**实验报告**

专业：计算机科学与技术

姓名：陈哲恺

学号：3210102035

日期：2022/9/30

课程名称： 图像信息处理 指导老师： 宋明黎 成绩：

实验名称： bmp文件读写及RGB和YUV色彩空间转化

# 一、实验目的和要求

熟悉bmp文件格式，RGB和YUV的图像存储方式。

成功读取一个彩色bmp文件。将其转换为YUV格式，并使其值在[0, 255]之间。输出该灰色bmp图像。修改Y的值，并将YUV格式转换回RGB格式。输出转换后的彩色bmp文件。

# 二、实验内容和原理

对bmp文件进行读写和处理，除了基本的C/C++操作之外，主要涉及bmp文件的保存形式，RGB与YUV的公式转化。

**BMP格式的文件从头到尾依次是如下信息：**

* bmp文件头（bmp file header）：共14字节；
* 位图信息头（bitmap information）：共40字节；
* 调色板（color palette）：可选；
* 位图数据；

1、bmp文件头包含如下信息：

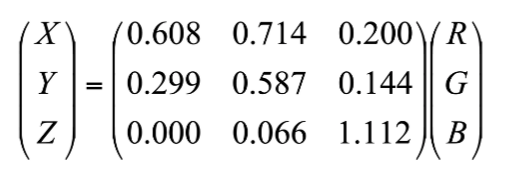
* bfType：2字节，文件类型；
* bfSize：4字节，文件大小；
* bfReserved1：2字节，保留，必须设置为0；
* bfReserved2：2字节，保留，必须设置为0；
* bfOffBits：4字节，从头到位图数据的偏移；

2、位图信息头一共40字节，包含如下内容：

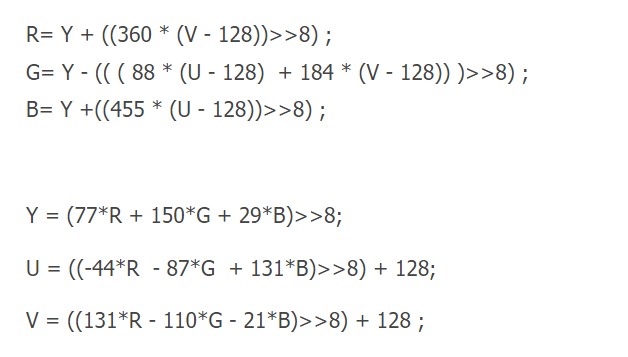
* biSize：4字节，信息头的大小，即40；
* biWidth：4字节，以像素为单位说明图像的宽度；
* biHeight：4字节，以像素为单位说明图像的高度，同时如果为正，说明位图倒立（即数据表示从图像的左下角到右上角），如果为负说明正向；
* biPlanes：2字节，为目标设备说明颜色平面数，总被设置为1；
* biBitCount：2字节，说明比特数/像素数，值有1、2、4、8、16、24、32；
* biCompression：4字节，说明图像的压缩类型，最常用的就是0（BI\_RGB），表示不压缩；
* biSizeImages：4字节，说明位图数据的大小，当用BI\_RGB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；

3、调色板就是一个颜色的索引，这里是8位色图，一共有256中颜色，由于每个颜色都有RGB三原色，也就是要3个字节表示，这样的话256个颜色就不能表示所有的颜色，所以就需要一个索引，用一个字节的索引指向4个字节表示的颜色（RGB加上Alpha值）。

**RGB与YUV转换**

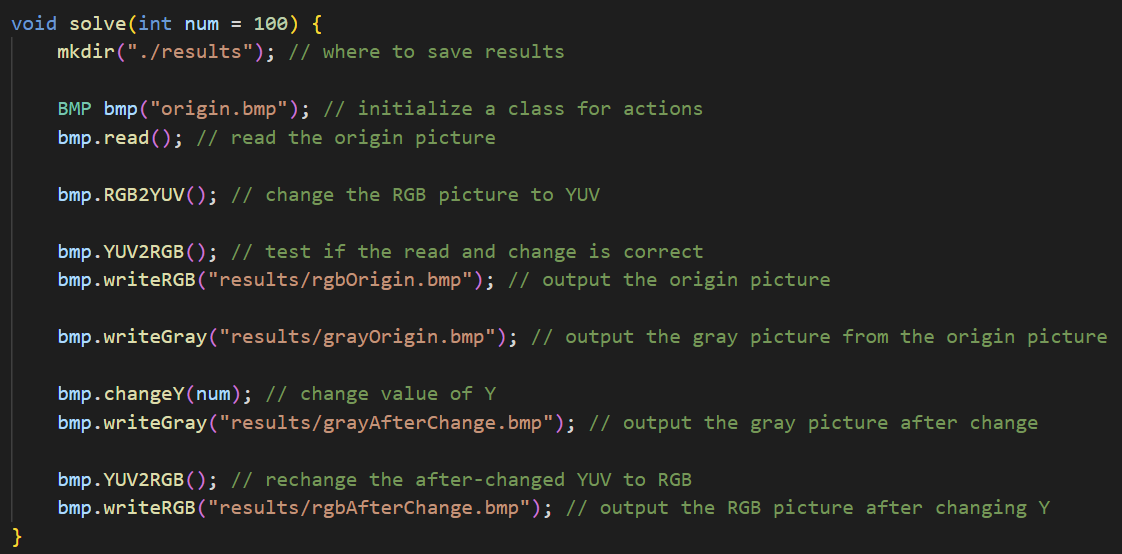


便于计算的格式：



# 三、实验步骤与分析

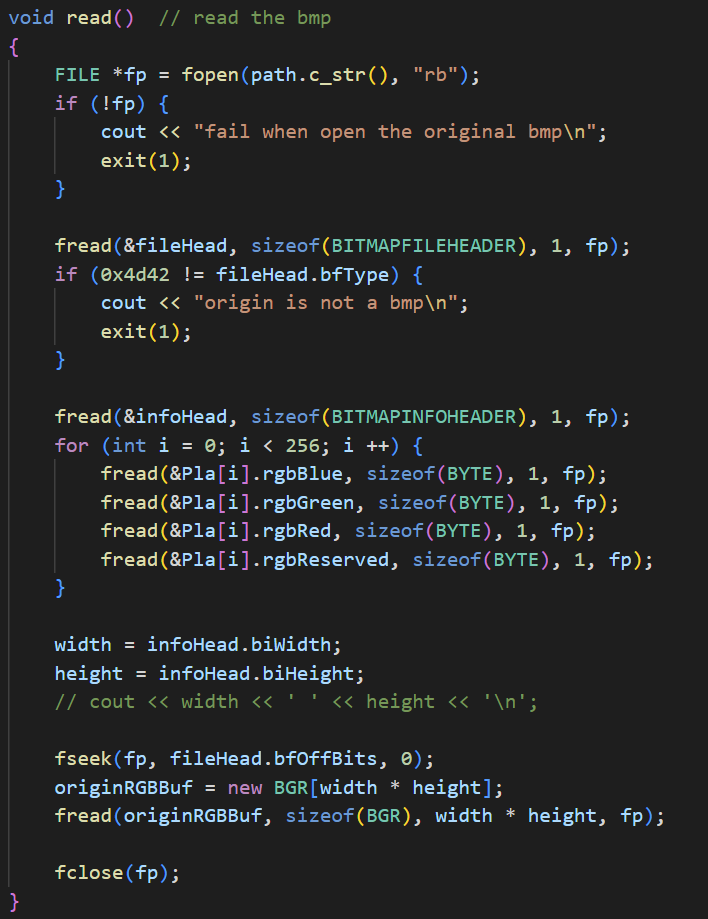
**全部步骤展示**



**详细步骤说明**

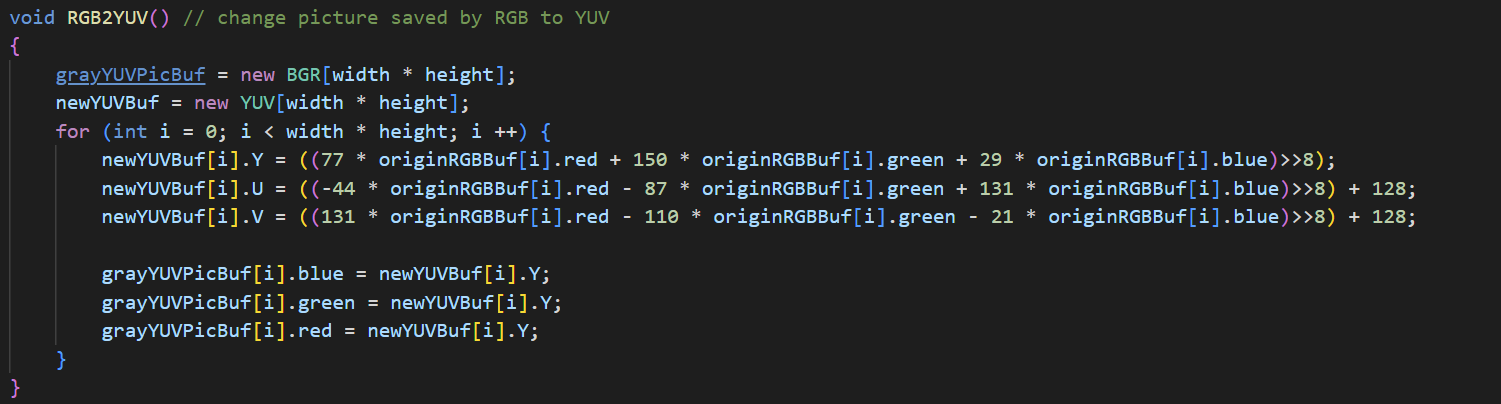
1、读取文件头、信息头和调色板

首先读取文件头判断是否为bmp文件，然后获取信息头，获取infoHead.biWidth和infoHead.biHeight从而判断所需读取大小，并创造对应大小的数组进行信息记录



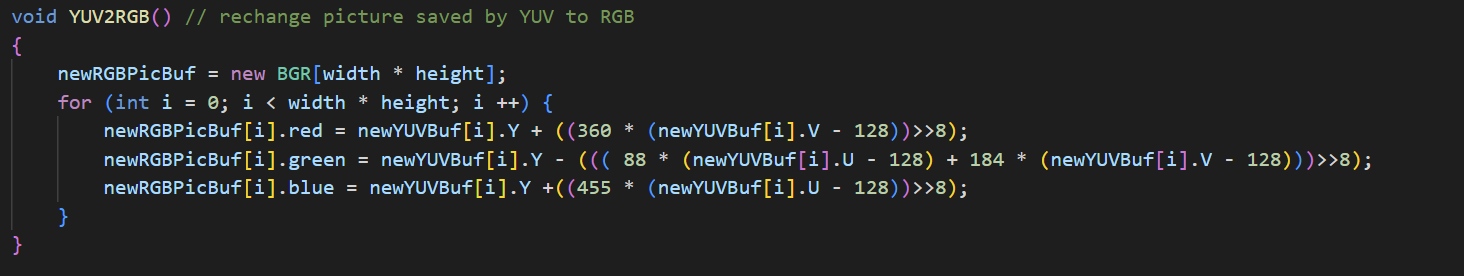
2、修改RGB值为YUV值

通过公式将RGB值转换成YUV值，并将Y值赋给RGB

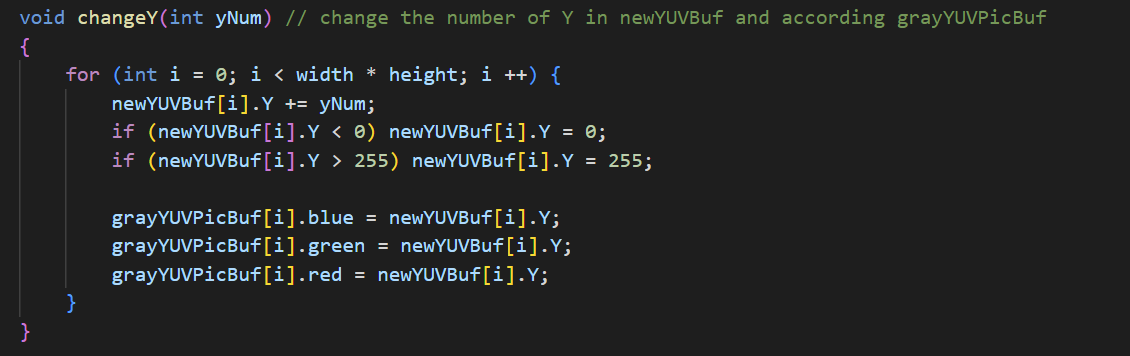


4、将YUV转换为新的RGB值并写入数组

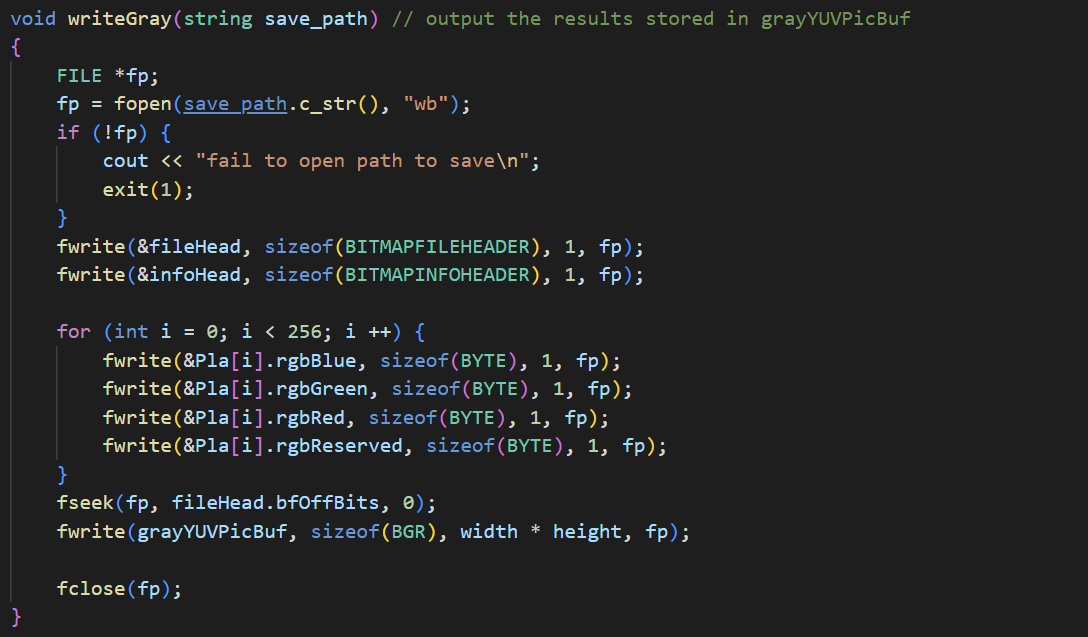
将YUV转回RGB并存入保存新生成图像RGB的数组，从而测试是否正确读取以及转换是否正确

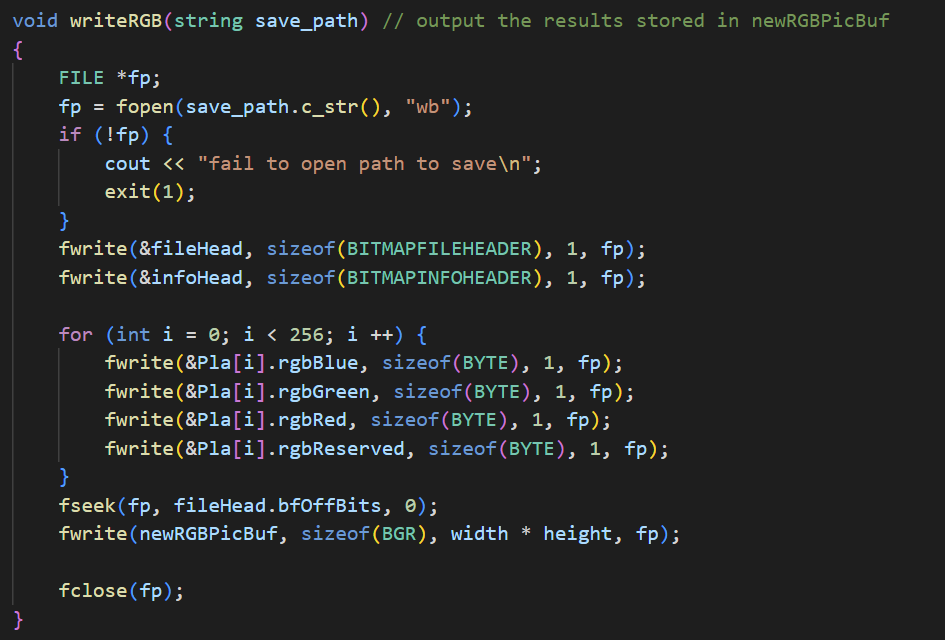


5、修改Y值并输出修改后图像



PS：写入bmp文件方法





# 四、实验环境及运行方法

如果希望使用default.bat脚本（windows）进行自动编译运行以及展示成果，需确保环境下有g++编译器，以及python环境，其中python需确保安装了PIL和matplotlib包。此时，可直接使用在本文件夹下使用./default.bat进行运行。

如果希望自己进行编译等，请确保环境下至少有一个C/C++编译器，在本文件夹下使用

{C编译器} code.cpp -o program

即可编译，然后使用./program进行运行测试，结果会保存在results文件夹中，也可以使用

Python show.py

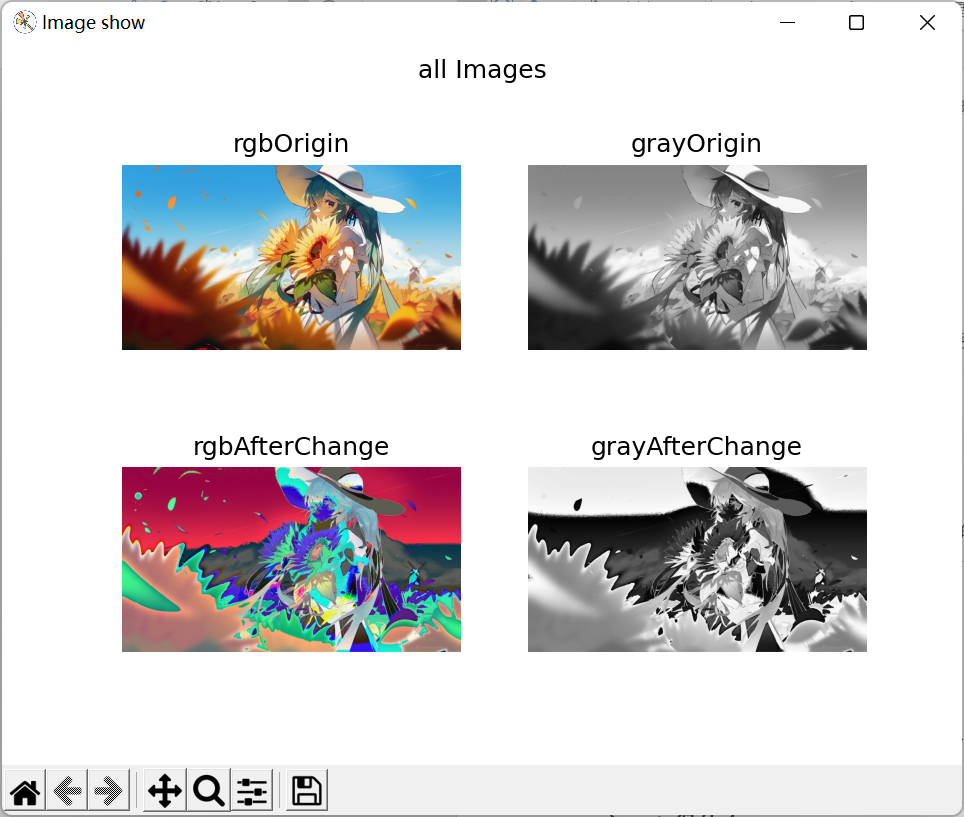
进行展示，但需要确保含有python环境以及python包PIL、matplotlib

如果需要更换源bmp文件，直接更改本文件夹下origin.bmp为所需文件即可，具体使用说明可参考README.md和default.bat脚本

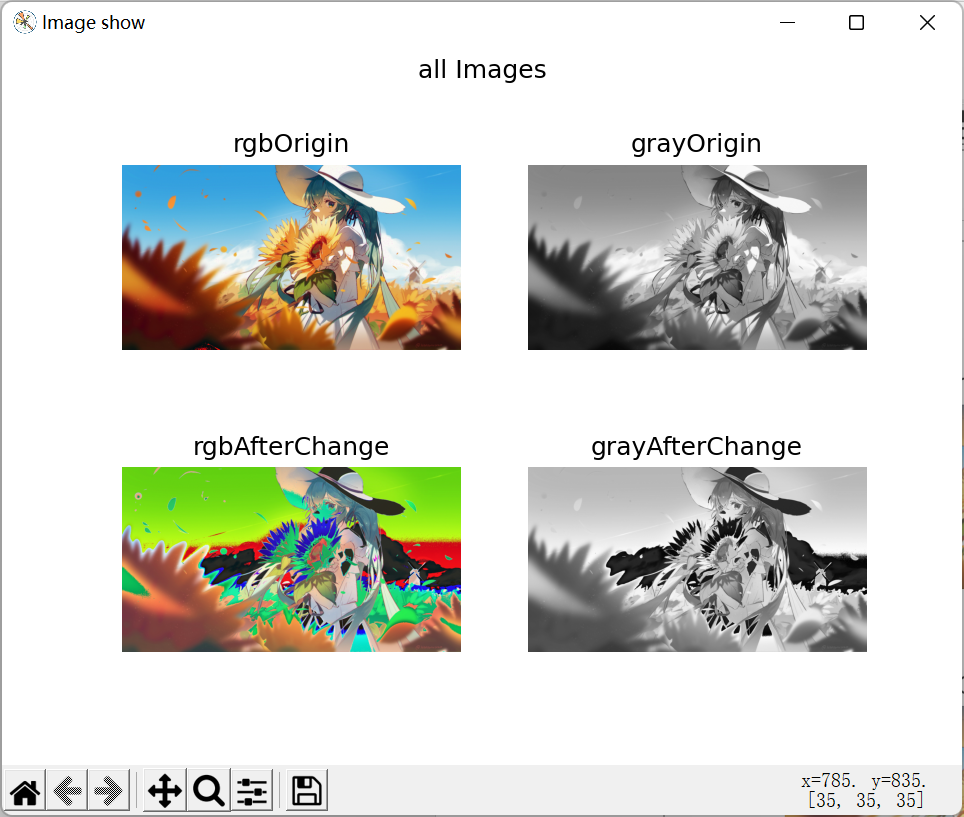
PS：编写的程序需要传入一个参数arg1，此参数代表对Y值进行的操作，具体为Y+={arg1}，此参数可为正数也可为负数，分别代表对整体Y值的加减。程序会限制Y的范围为[0, 255]，当修改后超过范围，将修改Y至上（下）限。默认参数为100.

# 五、实验结果展示

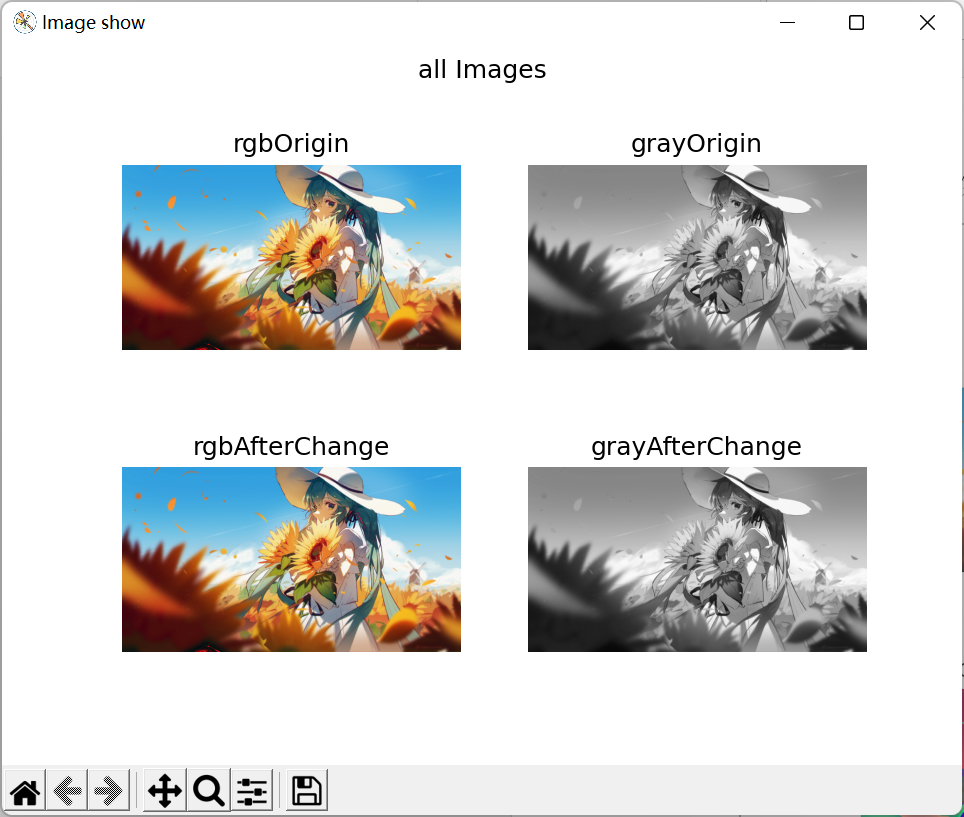
1、./default （不传参数，会使用默认值100，将Y值均加100）



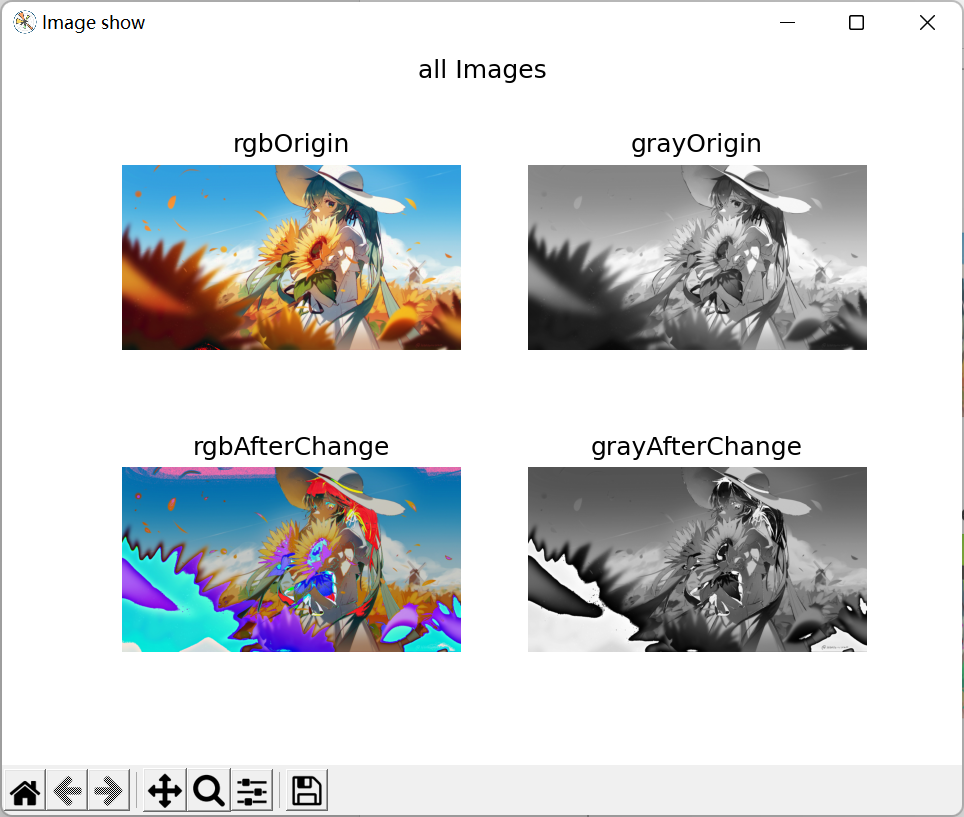
2、./default 50 （将Y值均加50）



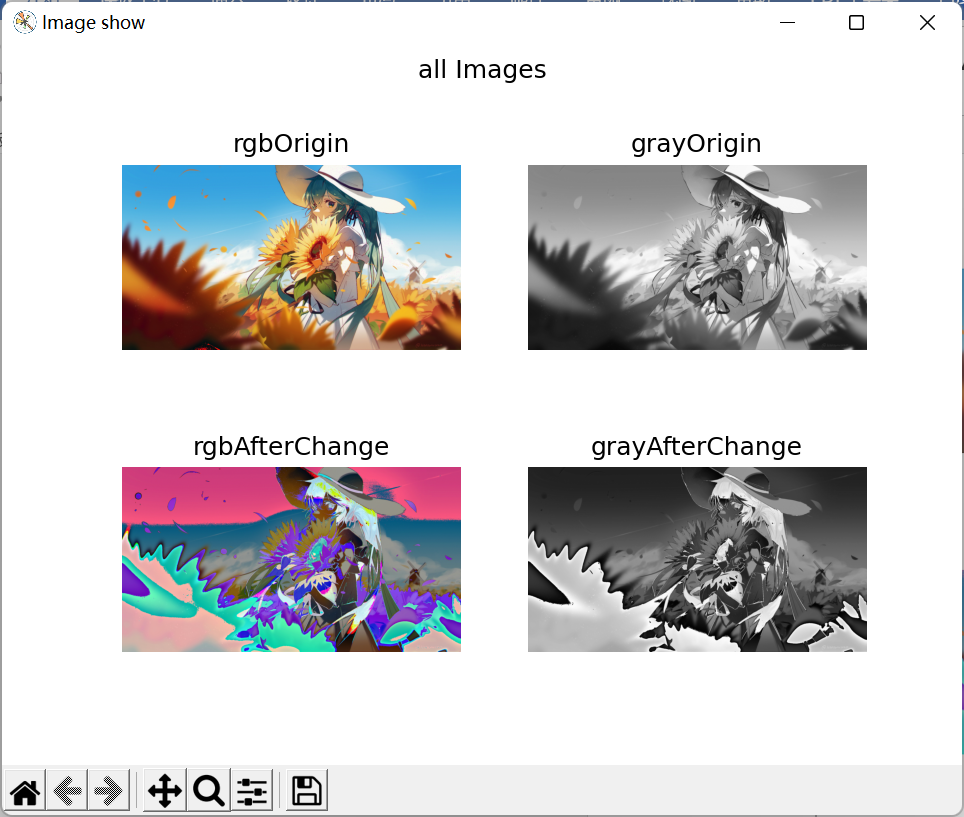
3、./default 0 （不进行修改）



4、./default -50 （将Y值均减50）



5、./default -100 （将Y值均减100）



# 六、心得体会

通过这次实验，了解了bmp图片的储存格式，以及RGB的获取和转换。也在自己探索的过程中，对C++的类、bat脚本、python的一些功能得到了了解，感觉学到了很多。处理图像确实是很有趣的一件事，修改一点灰度值，就能得到各种各样风格迥异的图片。希望能进一步学习相关知识，如果有机会的话，也希望学习到CV的相关内容。