**HACETTEPE UNIVERSITY**

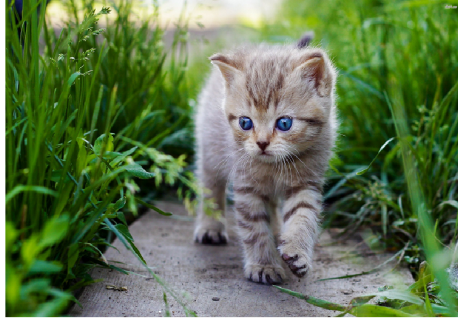Department of Computer Engineering

**Due Date: 23:59 pm on Monday, November 14th, 2022**

## Color Transfer Between Images



Source Image     Target Image     Result Image

Figure 1: Sample images for color transfer.

## Background

Color manipulation in photographs is one of the most fundamental task for the image editing tools such as Adobe Photoshop, Gimp, Instagram, etc. However, it is complicated job to give desired color tones in strongly correlated color spaces such as RGB. More precisely, if we want to change the appearance of a pixels color in a coherent way, we must modify all color channels in tandem.For this reason, new color spaces are developed to minimize the correlation between channels for natural scenes such as lab color space of Ruderman et al.[1] In this color space the l axis represents an achromatic channel,while the a and b channels are chromatic yellowblue and redgreen opponent channels. Such a color space makes more easier to give desired color effect to images. One interesting method uses lab color space was proposed by Reinhard et al.[2] to make a new image which takes another image's color tones as shown in Figure 1. The method uses simple statistics of lab channels to transfer color between source and target images.

## Overview

In this assignment, your goal is to implement the color transfer method of Reinhard et al. and analyze RGB histograms of before and after images. Moreover, you will improve this method by transferring the color between the similar regions of source and target images. Your program will take source $I_s$ and target $I_t$ images and produce a new image which contains the similar color tones with target image on source image as in Figure 1. It should have the following structure:

### Part 1

1. Convert the source and target images to $l\alpha\beta$ color space representation from RGB color space.

2. For each channel of source and target images, compute means $(\mu_l, \mu_\alpha, \mu_\beta)$ and standard deviations $(\sigma_l, \sigma_\alpha, \sigma_\beta)$.

3. Subtract means from the data points for target image:

$$l^* = l - \mu_l$$
$$\alpha^* = l - \mu_\alpha$$
$$\beta^* = l - \mu_\beta$$

4. Scale new data points according to the relative standard deviations of target and source images:

$$l' = l^* \frac{\sigma_s^l}{\sigma_t^l}$$

$$\alpha' = \alpha^* \frac{\sigma_s^\alpha}{\sigma_t^\alpha}$$

$$\beta' = \beta^* \frac{\sigma_s^\beta}{\sigma_t^\beta}$$

5. Add the averages computed for the source to scaled data points.

## Part 2

Differently from Part 1, you will divide the source and target images to equals rectangular regions and transfer colors between the most similar regions between source and target images. You can find the true number of regions experimentally (You must also report these experiments). You can use the following measures to find similar regions:

- Sum of squared differences (SSD)

- Normalized cross-correlation (NCC)

## The Report

You must analyze and show the histograms of input and output images. You must use similarity measures on both color space and compare the results in your report for Part 2. You must show failed examples and comment why you did not get satisfactory results. You can devise a new algorithm or modify the method of Reinhard et al.[2] to get better results to earn extra points. Your LATEX report should contain a brief overview of the problem, the details of your approach, and the results of your algorithm on at least 3 images with your comments. Show the results of all of the main steps. If your algorithm failed to give a satisfactory result on a particular image, provide a brief explanation of the reason(s).

## What to Hand In

Your submission format will be:

- README.txt *(give a text file containing the details about your implementation, how to run your code, the organization of your code, functions etc.)*

- code/ *(directory containing all your code)*

- report.pdf *(PDF report - LATEX)*

Archieve this folder as **b<studentNumber>.zip** and submit to `https://submit.cs.hacettepe.edu.tr`.

## Grading

The assignment will be graded out of 100:

- CODE: 0 (no implementation), 40 (a partial solution), 60 (a correct solution) and REPORT : 40

## Academic Integrity

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) will not be tolerated. In short, turning in someone else's work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else.

# References

[1] D.L. Ruderman, T.W. Cronin, and C.C. Chiao, Statistics of Cone Responses to Natural Images: Implications for Visual Coding, J. Optical Soc. of America, vol. 15, no. 8, 1998, pp. 2036-2045.

[2] Erik Reinhard, Michael Ashikhmin, Bruce Gooch and Peter Shirley, 'Color Transfer between Images', IEEE CG&A special issue on Applied Perception, Vol 21, No 5, pp 34-41, September - October 2001.

[3] Neural color transfer comparisons, `https://mingminghe.com/neural_color_transfer/comparison.html`