**Computer Vision HW1 report**

電機四 B04901117 毛弘仁

Description of my code

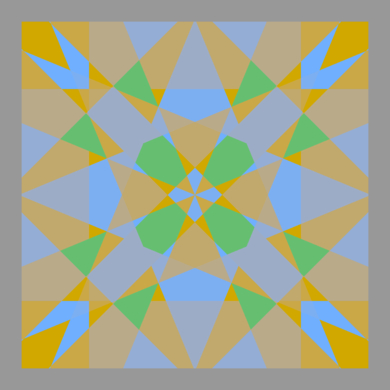
rbg2gray.py reads an input image (the image filename should be specified inside the code), feeds it through calculated bilateral and joint bilateral filters, then saves the cost function values in log files. The padding method used was BORDER\_REPLICATE, as I believe it makes sense when used with range kernels.

parse\_logs.py processes these log files, finding the RGB weight configuration local minima and tallying vote counts for them. My method of locating the local minima is to compare each point to its nearest 6 neighboring points (they are all distanced away). If a point is on the boundary (e.g. point ) , then it will have less neighboring points to compare with.

When writing my code, I was careful to avoid as many for loops as possible, and instead use broadcasting features of numpy which speeds up the calculation process. Also, I purposely incorporated information logging so that if something went wrong, I would not have to restart calculations from scratch.

Results

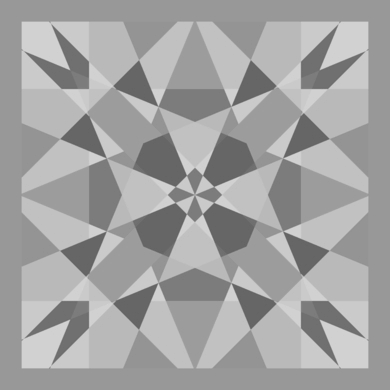
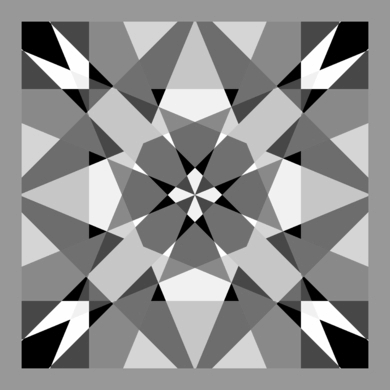
0a best candidates (the outputs are shown below from left to right):



|  |  |  |  |
| --- | --- | --- | --- |
| w\_b | w\_g | w\_r | votes |
| **0.0** | **0.0** | **1.0** | **9** |
| **1.0** | **0.0** | **0.0** | **9** |

input

conventional rgb2gray

0b best candidates (the outputs are shown below from left to right):



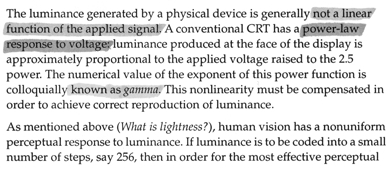
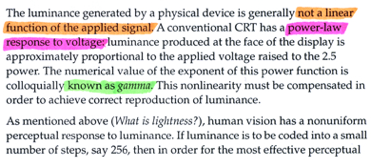
|  |  |  |  |
| --- | --- | --- | --- |
| w\_b | w\_g | w\_r | votes |
| **0.0** | **0.0** | **1.0** | **3** |
| **0.0** | **0.2** | **0.8** | **3** |
| **0.1** | **0.2** | **0.7** | **2** |

input

conventional rgb2gray

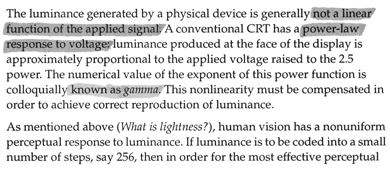
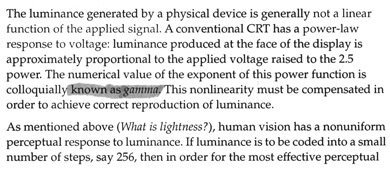
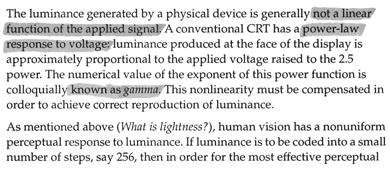
0c best candidates (the outputs are shown below from left to right):



input

conventional rgb2gray

|  |  |  |  |
| --- | --- | --- | --- |
| w\_b | w\_g | w\_r | votes |
| **0.3** | **0.4** | **0.3** | **3** |
| 0.0 | 0.0 | 1.0 | 2 |
| **0.3** | **0.5** | **0.2** | **1** |



I was surprised that the second best candidate was unable to clearly show all the highlight colors!