

# 彩色图像灰度化 (Color2Gray)

袁雨 001139.01 PB20151804

2023 年 3 月 20 日

## 摘要

本文基于彩色图像灰度化问题，综述了三种数学建模方法，包括线性量化模型、线性降维模型和非线性模型。其中，线性量化模型采用平均值法、最大值法和分量法等方法，将颜色空间中的三个分量线性加权变换为一维灰度数值，不仅计算量少，而且简洁易懂，底层支持广泛。线性降维模型采用主成分分析（PCA）方法，通过把多个高维颜色空间中的最主要特征映射到一维空间中，在保留图像主要信息的同时，抑制噪声和冗余信息，具有较好的效果。非线性模型采用 2005 年 Gooch 等人提出的 Color2Gray 算法和 2017 年 Nafchi 等人提出的一种基于图像相关性算法。Color2Gray 算法的主要优点是能更多地保留原始彩色图像的颜色对比信息。相关性算法是直接原图像估算投影系数。

本文实验证明，以上三种模型都能够有效地将彩色图像转换为灰度图像。在对比实验中，线性量化模型处理速度最快，但失真率较高；线性降维模型在不同类型的图片上效果不同，大多数较好，但计算量较大；而非线性模型的效果类似于线性降维模型，但同时能够更好地保留图像细节和特征。因此，选择灰度化模型应该综合考虑准确度、速度和性能等方面的因素，选择最适合实际应用场景的方法。

## 一、前言（问题的提出）

彩色图像涉及到我们生活的方方面面，在电视、摄影、印刷等日常活动中扮演着重要的角色。随着数字图像处理和现代科技的不断进步，灰度图像因其输入信息量小，能提高模式识别和图像处理系统预处理阶段的系统运行速度，降低对系统内存大小的占有量，从而被广泛的用于指纹档案管理、运动目标追踪、医学影像、人脸检测和黑白打印等领域。因此，彩色图像灰度化是数字图像处理中一个极为重要的研究方向，具有非常重要的研究意义。

在计算机中，彩色图像可以表达为彩色数字图像，且一幅数字图像是由大量图像像素，即离散的点的集合构成。灰度图像在每个像素点有唯一确定的值，这个值决定了像素的灰度值。不同的是，灰度图像仅用一个确定值来描述像素点的信息，而彩色图像的像素是一个矢量函数，通常包含了 3 个值，其取值范围构成了一个向量空间，这个空间被称为彩色空间。彩色图像灰度化研究的问题就是如何用灰度值最大可能地表达彩色像素的矢量函数的内容。

## 二、相关工作

随着灰度图像应用的不断扩大，彩色图像转灰度图像的研究已经受到国内外越来越多研究者的关注。在短短几年的时间里，国内与国外均呈现出大量的针对彩色图像灰度化处理的研究工作，为了让输出灰度图能更多地保留原始彩色图像的对比度信息，学者们提出了许多具有针对性的灰度化算法，彩色图像灰度化技术随之也层出不穷。

## 三、问题分析

### 3.1 基本色彩空间

色彩是客观存在的也是可以主观感知的。在数字图像处理中，为了能够更加直观便捷地表示颜色，人们提出了很多种色彩模型。

#### 3.1.1 RGB 色彩空间

RGB 颜色模型广泛用于彩色监视器和计算机，它以红、绿、蓝三种颜色为原色。一个颜色显示的描述是由三个数值控制的，分别为 R、G、B。但三个数值位为最大时，显示为白色，当三个数值最小时，显示为黑色。使用 0 到 255 之间的整数，八位数字表示，通常表示为十进制和十六进制的数值，如 (255, 0, 0) 或 FF0000 (十六进制)。颜色通常都是用三种成分来定义的，于是便采用三维空间来进行描述，把三种成分的数值当做欧几里得空间中普通笛卡尔坐标系的坐标值。一个 RGB 组合 (r,g,b) 表示代表一个给定颜色的点在立方体内部、表面或者边上的三维坐标。这种表示方法使得在计算两个颜色相近程度时只需简单计算它们之间的距离：距离越短颜色越接近。图 1 呈现了三原色叠加的效果，图 2 为该模型所

考虑的彩色子空间，其中 X、Y、Z 轴分别表示红色、绿色和蓝色。三种原色的值分别居于 3 个角上，黑色居于原点，白色则处于距离原点最远的角上。

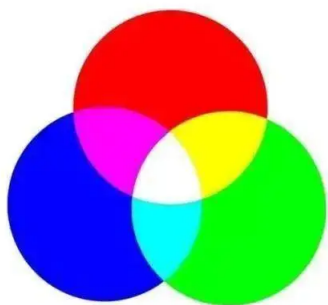


图 1: RGB 三原色

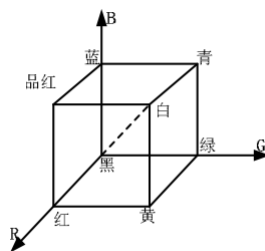


图 2: RGB 立方体

### 3.1.2 CIE LAB 色彩空间

CIE LAB 将颜色用三个值表达示：“L\*”代表感知的亮度、“a\*”和“b\*”代表人类视觉的四种独特颜色：红色、绿色、蓝色和黄色。LAB 模型中的纵轴 L 分量用于表示像素的亮度，取值范围是  $[0, 100]$ ，表示从黑色到白色；AB 平面代表图像的色度信息，A 轴表示从红色变化到绿色，取值范围是  $[-127, 127]$ ；B 轴表示从黄色到蓝色的范围，取值范围为  $[-127, 127]$ 。

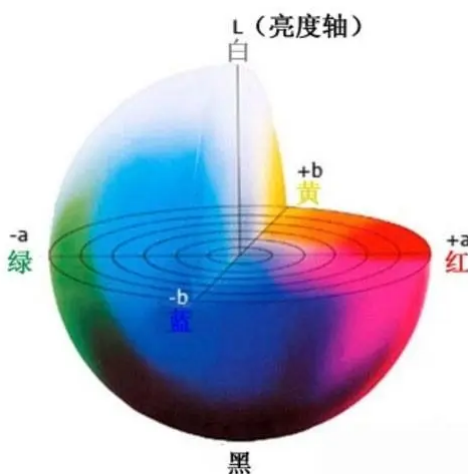


图 3: CIE LAB 空间

## 3.2 评价方法

### 3.2.1 主观评价方法

主观评价方法一般情况下是通过评价人员的视觉观察和主观感觉来对彩色转灰度图像的好坏直接进行评价。主观评价方法容易实现，而且比较直观，但是主观评价结果极易受到评价人员及其他外界因素影响。

### 3.2.2 客观评价方法

客观评价方法是对灰度图像进行某些方面的定量分析, 消除主观因素的影响, 对所采用的彩色图像灰度化技术在有效性方面进行更加客观、科学和定量的分析与评价。

#### (1) 结构相似度 (SSIM)

2004 年, Wang 和 Bovik 两人提出了一种新的图像质量客观评价指标——结构相似度 (SSIM) <sup>[47]</sup>。图像  $x$  与  $y$  二者之间的结构相似度被定义为:

$$SSIM(x, y) = [l(x, y)]^\alpha \cdot [c(x, y)]^\beta \cdot [s(x, y)]^\gamma$$

其中,  $\alpha \beta \gamma$  均是大于 0 的常数, 用来代表对应部分在 SSIM 中的重要程度;  $l(x, y)$  表示图像  $x$  与  $y$  二者的亮度比较函数、 $c(x, y)$  表示图像  $x$  与  $y$  二者的对比度比较函数、 $s(x, y)$  表示图像  $x$  与  $y$  二者的结构比较函数, 定义如下:

$$l(x, y) = \frac{2\bar{x} \cdot \bar{y} + C_1}{\bar{x}^2 + \bar{y}^2 + C_1}, \quad c(x, y) = \frac{2\sigma_x \cdot \sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \quad s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x \cdot \sigma_y + C_3}$$

其中,  $\bar{x}$  代表图像  $x$  的灰度平均值;  $\bar{y}$  代表图像  $y$  的灰度平均值;  $\sigma_x$  和  $\sigma_y$  分别代表了图像  $x$  及图像  $y$  的方差;  $\sigma_{xy}$  代表的则是图像  $x$  和图像  $y$  的协方差; 数  $C_i (i = 1, 2, 3)$  为常数。在实际使用时, 通常取  $\alpha = \beta = \gamma = 1, C_3 = C_2/2$ , 因此式 (2-46) 可表示为:

$$SSIM(x, y) = \frac{2\bar{x} \cdot \bar{y} + C_1}{\bar{x}^2 + \bar{y}^2 + C_1} \cdot \frac{2\sigma_{xy} + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}$$

其中, SSIM 的取值范围为  $[-1, 1]$ , SSIM 取值越接近 1, 则说明图像  $x$  与图像  $y$  的相似程度越高。

## 四、建模的假设

像素的颜色值可以表示为三原色 (红、绿、蓝) 的组合。我们可以进行数学建模, 并通过实验验证和优化来不断完善和优化模型。

## 五、符号说明

表 1: 符号说明

符号	说明	单位
$R$	颜色的 $R$ 分量	1
$G$	颜色的 $G$ 分量	1
$B$	颜色的 $B$ 分量	1

## 六、数学模型建立

### 6.3 线性量化模型<sup>[1]</sup>

#### 6.3.1 平均值法

平均值法的基本思想是将彩色图像的每个像素的颜色通道中的平均值作为该像素的灰度值。具体公式如下:

$$R' = G' = B' = (R + G + B)/3 = 0.333R + 0.333G + 0.333B$$

#### 6.3.2 YUV 算法

YUV 算法是经典的加权平均法,是通过 RGB 对当前 CRT 荧光屏校准得到的。该算法被广泛应用在数字图像处理软件中,例如图像编辑软件 Matlab 中的 `rgb2gray` 函数,具体公式如下:

$$R' = G' = B' = 0.2126R + 0.7152G + 0.0722B$$

#### 6.3.3 Luma 算法

Luma 算法也是加一种权平均法,但是和 YUV 算法中每个通道的权重不同,它是由标准的 RGB 三元色通过 Gamma 矫正得到的。具体公式如下:

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

#### 6.3.4 最大值算法

最大值法是一种简单的彩色图像灰度化方法,其基本思想是将彩色图像的每个像素的颜色通道中的最大值作为该像素的灰度值。具体公式如下:

$$R' = G' = B' = \max\{R, G, B\}$$

#### 6.3.5 分量算法

将彩色图像中的三分量的亮度作为三个灰度图像的灰度值,可根据应用需要选取一种灰度图像。具体公式如下:

$$R' = G' = B' = R$$

$$R' = G' = B' = G$$

$$R' = G' = B' = B$$

## 6.4 线性降维模型（主成分分析 PCA）

假设有一个 RGB 颜色模型的图像，可以将这个图像表示为一个三维矩阵，其中每个元素表示一个像素的 RGB 值。例如，对于一个具有  $m$  行  $n$  列像素的图像，可以表示为一个形状为  $m \times n \times 3$  的张量。要将彩色图像转换为灰度图像，可以使用 PCA 来提取出具有最高方差的 RGB 组合。这些组合将包含最多的图像信息，因此可以用它们来表示灰度图像。

具体而言，我们可以将 RGB 通道表示为一个矩阵  $X$ ，其中每行表示一个像素，每列表示一个通道。然后，我们可以对  $X$  进行 PCA 分解，以得到  $X$  的主成分。这些主成分将是一组新的归一化 RGB 通道，它们是  $X$  的线性组合。我们可以使用这些新的归一化通道来表示灰度图像。

更具体地说，第一步，设  $X$  为一个  $m \times n$  的矩阵，其中每行表示一个像素的 RGB 值。将  $X$  的每一列减去该列的平均值，使得每个通道的平均值为零。

第二步，计算  $X$  的协方差矩阵  $C$ ，其元素为： $C = \frac{1}{n-1} X X^T$

第三步，对  $C$  进行特征值分解，得到其特征向量和特征值。我们可以按照特征值的大小对特征向量进行排序，并选择前  $k$  个最大的特征值对应的特征向量。这些特征向量构成了新的归一化 RGB 通道，记为  $U_k$ 。

第四步，将原始 RGB 图像表示为一个张量  $X_{m \times n \times 3}$ ，然后将其重塑为一个  $m \times n$  的矩阵  $X'$ ，其中每行表示一个像素。然后，我们可以将每个像素的 RGB 值表示为一个行向量，并将其乘以  $U_k$ ： $Y = X' U_k$  其中  $Y$  是一个  $m \times k$  的矩阵，每行表示一个像素的灰度值。最终的灰度图像可以通过对  $Y$  中的每个元素进行缩放和平移来获得： $I = aY + b$  其中  $a$  和  $b$  是用于缩放和平移的常数， $I$  是输出的灰度图像。

需要注意的是， $U_k$  的选择是一个关键问题。如果我们选择的  $k$  太小，那么我们可能会失去一些重要的图像信息；如果我们选择的  $k$  太大，那么我们可能会保留一些不必要的信息，从而导致噪声增加。在实际应用中， $k$  的选择通常是通过试验来确定的。

## 6.5 非线性模型

### 6.5.1 color2gray 算法

2005 年 Gooch 等人提出了 color2gray 算法<sup>[2]</sup>，该算法在 Lab 色彩空间中通过相邻像素之间的色度和亮度差异来构造目标差异，主要有以下三个步骤：

第一步：把原始彩色图像从 RGB 颜色模型转化至 CIE Lab 颜色模型：首先从 RGB 颜色模型中转化至 XYZ 颜色模型：

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

再从 XYZ 色彩空间转换到 CIE Lab 空间：

$$\begin{aligned} L &= 116 * f(Y/Y_n) - 16 \\ a &= 500 * [f(X/X_n) - f(Y/Y_n)] \\ b &= 200 * [f(Y/Y_n) - f(Z/Z_n)] \\ f(t) &= \begin{cases} t^{1/3} & \text{if } t > (\frac{6}{29})^2 \\ \frac{1}{3} * (\frac{29}{6})^2 * t + \frac{4}{29} & \text{otherwise} \end{cases} \end{aligned}$$

为保证 XYZ 和 RGB 同等范围的映射， $X_n, Y_n, Z_n$  的取值分别为 0.950456 1.0 和 1.088754。

第二步：计算相邻像素之间的色度差异：  
给定：

$$crunch(x) = \alpha^* \tan(x/\alpha)$$

$$\vec{v}_\theta = (\cos \theta, \sin \theta)$$

然后：

$$\delta(\alpha, \theta)_{ij} = \begin{cases} \Delta L_{ij} & \text{if } |\Delta L_{ij}| > crunch(\|\Delta \vec{c}_{ij}\|) \\ crunch(\|\Delta \vec{c}_{ij}\|) & \text{if } \vec{c}_{ij} \cdot \vec{v}_\theta \geq 0 \\ crunch(-\|\Delta \vec{c}_{ij}\|) & \text{otherwise} \end{cases}$$

其中,  $\delta(\alpha, \theta)_{ij}$  为目标偏差, 如果绝对亮度差异小于色度差异, 则将其设置为色度差异, 否则将  $\delta(\alpha, \theta)_{ij}$  设置为亮度差异;  $L_i$  代表像素  $i$  的亮度,  $L_{ij}$  则表示  $(L_i - L_j)$  的缩写, 代表像素  $i$  和像素  $j$  之间的亮度差异,  $\Delta \vec{c}_{ij}$  表示彩色图像中相邻像素之间的色度差异;  $\alpha$  是一个控制色度差异取值范围的参数; 色差的符号由  $(\Delta \vec{c}_{ij}, \vec{v}_\theta)$  的符号决定, 其中  $\vec{v}_\theta$  是由  $\theta$  相对于  $\Delta a$  轴定义的归一化向量。

第三步：构造并求解目标函数：

$$f(g) = \sum_{(i,j) \in k} ((g_i - g_j) - \delta_{ij})^2$$

其中  $k$  表示相邻像素集,  $g$  表示源图像的亮度信息。初始化  $g$  为源图像的亮度通道, 然后使用共轭梯度法迭代下降到最小。

第四步：对求得的  $g$  进行平移, 使得求得的图像与源图像亮度最接近。

## 6.5.2 基于图像相关性的灰度化算法

2017 年, Nafchi 等人提出了一种基于图像相关性的彩色图像转灰度图像算法<sup>[3]</sup>, 此算法先在  $RGB$  颜色模型中构造了一个可能保持输入彩色图像对比度信息的对比图像; 然后再分别计算输入彩色图像的三个单通道图像和对比图之间的相关性; 最后按照一定的规则将求出的三个相关性指数映射为对应通道的加权参数, 从而得出灰度图像。具体算法如下:

首先, 通过输入彩色图像的平均图像  $\mu$  以及标准偏差图像  $\sigma$  构造一个能反映原彩色图像对比度的灰度图像  $Q$ , 具体公式如下:

$$\mu(i, j) = \frac{1}{3} \sum_{c=R,G,B} I_c(i, j)$$

$$\sigma(i, j) = \sqrt{\frac{1}{2} \sum_{c=R,G,B} |I_c(i, j) - \mu(i, j)|^2}$$

$$Q(i, j) = \mu(i, j) \times \sigma(i, j)$$

其次, 再通过下式分别计算出原彩色图像的三个单通道图像与图像  $Q$  的相关性指数  $\rho = \{\rho_{RQ}, \rho_{GQ}, \rho_{BQ}\}$

:

$$\rho_{X,Y} = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum (X_i - \bar{X})^2 \sum (Y_i - \bar{Y})^2}}$$

然后, 将相关性指数  $\rho$  映射为对于通道的加权系数, 映射规则如下:

$$\beta_c = \frac{|\rho_c|}{\sum |\rho|}$$

$$\gamma_c = \frac{\rho_c - \min \rho}{\max \rho - \min \rho} - 0.5$$
$$\lambda_c = |\beta_c + \min(\beta_c, \gamma_c)|$$

其中,  $c$  为图像通道顺序,  $\sum_{c=\{r,g,b\}} \beta_c = 1$ ,  $\gamma_c$  用于减小与对比度图  $Q$  反相关通道的加权参数, 且  $\gamma_c \in [-1, 1]$ 。

最后, 对参数  $\lambda$  进行归一化, 得到对应通道的加权系数, 从而得到最终的灰度图。

## 七、结果（与对比）

### 7.6 结果展示



图 4: 原图 1



图 5: 平均值



图 6: 最大值

### 7.7 结果对比

这里展示两幅测试图片。

第一副图是莫奈的画。r 分量法、Color2Gray 算法表现较好, 能区分出太阳、船、水、树木等; 最大值法反映出了颜色鲜明的太阳, 但忽略了其他颜色信息; g 分量、b 分量则丢失了较多信息; 平均值算法、加权平均值算法、PCA 算法中的太阳显示不明显; 相关性算法突出了太阳, 但将其显示为了深色。





图 7: 加权平均值 1



图 8: 加权平均值 2



图 9: r 分量



图 10: g 分量



图 11: b 分量



图 12: PCA



图 13: Color2Gray



图 14: 相关性



图 15: 原图 2

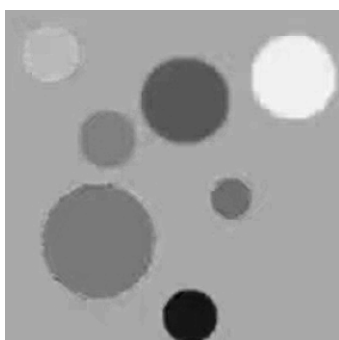


图 16: 平均值

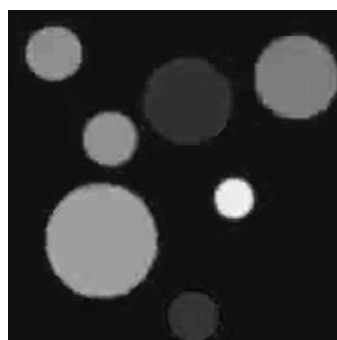


图 17: 最大值

第二幅图是圆形色块。PCA 算法表现较好，每个颜色都有区分度，颜色相近的色块深浅相近，且起到了一定的降噪作用；最大值法、分量法、相关性法丢失了一些颜色分量的信息；平均值算法、加权平均值算法、Color2Gray 算法中不同颜色的区别不大。

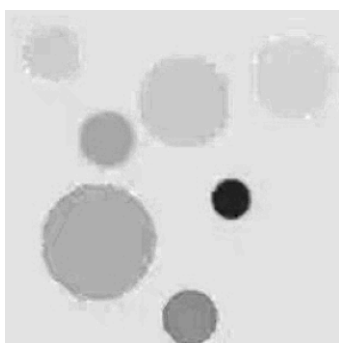


图 18: 加权平均值 1

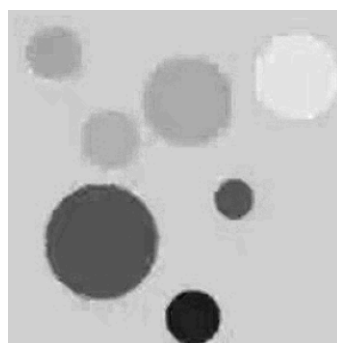


图 19: 加权平均值 2

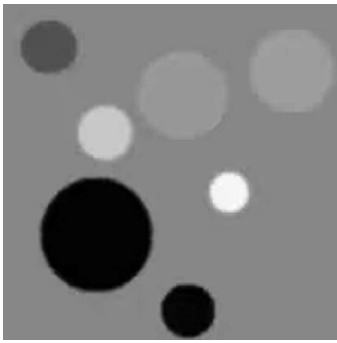


图 20: r 分量

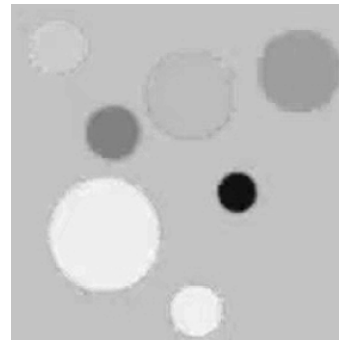


图 21: g 分量



图 22: b 分量



图 23: PCA

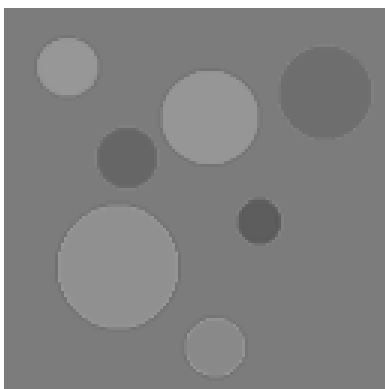


图 24: Color2Gray



图 25: 相关性

## 八、结论

本文围绕彩色图像灰度化问题，介绍了线性量化模型、线性降维模型和非线性方法，并通过实验对比分析了它们在不同图像上的效果。

实验结果显示，线性量化模型的优点是简单易懂，计算量较小，计算简单，在计算机内存和处理器速度都比较快，适合处理大量数据；其中平均值算法不能考虑每个颜色通道的亮度差异，可能会导致灰度图像的不准确。最大值算法能够较好地反映图像中色彩最鲜艳、最饱和的部分，但是可能会忽略掉其他颜色通道中的重要信息，造成灰度图像的亮度失真。分量算法会忽略掉其他分量的信息，造成信息损失。

PCA 算法的优点是可以提取出数据集的主要特征，比其他灰度化方法能更好地保留原始图像的信息。但对于特定颜色、光线、纹理变化、噪声比较大的彩色图像，可能会在灰度化后失去一些重要信息，导致 PCA 算法效果不佳。且由于其需要计算协方差矩阵等复杂运算，所以通常计算成本较高，不适用于实时场景。

Color2Gray 算法的主要优点是能更多地保留原始彩色图像的颜色对比信息，适合于具有少量不同色度值的有较大分隔区间的图像。缺点是由于该算法复杂度达到了  $O(n^2)$ ， $n$  为输入图像的像素数，故不太适用于实时处理和大批量处理。且对于自然风景之类的高动态图片以及色彩丰富的图片，灰度化效果并不理想。

相关性算法的优点是直接从原图像估算投影系数，不需要通过离散搜索的方式从有限的备选灰度图中寻找最优投影方向，简单便捷、运算效率高。缺点是在计算线性投影系数时只考虑了彩色图像三个单通道图像与对比图的相关性，而未考虑彩色图像三个单通道图像与对比图的亮度和对比度之间的关系，将绿色和红色几乎映射到了同一灰度级上，从而部分图像在灰度化过程中会出现特征信息丢失的情况，灰度化效果不稳定。

## 九、问题

在解决彩色图像灰度化的问题时，发现了一些问题和需要改进的地方。

线性量化模型虽然在某些情况下表现良好，但其对亮度和对比度变化的适应性有限。

PCA 算法在处理大型图像数据时，可能需要大量的计算资源，并且可能会受到噪声和不良样本的影响，导致降维效果不佳。

非线性方法可以在保留更多图像细节的同时产生高质量的灰度图像，但其计算量往往比线性方法高得多。Color2Gray 算法还有一些可以改进的地方，比如说可以去除手动指定值的需要，自动匹配值。或许有一些多尺度方法，可以提高算法速度，从而应用到更高分辨率的图案。在计算线性投影系数时只考虑了彩色图像三个单通道图像与对比图的相关性，而未考虑彩色图像三个单通道图像与对比图的亮度和对比度之间的关系。

因最近发烧耽误了时间，没能使用更多评价方法来对算法效果进行充分评估，在今后的学习中若有机会会去完善。

## 参考文献

- [1] 刘利刚, “数学建模 ppt,” 2023.
- [2] A. A. Gooch, S. C. Olsen, J. Tumblin, and B. Gooch, “Color2gray: salience-preserving color removal,” *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 634–639, 2005.
- [3] H. Z. Nafchi, A. Shahkolaei, R. Hedjam, and M. Cheriet, “Corrc2g: Color to gray conversion by correlation,” *IEEE Signal Processing Letters*, vol. 24, no. 11, pp. 1651–1655, 2017.