CmpE 537
Fall 2019
Assignment 1: Color Quantization

Gurkan Demir
2015400177

October 6, 2019

1 Introduction

Color quantization is the process of reducing number of colors used in an image while trying to maintain the visual appearance of the original image. In general, it is a form of cluster analysis, if each RGB color value is considered as a coordinate triple in the 3D colorspace.

In this homework, we are expected to implement a function **quantize(img, K)** which quantizes given image into K colors using Python. In order to accomplish project's main task, we are expected to use K-Means algorithm that we have learned in course. We can choose initial cluster centers by clicking on the image or randomly.

2 Implementation

1. Convert input image to a matrix of pixels.
2. Pick K initial colors to begin quantization by clicking on the image or randomly.
3. Implement K-Means Algoritm. Repeat followings 10 times.
   (a) Assign each pixel to closest clusters by comparing their R, G, B values.
   (b) Find clusters' new R, G, B values by getting means of assigned pixels' values.
4. Generate an output image.

3 How to Run?

In order to run color quantizer, execute the following from the command line:

python3 colorQuantizer.py [IMG] [K] [TYPE]
where;

1. **IMG :** Path of image want to be quantized.
2. **K :** Number of colors in an image we want to quantize.
3. **TYPE :** Type of selecting cluster centers.
   - **1** for choosing cluster centers by clicking on photo.
   - **2** for choosing cluster centers randomly.

For example, if you want to quantize **cat.png** into **8** colors, by choosing **random** cluster centers. You have to execute following command: **python3 colorQuantizer.py cat.png 8 2**

4 Outputs

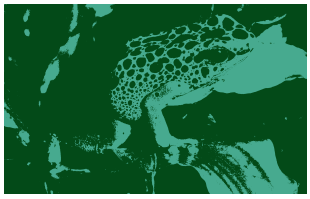
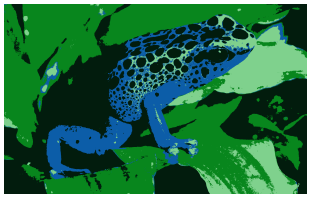
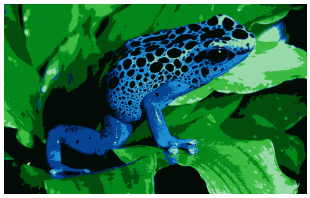
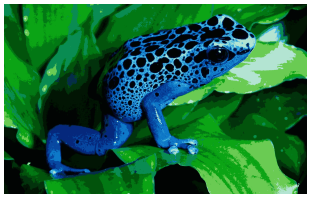
Inputs			
K = 2			
K = 4			
K = 8			
K = 16			
K = 32			

Table 1: Program outputs for given inputs with different K values.

5 Conclusion

The main reason we may want to perform this kind of compression is to enable the rendering of an image in devices supporting only a limited number of colors, usually due to memory limitations. Obviously, all compressions come with a cost. In this case, the resulting image may differ too much from the original one. The goal of the color quantization is to obtain a compressed image as similar as possible to the original one, therefore we need to increase K value. As can be seen in the figure above, increasing in K value results in better output. However, there is always trade-off, when we increase K, it takes much more time to complete execution. Also, computation time depends on size of image, if image has bigger width and height, it takes more time.