

计算机视觉 课程实验报告

学号：201900130151	姓名：莫甫龙	
-----------------	--------	--

实验题目：图像代数运算

实验过程中遇到和解决的问题：

（记录实验过程中遇到的问题，以及解决过程和实验结果。可以适当配以关键代码辅助说明，但不要大段贴代码。）

对于第一个实验，亮度这块十分好解决，只需要对整张图片全部加减就好（同时考虑溢出问题），但是对于对比度，我设置的 S 函数是在逻辑回归函数的基础上进行修改的，但是我没有考虑到这个函数的值域为 $(0, 1)$ ，所以用了 int 型，导致这个值无论如何都为 0，使得显示出来的图显示错误。后面我就用 double 型来存储该值，就能完美解决这一问题。

在写代码的时候，我开始是直接将修改后的值赋给了 image2，但是却显示一个窗口而已，这就让我一头雾水，后面上网搜索发现需要给 image2 申请和 image 一样的尺寸，让 image2 成为一张和 image 一样属性的空白图。

```
image2 = Mat::zeros(image.size(), image.type());
```

```
void modify(int, void *) {
    for (int x = 0; x < image.rows; x++) {
        for (int y = 0; y < image.cols; y++) {
            for (int c = 0; c < 3; c++) {
                double t=1/(1+exp(-X: (-1.0*image.at<Vec3b>(x, y)[c]+128)/k)); //S型函数，就是对比度
                image2.at<Vec3b>(x, y)[c] = saturate_cast<uchar>((v: t*(
                    image.at<Vec3b>(x, y)[c])+b)); //线性函数，其中斜率为对比度，截距为整张图加的b
            }
        }
    }
    imshow( winname: "1", mat: image2);
}

int main() {
```

此为主要的代码，首先三层循环来找到每个像素点的每一个通道，然后对其进行修改。

对于第二个实验，因为有了第一个实验的基础，所以只需要按照思路直接写就行了。

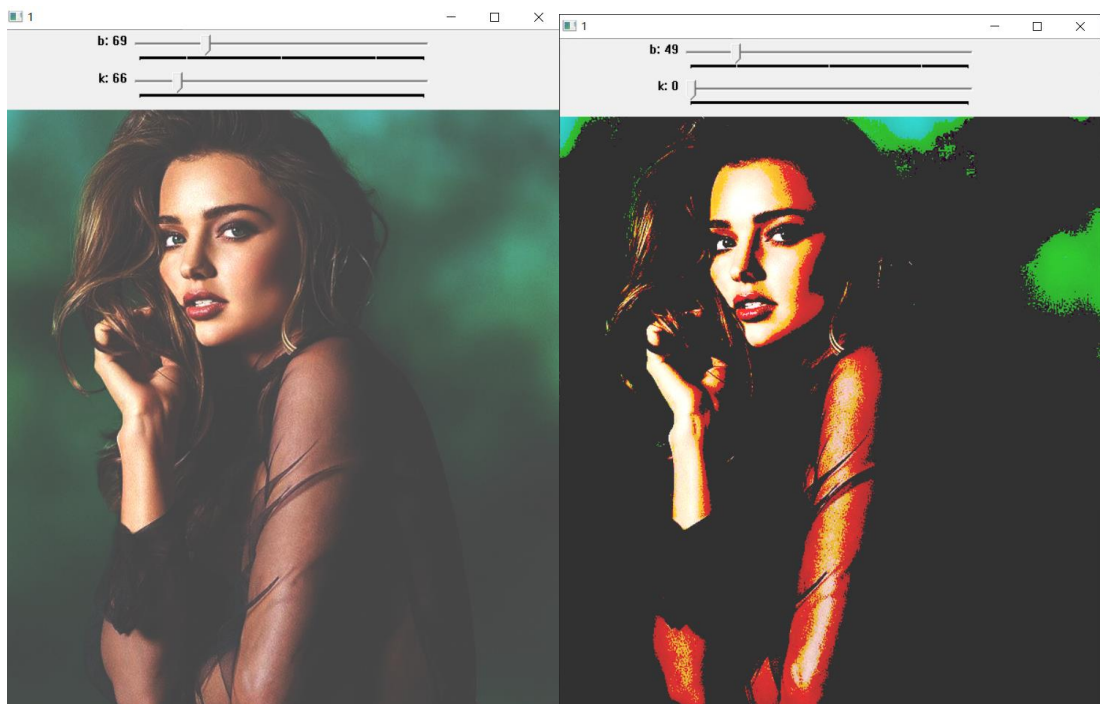
```

void modify(int, void *) {
    for (int x = 0; x < image.rows; x++) {
        for (int y = 0; y < image.cols; y++) {
            double diff = 0;
            for (int c = 0; c < 3; c++) {
                diff += pow((image.at<Vec3b>(x, y)[c] - image2.at<Vec3b>(x, y)[c]), 2);
            }
            diff = sqrt(diff);
            if (diff > k)
                for (int c = 0; c < 3; c++)
                    result.at<Vec3b>(x, y)[c] = 255;
            else
                for (int c = 0; c < 3; c++)
                    result.at<Vec3b>(x, y)[c] = 0;
        }
    }
    imshow( winname: "1", mat: result);
}

```

其中 diff 为图片 and 背景每个像素平方的和，然后对其进行开方，以此来判断该值是否大于阈值（阈值可以通过滑动条来进行改动），如果大于，那就是需要裁出来的图，赋值为 255，为白色，否则就是背景图，为 0，为黑色。

两个代码的结果如下：





结果分析与体会：

通过该实验，将课堂上十分抽象的公式和知识化为了可能，能更好地理解到课堂上的知识。