Problem Set #1, SP part

Issue date: Nov. 21, 2021; Deadline: 23:59, Dec. 5, 2021

Student Name:_____Student No.: _____

1. *Demosaicing and color balance*

(1) *Demosaicing*

The initial image is given as "Julia.mat". Please use the demosaicing algorithm to convert this single-channel image to a three-channel RGB image. Name the output image as "RGBImage".

(2) *White balance*

Apply the white balance on "sky.jpg" and name the output image as "WhiteBalanceImage".

(3) *Maximum value balance*

Apply the maximum value balance on "sky.jpg" and name the output image as "MaxValueBalanceImage".

For figure.1:
Display "Julia.mat" in subplot(2,2,1); Display "RGBImage" in subplot(2,2,2);
Display "WhiteBalanceImage" in subplot(2,2,3); Display "MaxValueBalanceImage" in subplot(2,2,4).

White balance:

$$I(x,y) = 0.299f_R(x,y) + 0.587f_G(x,y) + 0.114f_B(x,y)$$

$$k_R = \frac{\bar{I}}{\bar{f_R}} \quad k_G = \frac{\bar{I}}{\bar{f_G}} \quad k_B = \frac{\bar{I}}{\bar{f_B}}$$

$$\begin{bmatrix} g_R(x,y) \\ g_G(x,y) \\ g_B(x,y) \end{bmatrix} = \begin{bmatrix} k_R & & \\ & k_G & \\ & & k_B \end{bmatrix} \begin{bmatrix} f_R(x,y) \\ f_G(x,y) \\ f_B(x,y) \end{bmatrix}$$

Maximum value balance

$$S_{RGB} = \min[R_{max}, G_{max}, B_{max}]$$

$$k_R = \frac{S_{RGB}}{T_R} \quad k_G = \frac{S_{RGB}}{T_G} \quad k_B = \frac{S_{RGB}}{T_B}$$

$$\begin{bmatrix} g_R(x,y) \\ g_G(x,y) \\ g_B(x,y) \end{bmatrix} = \begin{bmatrix} k_R & & \\ & k_G & \\ & & k_B \end{bmatrix} \begin{bmatrix} f_R(x,y) \\ f_G(x,y) \\ f_B(x,y) \end{bmatrix}$$

For white balance part, **f(x,y)** is the initial image and **g(x,y)** is the output image.

For maximum value balance part, **S** is the smallest max value in all color channels. Then calculate the number of intensities **Nr**, **Ng**, **Nb** that larger than **S** in each color channel and choose the largest number **N*max***. Sort the intensities in each channel and find the **N*max***th intensity value [**Tr**, **Tg**, **Tb**] in each color channel.

2. *Noise generation and degeneration*

(1) *Noise generation*

Add the following noise to the initial image separately:
Gaussian noise: mean and variance are 25 and 25 respectively;
Salt-and-pepper noise: intensities are 0 and 255 with probabilities **P*salt*** = **P*pepper*** = 0.05;
Name the output images as "NoiseImage1", "NoiseImage2" and display both images. You are free to use the function "imnoise" or other built-in functions to generate the noise.

(2) *Noise degeneration*

Please design a median filter and an average filter, and apply both your filters to the "NoiseImage1" and "NoiseImage2". Then display all your output image results.

3. *Histogram equalization and histogram matching*

(1) *Histogram*

Develop a function which computes the gray-level histogram of the original image. Then using this function to find the histogram and display it.

(2) *Histogram equalization*

Develop a function to apply histogram equalization on the original image. Then using this function to equalize the histogram which given out by 3.1. Display the output image and its histogram.

(3) *Histogram matching*

Develop a function to apply the histogram matching between the source image and the target image. Use this function to display the output image. And you also need to display the histogram of the source image, the target image and the output image.

- You are free to use the built-in function "imhist" to get the histogram in 3.2 and 3.3. However, in 3.1 **you must use** the function which designed by yourself to get the histogram.

** Please save your matlab codes as a PDF and submit it on gradescope.*

** Discussion on methodology is allowed, yet, the assignment should be done individually. Plagiarism, once found, grades zero for the whole homework assignment!!!*