

BMP文件读写及RGB和YUV颜色空间转换

实 验 报 告

姓名：欧翌昕

专业：软件工程

学号：3190104783

课程名称：图像信息处理

指导老师：宋明黎

2020~2021秋冬学期 2020 年 10 月 25 日

目录

[1 实验目的和要求 2](#_Toc54202613)

[2 实验内容和原理 2](#_Toc54202614)

[2.1 BMP格式 2](#_Toc54202615)

[2.1.1 BMP文件结构 2](#_Toc54202616)

[2.1.2 位图文件头 2](#_Toc54202617)

[2.1.3 位图信息头 3](#_Toc54202618)

[2.1.4 调色板 4](#_Toc54202619)

[2.1.5 位图信息 4](#_Toc54202620)

[2.2 RGB和YUV颜色空间转换 4](#_Toc54202621)

[2.2.1 RGB转换为YUV 4](#_Toc54202622)

[2.2.2 YUV转换为RGB 4](#_Toc54202623)

[3 实验步骤与分析 5](#_Toc54202624)

[3.1 读取BMP文件 5](#_Toc54202625)

[3.2 写入BMP文件 6](#_Toc54202626)

[3.3 RGB转换为YUV 6](#_Toc54202627)

[3.4 YUV转换为RGB 7](#_Toc54202628)

[4 实验结果 8](#_Toc54202629)

[4.1 原始图 8](#_Toc54202630)

[4.2 灰度图 9](#_Toc54202631)

[4.3 处理图 9](#_Toc54202632)

[5 心得体会 10](#_Toc54202633)

# 1 实验目的和要求

1.读入一个 BMP 文件，由 RGB 转换为 YUV色彩空间

2.将 Y 值调整到[0,255]范围

3.写出灰度 BMP 格式图像

4.改变 Y 值，再转换为 RGB色彩空间 的 BMP 图像输出

# 2 实验内容和原理

## 2.1 BMP格式

### 2.1.1 BMP文件结构

位图文件头(Image file header)：共14字节；

位图信息头(Image information header)：共40字节；

调色板(Palette)：可选；

位图数据(Image data)；

### 2.1.2 位图文件头

bfType：2字节，文件类型；

bfSize：4字节，文件大小；

bfReserved1：2字节，保留，必须设置为0；

bfReserved2：2字节，保留，必须设置为0；

bfOffBits：4字节，从头到位图数据的偏移；

### 2.1.3 位图信息头

biSize：4字节，信息头的大小，即40；

biWidth：4字节，以像素为单位说明图像的宽度；

biHeight：4字节，以像素为单位说明图像的高度，同时如果为正，说明位图倒立（即数据表示从图像的左下角到右上角），如果为负说明正向；

biPlanes：2字节，为目标设备说明颜色平面数，总被设置为1；

biBitCount：2字节，说明比特数/像素数，值有1、2、4、8、16、24、32；

biCompression：4字节，说明图像的压缩类型，最常用的就是0（BI\_RGB），表示不压缩；

biSizeImages：4字节，说明位图数据的大小，当用BI\_RGB格式时，可以设置为0；

biXPelsPerMeter：表示水平分辨率，单位是像素/米，有符号整数；

biYPelsPerMeter：表示垂直分辨率，单位是像素/米，有符号整数；

biClrUsed：说明位图使用的调色板中的颜色索引数，为0说明使用所有；

biClrImportant：说明对图像显示有重要影响的颜色索引数，为0说明都重要；

### 2.1.4 调色板

调色板中的数据每4字节一组，分别表示蓝、绿、红和Alpha值。本实验中只考虑不含调色板的24位位图。

### 2.1.5 位图信息

位图信息的大小取决于图像大小和颜色深度。它存储取决于颜色深度的调色板的索引号或RGB值。

biHeight为正数时图像倒立存储在.bmp文件中，即从左下角开始到右上角，以行为主序排列。但不需要手动翻转行，这个过程会由显示位图的API函数自动进行。

## 2.2 RGB和YUV颜色空间转换

### 2.2.1 RGB转换为YUV

Y = 0.299\*R + 0.587\*G + 0.114\*B

U = -0.147\*R - 0.289\*G + 0.435\*B = 0.492\*(B- Y)

V = 0.615\*R - 0.515\*G - 0.100\*B = 0.877\*(R- Y)

### 2.2.2 YUV转换为RGB

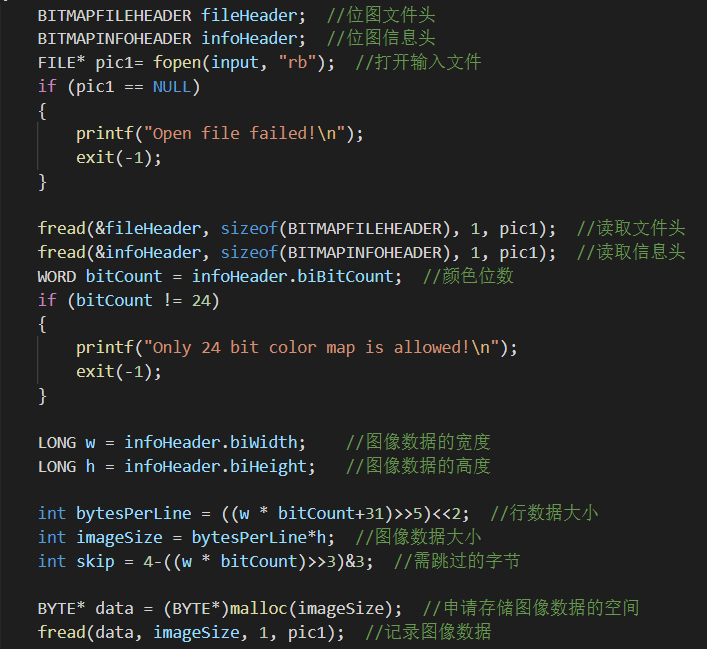
R = Y + 1.140\*V

G = Y - 0.394\*U - 0.581\*V

B = Y + 2.032\*U

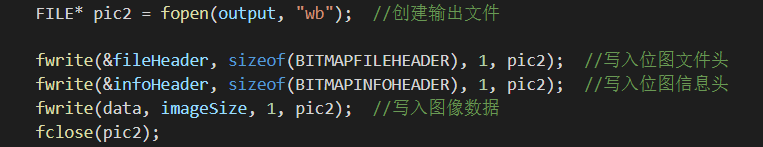
# 3 实验步骤与分析

## 3.1 读取BMP文件



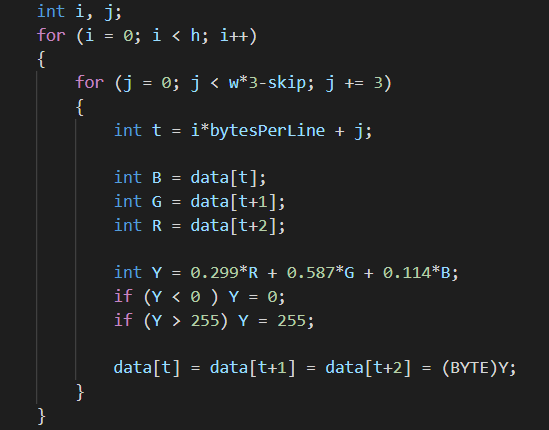
Windows默认的扫描的最小单位是4字节， BMP图像要求每行的数据的长度必须是4的倍数，如果不够需要进行比特填充（以0填充），这样可以达到按行的快速存取。

## 3.2 写入BMP文件



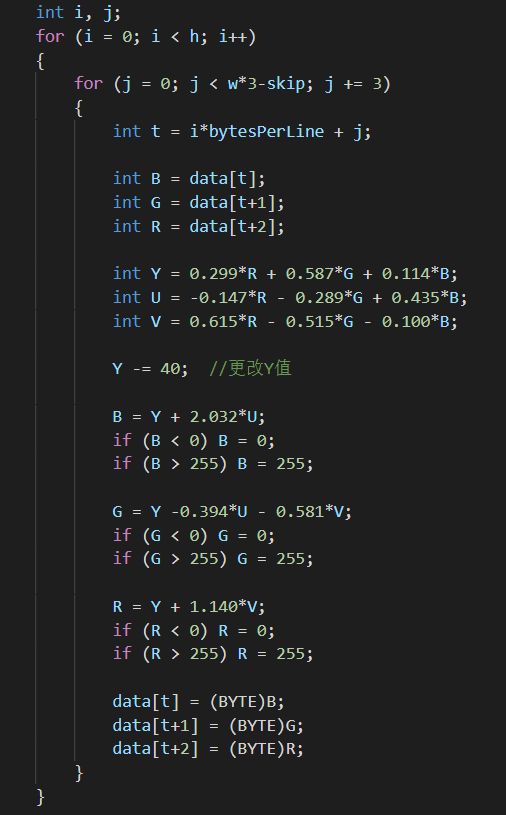
更改位图文件头、位图信息头、图像数据后，创建输出文件将它们写入，两次输出同理进行。

## 3.3 RGB转换为YUV



图像数据中每个像素以B、G、R的顺序用三个字节存储，灰度图即将三个数值均改成Y的值，Y值可由公式和RGB的值求出。Y值求出后将其调整到[0,255]的范围内。

## 3.4 YUV转换为RGB



以B、G、R的顺序从图像数据中读取像素信息，运用公式求出Y、U、V的值，调整Y之后再通过公式代回R、G、B，并将各值调整到[0,255]的范围内。

# 4 实验结果

## 4.1 原始图



## 4.2 灰度图



## 4.3 处理图



# 5 心得体会

在此次实验中，我进一步了解了BMP文件结构，掌握了用C语言对BMP文件进行读写的基本方法，还加强了对RGB和YUV色彩空间及其互相转换的认识。

虽然都是比较基础的知识，但我觉得一个图像处理问题，往往都会分成很多个简单的小问题来解决。通过简化问题，一步步从图像中获取我们需要的信息或者将图像处理成我们需要的效果。更多的知识还有待我们今后更加深入的学习。