

# 数字图像处理作业

## (第一次)

姓名：周瑞虎

班级：自动化 61

学号：2160504027

提交日期：2019/3/4

摘要：人们每天接触大量的信息，而其中很大一部分就是图像，而且在人工智能的带动下，有关图像的各种机器视觉都面临新的挑战 and 机遇，因而，掌握一些基本的图像知识，这在今天看来是一门必修课，加深我们对于计算机处理问题的流程和理解，是大有裨益的。本次实验使用 matlab2018 为实验平台，通过自己编写代码，实现了 8 位图的不同比特数的量化，还有图像的旋转操作，shear 操作，还有三种使用频繁的插值算法的使用，更好的理解了位图的格式，和计算机处理图像问题的基本流程。

## 1、Bmp 图像格式简介,以 7.bmp 为例说明;

Bmp 图格式一般包括四部分

- 位图头文件数据结构, 它包含 BMP 图像文件的类型、文件标识、文件大小, 位图数据偏移等信息, 共包含 14Bytes。7.bmp 文件标识为 BM, 大小为 1134 字节, 位图数据偏移量是 1078 字节。
- 位图信息数据结构, 它包含有 BMP 图像的宽、高、压缩方法、位深、颜色数等信息; 7.bmp 图像高为 7, 宽为 7, 未压缩, 位深为 8, 颜色数为 256 种颜色。
- 调色板, 这个部分是可选的, 有些位图需要调色板, 有些位图, 比如真彩色图 (24 位的 BMP) 就不需要调色板; 7.bmp 文件包含有 256 个表项即颜色数据, 其中每一个表项的结构为

```
typedef struct tagRGBQUAD{  
    BYTE rgbBlue;//蓝色的亮度 (值范围为 0-255)  
    BYTE rgbGreen;//绿色的亮度 (值范围为 0-255)  
    BYTE rgbRed;//红色的亮度 (值范围为 0-255)  
    BYTE rgbReserved;//保留, 必须为 0  
}__attribute__((packed)) RGBQUAD;
```

- 位图数据, 这部分的内容根据 BMP 位图使用的位数不同而不同, 在 24 位图中直接使用 RGB, 而其他的小于 24 位的使用调色板中颜色索引值。在 7.bmp 图中, 使用的是索引值即 1 至 256, 调用颜色板里的数据, 以完成显示图片的目的。

7.bmp 文件大小是 1024 (调色板) + 54 (文件头) + 7 (行数) \* 8 (列数 windows 规定必须是 4 的倍数) = 1134 字节。

2、把 lena 512\*512 图像灰度级逐级递减 8-1 显示；

问题分析：lena 图本身是 8bits 量化的位图，量化比特位越多，显示的效果越好，实际结果也是如此。

实验过程：使用 matlab 提供的 imread 函数直接读取原始数据，之后每个 pixel 逐次处以 2 的 n 次方，并以 imshow (source\_data,[low high]) 函数显示，low 和 high 以这两个数据为起点，将数据拉伸至 0 至 255 显示。结果

实验结果：



图 1

不难看出，4 至 8bit 量化的图人眼看起来差距不大，但是从 3bit 开始有了严重区别，可以看出，量化位数越高，感官效果更好。

3、计算 lena 图像的均值方差;

问题分析:

Bmp 图的均值方差的计算公式是 (其分辨率是  $M * N$ )

$$\text{均值 } m = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N f(i,j)$$

$$\text{方差 } \sigma = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (f(i,j) - m)^2$$

实验过程: 直接调用 matlab 中的 mean 和 std 函数进行求值即可。

实验结果:

```
mean_value =  
  
    99.0512  
  
var_value =  
  
    2.7960e+03
```

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

问题分析：近邻插值算法就是把未知的坐标点经过变换矩阵得到的原图中的坐标值，然后把距离原图中最近点的像素值赋给它，双线性和三线性与之的区别是利用周围更多的点的像素值计算目标的像素值。

双线性插值的计算公式是

$$f(x + u, y + v) = [f(x, y), f(x, y + 1), f(x + 1, y), f(x + 1, y + 1)] *$$

$$[u * v, (1 - u) * v, u * (1 - v), (1 - u) * (1 - v)]'$$

双三次内插使用的公式是

$$f(x, y) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} x^i y^j$$

三种插值方法，计算复杂度依次提升，但是视觉效果也会越来越好，最近邻插值方法会有一些棋盘格效应，影响图像质量，但是速度快，双线性插值则是一种折中，在保持图像细节等方面，双三次插值是最好的。

其本次实验中有跃出边界等非法值的行为均是通过将图像扩展所解决的。

实验结果：



图 2 从左至右是最近邻，双线性，三线性

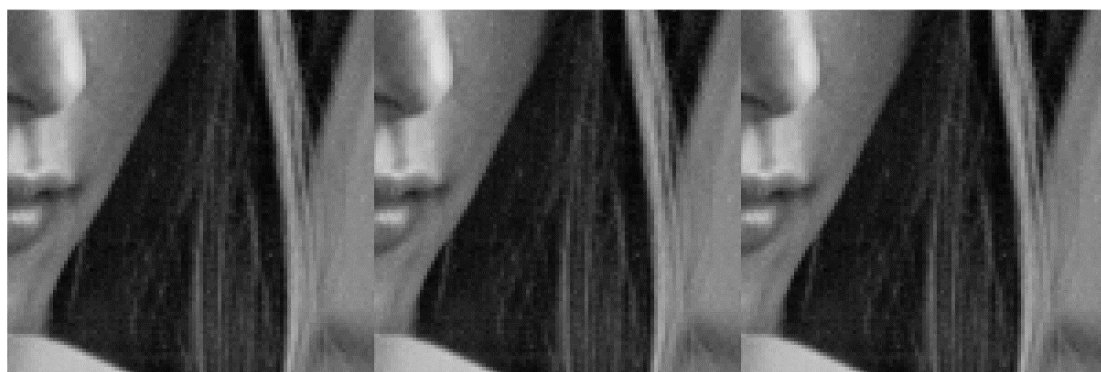


图 3 从左至右分别是最近邻，双线性，三线性

不难看出，在 lena 的鼻子轮廓上的纹理细节，第三副图的表现是最好的。这样比较有点不直观，但是如果把两张图放在文件夹下，然后打开图片管理器利用鼠标滚轮切换图片可以明显感觉到三线性插值效果最好，这也印证了我们的推测。

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;

水平 shear 和旋转变换都可以通过操作矩阵来实现, 设图像原坐标为(v,w,1)原坐标为(x,y,1),变换矩阵是 T。

$$\begin{bmatrix} x & y & 1 \end{bmatrix} = \begin{bmatrix} v & w & 1 \end{bmatrix} * T$$

对于旋转变换,

$$T = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

对于水平剪切

$$T = \begin{bmatrix} 1 & s_h & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

需要注意的是, 由于旋转变换是以图像左上角为原点的, 因而旋转会导致出现超出图像边界, 引起非法值, 因而比较好的做法是, 对变换之后有一定的偏移, 将矩阵变换为

$$T = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ M * \sin \theta & 1 & 1 \end{bmatrix}$$

而且还要注意提前计算好旋转会超出的边界, 加上之后就可以正常显示了。

下面 4-6 分别表示对 lena 水平 shear 之后, 分别最近邻, 双线性, 三线性插值后的结果, 7-9 分别表示 Elain1 旋转之后之后, 分别最近邻, 双线性, 三线性插值后的结果。

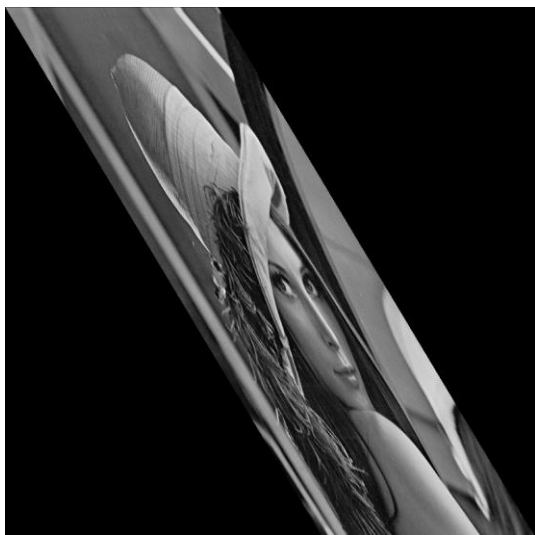


图 4

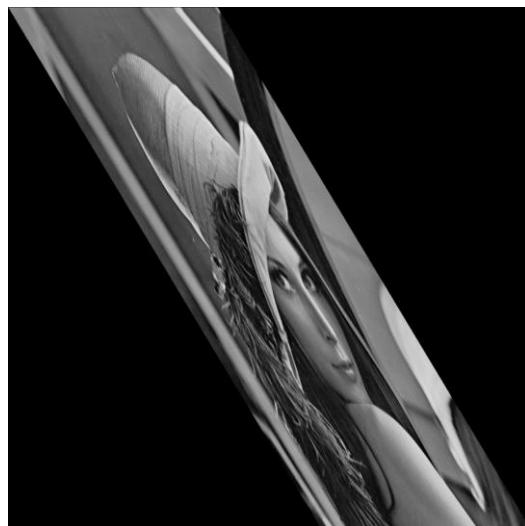


图 5

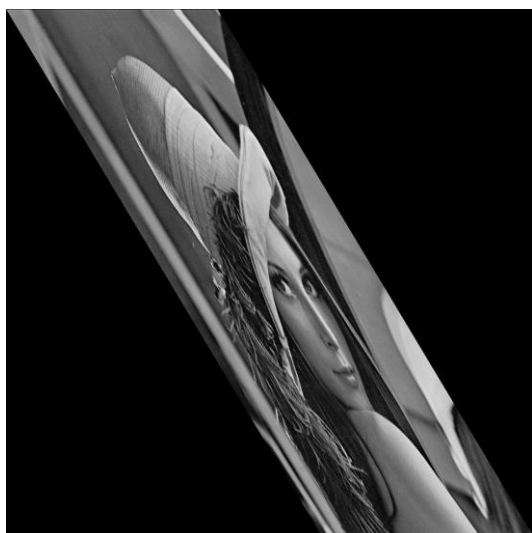


图 6



图 7



图 8



图 9



参考文献:

冈萨雷斯 数字图像处理 (第三版) 电子工业出版社