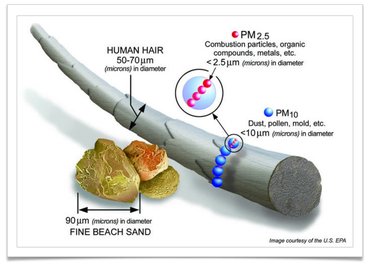
Image Processing Algorithms for Haze(PM2.5) Mapping

Qin Li, Sophomore, Electrical Engineering, ENG
Nuochen Lyu, Sophomore, Electrical Engineering, ENG

ILLINOIS

Introduction

Project overview:
—Building a Matlab based image processing algorithm —Analyze the picture taken from remote satellites.
—Drawing a 2D air quality map depicting air pollution distribution by image digitalization and pattern recognition.



What is PM2.5?
—Particulate matter (PM) is a term used to describe the mixture of solid particles and liquid droplets in air.
—PM2.5 means the mass per cubic meter of air of particles with a size (diameter) less than 2.5 (μm).
—PM2.5 index determine the air quality

Why dangerous?
— stay longer in the air (small and light)
— able to bypass the nose and throat and penetrate deep into the lungs

PM2.5 = Air Pollution

Pros

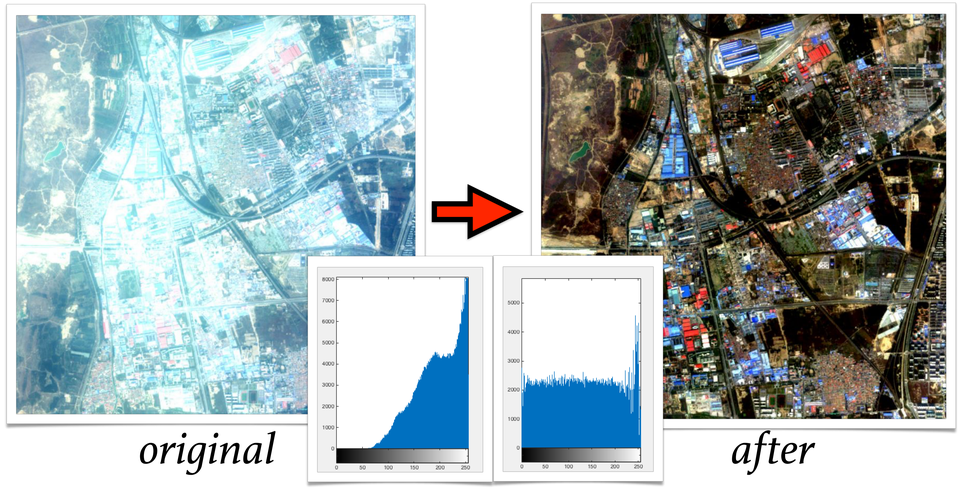
very fast
—matlab base code involving lots of built in library

zero cost
—need only one picture as input
—can run on any personal computer

direct
—good approximation on haze

Cons

inaccuracy
—unable to see a digit in ones place
—that is all…

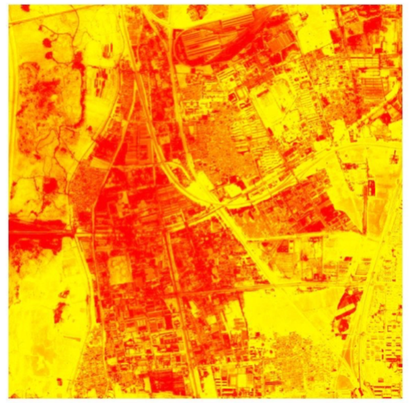


Step1:Defog

The original picture is hazy with RGB value centered around 200-255 (The reason why the picture appears white). Use histogram equalization function to defog the picture. The function will force the RGB value uniformly distributed from 0 to 255 so the pale picture becomes colorful and the contrast of the picture increased.

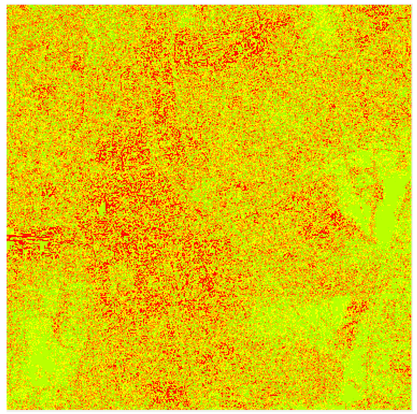
Step2:Segmentation

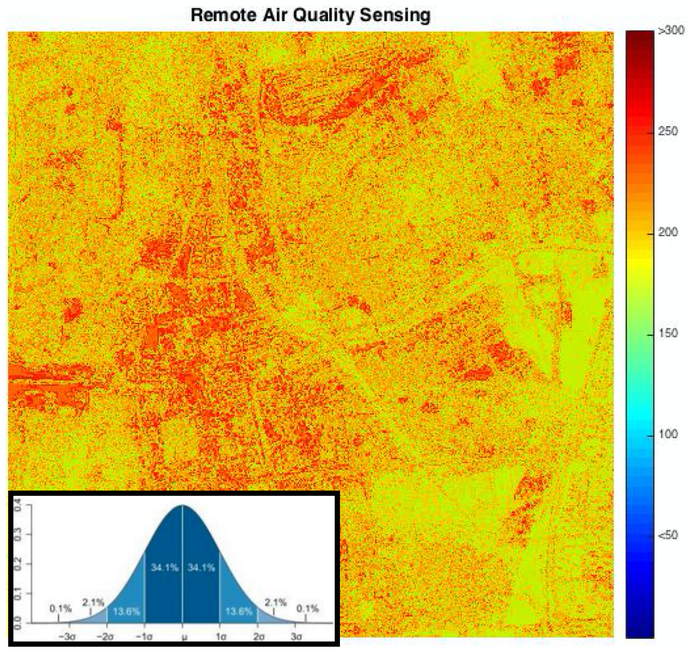
—Then we compare the defogged picture with the original picture on their RGB histogram to get a general picture of fog pattern and terrain patterns.
—The urban is segregate into different areas such as forests, sweets and houses.
—The boundary of different area is clear
—The fog pattern needs further process.



Step3:Discrimination

—In this step we generate another two picture with different degrees of histogram equalization 200 and 199
—This allows us to see the tiny difference and how distinct the RGB value change during the hiseq()
—We can get a preliminary distribution of fog in this urban area
—However, there are still error in this picture that needs correction. Error happens mostly in mountain area because the low reflection of grassland.





Final step:GUI design & error adjustment

—We combine the picture in step2 & 3 together to correct the error in step3
—Then we normalize the histogram data and assign these value according to Gaussian distribution
—Also we take the AQI index in the real world to adjust our distribution scale
—With normalize data, we can use the algorithm to process all the pictures
—Finally we pack the code into a GUI interface so everyone can use it.