# Image Dehazing for Object Recognition using Faster RCNN

Bhanu Teja Nalla, Teena Sharma, Nishchal K. Verma, S. R. Sahoo
Dept. of Electrical Engineering, Indian Institute of Technology Kanpur, India - 208016
Email: bhanutn@iitk.ac.in, tee.shar6@gmail.com, nishchal@iitk.ac.in, srsahoo@iitk.ac.in

***Abstract*—Object recognition in hazy conditions is quite difficult due to illumination variance. The challenge arises in finding out features from such images. Herein, we have proposed the method to deal with such images. The input image needs to be dehazed before applying the recognition algorithms. On the other hand, dehazing a non-hazy image makes it dark resulting in loss of features. Hence, a decision is to be made whether or not the image should be dehazed before recognition. Also, for a very dense haze, even dehazing doesn't help in object recognition. In order to tackle this issue, this paper presents a novel method to quantitatively estimate the amount of haze in the image– also termed as haze degree– using dark channel prior of the input images. We compared our values with the existing method using FRIDA dataset. The estimated haze degree is used to decide whether input image need to be dehazed or not. We use DehazeNet and Faster RCNN for dehazing and recognition, respectively. We test our method on real time hazy images to set a threshold on haze degree to classify the image as light, moderate or densely hazed. We used the Static Scenes dataset from Color Hazy Images for Comparison (CHIC) database to obtain the threshold values.**

***Keywords*—Haze Degree, DehazeNet, Faster RCNN, Dark Channel Prior, DCP**

## I. Introduction

Artificial object recognition is gaining pace in today's world with increasing automation. Also, the accuracy levels are reaching close to that of humans'. Yet, Recognition in hazy conditions remain a bottleneck with recognition algorithms performing poorly for hazy images. Hence, there is a need for a mechanism to automatically detect haze, and dehaze it if necessary before recognition. The image at hand could be non hazy, moderately hazed or heavily hazed. In non hazy case, there is no need to dehaze and can be operated on directly for object recognition. In the latter cases, the performance degrades with increasing haze. In heavily hazed conditions, the dehazing doesn't work as the image is entirely covered with white patches leaving us no clue about it's contents. Hence, we propose a novel method to detect the degree of haze in an image so as to decide whether the image can be dehazed and if yes, is there a need to dehaze. The flow diagram is shown in Fig. 1.

We use the atmospheric scattering model as in [3] to model the images with haze. The atmospheric scattering model is given by,

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

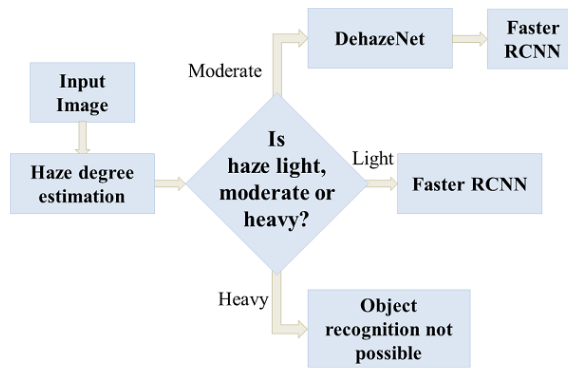


Fig. 1: Proposed Algorithm

where, $I(x)$ is resultant hazy image, $J(x)$ is the image without haze, $A$ is Atmospheric Light, $t(x)$ is transmission, $t(x) = e^{-\beta d(x)}$, $d(x)$ is the depth at pixel $x$, $\beta$ is scattering coefficient.

Our objective is to provide the haze free image $J(x)$ for object recognition network in least possible time. In most of the haze removal algorithms [1][3][4], the transmission $t(x)$ is found out using various properties of hazy images like dark channel prior (DCP), lower contrast, color attenuation, etc. Then, using transmission, image is dehazed. In this paper, we used DCP [3] to find the haze degree of the image which is explained in the further sections. For the purpose of dehazing, we have used CNN based DehazeNet [1] proposed by Bolun Cai *et al.* due to its superior performance pertaining to proximity of output images to real world images. Also, DehazeNet is fairly fast compared to other dehazing algorithms based on computer vision techniques. For the sake of object recognition, we use Faster RCNN [2] due to its superior speed.

Object recognition has made lot of progress in recent years, thanks to the advent of convolutional neural networks (CNN) and artificial neural networks (ANN) [23]. There have been many variants of detection networks [22], each with improved accuracy compared to the previous. Faster RCNN [2] proposed by Shaoqing Ren *et al.* is one of the faster object recognition algorithms available. It is developed based on regional based CNN (RCNN) [8] and Fast-RCNN [11] by Ross Girshick *et al.* which increased the speed of recognition by manifold.

Kaiming He *et al.* proposed the concept of DCP [3] based on the observations that non-hazy images have pixels close to the zero in at least one of the color channel in a local patch.

978-1-5090-6014-6/18/$31.00 ©2018 IEEE

They used the DCP obtained as $t(x)$ and atmospheric light, $A$ is obtained from DCP itself. However, the soft matting process used to refine the Transmission map is time consuming. It is overcome by Zhang *et al.* [10] by using guided filter. Fattal [4] used Independent Component Analysis (ICA) algorithm for single image dehazing. Liu *et al.* [5] used CNN to dehaze the images. They break the image into a number of superpixels and send them through CNN to estimate the transmission. Young-Sik Shin *et al.* [6] used CNN to find Ambient light which greatly aids dehazing in underwater images. Based on hazy image properties, Bolun Cai *et al.* [1] proposed a new architecture called DehazeNet which incorporates maxout network [12] proposed by Goodfellow *et al.* and BReLU.

Jun Mao *et al.* [14] proposed algorithm for quantifying the haze based on maximum and minimum values of the image. Chi Wei Wang *et al.* [15] proposed an algorithm for detection and haze degree estimation using DCP and contrast in the image. They use features obtained in these in SVM to obtain haze degree.

The paper is divided as follows. Section II explains about proposed method. Section III describes about the Datasets used in this paper. Section IV describes about the Experiments conducted and Results obtained. Section V has Conclusion and Future Work.

## II. Proposed Method

The proposed architecture consists of three parts - (1) Haze Degree Estimation, (2) Image Dehazing Network and (3) Object Recognition Network. Haze degree estimation measures the amount of haze in input image. If the image is found to be hazy, then it is first dehazed, before sending it to object recognition network. The second part is image dehazing network which is inspired from DehazeNet. The third part consists of object recognition network inspired from Faster RCNN. These parts are explained below.

### A. Haze Degree Estimation

In this section, we discuss about the method proposed in [14] before going into proposed method. First, calculate the minimum and maximum values of the the input image across the color channels at each pixel and then calculate the average of maximum and minimum.

$$p^I(x) = \max_{c\in(r,g,b)}(I_c(x)) \tag{2}$$

$$q^I(x) = \min_{c\in(r,g,b)}(I_c(x)) \tag{3}$$

$$r^I(x) = p^I(x) - q^I(x) \tag{4}$$

$$p = \frac{\Sigma p^I(x)}{N};\ q = \frac{\Sigma q^I(x)}{N} \tag{5}$$

$$r = p - q \tag{6}$$

where $N$ is number of pixels in the image.
Calculate Atmospheric light,$A$ using

$$A = \lambda max(p^I(x)) + (1-\lambda)p \tag{7}$$

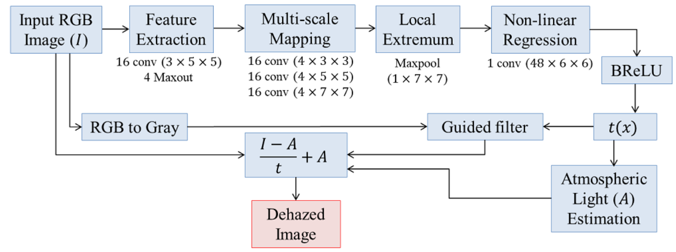


Fig. 2: Dehazing Network

where $\lambda = 1/3$. The haze degree $\omega$ is calculated using

$$\omega = e^{-0.5(\mu t_1 + \nu t_2 + \sigma)};\ t_1 = \frac{A-q}{A};\ t_2 = \frac{r}{A} \tag{8}$$

where $\mu = 5.1$, $\nu = 2.9$, $\sigma = 0.2461$ which are heuristically determined by authors.

Coming to proposed method, it has been observed in the previous work that hazy images have less contrast and the corresponding Dark Channel Prior (DCP) is relatively brighter. Contrarily, for a clear image without haze, the DCP is very dark and pixel values are close to zero, unless there exist any white regions or sky regions in the image. We use this property of DCP to find the amount of Haze in an image. First, we find the DCP of the image which is given by,

$$J_{Dark}(x) = \min_{c\in(r,g,b),x\in\omega}(I_c(x)) \tag{9}$$

where, $\omega$ is a patch of size $3 \times 3$.

It is vectorized and sorted in ascending order. In order to avoid the white patches and sky regions (if any) we take the mean of darkest 1% pixels(the first 1% of pixels arranged in ascending order) which will serve as Haze degree. We observed that, for an image without haze, the darkest 1% of the pixels will be very close to 0 (less than 1 in most of the cases). In case of hazy images, almost all the pixels in the DCP carry some brightness, leading the pixel value to be greater than 0. A threshold, $thresh$ is set to classify the images as hazy or non-hazy.

Calculating the DCP requires exhaustively calculating the minimum of a local patch at each and every pixel. In order to avoid that, we randomly select sufficiently large patches from the image and calculate the minimum of three channels at those pixels. These values are sorted and mean of 1% of smallest values is calculated. Our experiments show that this value is sufficiently close to the value calculated using exhaustively.The algorithm is shown in Algorithm1.

### B. DehazeNet

The dehazing network consists of 4 layers namely feature extraction layer, multi-scale mapping layer, local extremum layer and non-linear regression layer. Input image passes through these layers which finally gives out the transmission estimate $t(x)$ required to remove the haze content. The block diagram of DehazeNet is shown in Fig. 2. The layers are explained as follows.

```
Result: Haze Degree
numPatches ← height * width/100;
i ← 1;
J ← [];
while i < numPatches do
    randomly select a patch ω of size 3 × 3;
    minimum ← min_{c∈(r,g,b),x∈ω}(I_c(x));
    J ←[J,minimum];
end
J_sort ← sort(J);
numValues ← numPatches/100;
d_haze ← mean(J_sort[1 : numValues]);
if (d_haze < thresh) then
    Faster RCNN;
else
    DehazeNet;
    Faster RCNN;
end
```

**Algorithm 1:** Algorithm to calculate Haze Degree ($d_{haze}$)

*1) Feature Extraction:* The feature extraction layer performs convolution operation on the input image giving out 16 layers. Maxout operation is performed on these layers to give out 4 layers. Under maxout operation, 16 layers are divided into 4 sets of 4 layers each. In each set, the 4 layers are stacked upon each other and maximum of the corresponding pixels is taken. Thus feature extraction gives out 4 layers. The maxout operation is as shown.

$$F^i(x) = \max_{j\in[1,4]} f^{i,j}(x) \tag{10}$$

$$f^{i,j} = W^{i,j} \circledast I + b^{i,j}; i \in [1, 4]$$

where, $W, b$ are Weights and bias respectively, $F, f$ are output and input layers respectively.

*2) Multi-scale Mapping:* The layers obtained in previous section are convolved with kernels of size $3 \times 3$, $5 \times 5$ and $7 \times 7$. This is performed to achieve cale invariance. The layers are padded accordingly so that all the output layers have same size. There exist 16 kernels of each size thus taking the number of output layers to 48 which are all concatenated.

*3) Local Extremum:* This layer is akin to Max Pooling layer where the maximum element in a local patch is taken. The only difference is that this operation is carried out at every pixel so as to maintain the size of the layer. Each max pooling patch is of size $7 \times 7$.

$$F(x) = \max_{y\in\omega} f(y) \tag{11}$$

where, $\omega$ is $7 \times 7$ patch centered at $x$,$F, f$ are output and input respectively.

*4) Non-linear Regression:* This is an activation function. The commonly used ReLU activation function is modified so as to saturate the input signal from both ends. It is called Bilateral Rectified Linear Unit (BReLU). The operation is as shown.

$$output = min(t_{max}, max(t_{min}, W \circledast F + b) \tag{12}$$

where, $F$ is input, $W$ is weight kernel of size $48 \times 6 \times 6$, $b$ is bias of size $48 \times 1 \times 1$, $t_{max}, t_{min}$ are maximum and minimum thresholds, respectively.

The output of the last layer gives the transmission estimate $t(x)$ of the given hazy image. The transmission estimate is not continuous and hence is smoothened using guided filter. To calculate Atmospheric light($A$), the 0.1% brightest pixels in the DCP are found out by vectorizing and sorting. Then, the corresponding pixels of the input image are taken and their mean in each channel is calculated. This mean value is taken as $A$. Then finally, using $t(x)$ and $A$, dehazed image is obtained by,

$$J(x) = \frac{I(x) - A}{t(x)} + A \tag{13}$$

### C. Faster RCNN

The third part is responsible for Object recognition which is achieved using Faster RCNN. It takes image as input and gives bounding box around the detected objects as output. It also gives the probability of the class of the object detected. Faster RCNN consists of 3 sections. The block diagram is shown in Fig. 3

*1) Object Detection Network:* The input image is first passed through an Object Detection Network. Existing networks like AlexNet, ZF Net, VGG Net can be used for this purpose. Here, we use VGG Net as it promises higher accuracy. VGG Net consists of series of convolutional layers, each followed by ReLU activation layers and Max Pooling layers occurring intermittently. There exists a total of 13 weight layers in this.

*2) Region Proposal Network (RPN):* RPN is the section of network responsible for higher speed of Faster RCNN. It gives a number of Object proposals which are evaluated in further sections. RPN consists of a sliding layer network which acts on last layer of VGG Net and gives out $512 - D$ vector. This vector is fed into class prediction and bounding box regression layers. RPN also generates anchors boxes in the input image with three scales ($1:1$, $1:2$, $2:1$) and three resolutions (128, 256, 512), thus generating 9 anchor boxes for each position. These anchor boxes are further filtered out by eliminating cross boundary images. The remaining are sent through RPN class score layer and RPN bounding box regression layer. The RPN class score layer determines whether the anchor box has valid object or not using softmax layer. Anchor boxes that are close to each other and having similar class scores are reduced using non-maximum suppression. The remaining valid boxes are forwarded as Object Proposals.

*3) Bounding Box Regression:* The Object Proposals are then fed into (Region of Interest) ROI Pooling layer. It is a Max Pooling layer which divides the input into various patches such that output is of fixed size (in this case, $7 \times 7$). These ROIs are fed into three fully connected layers of size $4096$, $4096$ and $1000$. Then, a class score layer gives the probability of the object belonging to each class and a bounding box regression layer gives the bounding box around the detected object.

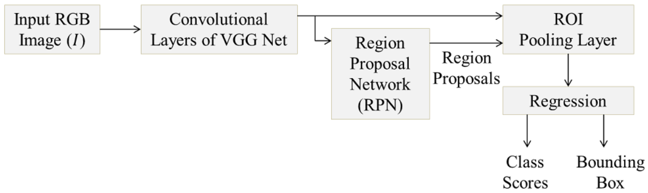


Fig. 3: Faster RCNN Block Diagram

## III. Datasets

Hazy Image datasets are quite difficult to obtain and the existing ones too contain limited number of images. For the experiments, we used Static Images Dataset from Color Hazy Images for Comparison (CHIC) database[18]. They created the haze using fog machine and took images of same scene at different levels of haze (total of $10$) varying from absolutely no haze to image completely covered with haze. Since this dataset provides us with same image and lightning conditions with ten degrees of haze, the comparision is justified. The images are taken with two levels of light and in two scenes thus making a total of $40$ images. We used Foggy Road Image Dataset (FRIDA) [19] to compare the Haze degree obtained using proposed method with that of Jun Mao *et. al*. We used images from IVC Dehazing [17] Dataset to test the Haze degree. It contains artificially generated graphic images both hazy and non hazy. It contains hazy images in indoor, outdoor, urban, wild,day and night environments. We used some images from Test Cricket Matches available online [16] to test the haze degree with predominantly white patches. Example images from the datasets are shown in figures 4, 5, 6. Figures 6a, 6b contain sample images from IVC dataset and Test cricket images respectively.

Fig. 4: Images with haze levels 1 to 10 (from left to right) without inside light [18]

Fig. 5: Images with haze levels 1 to 10 (from left to right) with inside light [18]

(a) Hazy Images from IVC dataset [17]

(b) Non-hazy images which are predominantly white [16]

Fig. 6: Dataset Images

## IV. Experiments and Results

We used the pre-trained networks for DehazeNet [21] and Faster RCNN [20] available online. The Faster RCNN is trained on 20 classes like cat,dog, potted plant, etc. We used Python and Matlab on Intel Quadcore Processor with 4GB RAM to carry out experiments. We have calculated the haze degree of all the images collected. Patch sizes of a $3 \times 3$, $5 \times 5$, $7 \times 7$, $15 \times 15$ are used to find the Dark Channel Prior. Of all those, patch size of $3 \times 3$ is found to give clear distinction between non-haze and hazy images. Bigger patch sizes tend to bring down the haze degree of hazy images, thus introducing noise. Table I shows a sample $d_{haze}$ values of Static Scenes dataset calculated using Exhaustive Method (calculated at every pixel) and Approximate Method (calculated at $1\%$ of pixels). In most of the cases, the relative error is not more than $10\%$ which is not sufficient enough to wrongly classify the image as hazy or non-hazy. We compared the haze degree obtained using proposed method with that of Jun Mao *et. al*. For this purpose, we use FRIDA database, which they have used for their experimentation. Table II shows the estimated values obtained using proposed method and Jun Mao *et. al* method. In the last column, it can be seen that the values from proposed method are close to zero for clear images. Jun Mao *et. al*'s method expects the haze degree value of haze free images to be close to zero which is not the case from our experiments. They are close to $0.4$ as shown in Column-VI of Table II. Though the haze degree of [14] decreases from hazy to corresponding non-hazy image, it is not sufficient to determine a threshold value to classify the image as hazy or non-hazy. As seen from the first two rows of Table II, the change in values for [14] (from 0.76 to 0.67) is not as high as proposed method (70 to 12). With increased variance of haze degree with proposed method, it is easier to comprehend haze and assign a threshold. When tested on other datasets[16][17], Jun Mao *et. al*'s method gave lowest haze degree close to 0.3 as opposed to the expected value of 0, whereas the proposed method gave results as expected i.e all the haze free images almost always gave haze degree close to zero.

TABLE I: Haze Degree comparison for exhaustive and approximate estimation

| Haze Level | Exhaustive | Approximate |
|---|---|---|
| 1 | 63.6 | 64.3 |
| 2 | 73.7 | 73.76 |
| 3 | 74.5 | 74.67 |
| 4 | 73.3 | 73.65 |
| 5 | 58.6 | 57.8 |
| 6 | 48.8 | 49.9 |
| 7 | 43.2 | 43.3 |
| 8 | 15.6 | 14.8 |
| 9 | 3 | 2.8 |
| 10 | 0.18 | 0.17 |

We sent the hazy images from Static Scenes through Faster RCNN Network. The images are labeled as level 1 to 10 by the owners with level 1 being highest and 10 being lowest as shown in 6a, 6b. Images from haze level 1 to 5 are densely hazed no objects are recognized in them. Also, the number of object proposals by the RPN network is very low as compared to that of Original Image. These densely hazed images have haze degree above 55. The network is able to recognize objects in the images from level 6 to 8, but not all objects are detected. The number of object proposals are also less than that of Original image, which gives a total of 300 proposals which is also the maximum allowed by the current Faster RCNN network. These images have haze degree from 10 to 50. The images with haze level 9, 10 are close to original image and network performs well on them on par with that of original image.

Next, we dehazed all the images from Static Scenes using DehazeNet and then sent them through Faster RCNN. This time, the performance on images of level 1, 2 stayed the same. There is no object recognition in images of level 3, 4 but the number of object proposals have increased. The performance on the rest of the images have improved with all the objects being recognized and the number of proposals increasing. The probability of the class too increased compared to their hazy counterparts. Figure 7 shows the performance of Faster RCNN on hazed and dehazed images. The top row contains the hazy images of level 5, 6, 7 (with inside light) respectively and the bottom row contains the corresponding dehazed images. The number over each bounding box (colored in red) contains the probability of the class detected. The number on top of each image is the number of proposals by RPN network. It can be seen clearly that the number of proposals, probability of class detected and the number of objects detected improved after dehazing. Figure 8 shows the same for IVC dataset.

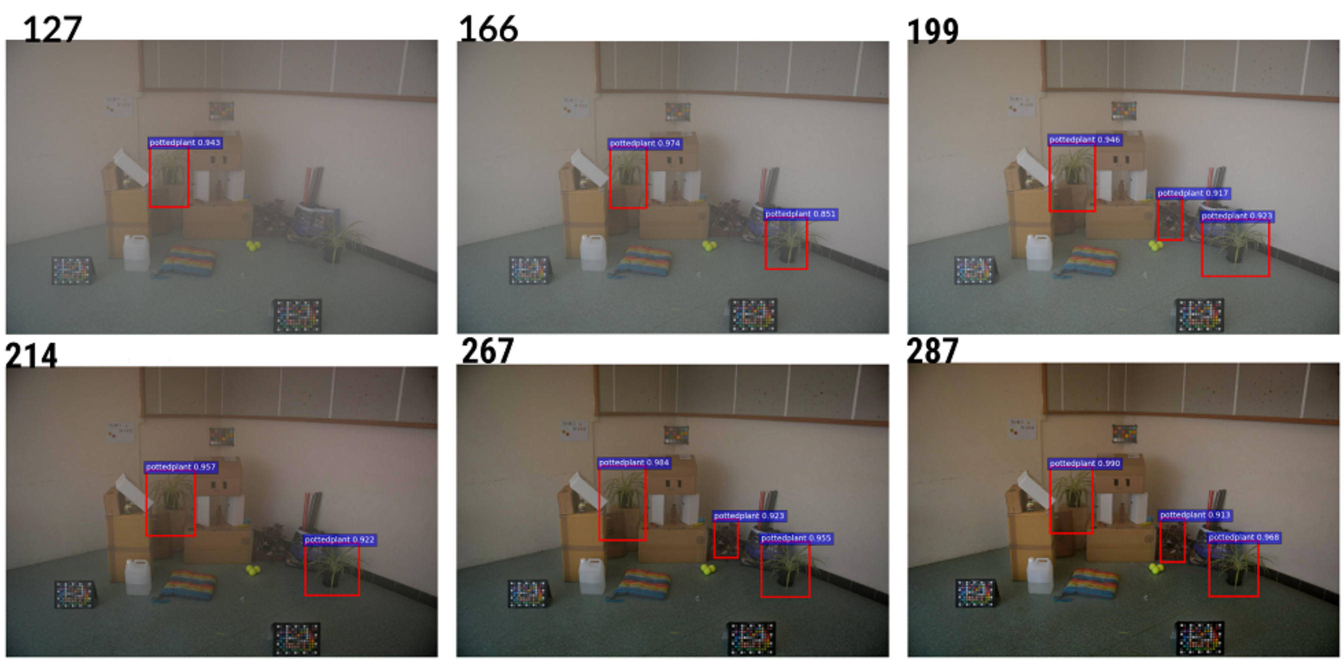


Fig. 7: Faster RCNN on Hazy images (top row) from Static Scenes Dataset and corresponding dehazed images (bottom row)

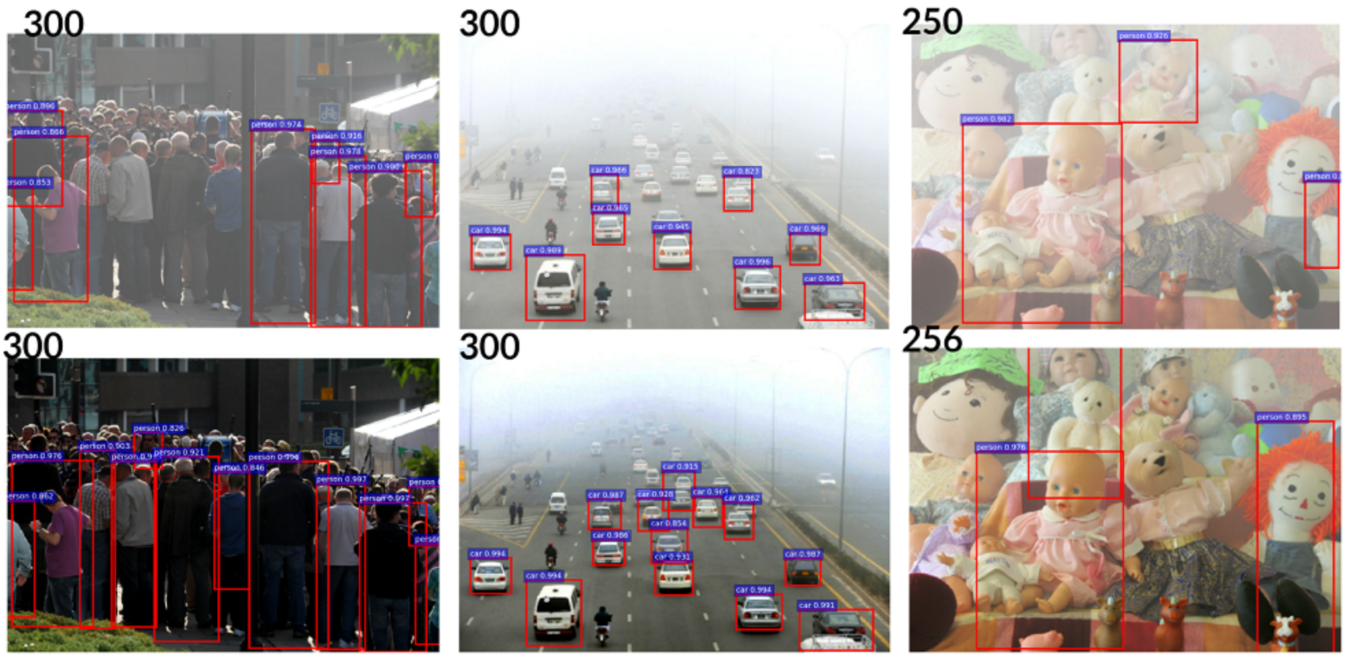


Fig. 8: Faster RCNN on Hazy images (top row) from IVC Dataset and corresponding dehazed images (bottom row)

The Table III shows the comparison hazy and dehazed images with respect to the number of object proposals by RPN and number of detected objects. It can be observed that, for animal2 image, the number of proposals decreased after dehazing. Similarly, for plant2 image, number of objects detected decreased after dehazing. These two images have low haze degree. For human3 and transportation1 images(first two images in Fig 8, the number of objects detected greatly improved.

*Setting $d_{haze}$:*

After making similar observations discussed above on a number of images, the threshold for haze degree,$d_{haze}$ is set at 10 for indoor, 25 for outdoor for moderately hazed indoor and outdoor images respectively. For heavy haze, it is set at 60 for indoor and 120 for outdoor daylight images. However, the upper and lower threshold varies depending upon illumination. For indoor images, we took the lower threshold where the number of proposals for Faster RCNN reached maximum $i.e.$ 300. For a different Object Detection network or for a different usage, this value is bound to change. For outdoor images, it is much more difficult to come to a conclusion for a haze degree. It is observed that number of proposals is directly proportional to atmospheric light. Though we came to a conclusion heuristically based on limited dataset, it is bound to change as dataset becomes bigger. One method is to check when the number of proposals reach maximum and setting it as threshold for further images. Another method is to calculate atmospheric light intermittently and changing the threshold according to it.

TABLE II: Comparison of haze degree estimated using proposed method with Jun Mao *et. al* [14]

| S.No. | Hazy images [19] | Haze degree using Jun Mao *et. al* [14] | Haze degree using proposed method | Corresponding non hazy images [19] | Haze degree using Jun Mao *et. al* [14] | Haze degree using proposed method |
|---|---|---|---|---|---|---|
| 1 | K080-000001 | 0.7647 | 70.54 | LIma-000001 | 0.4719 | 0 |
| 2 | K080-000002 | 0.686 | 12.03 | LIma-000002 | 0.3451 | 0.8387 |
| 3 | K080-000003 | 0.79163 | 46.8 | LIma-000003 | 0.423 | 0.5484 |
| 4 | K080-000004 | 0.6832 | 50.77 | LIma-000004 | 0.3341 | 0 |
| 5 | K080-000005 | 0.7728 | 28.935 | LIma-000005 | 0.4641 | 0 |
| 6 | K080-000006 | 0.7446 | 56.6452 | LIma-000006 | 0.4113 | 0.0323 |
| 7 | K080-000007 | 0.6217 | 32.5484 | LIma-000007 | 0.3119 | 1.13 |
| 8 | K080-000008 | 0.7944 | 57.61 | LIma-000008 | 0.4822 | 0 |
| 9 | K080-000009 | 0.8671 | 69.09 | LIma-000009 | 0.471 | 0 |
| 10 | K080-000010 | 0.8680 | 55.9677 | LIma-000010 | 0.4939 | 0 |

TABLE III: Comparison of Object Proposals and Detections for Hazy and Corresponding Dehazed images

| S.No. | Image Name [17] | Haze Degree | Number of Proposals | | Number of Objects Detected | |
|---|---|---|---|---|---|---|
| | | | Before Dehazing | After Dehazing | Before Dehazing | After Dehazing |
| 1 | animal1 | 18 | 300 | 300 | 2 | 2 |
| 2 | animal2 | 7 | 205 | 195 | 1 | 1 |
| 3 | human1 | 35 | 113 | 135 | 1 | 1 |
| 4 | human2 | 54 | 300 | 300 | 2 | 2 |
| 5 | human3 | 87 | 300 | 300 | 8 | 12 |
| 6 | human4 | 17 | 300 | 300 | 5 | 5 |
| 7 | plant2 | 5 | 244 | 248 | 2 | 1 |
| 8 | static2 | 95 | 250 | 256 | 2 | 3 |
| 9 | static3 | 125 | 237 | 254 | 1 | 1 |
| 10 | transportation1 | 59 | 300 | 300 | 10 | 15 |

## V. Conclusions and Future Work

In this paper, we have proposed a novel method to quantitatively estimate haze in an image thereby facilitating us to know beforehand whether it should be dehazed. Also, we proposed a flow diagram to be used on images based on the haze degree. With moderate amount of haze, the algorithm works well with DehazeNet preserving most of the features intact which helps Faster RCNN to find more proposals and enhanced object recognition. The proposed method helps in bypassing the Dehazing Network when not required. By doing so, it helps in avoiding loss of information due to Dehazing and also saves time thus reducing computational complexity. As discussed earlier and as also evident from images, Dehazing results in darkening of images that hampers the recognition process. This has a negative effect on small objects in the image. Apart from this, knowledge of atmospheric haze can be used for controlling other sensors which makes the system more robust. For example, depth sensors, infrared sensors can be switched on when visual information is not enough due to presence of haze. In that way energy consumption and computational complexity can be brought down.

Although Faster RCNN is better alternative for object recognition in terms of speed, it is still takes lot of time on a regular CPU. Coupled with DehazeNet, this time increases further hindering their use in real-time applications. Instead of cascading the two networks, the layers can be altered so that the features learned in dehazing layers can be directly used for object recognition without having to reconstruct the image. An alternative method needs to be developed which automatically estimates the threshold for haze degree.

## References


[1] B. Cai, X. Xu, K. Jia, C. Qing and D. Tao, "Dehazenet: An end-to-end system for single image haze removal," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5187-5198, 2016.

[2] S. Ren, K. He, R. Girshick and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *In Advances in neural information processing systems*, pp. 91-99, 2015.

[3] K. He, J. Sun and X. Tang, "Single image haze removal using dark channel prior," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 12, pp. 2341-2353, 2011.

[4] Fattal, Raanan. "Single image dehazing." ACM transactions on graphics (TOG) 27.3 (2008): 72.

[5] F. Liu, C. Shen and Guosheng Lin, "Deep convolutional neural fields for depth estimation from a single image," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[6] Y. S. Shin, Y. Cho, G. Pandey and A. Kim, "Estimation of ambient light and transmission map with common convolutional architecture," *In OCEANS 2016 MTS/IEEE Monterey*, pp. 1-7, 2016.

[7] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint*, arXiv: 1409.1556, 2014.

[8] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation" *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580-587, 2014.

[9] A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, 2012.

[10] Q. Zhang and X. Li, "Fast image dehazing using guided filter," *Communication Technology (ICCT), 2015 IEEE 16th International Conference on. IEEE*, 2015.

[11] R. Girshick, "Fast r-cnn," *arXiv preprint*, arXiv: 1504.08083, 2015.

[12] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville and Y. Bengio, "Maxout networks," *arXiv preprint*, arXiv: 1302.4389, 2013.

[13] J. E. Khoury, J. B. Thomas and A. Mansouri, "A color image database for haze model and dehazing methods evaluation," *International Conference on Image and Signal Processing. Springer, Cham*, 2016.

[14] J. Mao, U. Phommasak, S. Watanabe and H. Shioya, "Detecting foggy images and estimating the haze degree factor," *Journal of Computer Science & Systems Biology*, vol. 7, no. 6, pp. 1, 2014.

[15] C. W. Wang, J. J. Ding and L. A. Chen, "Haze detection and haze degree estimation using dark channels and contrast histograms," *Information, Communications and Signal Processing (ICICS), 2015 10th International Conference on. IEEE*, 2015.

[16] Test Cricket Images link: http://www.hdwalle.com/2012/12/rahul-dravid.html.

[17] IVC Dataset link: http://ivc.uwaterloo.ca/database/Dehaze/Dehaze-Database.php.

[18] Static image dataset from CHIC: http://chic.u-bourgogne.fr/.

[19] Frida Images Database: http://perso.lcpc.fr/tarel.jean-philippe/bdd/frida.html.

[20] Pre-trained Faster RCNN Network: https://github.com/rbgirshick/py-faster-rcnn.

[21] Pre-trained DehazeNet: https://github.com/caibolun/DehazeNet.

[22] N. K. Verma, T. Sharma, S. D. Rajurkar and A. Salour, "Object identification for inventory management using convolutional neural network" IEEE Applied Imagery Pattern Recognition Workshop (AIPR), Washington DC, USA, Oct. 18-20, 2016.

[23] N. K. Verma, "Future Image Frame Generation Using Artificial Neural Network with Selected Features," *IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pp. 1-8, 2012.