Day/Night Unconstrained Image Dehazing

Sanchayan Santra
Under the supervision of Prof. Bhabatosh Chanda

Electronics and Communication Sciences Unit
Indian Statistical Institute, Kolkata, India

Presented at: ICPR 2016, Cancun, Mexico

Introduction

Image Dehazing

(a) Original Image (b) Dehazed Image

Figure: Image and its dehazed version

Dehazing Methods (Day time)

Figure: Daytime dehazaing method of Fattal[1] failing for night time images. Include miri haze here

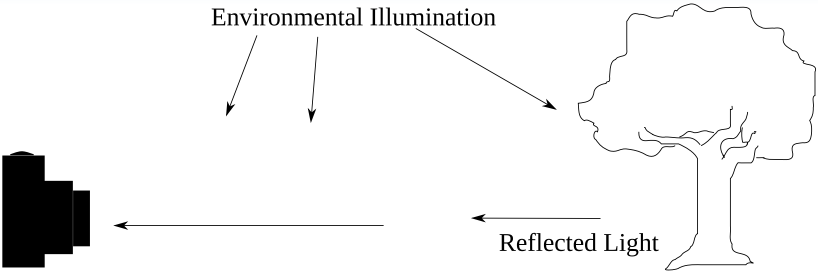
Dehazing Methods (Night Time)

Figure: Nighttime dehazaing method of Li et al.[2] failing for day time images

Our Method

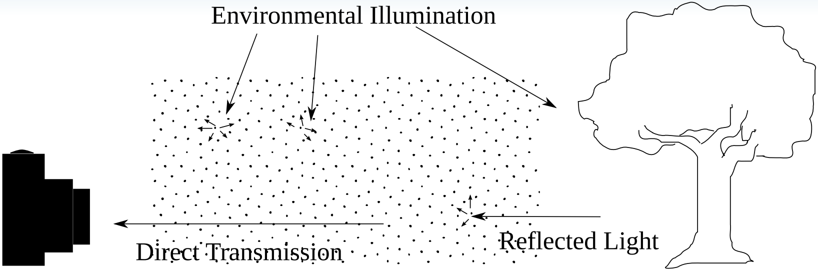
Figure: Our method works for both cases

Image Formation



$$\underbrace{I(\mathbf{x})}_{\text{Observed Intensity}} = \underbrace{J(\mathbf{x})}_{\text{Reflected Intensity}} \quad (1)$$

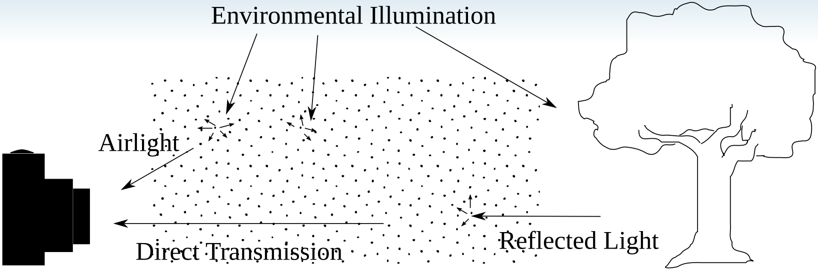
Image Formation in Haze



$$\underbrace{I(\mathbf{x})}_{\text{Observed Intensity}} = \underbrace{J(\mathbf{x})t(\mathbf{x})}_{\text{Direct Transmission}} + \cdots \quad (2)$$

$t(\mathbf{x})$ is the scene transmittance.

Image Formation in Haze



$$\underbrace{I(\mathbf{x})}_{\text{Observed Intensity}} = \underbrace{J(\mathbf{x})t(\mathbf{x})}_{\text{Direct Transmission}} + \underbrace{(1 - t(\mathbf{x}))A}_{\text{Airlight}} \tag{3}$$

$t(\mathbf{x})$ is the scene transmittance.
A is the environmental illumination.
Dehazing methods try to recover $J(\mathbf{x})$.

Haze Imaging Equation: Assumptions

$$I(\mathbf{x}) = J(\mathbf{x})t(\mathbf{x}) + (1 - t(\mathbf{x}))A \tag{4}$$

Assumptions,

- For RGB image $I(\mathbf{x})$, $J(\mathbf{x})$ and A are 3x1 vectors and $t(\mathbf{x})$ is a scalar, assuming it to be constant across color channels.
- The environmental illumination (A) is constant in the whole scene.

Environmental Illumination

Figure: Atmospheric Light is not constant throughout.

Relaxed Imaging Equation

$$I(\mathbf{x}) = J(\mathbf{x})t(\mathbf{x}) + (1 - t(\mathbf{x}))A \tag{5}$$

is changed to

$$I(\mathbf{x}) = J(\mathbf{x})t(\mathbf{x}) + (1 - t(\mathbf{x}))A(\mathbf{x}) \tag{6}$$

Where, A is changed to $A(\mathbf{x})$ to account for the space-variant illumination within an image.

In our method, we try to solve the following,

$$\begin{align} I(\mathbf{x}) &= J(\mathbf{x})t(\mathbf{x}) + (1 - t(\mathbf{x}))m(\mathbf{x})\hat{A}(\mathbf{x}) \tag{7} \\ &= J(\mathbf{x})t(\mathbf{x}) + a(\mathbf{x})\hat{A}(\mathbf{x}). \tag{8} \end{align}$$

Here we decompose $A(\mathbf{x})$ into its magnitude $m(\mathbf{x})$ and direction $\hat{A}(\mathbf{x})$.

Relaxed Imaging Equation

$$
\begin{aligned}
I(\mathbf{x}) &= J(\mathbf{x})t(\mathbf{x}) + (1 - t(\mathbf{x}))m(\mathbf{x})\hat{A}(\mathbf{x}) \qquad (9)\\
&= J(\mathbf{x})t(\mathbf{x}) + a(\mathbf{x})\hat{A}(\mathbf{x}). \qquad (10)
\end{aligned}
$$

This equation underconstrained.
Only $I(\mathbf{x})$ is known and all other quantities ($J(\mathbf{x})$, $t(\mathbf{x})$, $m(\mathbf{x})$, $\hat{A}(\mathbf{x})$) are unknown.

So we do two things,

- We assume within a small patch $t(\mathbf{x})$, $m(\mathbf{x})$ and $\hat{A}(\mathbf{x})$ is constant.
- We use color line prior to introduce more constrains. This has been used previously by Fattal[1], but in the context of daytime dehazing.

Color line

If we consider a small patch of a natural image, the pixel will ideally lie on a line passing through the origin. But due to camera and other distortions they form elongated color clusters.[3]

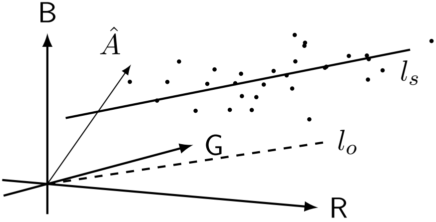
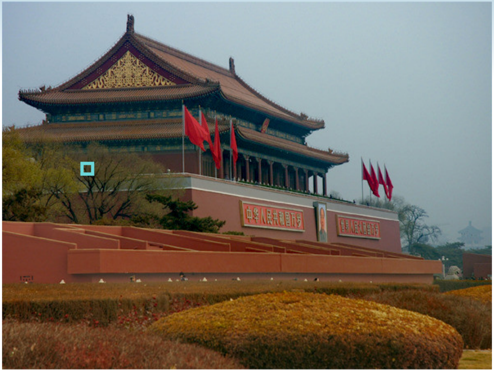
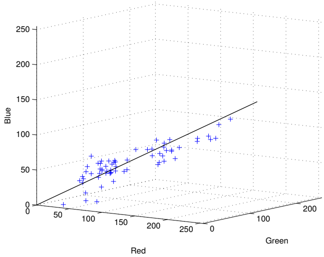


Figure: Colors in a patch as points in RGB space. The original line l_o got shifted in the direction of $\hat{A}$ to make l_s.

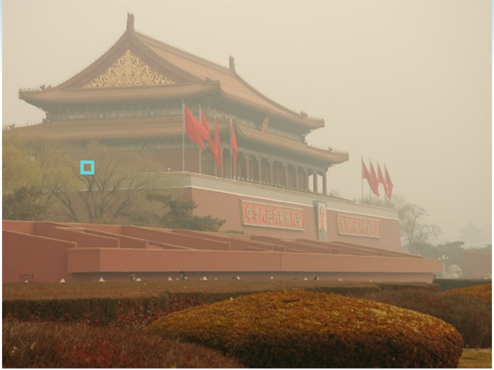
In case of haze this line gets shifted by the airlight component $((1 - t(\mathbf{x}))m(\mathbf{x})\hat{A}(\mathbf{x}))$.

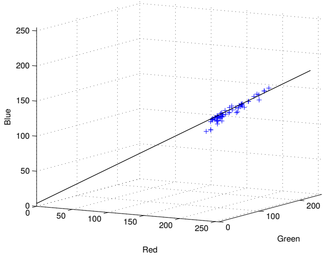
Color line: Haze free





Color line: Haze





Line fitting

We apply RANSAC in each image patch and obtain,

- Two points on the line $(I_1(\mathbf{x}),\ I_2(\mathbf{x}))$
- Normal to the plane $(\hat{n})$ containing these two points and the origin. This plane will contain the $\hat{A}$ that this patch got affected with.

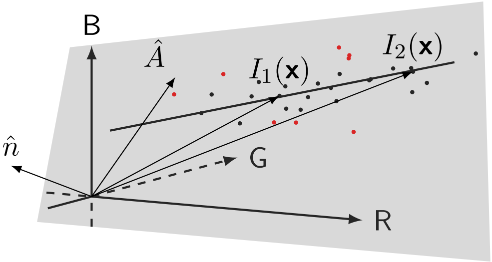


Figure: Colors in a patch as points in RGB space and the fitted line.

$\hat{A}$'s computation

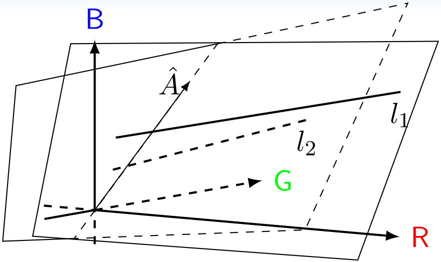


Figure: Fitted lines l_1 and l_2 and the corresponding patch planes obtained from two image patches. $\hat{A}$ lies in the intersection of the planes.

$\hat{A}$ can be computed as the normal to the two normals of the planes.

Single $\hat{A}$ computation

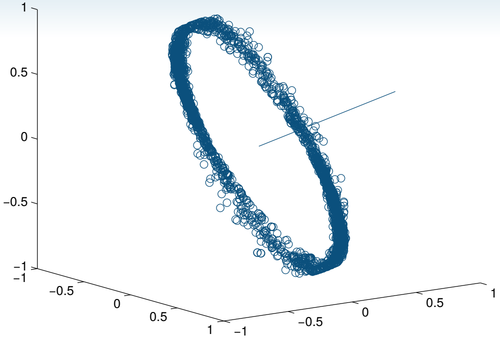
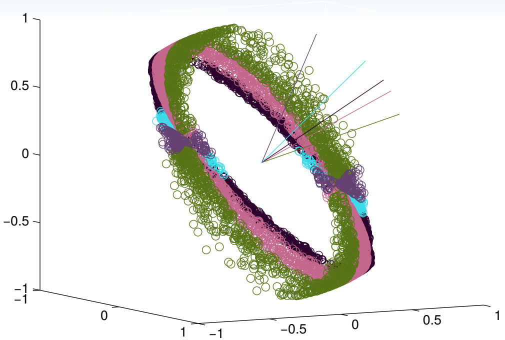


Figure: The normals obtained from patches affected by same $\hat{A}$, are plotted as circles and the line segment denote their normal i.e. $\hat{A}$

$\hat{A}$ can be computed by fitting a plane considering these circles as points.

$\hat{A}$'s computation



Problem-

- Number of $\hat{A}$ is unknown

Finding all $\hat{A}$'s

To fit an unknown number of planes, first we represent the plane using the following form,

$$n_x \cos\theta \sin\phi + n_y \sin\theta \sin\phi + n_z \cos\phi = 0, \quad (11)$$

where $[n_x, n_y, n_z]^T = \hat{n}$ (the normal computed from patches) and $(\cos\theta \sin\phi, \sin\theta \sin\phi, \cos\phi)$ denotes the normal of plane we are fitting now.
We use Hough Transform with parameters θ and ϕ and find modes in the Hough space to get $\hat{A}$'s.

$a(\mathbf{x})$ Estimation

The shift of the line from the origin is computed using,

$$E_{line}(\rho, s) = \min_{\rho, s} \|P_0 + \rho D - s\hat{A}\|^2 \tag{12}$$

where the fitted line is represented as $P = P_0 + \rho D$ and s provides the shift $((1 - t(\mathbf{x}))m(\mathbf{x}))$ of this line in the direction $\hat{A}$.

Discarding patches

The estimates could be bad if,

- $t(\mathbf{x})$ not constant in a patch
- The variability of the pixels are too low to fit line
- The color of the patch is similar to $\hat{A}$.

So, some patches are discarded in the process. But we need the estimate at each pixel. So, we interpolate $\hat{A}$'s and $a(x)$ at positions where the estimates are not computed.

$\hat{A}$ Interpolation

We denote each one of the $\hat{A}$'s by a label and compute their influence at all the pixels. The influence of each label is obtained by minimizing the function

$$E_{infl}(F) = (F - P)^T (F - P) + \frac{\lambda}{2} F^T L F, \tag{13}$$

where F is a matrix of size *numpixel* $\times$ *numlabel* with entry $F(i, j)$ denoting the influence of j-th $\hat{A}$ on i-th pixel. P is also a *numpixel* $\times$ *numlabel* matrix with $P(i, j) = 1$ if j-th $\hat{A}$ is assigned to i-th pixel. L is the graph laplacian constructed from the input image.

The final interpolated $\hat{A}(\mathbf{x})$ is a normalized weighted sum of the $\hat{A}$'s where the weights are the influences.

$a(\mathbf{x})$ Interpolation

The interpolation of $a(\mathbf{x})$ is done minimizing the function

$$E_{airlight}(a) = (a - \tilde{a})^T \Sigma (a - \tilde{a}) + \alpha a^T L_g a + \beta b^T a, \qquad (14)$$

where $\tilde{a}$ is estimated magnitudes of airlight component and 'a' is its interpolated value, both in vector form (*numpixel* $\times$ 1). $\tilde{a}$ is zero where the estimate is not there. Σ is a diagonal matrix with its diagonal containing the error variance where $a(\mathbf{x})$ is estimated and 0 otherwise.

Recovery

We now compute airlight removed original image:

$$J(\mathbf{x})t(\mathbf{x}) = I(\mathbf{x}) - (1 - t(\mathbf{x}))m(\mathbf{x})\hat{A}(\mathbf{x}). \tag{15}$$

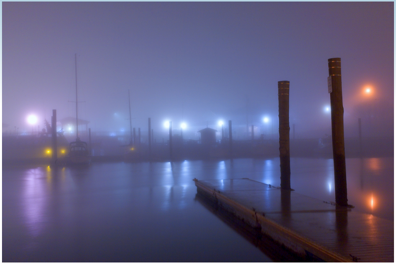
As we don't have $t(\mathbf{x})$, we enhance the contrast in the following way,

$$J'(\mathbf{x}) = \frac{I(\mathbf{x}) - (1 - t(\mathbf{x}))m(\mathbf{x})\hat{A}(\mathbf{x})}{1 - Y((1 - t(\mathbf{x}))m(\mathbf{x})\hat{A}(\mathbf{x}))} \tag{16}$$

$$Y(I(\mathbf{x})) = 0.2989 I_R(\mathbf{x}) + 0.5870 I_G(\mathbf{x}) + 0.1140 I_B(\mathbf{x}). \tag{17}$$

Though this works well for some images, its good performance can't be guaranteed.

Night-time Result



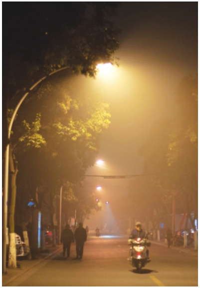
(a) Image fog on the bay

(b) Li et al.

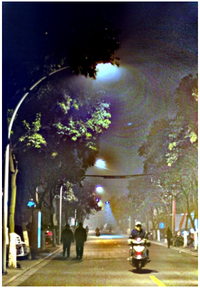
(c) Our method

(d) Airlight

Night-time Result



(a) Image night road



(b) Li et al.

(c) Our method

(d) Airlight

Daytime Result

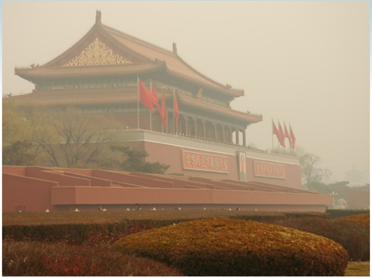
(a) Image ny12

(b) Li et al.

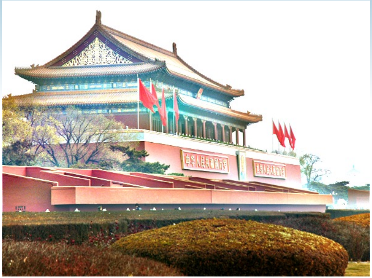
(c) Fattal color line

(d) Our method

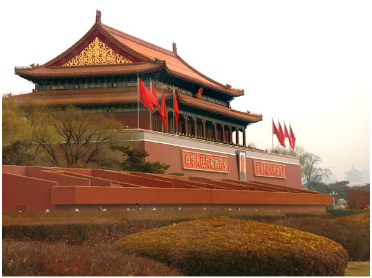
Daytime Result



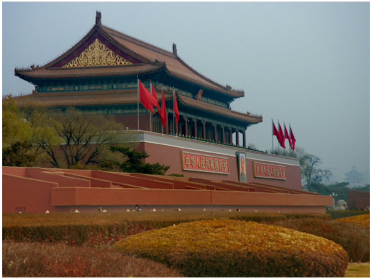
(a) Image tiananmen



(b) Li et al.



(c) Fattal Color line



(d) Our method

Result

Table: FADE[4] value of images. Lower value indicates less fog.

Image	**Time**	**Original**	**Li et al.**	**Fattal**	**Our**
fog on the bay	night	1.1555	**0.3799**	-	0.4221
night road	night	0.6753	0.3062	-	**0.2308**
ny12	day	0.7344	0.3017	**0.1715**	0.2057
tiananmen	day	1.3454	0.4058	0.4329	**0.3928**

Conclusion

- We have proposed a generalized dehazing method that works for both day and night time images.
- Although This requires relaxation of the model, the solution can still be obtained.
- More results can be obtained from our website. (`http://www.isical.ac.in/~sanchayan_r/day_night_dehaze`)

References

1. R. Fattal, “Dehazing using color-lines,” ACM Trans. Graph., vol. 34, no. 1, pp. 13:1-13:14,
2. Y. Li, R. T. Tan, and M. S. Brown, “Nighttime haze removal with glow and multiple light colors,” in 2015 IEEE International Conference on Computer Vision (ICCV), Dec 2015, pp. 226-234.
3. I. Omer and M. Werman, “Color lines: Image specific color representation”, CVPR 2004.
4. L. K. Choi, J. You, and A. C. Bovik, “Referenceless prediction of perceptual fog density and perceptual image defogging,” IEEE Transactions on Image Processing, vol. 24, no. 11, pp. 3888-3901, Nov 2015

Thank You