hazy image, the atmospheric scattering model is first proposed by McCartney [1], which is further developed by Narasimhan and Nayar [2], [3].The atmospheric scattering model can be formally written as

$$I(x) = J(x)t(x) + \alpha(1 - t(x)) \tag{1}$$

where I(x)is the observed hazy image, J(x)is the real scene to be recovered, t(x) is the medium transmission ,α is the global atmospheric light, and x indexes pixels in the observed hazy image I. There are three unknowns in equation (1), and the real scene J(x) can be recovered after α and t(x) are estimated.
The medium transmission map t(x)describes the light portion that is not scattered and reaches the camera. t(x) is defined as

$$t(x) = e^{-\beta d(x)} \tag{2}$$

where d(x) is the distance from the scene point to the camera, and β is the scattering coefficient of the atmosphere. Equation (2) suggests that when d(x) goes to infinity, t(x) approaches zero. Together with equation (1) we have

$$\alpha = I(x), d(x) \to Inf \tag{3}$$

In practical imaging of a distance view, d(x) cannot be infinity, but rather be a long distance that gives a very low transmission t_0 . Instead of relying on equation (3) to get the global atmospheric light α , it is more stably estimated based on the following rule

$$\alpha = \max_{y \in \{x | t(x) \leq t_0\}} I(y) \tag{4}$$

There are only two unknown parameters in Eq.(1). If we can obtain transmission t and atmospheric light value α , then restored image J will be obtained. Estimation of the atmosphere light α by selecting 0.1% darkest pixels in a transmission map t(x). Among these pixels, the one with the highest intensity in the corresponding hazy image I is selected as the atmospheric light. To recover a clean scene(i.e., to achieve haze removal), it is the key to estimate an accurate medium transmission map.

2.2 CNN model for Transmission Map

This section approach is from 'Single Image Dehazing via Multi-Scale Convolutional Neural Networks' [4] use a multi-scale CNN to learn effective features from hazy images for the estimation of scene transmission map. The scene transmission map is first estimated by a coarse-scale network and then refined by a fine-scale network. To learn the network, they develop a benchmark dataset consisting of hazy images and their transmission maps by synthesizing clean images and ground truth depth maps from the NYU Depth database [13] . Although the network is trained with the synthetic dataset, they show the learned multi-scale CNN is able to dehaze real-world hazy images well.

NYU Depth database contain clean images and their corresponding ground truth depth map. So given a clear image J and the ground truth depth d, they synthesize a hazy image using the physical model (1). Then generate the random atmospheric light α = [k, k, k], where k $\in$ [0.7, 1.0], and sample three random $\beta \in$ [0.5, 1.5] for every image. We do not use small $\beta \in$ (0, 0.5) because it would lead to thin haze and boost noise . On the other hand, we do not use large $\beta \in$ (1.5, Inf) as the resulting transmission maps are close to zero.

-

Pretrained model available so We run this model on ISI server and get the results. Just need to add VGG weights and code was in Matlab.

Figure below show that algorithmic view :(A in image is α here)

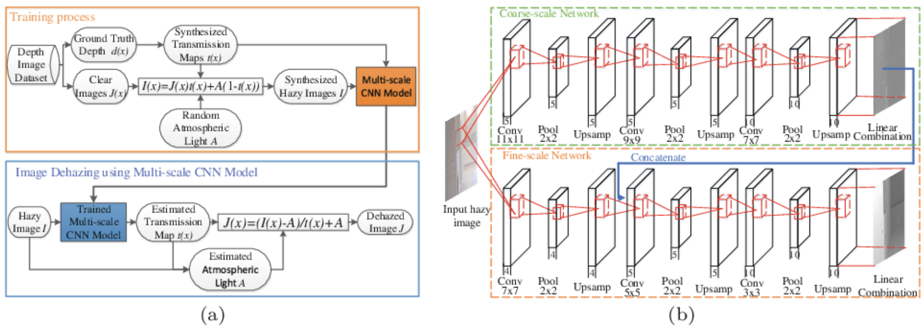

Figure 2: (a)MSCNN algorithm and (b) MSCNN Model to generate transmission map [4]

Haze removal equation :

$$J(x) = \frac{I(x) - \alpha}{\max\{0.1, t(x)\}} + \alpha \tag{5}$$

after getting transmission map from above algorithmic figure this equation used to generate clear image in MSCNN.

2.3 Nighttime dehazing Method

In this section we give equation for night time dehazing from Nighttime Haze Removal with Glow and Multiple Light Colors [5] which will different from day time image dehazing. Nighttime scenes typically have active light sources that can generate glow when the presence of particles in the atmosphere is substantial. Glow describe as light from sources that gets scattered multiple times and reaches the observer from different directions. They model this glow as an atmospheric point spread function *(APSF)*. So now equation (1) modified as

$$I(x) = J(x)t(x) + \alpha(1 - t(x)) + \alpha_g \times ASPF \tag{6}$$

where α_g is the active light sources, which the intensity is convolved with the atmosphere point spread function, $APSF$. This model no longer globally uniform, and thus can change at different locations. This is because various colors from different light sources can contribute to the atmospheric light as a result of the scattering process. So make Night Time Haze removal model standard for glow decomposition they rewrite as

$$I(x) = R(x) + G(x) \tag{7}$$

where R = J(x)t(x) + α(1 - t(x)) and G(x) = $\alpha_g \times ASPF$
So R symbolise as the nighttime haze image, and G as glow image. In this form, decoupling glow becomes a layer separation problem, with the two layers: R and G, which need to be estimated from a single input image I.

2.4 Nighttime Haze and Glow removal using CNN

They evaluated proposed CNN algorithm on both synthetic and natural night hazy images and compared results with various night haze methods. To train DeGlow-DeHaze model [6] we generated synthesized hazy image I(x) dataset and their transmission maps t(x) .
Dilated CNN model first transforms input night hazy images into a feature space through a series of convolutions (Figure 3). The dilated convolution weighs pixels with a step size equal to the dilated factor (DF), and it increases the receptive field without losing resolution. Three dilated paths P1, P2, and P3 are shown on the (Fig.2) and each consist of three convolutions with a kernel size of 3x3. The above paths uses different DF's i.e. DF = 1, DF = 2, DF = 3 and obtain their expanded receptive field as 7x7, 13x13, and 17x17 respectively.

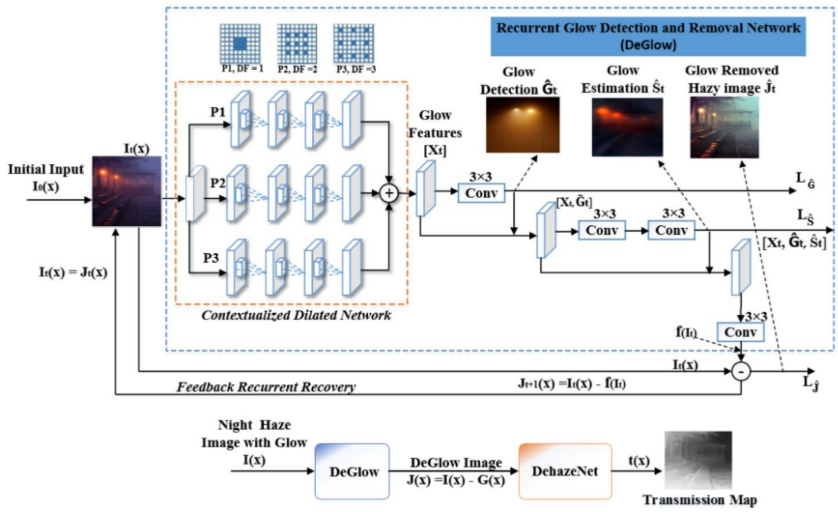

Figure 3: Architecture of Dilated CNN for deglow-dehaze [6]

3 Proposed Approach

In practical situations, nighttime hazy images may be further affected by the presence of noise due to availability of light and camera quality. Hence Equation 6 may be further modifies as

$$I(x) = J(x)t(x) + \alpha(1 - t(x)) + \alpha_g \times ASPF + \epsilon(x) \tag{8}$$

where $\epsilon(x)$ denotes a spatially varying noise component. We propose a two-step solution to the dehazing of noisy, nighttime hazy images:

- Denoising of the nighttime hazy images using an Autoencoder network.
- Dehazing of the denoised images so obtained from the Autoencoder network by further using a Generative Adversarial Network.

3.1 Image Denoising

One of the fundamental challenges in the field of image processing and computer vision is image denoising, where the underlying goal is to estimate the original image by suppressing noise from a noise-contaminated version of the image.

3.1.1 Theory

Image denoising is removing noise from noisy image.
Our proposed approach is using Autoencoder with MNIST fashion dataset. Autoencoders are a class of neural networks used for feature selection and extraction, also called dimensionality reduction. Fashion-MNIST is a dataset of Zalando's article images—consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image, associated with a label from 10 classes.
Equation used in this experiment to add noise in image :

$$noisy\_image = clean\_image + noise\_factor \times clean\_image\_shape \tag{9}$$

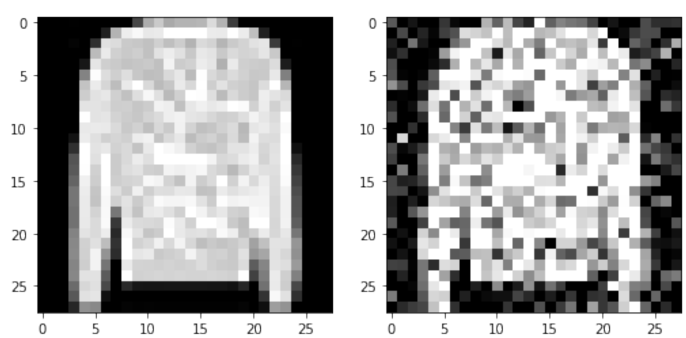

Figure 4: Image in MNIST dataset (left) and image after adding noise (right)

3.1.2 Autoencoder

Autoencoders are a specific type of feed-forward neural networks. In autoencoder, the input is the same as the output. They compress the input into a lower-dimensional code and then reconstruct the output from this representation [9, 10].

An autoencoder consists of 3 components: encoder, code and decoder. The encoder compresses the input and produces the code, the decoder then reconstructs the input from this code.

First the input passes through the encoder, which is a fully-connected Neural Network, to produce the code. The decoder, which has the similar structure,then produces the output only using the code. The goal is to get an output identical with the input [10].

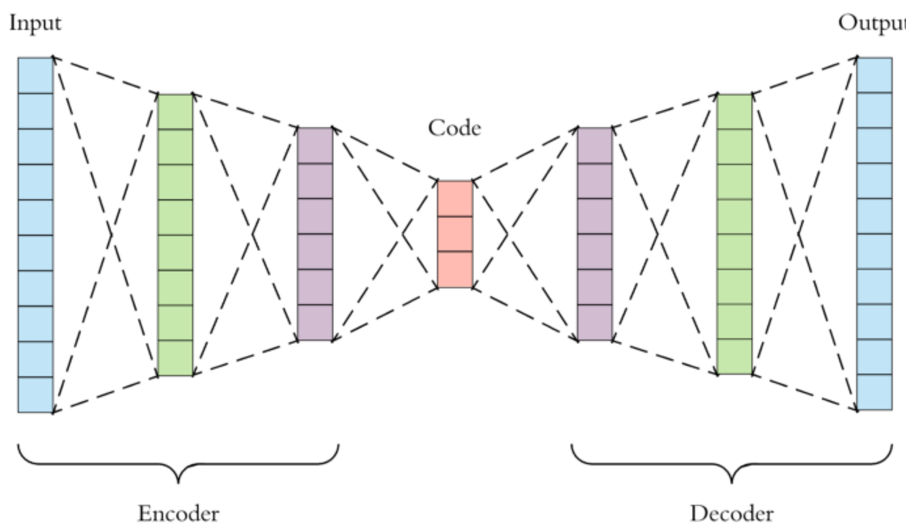

Figure 5: Schematic diagram of Autoencoder[10].

3.2 Dehazing of denoised images

The block diagram shown below is that of a Generative Adversarial Network whose input is the denoised, hazy image. The function of the generator is to produce an improved image from an input hazy image. So, it should not only preserve structure and detail information of an input image but also remove haze as much as possible.

The Discriminator is used to distinguish whether a generated image is real or fake. It tries to compare the output of the generator to a real image . Depending upon Discriminator's output, the GAN trains itself through back-propagation.The Discriminator checks the quality after getting hazy image from generation as compared to ground truth and tries to minimize the loss.

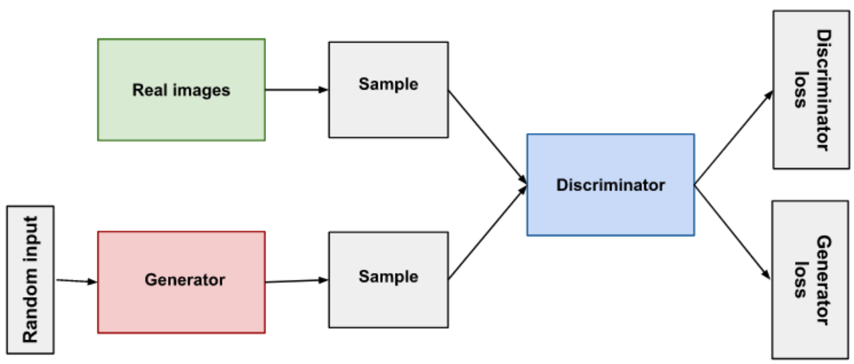

Figure 6: Schematic diagram of a GAN [8].

In training process of GAN , The discriminator outputs a value D(x) indicating the chance that x is a real image. Our objective is to be maximize the chance to recognize real images as real and generated images as fake. i.e. the maximum likelihood of the observed data. The objective function for Discriminator is:

$$\max_{\theta_D} V(D) = \mathbb{E}_{x\sim p_{data}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))] \tag{10}$$

The Generator wants to maximize the value D(G(z)). Its objective function is:

$$\min_{\theta_G} V(G) = \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))] \tag{11}$$

Combining equation 1 and 2, we get the objective function of GAN as:

$$\min_{\theta_G} \max_{\theta_D} V(G, D) = \mathbb{E}_{x \sim p_{data}}[\log D(x)] + \mathbb{E}_{z \sim p_z}[\log(1 - D(G(z)))] \tag{12}$$

The two network play a **min-max** game . The discriminator tries to maximize the objective function $V(G, D)$ by only controlling it's parameter θ_D. On the other hand, the generator tries to minimize the objective function $V(G, D)$ by only controlling it's parameter θ_G.

4 Implementation and Results

4.1 Denoising

4.1.1 Architecture

An Autoencoder is used as Network . We used Convolutional Neural Network(CNN) in both encoder and decoder of the Autoencoder. With kernal size 3×3 but number of filter changed in each layer. We used normalised data so Sigmoid activation used in last layer of decoder.We used stride s = 2 in each layer except the code layer where stride s = 1

Figure given below shows the architecture of encoder.

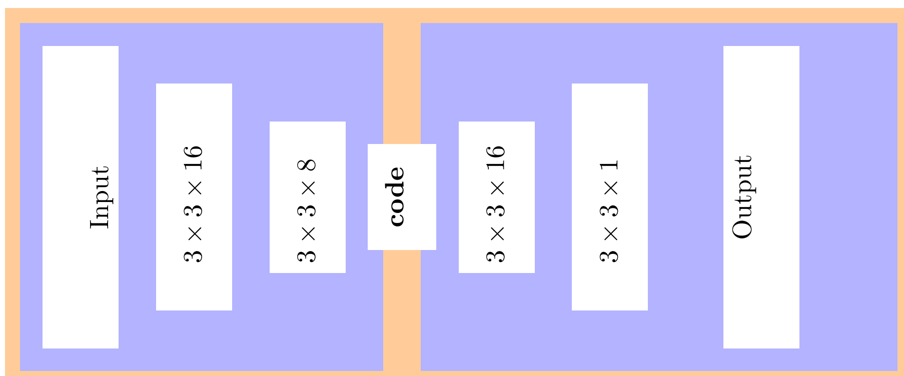

Figure 7: Autoencoder architecture. the tuple inside a box indicate kernel size(row × column) × number of filters.

Output Shape: Summary of Autoencoder The architecture of network

After each convolution layer we did not used Batch Normalization, and Sigmoid is used as activation function in last decoder layer. We didn't used MaxPooling after convolution layers. and padding same for each layer .

We trained our model on **MNIST** dataset which contains 28×28, single channel, 60,000 training images. The model is implemented using Tensorflow Framework 2.x and Python 3.x and run on **Google Colab**.

Table 1: Autoencoder Summary

Layer Name	**Output Shape**	**Param**
Conv2D	(None, 14, 14, 16)	160
Conv2D_1	(None, 7, 7, 8)	1160
Conv2D_2	(None, 7, 7, 8)	584
Conv2D transpose	(None, 14, 14, 16)	1168
Conv2D transpose_1	(None, 28, 28, 1)	145

Total params: 3,217
Trainable params: 3,217
Non-trainable params: 0

Hyper-parameter: Random Noise of noise factor = 0.3 is added to input images in dataset and make two set noisy and clean dataset. For test images noise factor = 0.1 used. We used Adam optimizer with learning rate=0.001. The size of minibatch is set to 200 and number of epochs is set to 10. And loss function used was Binary Cross Entropy [12].

4.1.2 Result

MNIST dataset consist of 60,000 fashion images from label '0' = T-shirt/top to label '9' = Ankle boot used for training. Each of the ten categories of fashion wears is taken as novel data.

Autoencoder after 10 epochs of training generate loss: 0.2975 and val_loss: 0.2840, Test dataset contain 10000 images . After evaluating trained model it generate test accuracy of 0.284 which is okay. Below show you result in grid contain noisy and denoised images

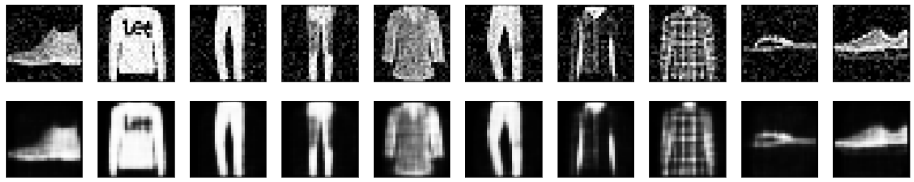

Figure 8: first row show noisy images and second row show corresponding denoising image

4.1.3 Analysis of Denoising results

From the above result we can see Autoencoder able to denoise noisy image to some extent but it also generate some artifacts if you compare it to clean images. Test accuracy is also not high so architecture need to have better feature extraction in convolution layer.

If this experiment done on real world images then it would not give good result as real world images have different type of noise. This experiment is only for grey scale images.

4.2 Dehazing using GAN

We used the NYU Depth dataset for experimentation. The generator consisted of encoding and decoding modules. The encoding process in the generator consisted of convolution, batch normalization and LeakyReLU. The decoding process was composed of deconvolution, batch normalization and ReLU. The Discriminator consisted of convolution, batch normalization and LeakyReLU and last layer sigmoid activation function. The network parameters were selected as given below:

- Size of input and output image from generator was set to $256 \times 256 \times 3$.
- The size of input in discriminator is $256 \times 256 \times 6$ and size of output is $256 \times 256 \times 1$
- The learning rate was set to be 2×10^{-3}. The update ratio of Generator G and Discriminator D is set to be 1.

We intended to use Adam optimization method to train our network. Implementation of the proposed approach was planned to be attempted using Tensorflow, Keras and GoogleColab.

5 Conclusion and Scope for future work

In this dissertation, we have proposed a two-step approach for obtaining dehazed nighttime images. The first stage employs an Autoencoder for denoising Though we are able to obtain denoised images,the performance of the autoencoder is not satisfactory and needs to be improved further.

The second stage inputs the denoised images to a GAN network with parameters as indicated. Unfortunately, we were unable to complete its implementation due to the outbreak of COVID-19, the ensuing lockdown and unavailablility of computing resources and network connectivity from my home.

Besides implementation and modification for better performance of our proposed networks, we also plan to utilize GAN to map daytime images to corresponding nighttime images.

References

[1] E. J. McCartney. *Optics of the atmosphere: Scattering by molecules and particles*

[2] S. G. Narasimhan and S. K. Nayar *Vision in bad weather*

[3] ZS. G. Narasimhan and S. K. Nayar *Contrast restoration of weather degraded images*

[4] Wenqi Ren , Si Liu , Hua Zhang, Jinshan Pan , Xiaochun Cao and Ming-Hsuan Yang *Single Image Dehazing via Multi-Scale Convolutional Neural Networks* http://www.eccv2016.org/files/posters/P-1B-14.pdf

[5] Yu Li, Robby T. Tan, Michael S. Brown *Nighttime Haze Removal with Glow and Multiple Light Colors* https://www.cv-foundation.org/openaccess/content_iccv_2015/papers/Li_Nighttime_Haze_Removal_ICCV_2015_paper.pdf

[6] Shiba Kuanar, K.R. Rao, Dwarikanath Mahapatra, Monalisa Bilas *Night Time Haze and Glow Removal using Deep Dilated Convolutional Network* https://arxiv.org/pdf/1902.00855.pdf

[7] Bolun Cai, Xiangmin Xu, Kui Jia, Chunmei Qing, Dacheng Tao *DehazeNet: An End-to-End System for Single Image Haze Removal* https://arxiv.org/abs/1601.07661

[8] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio *Generative Adversarial Networks* https://arxiv.org/pdf/1406.2661.pdf

[9] Jake Krajewski *Autoencoder neural networks: what and how?* Article

[10] Arden Dertat *Applied Deep Learning - Part 3: Autoencoders* Article

[11] Yong Xu, Jie Wen, Lunke Fei AND Zheng Zhang *Review of Video and Image Defogging Algorithms and Related Studies on Image Restoration*

and Enhancement https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7365412

[12] Daniel Godoy *Understanding binary cross-entropy / log loss: a visual explanation* Article

[13] Nathan Silberman *NYU Depth Dataset V2* Article