数字图像处理基础实验报告

自动化 65 姚润昭 2160504132

摘要

本报告对 bmp 的图像格式进行了介绍，实现了改变灰度图的灰度等级、求灰度图方差均值、用三种插值方法对图像缩放、对图像进行拉伸和旋转等等操作。

1.bmp 图像格式

BMP（全称 Bitmap）是 Windows 操作系统中的标准图像文件格式，可以分成两类：设备有向量相关位图（DDB）和设备无向量相关位图（DIB），使用非常广。它采用位映射存储格式，除了图像深度可选以外，不采用其他任何压缩，因此，BMP 文件所占用的空间很大。BMP 文件的图像深度可选 lbit、4bit、8bit 及 24bit。BMP 文件存储数据时，图像的扫描方式是按从左到右、从下到上的顺序。由于 BMP 文件格式是 Windows 环境中交换与图有关的数据的一种标准，因此在 Windows 环境中运行的图形图像软件都支持 BMP 图像格式。

Bmp 文件的数据从开头到结尾分为四个部分：bmp 文件头，文件信息头，调色板和位图数据，其大小如下：

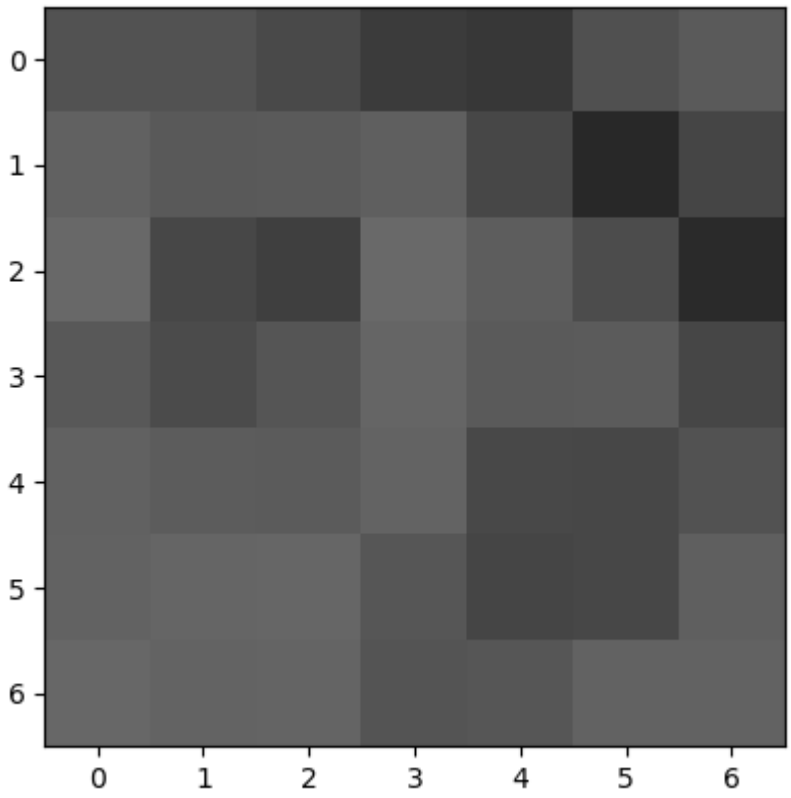
数据段	大小	内容
bmp 文件头	14byte	提供文件的格式、大小等信息
文件信息头	40byte	提供图像数据的尺寸、位平面数、压缩方式、颜色索引等信息
调色板	由颜色索引数决定	可选，如使用索引来表示图像，调色板就是索引与其对应的颜色的映射表
位图数据	由图像尺寸决定	图像数据

读取 7.bmp 文件头信息：

```
C:\Users\DELL\PycharmProjects\DIP\venv\Scripts\python.exe C:/Users/DELL/PycharmProjects/DIP/Img_trans.py
b'BMn\x04\x00\x00\x00\x00\x006\x04\x00\x00(\x00\x00\x00\x07\x00\x00\x00\x07\x00\x00\x00\x01\x00\x08\x00'
7 7 8
```

可知：图像为 7*7 像素为 8

打印图片：



7.bmp

2、把 lena 512*512 图像灰度级逐级递减 8-1 显示

1）问题分析：lena 图像是 8 位 0-255 的灰度图像，可以用如下公式调整灰度等级为 k：$n\_p(x,y) = floor\left(\frac{p(x,y)}{2^{8-k}}\right)$ （1），其中 p（x,y）为调整前像素点的值，n_p(x,y)为调整后像素点的值，floor（）函数为向下取整。

2) 实验过程: 利用 matlab 编程编写函数 A=change_scale(filename,scale), 其中，filename 为图片路径，scale 为调整后的灰度级别，A 为调整后的矩阵，具体实现

为：用 imread（）函数读取图片，用公式（1）调整像素点的值，再用 imshow()展示图片。

3）实验结果

8 bits

7bits

6bits

5bits

4bits

3bits

2bits

1bit

4）结果分析：图像用 5-7bit 表示时，失真程度较小，从 4bit-1bit 过程中，图像失真程度明显增大。可以发现当图片灰度的级别降低时，细节丰富的部分（比如头发），失真较小，而细节少的部分（如帽子），失真较大。

3. 算 lena 图像的均值方差.

1)问题分析：读取图像后，求每一个像素点累加求均值，再求方差。

2）实验过程，用 matlab 编写函数[mean_c,var_c]=calculate_mean_var(filename)

其中 filename 为图片路径，mean_c 返回均值，var_c 返回方差。具体实现为：

用 imread（）读取图片信息，用 mean（）函数求均值，用 var（）函数求方差。

3）实验结果：

```
>> [mean_c,var_c]=calculate_mean_var('C:\Users\DELL\Desktop\hw\lena.bmp')

mean_c =

    99.0512

var_c =

   2.7958e+03
```

4、把 lena 图像用近邻、双线性和双三次插值法 zoom 到 2048*2048

1）题目分析：lena 图像大小为 512*512，分别用临近、双线性和三次插值法将图片长宽扩大四倍。

2）实验过程：用 matlab 编写函数：

```
[img_nearest,img_bilinear,img_bicubic]=resize_img(filename,times)
```

其中，filename 为文件名，times 为放大的倍数，返回值分别为临近插值、双线性插值和双三次插值得到的像素矩阵。具体实现为：先用 imread（）读取图片，再用函数 imresize（）进行插值，最后打印图片

3）实验结果：

original img

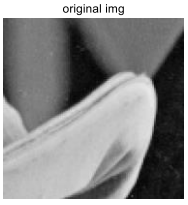
nearest img

bilinear img

bicubic img

细　　节　　展　　示　　：

original img



nearest img

bilinear img

bicubic img

分析：原图像经过临近插值后，直线部分会出现锯齿状，线条不够平滑，而经过双线性插值后，线条较为平滑，没有出现锯齿状的情况，双三次插值的想过比其他插值方法更好一些，这是因为，临近插值是选取邻域内最近一个像素点，线条自然会出现锯齿状，而双线性插值，和双三次插值，分别利用 4 个临近像素点和 16 个临近像素点，效果自然会更好。

5. 把 lena 和 elain 图像分别进行水平 shear（参数可设置为 1.5，或者自行选择）和旋转 30 度，并采用用近邻、双线性和双三次插值法 zoom 到 2048*2048

1）问题分析：将图像沿水平方向拉伸 1.5 倍，再旋转 30°（双三次插值）最后用三种插值方法将图片放大到 2048*2048

2）试验过程：

用matlab编写函数：`m_imrotate(filename,h,w,angle,shape)`

其中 filename 为图片名，h，w 为调整长度后图片的长宽，angle 为旋转的角度，shape 为最终插值后图片的大小。具体实现为，先用 imread（）读取函数，再用 imresize（）调整图片大小，再用 imrotate（）旋转图片，最后调用实验三中的函数 resize_img 进行插值

3）实验结果

original img

nearest img

bilinear img

bicubic img

original img

nearest img

bilinear img

bicubic img

4）结果分析

最终得到的图像经过了拉伸、旋转和插值放大，三种插值方式同样存在实验 4 中所述的特点。