|  |  |  |
| --- | --- | --- |
| **得分** | **教师签名** | **批改日期** |
|  |  |  |

课程编号： IB00078

****

**深圳技术大学实验报告**

**课程名称： 数字图像处理**

**实验名称： 实验一 图像增强实验**

**班 级： 计算机科学与技术2班**

**指导教师： 李蒙**

**报 告 人： 黄荣权 学号： 202002020213**

**合 作 者： 组号：**

**实验地点：**

**实验时间： 2022 年 4 月 13 日 星期 二**

**提交时间： 2022/4/13**

|  |
| --- |
| 1. **实验目的：** 2. **掌握读、写图像的基本方法。** 3. **掌握MATLAB语言或OpenCV中图像数据与信息的读取方法。** 4. **理解图像灰度变换处理在图像增强的作用。** 5. **掌握绘制灰度直方图的方法，理解灰度直方图的灰度变换及均衡化的方法。**   **二、实验原理：**  **灰度变换是图像增强的一种重要手段，它常用于改变图像的灰度范围及分布，是图像数字化及图像显示的重要工具。**   1. **图像反转**   **灰度级范围为[0, L-1]的图像反转可由下式获得**     1. **对数运算：**   **有时原图的动态范围太大，超出某些显示设备的允许动态范围，如直接使用原图，则一部分细节可能丢失。解决的方法是对原图进行灰度压缩，如对数变换：**  ***s* = *c*log(1 + *r*)，*c*为常数，*r*≥ 0**   1. **幂次变换：**      1. **对比拉伸：**   **在实际应用中,为了突出图像中感兴趣的研究对象，常常要求局部扩展与拉伸某一范围的灰度值，或对不同范围的灰度值进行不同的拉伸处理，即分段线性拉伸：**    **其对应的数学表达式为：**     1. **直方图均衡化**   **灰度直方图的横坐标是灰度级，纵坐标是该灰度级频率，它是图像最基本的统计特征。依据定义，在离散形式下，用*rk*代表离散灰度级，用*pr*(*rk*)代表*pr*(*r*)，且有下式成立：**    **式中，*nk*为图像中出现*rk*级灰度的像素数，*n*是图像像素总数，而*nk/n*即为频数。**  **直方图均衡化处理是以累积分布函数变换法为基础的直方图修正法。假定变换函数为**    **当灰度级是离散值时，可用频数近似代替概率值，即**    **式中，l是灰度级的总数目，*pr*(*rk*)是取第*k*级灰度值的概率，*nk*是图像中出现第*k*级灰度的次数，*n*是图像中像素总数。**  **所以积分可以表示为下列累计分布函数**    **三、实验内容步骤**（记录实验主要步骤，且在调试成功后，将结果截屏或拍照保存）  熟悉MATLAB语言或OpenCV中对图像数据读取，显示等基本函数。以MATLAB为例，需要熟悉下列命令：熟悉imread()函数、imwrite()函数、size()函数、subplot()函数、figure()函数。   1. 将文件夹中的任一图像文件读出，观察图像数据，将这个图像显示出来。读入不同的图像，请自己编程和调用Matlab或OpenCV函数用常用灰度变换函数对输入图像进行灰度变换，比较相应的处理效果，感受图像灰度变换处理在图像增强的作用。   import cv2  # 导入opencv库  lena\_color = cv2.imread("./images/lena.jpg")  # 读取图像lena.jpg  lena\_gray = cv2.cvtColor(lena\_color, cv2.COLOR\_BGR2GRAY)  # 将原图进行灰度化处理  # 通过窗口展示图片  cv2.imshow('lena\_color', lena\_color)  cv2.imshow('lena\_gray', lena\_gray)  cv2.waitKey(0)  # 设置waitKey(0),代表按任意键继续     1. 绘制图像灰度直方图的方法，对图像进行均衡化处理   将图像文件读出，用函数将彩色图像转化为灰度图像，记为变量B。请自己编程和调用函数完成如下实验。   1. 显示B的图像及灰度直方图，可以发现其灰度值集中在一段区域，用imadjust函数将它的灰度值调整到[0, 1]之间，并观察调整后的图像与原图像的差别，调整后的灰度直方图与原灰度直方图的区别。   close all  lena=imread("./images/lena.jpg");  lena\_gray1=rgb2gray(lena);  subplot(2,2,1); %将多个图画到一个平面上的工具  imshow(lena\_gray1);  title("灰色图像");  subplot(2,2,2); %将多个图画到一个平面上的工具  imhist(lena\_gray1);  title("调整前的灰度直方图");  lena\_gray2=imadjust(lena\_gray1,[0,0.4],[0,1],1); %对图像进行灰度调整  subplot(2,2,3);  imshow(lena\_gray2);  title("imadjust灰度变换调整后图像");  subplot(2,2,4);  imhist(lena\_gray2);  title("调整后的灰度直方图");     1. 对B进行直方图均衡化处理，试比较与原图的异同。   close all  %直方图均衡化  lena=imread("./images/lena.jpg");  lena1=rgb2gray(lena);  subplot(1,2,1);  imshow(lena1);  title("原图");  ylim("auto"); %自动设定纵坐标轴的取值范围和刻度线  g=histeq(lena1,256);  subplot(1,2,2);  imshow(g);  title("直方图均衡化图");  直方图均衡化图像与原图相比，像素值分布更加均匀，细节更加丰富。     1. 对B进行如图所示的分段线形变换处理，试比较与直方图均衡化处理的异同。     图1.1 分段线性变换函数  close all  lena=imread('./images/lena.jpg'); %读入原始图像  lena=rgb2gray(lena);  subplot(2,2,1);imshow(lena,[]);title('原图像');  %分段线性变换  [M,N]=size(lena);  lena=im2double(lena);  out=zeros(M,N);  X1=0.125;  Y1=0.250; %转折点1  X2=0.750;  Y2=0.625; %转折点2  for i=1:M  for j=1:N %遍历图像每个像素  if lena(i,j)<X1  out(i,j)=Y1\*lena(i,j)/X1;  elseif lena(i,j)>X2  out(i,j)=(lena(i,j)-X2)\*(1-Y2)/(1-X2)+Y2;  else  out(i,j)=(lena(i,j)-X1)\*(Y2-Y1)/(X2-X1)+Y1;  end  end  end  subplot(2,2,2);  imshow(out,[]);  title('变换的后图像')  %绘制直方图  subplot(2,2,3);  imhist(lena);  title('原始图像');  subplot(2,2,4);  imhist(out);  title('变换后的图像')  相同点：处理后使得图像更加清晰  不同点：实现图像增强方式不同，直方图均衡化处理使像素按值的概率均匀分布，而分段线形变换处理则是通过简单地对小像素值进行拉伸处理来实现增强图像对比度的目的。 |
| **四、实验总结与心得体会**  ①进一步加深了对数字图像处理的认识；  ②熟练使用熟悉imread()函数、imwrite()函数、size()函数、subplot()函数、figure()函数； |
|  |