**实验报告**

专业：计算机科学与技术

姓名：张三

学号：111111

日期：2022/10/31

课程名称： 图像信息处理 指导老师： 成绩：

实验名称： bmp文件对数操作和直方图均衡化处理

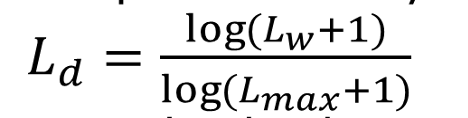
**一、实验目的和要求**

目的：通过对bmp文件的对数化和直方图均衡化操作来增强图片的可视效果。

实验要求：用C语言实现，不的调用相关的库函数进行实现。

**二、实验内容和原理**

对数操作：



上述公式中，Ld为显示出来的亮度，Lw为未进行对数操作前的亮度，而Lmax为全局最大亮度，这样的操作保证了场景中的最大亮度能够被映射为1，而其他的亮度能够进行平滑的变化，不至于因为过暗而难以分辨。

直方图均衡化：

假设一张图片的灰度分为0-255共256级，首先我们统计出每个灰度k对应的像素点的个数，并将其除以总像素数得到该灰度的像素所占的比例rk，那么每个灰度所对应的sk就为从0到当前的k值所有的灰度对应的像素比例的叠加，最终s255=1，我们就将原来的灰度映射到255\*sk所对应的灰度上，这样的话一些原本不同的灰度就会被合并，使得灰度与灰度之间的差异变大，也就是加强了对比度，增强了可视化效果。

**三、实验步骤与分析**

int find\_closest(int n){

    return (n +0.5)/1;//round off

}

unsigned char \*equalization(int \*rgbcount,int total,unsigned char \* rgbdata){

    double s\_rgbcount[256][3]={0};//store the sk

    double tot = (double) total;

    int adjusted\_rgbcount[256][3] = {0};

    for(int i=0;i<256;i++){

        if(i==0){//initial

            s\_rgbcount[0][0]= \*rgbcount/tot;

            s\_rgbcount[0][1]= \*(rgbcount+1)/tot;

            s\_rgbcount[0][2]= \*(rgbcount+2)/tot;

        }else{//latter

            s\_rgbcount[i][0] = s\_rgbcount[i-1][0] + \*(rgbcount+3\*i)/tot\*1.0;

            s\_rgbcount[i][1] = s\_rgbcount[i-1][1] + \*(rgbcount+3\*i+1)/tot\*1.0;

            s\_rgbcount[i][2] = s\_rgbcount[i-1][2] + \*(rgbcount+3\*i+2)/tot\*1.0;

        }

    }

    for(int i=0;i<256;i++){// find the adjusted intensity of rgb

        for(int j=0;j<3;j++){

            adjusted\_rgbcount[i][j] = find\_closest(256\*s\_rgbcount[i][j]);

        }

    }

    unsigned char\* final\_rgbdata = malloc(sizeof(unsigned char)\*3\*total);

    for(int i=0;i<total;i++){//adjust the rgbdata

        final\_rgbdata[3\*i] = adjusted\_rgbcount[rgbdata[3\*i]][0];

        final\_rgbdata[3\*i+1] = adjusted\_rgbcount[rgbdata[3\*i+1]][0];

        final\_rgbdata[3\*i+2] = adjusted\_rgbcount[rgbdata[3\*i+2]][0];

    }

    return final\_rgbdata;

}

int log\_the\_data(unsigned char \* logrgb,unsigned char\* rgbdata,int n,unsigned char\* max){

    for(int i=0; i<n;i++){

        logrgb[3\*i] = 256\*log(rgbdata[3\*i]+1)/log(max[0]+1);

        logrgb[3\*i+1] = 256\*log(rgbdata[3\*i+1]+1)/log(max[1]+1);

        logrgb[3\*i+2] = 256\*log(rgbdata[3\*i+2]+1)/log(max[2]+1);

    }

}

**四、实验环境及运行方法**

（说明程序的编译环境和具体测试方法）

直接用gcc编译运行，注意D：\test.bmp必须存在

**五、实验结果展示**

原图片：



对数操作：



直方图均衡化：



**六、心得体会**

本次实验算法上的实现较为简单，基本上只需要对图像数据进行一个步骤一个步骤的处理就好了，最终可以看到可视化效果得到了增强，但是画面整体偏白。