Image Dehazing and Guided Filtering

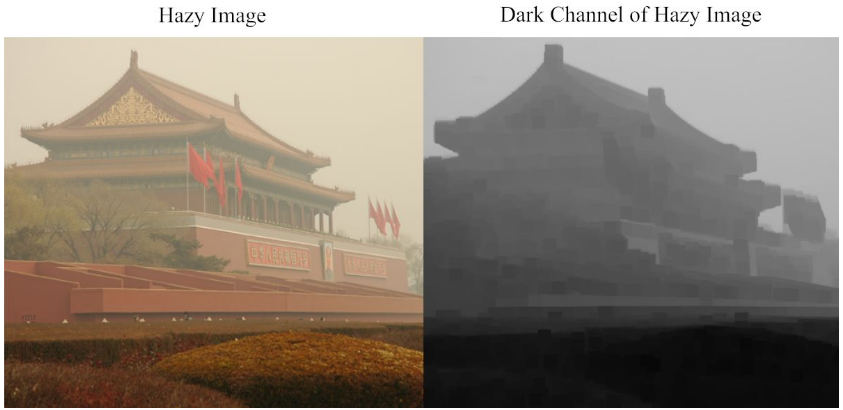
Brycen Westgarth and Jackie Burd

ECE 278A

Nina Miolane

Dataset

Dark Channel Prior



$$J^{dark}(\mathbf{x}) = \min_{c \in \{r,g,b\}} \left(\min_{\mathbf{y} \in \Omega(\mathbf{x})} \left(J^c(\mathbf{y}) \right) \right)$$

Atmospheric Transmission Map and Global Atmospheric Light

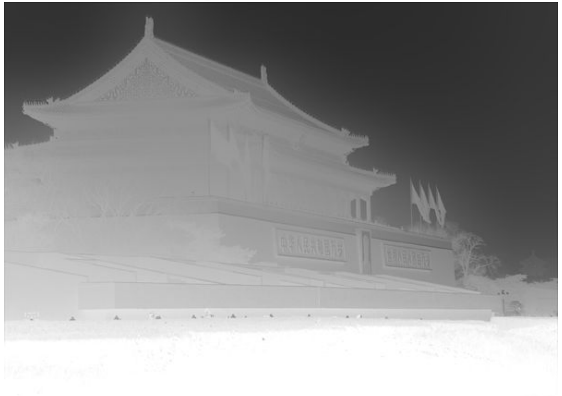
Global Atmospheric Light: color/intensity value that can be attributed to illumination of the image caused by the atmosphere

- Independent of the haze
- Helps avoid distorting the natural illumination of the image
- Identify the locations of the top 0.1% highest intensity pixels of the dark channel, and then where they correspond in the original image, take *those* intensity values as the global atmospheric light.

$$\tilde{t}(x) = 1 - \omega \min_{c \in r,g,b} (\min_{y \in \Omega(x)} (\frac{I_c(y)}{A_c}))$$

Transmission Map Refinement with Guided Filtering

- The Raw Transmission map computed from the dark channel does not accurately represent the edges and detail of the original image
  - 15x15 minimum filter is used to compute dark channel
  - Local minimum of region will be represented many times in area around its true location
- Original paper suggests using a complicated optimization called Levin's Soft Matting method
  - Complicated to implement
  - Not many resources that outline the procedure
- Instead we use a method called guided filtering to achieve similar results



Guided Filtering

Input: Rough Image (*p*), Guiding Image (*I*), Kernel Size (n)

Output: Image (*q*)

Goal: For each possible sqaure neighborhood of size (n x n), assign output window q_k where *k* is the widow index in the form:

$$q_i = a_k I_i + b_k, \forall i \in \omega_k$$

with linear coefficients (a_k, b_k) that minimize cost:

$$E(a_k, b_k) = \sum_{i \in \omega_k} ((a_k I_i + b_k - p_i)^2 + \epsilon a_k^2)$$

This can be solved by linear regression

$$a_k = \frac{\frac{1}{|\omega|}\sum_{i \in \omega_k} I_i p_i - \mu_k \bar{p}_k}{\sigma_k^2 + \epsilon}$$

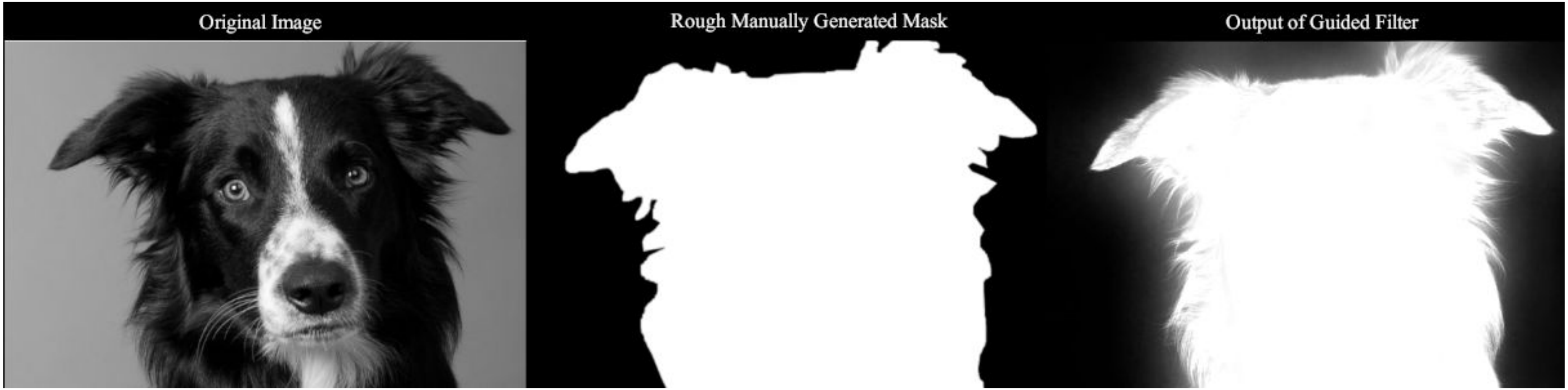
$$b_k = \bar{p}_k - a_k \mu_k$$

Combine all q_k into output image *q* by averaging all overlapping values of q_k ∀ *k*

Other Applications of Guided Filtering

Blurring with Edge Preservation

Binary Mask Refinement



Dehazing Results