

数学建模实验报告

实验一 图像颜色变换

姓名: 王世烜

学号: PB20151796

学院: 大数据学院

001139.01 数学建模
(春季, 2023)

中国科学技术大学

2023 年 3 月 16 日

摘要

图像颜色迁移 (Color Transfer) 是指对于给定的两幅图像 I_s, I_t , 我们要求在保留原图像 I_s (Source Image) 的内容的条件下根据目标图像 I_t (Target Image) 的颜色风格生成新的图像 I' 。

而在一幅图像中, 我们用像素颜色在颜色空间内的分布情况来表示图像的颜色风格。于是问题便转化为求解一个颜色空间的变换, 使得变换之后的新图像和目标图像在颜色空间内具有类似的分布。

本文基于"Color transfer between images"[1] 以及"Histogram matching"[2] 分别完成了两种不同的图像颜色迁移的解决办法。

关键词: 图像颜色迁移、直方图匹配

Abstract

Color Transfer refers to the process of generating a new image I' , based on a target image I_t 's color style while preserving the content of a given source image I_s . In an image, the distribution of pixel colors in the color space represents the image's color style. The problem is then to find a color space transformation that produces a new image with a similar color distribution to the target image in the color space, while preserving the content of the source image.

In this paper, we present two different solutions for image color transfer based on "Color transfer between images" [1] and "Histogram matching" [2], respectively.

Key Words: Color Transfer, Histogram matching

目 录

1 前言	4
2 问题分析	4
3 数学模型建立	4
3.1 Reinhard 提出的经典算法	4
3.1.1 颜色空间变换	4
3.1.2 拟合	5
3.2 直方图匹配算法	5
3.2.1 原理	6
3.2.2 执行过程	6
4 结果与对比	6
5 结论	7
6 问题	8
A 结果展示	9

1 前言

随着计算机图像处理技术的不断发展，图像颜色迁移技术已成为图像处理领域的热门研究方向之一。图像颜色迁移旨在将一幅图像的色彩风格转移到另一幅图像上，以改变图像的色彩表现形式，从而达到美化图像、增强图像视觉效果的目的。近年来，利用深度学习技术来实现图像颜色迁移，具有较好的迁移效果和计算速度，能够广泛应用于图像美化、图像风格迁移、图像特效制作等领域。

但是由于神经网络对于计算资源的需求较大，所以本文对于两种传统算法进行了复现和对比。

2 问题分析

图像颜色风格迁移也可以看作两个在 RGB 空间的点集进行拟合的过程，我们可以首先考虑它们的两个最主要的统计学特征。包括一-范数平均数和二-范数方差。但 RGB 空间对颜色性质的展现不够深入，也不接近人类感知。文章 [1] 采用了以下方法解决了这个问题：我们首先考虑将 RGB 空间转换为基于人类感知的 lab 色彩空间的方法。其中 l 分量用来表示颜像素的亮度，a 表示从红色到绿色的范围，b 表示从黄色到蓝色的范围。这样做可以保证各个通道之间的依赖性较低，从而为后面的操作带来便利。

而直方图匹配算法通过估算 source image 的灰度累计分布与 target image 的累积分布构造一个从 source image 到 target image PDF 中具有相同累积分布值的映射，从而达到颜色风格的迁移。

3 数学模型建立

3.1 Reinhard 提出的经典算法

在转换为 lab 颜色空间后，我们以像素在颜色空间中分布的均值和方差（标准差）两个量作为衡量这一分布的特征。对源图像 I_s 的颜色进行简单的平移和伸缩变换，使得对应的均值和标准差和目标图像 I_t 一致，这时，我们就认为得到图像 I 的颜色风格和 I_t 比较接近。

3.1.1 颜色空间变换

整个过程可由下述三步完成：

第一步从 RGB 空间转为 XYZ 空间：

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.5141 & 0.3239 & -0.1604 \\ -0.2651 & 0.6702 & 0.0641 \\ 0.0241 & 0.1228 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

第二步从 XYZ 空间转为 LMS 空间：

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3897 & 0.6890 & -0.0787 \\ -0.2298 & 1.1834 & 0.0464 \\ 0.0000 & 0.0000 & 1.0000 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

第三步从 LMS 空间转为 lab 空间：

这一步首先需要取一个对数变换：

$$\begin{cases} L = \log L \\ M = \log M \\ S = \log S \end{cases}$$

再结合一个线性变换：

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix}$$

便完成了从 LMS 空间转为 lab 空间。

3.1.2 拟合

拟合也可分为三步进行：

减去原平均值：

$$\begin{cases} l^* = l - \langle l \rangle \\ \alpha^* = \alpha - \langle \alpha \rangle \\ \beta^* = \beta - \langle \beta \rangle \end{cases}$$

拟合方差：

$$\begin{cases} l' = \frac{\sigma_t^l}{\sigma_s^l} l^* \\ \alpha^* = \frac{\sigma_t^\alpha}{\sigma_s^\alpha} \alpha^* \\ \beta^* = \frac{\sigma_t^\beta}{\sigma_s^\beta} \beta^* \end{cases}$$

加上新的平均值：

$$\begin{cases} l'' = l' + \langle l_t \rangle \\ \alpha'' = \alpha' - \langle \alpha_t \rangle \\ \beta'' = \beta' - \langle \beta_t \rangle \end{cases}$$

就完成了拟合。

3.2 直方图匹配算法

直方图匹配又称为直方图规定化，是指将一幅图像的直方图变成规定形状的直方图而进行的图像增强方法。即将某幅影像或某一区域的直方图匹配到另一幅影像上。使两幅影像的色调保持一致。可以在单波段影像直方图之间进行匹配，也可以对多波段影像进行同时匹配。两幅图像比对前，通常要使其直方图形式一致。

3.2.1 原理

考虑输入图像的灰度的概率密度函数为 $p_r(r)$, 我们所希望的输出图像应该具有的概率密度函数为 $p_z(z)$ (已指定)。

令 s 为一个有如下特性的随机变量:

$$s = T(r) = (L - 1) \int_0^r p_r(w) dw$$

接着, 我们定义一个有如下特性的随机变量 z :

$$G(z) = (L - 1) \int_0^z p_z(t) dt = s$$

由这两个等式可得到 $G(z) = T(r)$, 因此, z 必须满足以下条件:

$$z = G^{-1}(T(r)) = G^{-1}(s)$$

这样, 由输入图像估计出 $p_r(r)$ 并通过以上式子求出 s 以及 $G(z)$, 就可以利用 $z = G^{-1}(s)$ 求出输出后的 z 。

3.2.2 执行过程

在具体实现过程中, 完成以下步骤即可实现直方图匹配算法:

1. 计算 source image 以及 target image 的直方图 src-hist、tar-hist
2. 灰度级 g 在 src-hist 对应的值为 src-hist g , 在 tar-hist 中找出与它最接近的值 tar-hist G 所对应的灰度级 G , 建立 g 与 G 之间的映射
3. 根据 g 与 G 的映射关系得到 src 经过匹配之后的直方图
4. 输出转换之后的图像

4 结果与对比

首先给出 Reinhard 论文中使用的一组图片的结果:

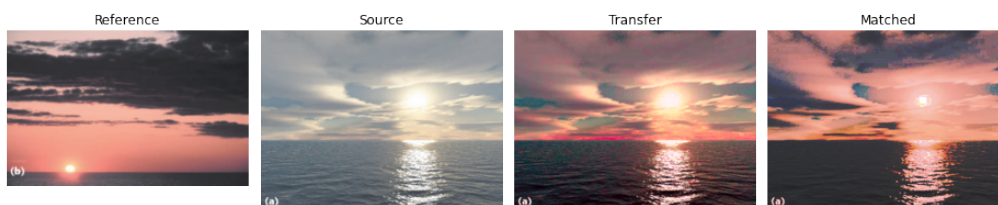


图 1: group1

给出 Reinhard 算法与直方图匹配算法中源图片、目标图片与输出图片的 RGB 分布直方图:

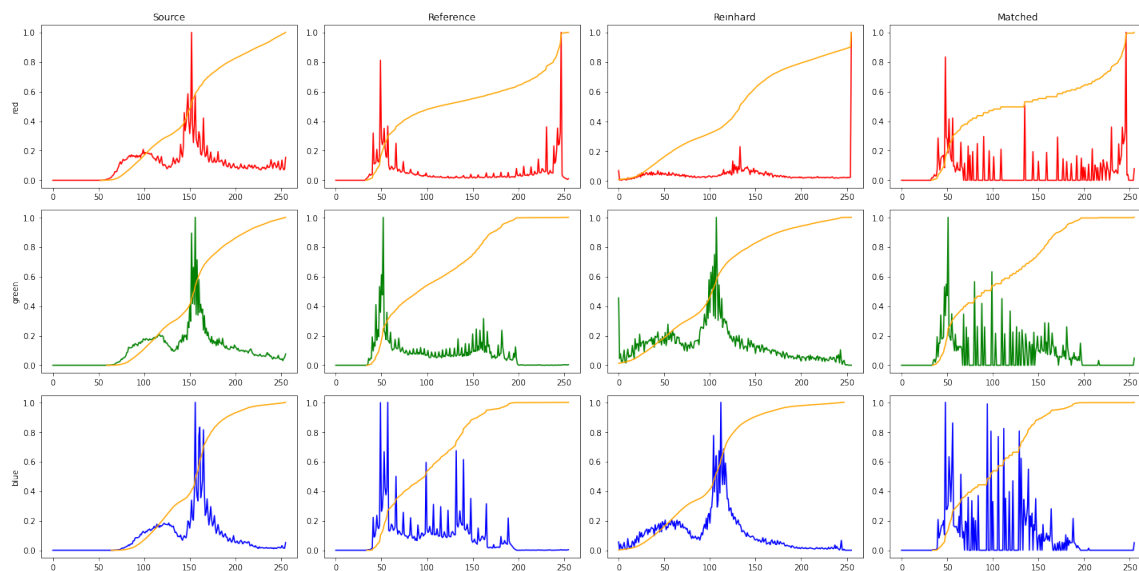


图 2: group1-rgb-hist

再给出四张图片的 RGB scatter plot:

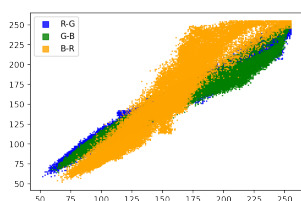


图 3: source

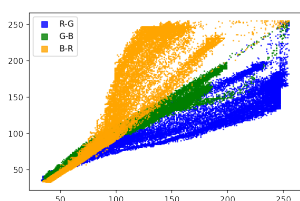


图 4: target

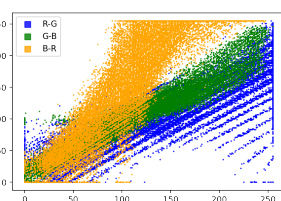


图 5: Reinhard

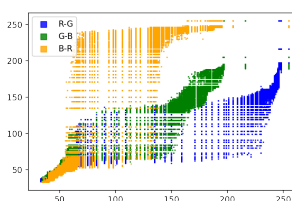


图 6: Matched

其他几组图片结果请参见附录A

从图1可以很明显的看出 Reinhard 算法相较于直方图匹配算法得到的结果更优, Reinhard 算法处理后的图片, 不仅对图像的颜色风格进行了迁移, 还很大程度的保留了原图像的形状, 而直方图匹配算法处理后的图片对于形状的保留则并没有十分明显。如图中的太阳部分, 直方图匹配算法得到的太阳部分明显缺少了边缘的光晕部分。

通过观察 RGB 的分布直方图图2, 我们发现 Reinhard 算法大体上保留了原图片的颜色分布 (如峰个数), 而直方图匹配算法则完全的将目标图像的直方图分布复制过去, 这样做虽然可以保证颜色风格的迁移, 但却忽略了原始图像的颜色分布, 就会导致生成图像的形状会与原图像有所差别。

5 结论

本文实现了 Reinhard 提出的算法以及直方图匹配算法并进行了对比。从上面的实验结果以及分析, 我们可以得到, Reinhard 提出的算法的效果要明显优于直方图匹配算法。Reinhard 提出的算法通过 RGB 空间到 lab 空间的变换和均值方差的拟合, 通过无论从直接的观感, 还是从 RGB 值变化, 都较迁移前图片与被迁移图片更加相似, 实现了源图片到目标图片的颜色流畅迁移。

6 问题

事实上在 Reinhard 提出的算法上可以进行改进。Reinhard 方法是应用于整幅图像上, 如果 I_s, I_t 的颜色分布比较单调, 已经可以得到一个比较好的结果。但是, 这种整体性的方法对于 I_s 和 I_t 具有复杂颜色分布的情形很难得到良好的表现, 可能会丢失一些原图像的细节信息。比较直接的想法是对 I_s 和 I_t 颜色空间的样本点分别进行聚类划分, 再由若干次局部的颜色变换得到 I , 从而降低复杂的颜色分布对颜色变换的影响。但是这样做会消耗大量的资源, 开销很大。

参考文献

- [1] E. Reinhard, M. Adhikhmin, B. Gooch and P. Shirley, "Color transfer between images," in IEEE Computer Graphics and Applications, vol. 21, no. 5, pp. 34-41, July-Aug. 2001, doi: 10.1109/38.946629.
- [2] Gonzalez RC Woods RE. "Digital Image Processing". Fourth ed. New York NY: Pearson; 2018.

附录 A 结果展示

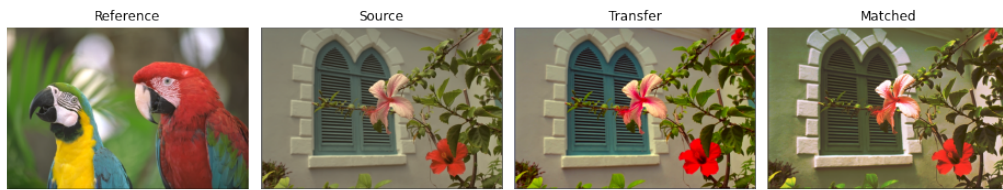


图 7: group2



图 8: group3

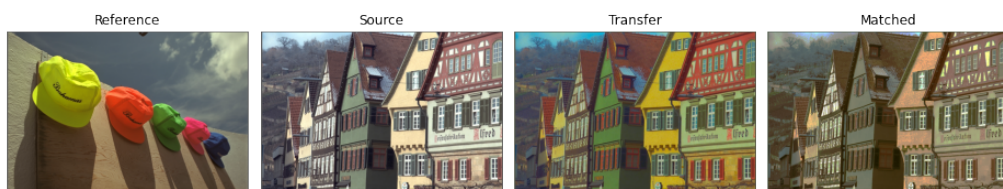


图 9: group4



图 10: group5