

# 云南大学

## 本科实验报告

课程名称： 图像理解与计算机视觉

实验名称： 实验一. 图像变换实验

学院（系）：

专 业：

班 级：

姓 名：

学 号：

指导教师：

成 绩：

评 语：

*Steven*

## 一. 实验目的

通过编程实现使学生能够读入图像数据、输出显示图像、生成图像直方图，掌握图像灰度变换的基本原理并能够完成图像的对比度变换操作，掌握直方图均衡化、直方图规定化的具体处理方法，能够完成图像均衡化处理算法。

## 二. 实验内容

编程实现图像变换操作，具体需要实现：

- (1) 读入图像数据并输出显示图像；
- (2) 生成图像直方图；
- (3) 完成图像的对比度变换操作；
- (4) 完成图像直方图均衡化处理算法。

## 三. 实验环境

Matlab软件是图像处理领域广泛使用的仿真软件之一。本实验基于Matlab 2022版本完成。

## 四. 实验代码（详细注释，Times New Roman/宋体 五号字体 单倍行距）

```
function Exp1(func, show)
    img_lena = imread("lena.jpg");
    img_huafen = imread("huafen.tif");

    % 不同的 4 个功能
    if func == "read"
        image(img_lena); % 展示灰度图
        title("原图");
        axis square;

    elseif func == "histogram"
        bin = input("请输入灰度级数: ");
        imhist(img_lena, bin); % 划分为 128 个灰度级
        title("灰度直方图");
        axis square;

    elseif func == "contrast"
```

批注 [短腿1]: 实际代码与文档中的代码功能效果相同，但写法有一定差异，建议以 Exp1.m 的内容为准

```
new_img = contrast(img_huafen);

if show == "show"
    subplot(1, 2, 1);
    imshow(img_huafen);
    title("原图");
    subplot(1, 2, 2);
    imshow(new_img);
    title("分段线性变换后的图");
    axis square;
end

elseif func == "equalization"
    bin = input("请输入灰度级数: ");
    new_img = equalizeImg(img_huafen, bin);

    if show == "show"
        subplot(3, 2, 1)
        imshow(img_huafen);
        title('原图');
        axis square;

        subplot(3, 2, 2)
        imhist(img_huafen);
        title('原图-灰度直方图');
        axis square;

        subplot(3, 2, 3)
        imshow(histeq(img_huafen, bin));
        title('histeq 灰度均衡化');
        axis square;

        subplot(3, 2, 4)
        imhist(histeq(img_huafen));
        title('histeq 均衡化-灰度直方图');
        axis square;

        subplot(3, 2, 5)
        imshow(new_img);
        title('自制灰度均衡化');
        axis square;

        subplot(3, 2, 6)
        imhist(new_img);
```

```

        title('自制灰度均衡化-灰度直方图')
        axis square;
    end
end
end

function new_grayscale = adjustGray(grayScale)
% 将灰度值在 100~120 之间的像素映射到 30~150 之间
if grayScale < 100
    new_grayscale = 0.3 * grayScale;
elseif grayScale > 120
    new_grayscale = round(106/118 * grayScale + 42.2);
else
    new_grayscale = round(6 * grayScale - 570);
end
end

function new_img = contrast(img)
new_img = img;

for row = 1:length(img)
    for col = 1:length(img)
        new_img(row, col) = adjustGray(img(row, col));
    end
end
end

function new_img = equalizeImg(img, bin)
% img:灰度图矩阵
% bin:灰度级数
new_img = img;
[n] = imhist(img, bin); % 灰度级的像素个数
gray2grayRank = [0:1:255; 0:1:255]'; % 完成原图的"灰度-灰度级"的对应

for gray = 0:255
    for grayRank = 0:bin - 1
        if gray >= 255 * (grayRank - 1.5) / (bin - 1) && gray < 255 * (grayRank - 0.5) / (bin
- 1)
            gray2grayRank(gray + 1, 2) = grayRank;
            break;
        else
            continue;
        end
    end
end
end

```

```

end

counts = n / sum(n); % 原始直方图
cum_counts = cumsum(counts); % 累积直方图
grayRank2grayRank = cum_counts;

for i = 1:length(counts)
    grayRank2grayRank(i) = round((bin - 1) * cum_counts(i));
end

grayRank2grayRank = [[0:1:bin - 1]', grayRank2grayRank]; % 灰度级变换，第 1 列转换为第 2 列
grayRank2gray = grayRank2grayRank;

for grayRank = 0:bin - 1
    grayRank2gray(grayRank + 1, 2) = round(((grayRank) * 256 / bin + (grayRank + 1) * 256 / bin) / 2);
end

for row = 1:length(img)

    for col = 1:length(img)
        new_img(row, col) =
grayRank2gray(grayRank2grayRank(gray2grayRank(img(row, col), 2), 2), 2);
    end
end
end

```

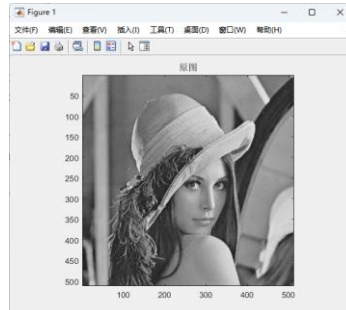


Exp1.m

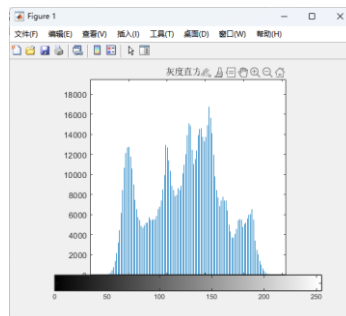
附件（.m文件）：

## 五. 实验结果

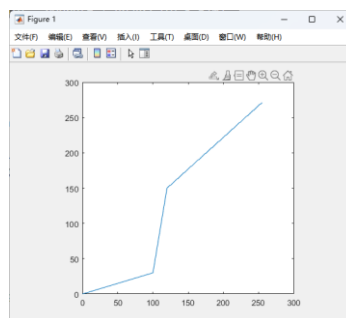
（1）原图像



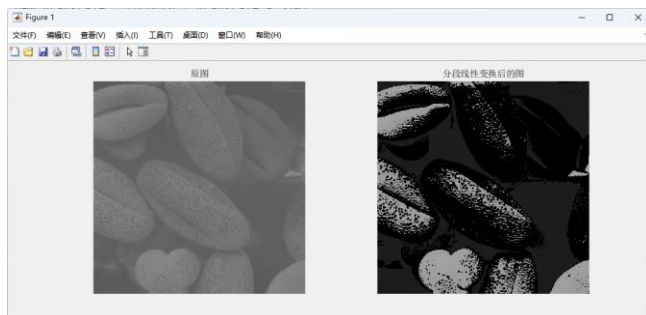
(2) 直方图



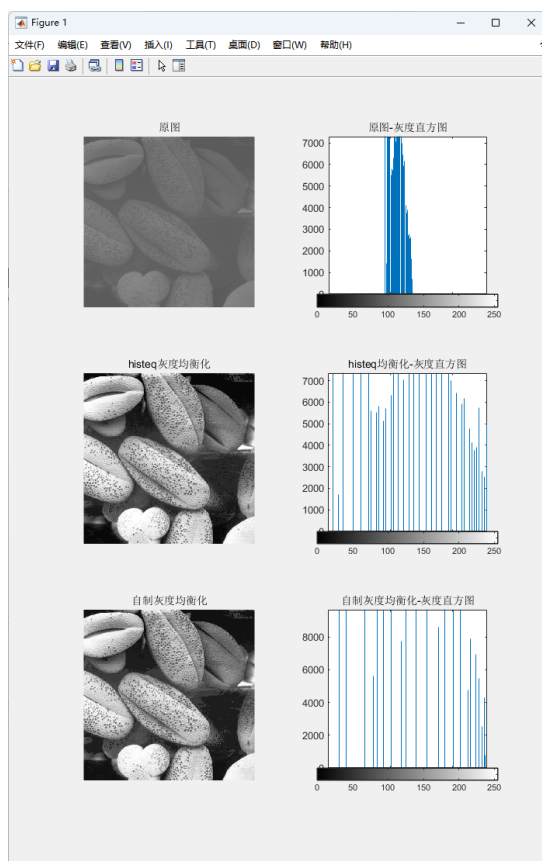
(3) 对比度变换函数



(4) 对比度变换结果



(5) 直方图均衡化结果



## 六. 结果分析及体会

在本次实验中主要练习了Matlab中图像的读取、输出，以及一些简单的灰度图像处理方法，大体上每题都得到了较为满意的结果。

虽然有的处理过程在Matlab中提供了自带库函数调用，如histeq，但按照数学步骤从头实现一遍依然是一件很有成就感的事，虽然花费的时间很长，但也确实理解了这一函数的实现逻辑。

此外，在equalizeImg函数中，grayRank2gray的取值参考了[官方文档](#)中的公式，因此才完成了该题，这同样也是对搜集信息能力的一种考验吧....