Computer Vision Homework 1 Report

B04901147 電機四 黃健祐

Usage

python3 rgb2gray.py <input image>

Example:

python3 rgb2gray.py 0a.png

The output images will be “0a_y.png”, “0a_y1.png”, “0a_y2.png”, and “0a_y3.png”.

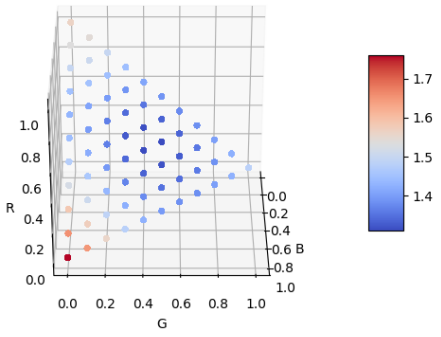
Implementation of local minima selection

Given a point (b_0, g_0, r_0), its neighbors are:

$$(b_0 - d, g_0 + d, r_0), (b_0 - d, g_0, r_0 + d)$$
$$(b_0 + d, g_0 - d, r_0), (b_0, g_0 - d, r_0 + d)$$
$$(b_0 + d, g_0, r_0 - d), (b_0, g_0 + d, r_0 - d)$$

where $d = 0.1$ in this homework.

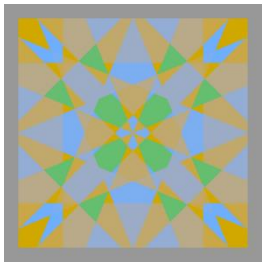
To search for local minima, a dictionary is created. The keys are BGR parameters while the values are costs. For each point, if its cost is less than all of the neighbors, then it is a local minimum and is recorded in a list. In case of edge points, when a given parameter does not exist in the dictionary, an infinity value will be returned.



- Cost plane for 0c.png($\sigma_s = 1$ $\sigma_r = 0.1$)

Result

■ Input Images

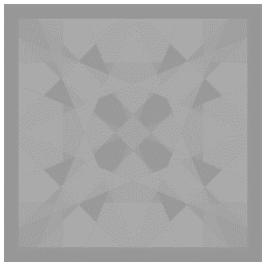



The luminance generated by a physical device is generally not a linear function of the applied signal. A conventional CRT has a power-law response to voltage; luminance produced at the face of the display is approximately proportional to the applied voltage raised to the 2.5 power. The numerical value of the exponent of this power function is colloquially known as *gamma*. This nonlinearity must be compensated in order to achieve correct reproduction of luminance.

As mentioned above (*What is lightness?*), human vision has a nonuniform perceptual response to luminance. If luminance is to be coded into a small number of steps, say 256, then in order for the most effective perceptual

■ Conventional RGB2GRAY Conversion

$$Y = 0.299R + 0.587G + 0.114B$$

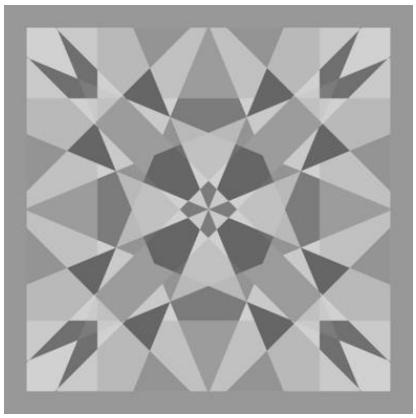



The luminance generated by a physical device is generally not a linear function of the applied signal. A conventional CRT has a power-law response to voltage; luminance produced at the face of the display is approximately proportional to the applied voltage raised to the 2.5 power. The numerical value of the exponent of this power function is colloquially known as *gamma*. This nonlinearity must be compensated in order to achieve correct reproduction of luminance.

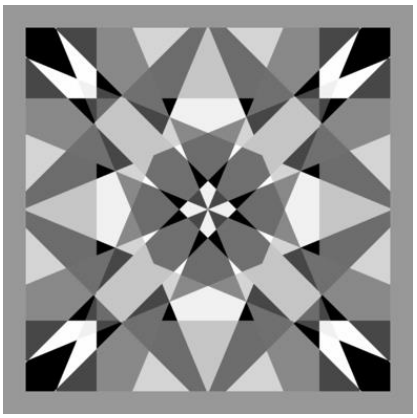
As mentioned above (*What is lightness?*), human vision has a nonuniform perceptual response to luminance. If luminance is to be coded into a small number of steps, say 256, then in order for the most effective perceptual

■ Advanced RGB2GRAY Conversion

1. 0a.png



$(w_b, w_g, w_r) = (0,0,1)$



$(w_b, w_g, w_r) = (1,0,0)$

◆ Number of votes

(w_b, w_g, w_r)	Vote
$(0,0,1)$	9
$(1,0,0)$	9

2. 0b.png

$(w_b, w_g, w_r) = (0,0,1)$

$(w_b, w_g, w_r) = (0,0.2,0.8)$

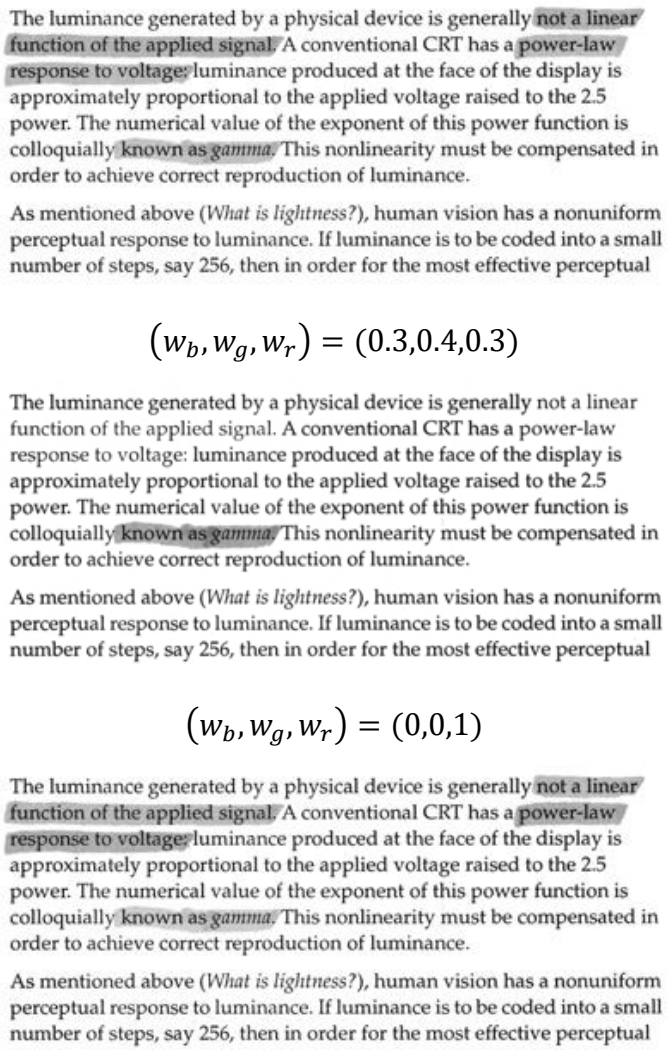
$(w_b, w_g, w_r) = (0,1,0)$

$(w_b, w_g, w_r) = (0.1,0.2,0.7)$

- Number of votes

(w_b, w_g, w_r)	Vote
(0,0,1)	3
(0,0.2,0.8)	3
(0,1,0)	2
(0.1,0.2,0.7)	2
(0,0.3,0.7)	1

3. 0c.png



The luminance generated by a physical device is generally not a linear function of the applied signal. A conventional CRT has a power-law response to voltage: luminance produced at the face of the display is approximately proportional to the applied voltage raised to the 2.5 power. The numerical value of the exponent of this power function is colloquially known as *gamma*. This nonlinearity must be compensated in order to achieve correct reproduction of luminance.

As mentioned above (*What is lightness?*), human vision has a nonuniform perceptual response to luminance. If luminance is to be coded into a small number of steps, say 256, then in order for the most effective perceptual

$$(w_b, w_g, w_r) = (0.3,0.4,0.3)$$

$$(w_b, w_g, w_r) = (0,0,1)$$

$$(w_b, w_g, w_r) = (0.2,0.5,0.3)$$

- Number of votes

(w_b, w_g, w_r)	Vote
(0.3,0.4,0.3)	3
(0,0,1)	2
(0.2,0.5,0.3)	2
(0,0.1,0.9)	1
(0.1,0.5,0.4)	1
(0.2,0.6,0.2)	1
(0.3,0.3,0.4)	1