Estimation of Environmental Illumination in Hazy Images using the Direction of Deviation from the Dehazed Image

A Project Report submitted to the
Indian Statistical Institute, Kolkata
in partial fulfilment of the requirements of

The Sixth Summer School

on

Computer Vision, Graphics and Image Processing

by

Satish Kumar Keshri
(RKMVERI, Belur)

and

Pratiti Das
(NIT Patna)

under the supervision of

Sanchayan Santra
SRF, ECSU, Indian Statistical Institute, Kolkata

Indian Statistical Institute
Kolkata - 700108

Abstract

Image taken under haze and fog usually suffers from low contrasts and reduces visibility. This is caused by the scattering of the light by the particles present in the atmosphere. The degradation is caused by mainly two reasons. The out scattering of the light that is reflected from an object, away from the observer, and in scattering of the environmental illumination. The first reason (out scattering) results in reduction of object intensity. This property is called the transmittance of the atmosphere. On the other hand, the whitish translucent appearance of haze is the outcome of the in scattering. This veil of haze is called the airlight. Although efforts have been made to accurately estimate the transmittance, estimation of the environmental illumination (the cause behind the airlight) has largely been ignored barring a few methods. In report we describe in detail how the environmental illumination may be computed in a very simple manner by looking at how the dehazed image deviates from the actual one when we try to dehaze it using different values of transmittance and environmental illumination. We have proved if we know the direction of deviation of the dehazed image from the actual one, the hazy image can accurately be dehazed. Since the actual deviation is very hard to compute without the corresponding haze free image, help of a Convolutional Neural Network is taken to get this deviation.

Contents

1. Introduction

Presence of fog and haze in the environment can significantly degrade the visibility of the scene. Images captured in a foggy environment usually suffer from low contrast and higher brightness. In a foggy weather, light rays coming from a scene get scattered by the haze particles and only a fraction of them get captured by a camera. This light deflection reduces the direct scene transmission and replaces it with a layer of previously scattered ambient light known as airlight or veiling light. The fraction of the actual light ray is known as transmittance and the ambient light is known as environmental illumination. If we can correctly estimate the environmental illumination and the transmittance, then we can reconstruct a dehazed image from the hazy input using the image formation model[4]:

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

where I is input image, J is dehazed image, t is transmittance, A is environmental illumination and x denotes pixel coordi- nates.

The main challenge lies in the fact that while the environmental illumination is more or less remains the same for the whole image, the transmittance depends on the color of the light ray and depth of the scene. As the depth of a scene increases, transmittance value gets reduced. In general, the scene transmittance t(x) depends on the depth at position x and the scattering coefficient β[4] and it follows

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

where $d(x)$ is the depth at position x. If the particle size is big then the β value will be independent of the color of the light ray. To completely dehaze an image we need to jointly estimate both the A and $t(x)$ from Eq. (1). It has been observed that inaccurate value of environmental effect leads to bad estimation of transmittance [6]. So, to dehaze an image properly we need accurate environmental illumination.

In this project, we have derived a method that can estimate the environmental illumination given a single-hazy image by measuring the direction of deviation of the dehazed image from the actual haze free image. Then using this environmental illumination we can estimate transmittance for each pixel. Although the theory that we have proposed can also be utilized to compute the transmittance and a method to do so has been outlined in the appendix. But designing and training a CNN to estimate this direction of deviation has proved to be difficult. So, we have focused only on the estimation of environmental illumination. The transmittance has been computed using the method of [6].

2. Related Work

Image dehazing is a very difficult problem as the haze of an image depends mainly on the transmittance of the atmosphere and environmental illumination and these two are unknown. He *et al.* [3] have proposed dark channel prior to estimate scene transmittance. Dark channel prior is based on the observation that in haze-free images, in most of the local regions not covering the sky, pixels often have low intensity in at least one color channel. In case of hazy images the intensity of those color channels is mainly contributed by the airlight. Fattal [1] tried to estimate the optical transmission of a given image, by which the scattered light is eliminated from hazy image. Zhu et al.[7] proposed color attenuation prior to model the scene depth. This prior is based on the observation that difference between the brightness and saturation can approximately represent the concentration of haze. So, they have modeled depth as a linear function of brightness and saturation. Fattal[2] adopted the idea of color line to image dehazing. In hazy condition, the local color line gets shifted in the direction of airlight. From this shift the transmittance is estimated. Recently, Santra *et al.* [6] tried to dehaze an hazy-image with some estimated transmittance, which is computed in each patch, and then comparing this dehazed image with the original image using CNN based patch quality comparator.

3. Proposed Solution

Using the image formation model 1, we tried to find out any pattern between the dehazed image and the computed image. Then using the difference between these two values we will try to minimize the loss. The value of A and t(x) for which this loss is minimum will be our estimated environmental illumination and transmittance.

3.1. Theory Behind Our Approach

Let us suppose at a particular location x our input image and our estimated image for any given channel are

$$\begin{aligned} I(x) &= J_{org}(x)t'(x) + (1 - t'(x))A' \\ I(x) &= J_c(x)t(x) + (1 - t(x))A, \\ 0 &\leq t(x), t'(x) \leq 1 \\ 0 &\leq A, A' \leq 1 \end{aligned} \tag{3}$$

where $J_{org}(x)$ is actual dehazed image, $t'(x)$ is actual transmittance, A' is actual environmental illumination, $J_c(x)$ is computed dehazed image, $t(x)$ is estimated transmittance , A is estimated environmental illumination.

From Eq (3) we obtain,

$$J_{org}(x) = A' - \frac{A' - I(x)}{t'(x)} \tag{4}$$

$$J_c(x) = A - \frac{A - I(x)}{t(x)} \tag{5}$$

At a particular position x, let us define

$$\Delta J(x) = J_{org}(x) - J_c(x) = \Big(A' - A\Big) + \Big(\frac{A - I(x)}{t(x)} - \frac{A' - I(x)}{t'(x)}\Big) \tag{6}$$

We can completely dehaze an image if we can estimate A and t(x) in such a way such that $\Delta J(x) = 0$ at each position x.

Using equation 6, we can have some geometric insights about our aim of making $\Delta J(x) = 0$. Now, the partial derivatives of ΔJ with respect to A and t are:

$$\frac{\partial \Delta J}{\partial A} = \frac{1}{t} - 1$$
$$\frac{\partial \Delta J}{\partial t} = -\frac{A - I}{t^2}$$

So the hessian at any specific position x, where $t = t'$ and $A = A'$ is given by:
$$H(x) = \begin{bmatrix} 0 & \frac{-1}{(t')^2} \\ \frac{1}{(t')^2} & 2\frac{A'-I}{(t')^3} \end{bmatrix}$$
$\therefore det(H(x)) = -\frac{1}{(t')^4} < 0$
Since, at any arbitrary but fixed position x across all the channels, $det(H(x)) < 0$, so the eigen values are of opposite signs. Therefore the quadratic surface generated by $\begin{bmatrix} A & t_1 \end{bmatrix} H \begin{bmatrix} A \\ t_1 \end{bmatrix}$ is a hyperbola. Also the hessian is indefinite in nature. So, the surface of ΔJ is not convex. So the solution of $\Delta J = 0$ is not unique.

3.2. Nature of $\Delta J = 0$

If we analyze the nature of $\Delta J = 0$ then we can see that the values of A and t, which satisfies this equation forms a hyperbola. More interestingly, the positions at which $I < A'$, we get $\Delta J > 0$ for $t < t'$ and $\Delta J < 0$ for $t > t'$. Similar but the opposite trend can be observed in case of the positions at which $I > A'$.A
So, the actual environmental illumination, A' lies in a region where for $t > t'$, $I < A'$ satisfies $\Delta J < 0$ and $I > A'$ satisfies $\Delta J > 0$. If we simply choose $t = 1$, then it will be very simple for us to find a region for A'. In this region, for every choice of A, we will get a t such that $\Delta J = 0$ at each position x. The situation can be easily visualized by the below figure:

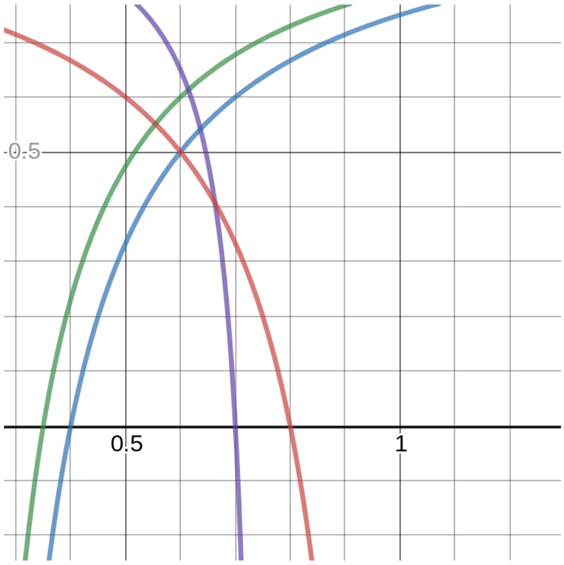


Figure 1: $\Delta J = 0$ curve for different t' and $A' = 0.6$

For different estimates of A and t the behaviour of ΔJ varies as per the table below:

	$A < A'$		$A = A'$		$A > A'$	
$t > t'$	$\Delta J > -k$ for $I < A$,where $k \in \mathbb{R}$	$\Delta J < 0$ for $A < I$	$\Delta J < 0$ for $I < A'$	$\Delta J > 0$ for $I > A'$	$\Delta J < k$ for $I < A$, where $k \in \mathbb{R}$	$\Delta J > 0$ for $A < I$
$t = t'$	$\Delta J < 0$		$\Delta J = 0$		$\Delta J > 0$	
$t < t'$	$\Delta J > -k$ for $I < A$, where $k \in \mathbb{R}$	$\Delta J < 0$ for $A < I$	$\Delta J > 0$ for $I < A'$	$\Delta J < 0$ for $I > A'$	$\Delta J > 0$ for $I < A$	$\Delta J < k$ for $A < I$

Using this table we have tried to formulate an algorithm for finding A for each channel separately. We have also formulated an algorithm for estimating transmittance (t) for each pixel across all channels, using the estimated A values.

3.3. Network Architecture

For dehazing an image we only need to know the sign of the deviation $(\Delta J(x))$ of our dehazed image from the ground truth at each position x and accordingly estimate A .

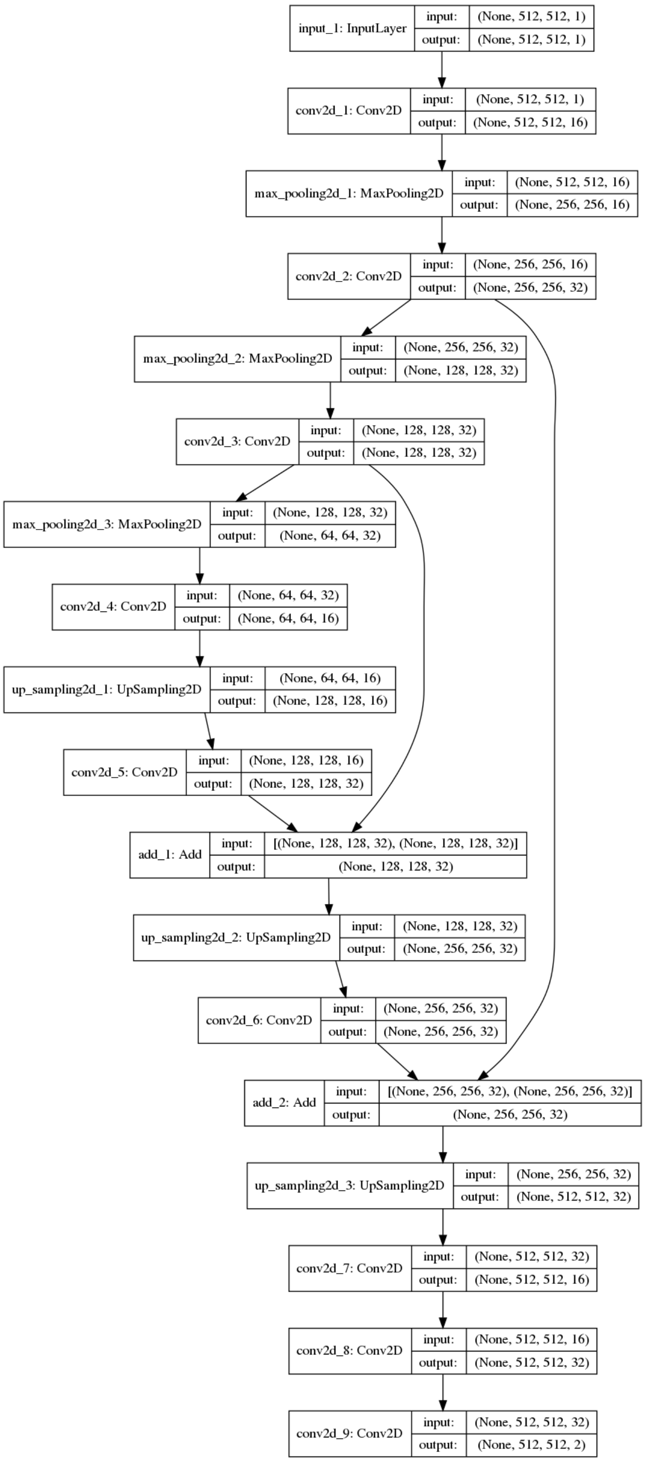


Figure 2: The Network

In the proposed algorithm, dehazing is done on the basis of the nature of the ΔJ, like $\Delta J > 0$ or $<= 0$. This ΔJ is calculated from the original image (ground truth) and the hazy image. In this algorithm, the input data size and target data size are same for the network. So, the network, which is used to train the hazy image data and the label for the ΔJ, is inspired by U-Net [5]. The architecture of the network is shown in Figure 2 and the details are given below:

- Fully Convolutional Network Model concept is used in this network. The convolutional layer is available but no dense layer is present in the network.It has two types of path, one is contracting path and another is expansive path.
- The contracting path is used to down sampling the data and up-sampling is done by the expansive path. In the contracting path typical convolution with 3×3 kernel size and *tanh* activation function and 2×2 max pooling is used.
- 2×2 up-sampling followed by convolution with 3×3 kernel size and *tanh* activation function is done in expansive path. At the last convolution layer the *softmax* activation function is used.
- There are one input for training the network, the hazy image and the target is the label as [0 1], when $\Delta J > 0$ and [1 0], when $\Delta J \leqslant 0$. The size of the input is $512 \times 512 \times 1$ and the target size is $512 \times 512 \times 2$. For every pixel these two labels are used to compute the nature of ΔJ properly. When the 1st target value is less than the 2nd target value, then it denotes $\Delta J > 0$, otherwise it is $\leqslant 0$.

3.4. Algorithm

After training the network, if a new hazy image comes then, the value of the environmental illumination, A can be estimated by Algorithm 1. The description is given below:

The inputs are the hazy image(I_p), size of the image(s_x, s_y) and some constants(c_1,c_2) and the output is the estimated A . Initially, A and t both are assumed as 1. The value of ΔJ is obtained as an output from the network. $r1$ is the set containing the values from I_p, where ΔJ is 1 at a particular pixel, and then it takes the top c_1 percentage of r_1. r_2 set is calculated in the same way,but it is obtained when $\Delta J = 0$,and then c_2 percentile of r_2 from is taken. The constants c_1 and c_2 are needed to reduce the error which is included due to the loss of the network. Now, if length of r_1 is greater than 0, then A_{min} stores the minimum value of r_1, otherwise, it stores 1. Similarly, if length of r_2 is greater than 0, then A_{max} stores the maximum value of r_2 set, otherwise, it is 1. At last the final A value is computed as the average of A_{min} and A_{max}. This algorithm has been executed individually for each color channel of RGB image, and the Neural Networks behind this algorithm, are trained differently for each color channel.

Algorithm 1 *A* searching algorithm

Input: I_p, c_1, c_2
Output: a

$a \leftarrow 1$
$t \leftarrow 1$
$\Delta J=$ `delta_j`(I_p)
$r_1 = \{i \in I_p : \Delta J(x) = 1 \ at \ x\}$
$r_1 =$ top c_1 percentage of r_1
$r_2 = \{i \in I_p : \Delta J(x) = 0 \ at \ x\}$
$r_2 = c_2$ percentile of r_2
if `length`$(r_1) > 0$ **then**
    $a_{min} = \min(r_1)$
else
    $a_{min} = 1$
end if
if `length`$(r_2) > 0$ **then**
    $a_{max} = \max(r_2)$
else
    $a_{max} = 0$
end if
$a \leftarrow (a_{min} + a_{max})/2$

4. Experimental Settings

4.1. Data Generation

To train the network and to predict the nature of ΔJ, one hazy image is taken as input and the labelled ΔJ is used as target, from which estimated environmental illumination (A) is calculated. The hazy image and the labelled ΔJ are computed in the following way.

1. At first one without haze image(J) and its corresponding actual transmittance ($t'(x)$) is taken as the input to the data generator and both are resized as 512×512.

2. Then a random environmental illumination(A) is generated for each color channel and by using eq.(1) a hazy image(I) is computed, which is used as the input to train the network.

3. To estimate the value of environmental illumination(A), ΔJ is assuming as actual image(J)-hazy image(I) and a label is assigned as per the nature of ΔJ
   - $\Delta J > 0$, then label is [0 1],
   - $\Delta J \leqslant 0$,then the label is [1 0].

4. The hazy image(I) and the labels for ΔJ are generated and used to train the network.

4.2. Training Details

The network, which is used, is inspired by U-Net. It is trained by the pair of hazy image(I) and the label, which is computed based on the nature of the ΔJ.

- The input to the network is the hazy image and its size is $512 \times 512 \times 1$. The label works as the target and the size of the label is $512 \times 512 \times 2$.
- At first convolution and down sampling is done on the input. Here the hazy image size is decreased.
- After that up sampling and convolution is done to increase the target size
- Lastly the prediction of the nature of the ΔJ is done on the basis of this training.
- The data set of eleven(11) images and corresponding actual transmittance which were used by Fattal [2] to train his approach, are also used here to train the network. These images are resized as 512×512 before using.

5. Result

The evaluation of our proposed approach is done over the data set with benchmark hazy images, which are used by Fattal [2] to evaluate his method. Using our proposed algorithm we have dehazed some synthetic and benchmark images which are used by Fattal [2]. We have estimated our own environmental illumination but Fattal has used some other methods to estimate it. The results are as follows:

5.1. Synthetic Images:

(a) Input (b) Fattal [2] (c) Our (d) True

Figure 3: Church A = [.494,.5714,.996], A' = [.5,.6,1.0]

(a) Input (b) Fattal [2] (c) Our (d) True

Figure 4: road2 A = [.50,.60,.994], A' = [.5,.6,1.0]

(a) Input (b) Fattal [2] (c) Our (d) True

Figure 5: lawn1 A = [.51,.577,.996], $A' = [.5,.6,1.0]$

In synthetic images our method estimates near to original values of environmental illumination. In case of road2 and church, our dehazed images outperform Fattal's colour lines[2].

5.2. Benchmark Images

:

(a) Input (b) Fattal [2] (c) Our (d) Transmittance

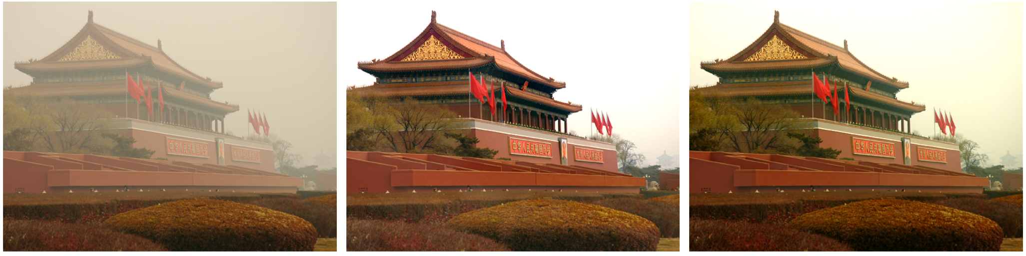
Figure 6: Dubai A = [.67,.70,.75]

(a) Input (b) Fattal [2] (c) Our

Figure 7: Florence A = [.67,.70,.75]

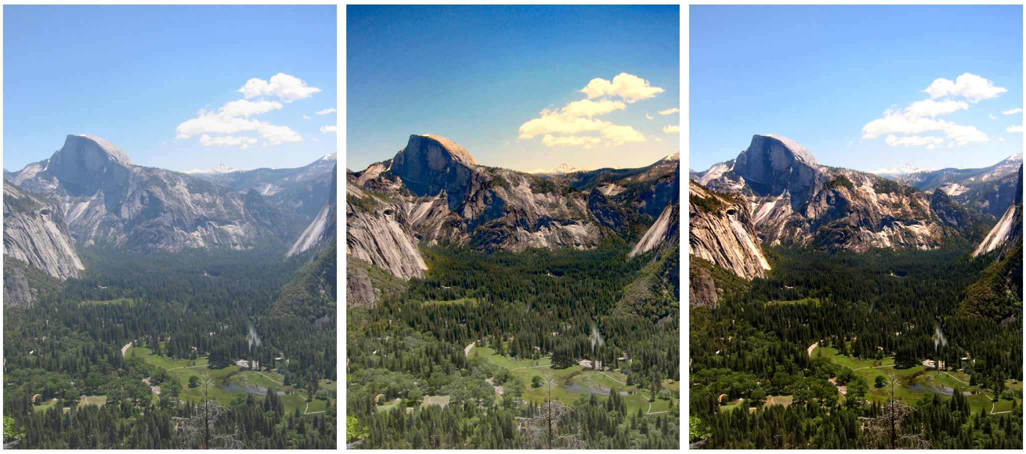
(a) Input (b) Fattal [2] (c) Our

Figure 8: Herzeliya A = [.54,.65,.69]



(a) Input (b) Fattal [2] (c) Our

Figure 9: tiananmen A = [.75,.69,.62]



(a) Input (b) Fattal [2] (c) Our

Figure 10: Yosemite A = [.54,.65,.78]

(a) Input (b) Fattal [2] (c) Our

Figure 11: Train A = [.57,.57,.56]

In case of benchmark images our method produces more natural images in comparison to Fattal[2]. In case of Yosemite our method can correctly dehaze the clouds and the trees but clouds in Fattal's image is yellowish. On the other hand, buildings in fattal's Dubai image are blackish but our method has estimated more natural buildings.

5.3. Failure Cases

Synthetic Images:

(a) Input (b) Fattal [2] (c) Our (d) True

Figure 12: Mansion A = [.50,.55,.58], $A' = [.5,.6,1.0]$

(a) Input (b) Fattal [2] (c) Our (d) True

Figure 13: Road1 A = [.53,.62,.74], $A' = [.5,.6,1.0]$

Here we can see that in some cases our algorithm might fail to correctly estimate environmental illumination in the blue channel,due to which sometimes we might not achieve the expected quality of dehazing.

Benchmark Images:

(a) Input (b) Fattal [2] (c) Our

Figure 14: Landscape A = [.62,.614,.65]

(a) Input (b) Fattal [2] (c) Our

Figure 15: Manhattan A = [.524,.628,.99]

In some rare instances, our algorithm could not correctly estimate environmental illumination in some benchmark images and that is reflected in the quality of dehazing in these two images.
Though our method could not perform well in case of some rare cases, we have produced more natural looking dehazed images in almost all cases..

6. Conclusion

In this report, a method of calculating the environmental illumination, which is a reason of the color of the haze in image, is proposed. It is computed on the basis of the deviation of dehazed image, which are dehazed by different transmittance and environmental illumination, from the actual image. The hazy image can be properly dehazed, when the direction of the deviation of the dehazed image from the actual image is known and this is proved in the report. A Convolutional Neural Network is used to compute the above mentioned deviation, as the actual image is not available. Some errors are included to the result because of the traing of Convalutional Neural Network. There is some lose during the training of the neural network so, some deviation is still there. Here we have trained our CNN over only red channels and tried to estimate the A for all channels. That may be a reason for wrong estimation of A incase of blue channels. In future, we will try to design a more sophisticated network for predicting ΔJ. We also try to estimate transmittance(t) using the table 3.2 using our own algorithm which can be found in the appendix section.

Appendix A Proof for Algorithm 1

This appendix contains the proofs necessary for developing our algorithm.
Since the value of the actual transmittance (t') is such that
$0 \leq t' < 1$, so

$$(1 - \frac{1}{t'}) < 0. \tag{7}$$

Proposition 1. *If $I < A'$ then for $t = 1$ and $A = 1$, we will get $\Delta J < 0$, where the actual transmittance t' satisfies $0 \leq t' < 1$.*

Proof. We have $A' > I$. So, $A' - I > 0$. Now,

$$\begin{aligned}
\Delta J &= (A' - A) + (\frac{A - I}{t} - \frac{A' - 1}{t'}) \\
&= (A' - 1) + (\frac{1 - I}{1} - \frac{A' - 1}{t'}) \quad [\because A = 1, t = 1] \\
&= A' - I - \frac{A' - I}{t'} \\
&= (A' - I)(1 - \frac{1}{t'}) \\
\therefore \Delta J &< 0 \; [\textit{using } Eq(7) \textit{ and } A' > I]
\end{aligned}$$

This concludes the proof. □

Proposition 2. *If $1 > I > A'$ then for $t = 1$ and $A = 1$, we will get $\Delta J > 0$, where the actual transmittance t' satisfies $0 \leq t' < 1$.*

Proof. We have $> A' > I$. So, $A' - I > 0$. Now,

$$\begin{aligned}
\Delta J &= (A' - A) + (\frac{A - I}{t} - \frac{A' - 1}{t'}) \\
&= (A' - 1) + (\frac{1 - I}{1} - \frac{A' - 1}{t'}) \quad [\because A = 1, t = 1] \\
&= A' - I - \frac{A' - I}{t'} \\
&= (A' - I)(1 - \frac{1}{t'}) \\
\therefore \Delta J &> 0 \; [\textit{using } Eq(7) \textit{ and } A' > I]
\end{aligned}$$

This concludes the proof. □

Appendix B Proofs for Algorithm 2

This section contains the theory necessary for estimating transmittance $t(x)$ at each position x.
After estimating A we can follow estimate t such that $\Delta J = 0$ for each position x.

Proposition 3. *For $A = A'$ sign of ΔJ depends on I and t. The relationships are as follows: When $I < A$ then*

1. $t < t' \implies \Delta J > 0$
2. $t = t' \implies \Delta J = 0$
3. $t > t' \implies \Delta J < 0$

Proof. 1. $t < t' \implies \frac{1}{t} > \frac{1}{t'}$.

$$\begin{aligned}\Delta J &= (A - I)(\frac{1}{t} - \frac{1}{t'}) \\ &> 0 \quad [\because A - I > 0]\end{aligned}$$

2. $t = t'$ If $A = A'$ and $t = t'$ then $\Delta J = 0$.
3. $t > t' \implies \frac{1}{t} < \frac{1}{t'}$.

$$\begin{aligned}\Delta J &= (A - I)(\frac{1}{t} - \frac{1}{t'}) \\ &< 0 \quad [\because A - I > 0]\end{aligned}$$

$\square$

Proposition 4. *For $A = A'$ sign of ΔJ depends on I and t. The relationships are as follows: When $I > A$ then*

1. $t < t' \implies \Delta J < 0$
2. $t = t' \implies \Delta J = 0$
3. $t > t' \implies \Delta J > 0$

Proof. The proof is similar to proposition 3 $\square$

Algorithm 2 t searching algorithm

Input: $I(x)$, $u, l, \delta t$
Output: t

```
a ← A and
if I < a then
    t = l
else
    t = u
end if
flag ← FALSE
ΔJ(x) = delta_j(I(x)
if ΔJ(x) = 0 then
    flag ← TRUE
else
    flag ← FALSE
end if
while flag = FALSE do
    if t = l then
        t = t + δt
        ΔJ(x) = delta_j(I(x)
        if ΔJ(x) = 0 then
            flag = TRUE
        else
            Continue
        end if
    end if
    if t = u then
        t = t − δt
        ΔJ(x) = delta_j(I(x)
        if ΔJ(x) = 0 then
            flag = TRUE
        else
            Continue
        end if
    end if
end while
```

References

[1] Raanan Fattal. Single image dehazing. *ACM transactions on graphics (TOG)*, 27(3):72, 2008.

[2] Raanan Fattal. Dehazing using color-lines. *ACM transactions on graphics (TOG)*, 34(1):13, 2014.

[3] Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *IEEE transactions on pattern analysis and machine intelligence*, 33(12):2341–2353, 2010.

[4] W. E. K. Middleton. Vision through the Atmosphere. In Julius Bartels, editor, *Geophysik II / Geophysics II*, Handbuch der Physik / Encyclopedia of Physics, pages 254–287. Springer Berlin Heidelberg, Berlin, Heidelberg, 1957.

[5] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[6] Sanchayan Santra, Ranjan Mondal, and Bhabatosh Chanda. Learning a patch quality comparator for single image dehazing. *IEEE Transactions on Image Processing*, 27(9):4598–4607, 2018.

[7] Qingsong Zhu, Jiaming Mai, and Ling Shao. A fast single image haze removal algorithm using color attenuation prior. *IEEE transactions on image processing*, 24(11):3522–3533, 2015.