

# 数字图像处理

## 第一次作业实验报告

姓名：张斐然 班级：自动化 62 学号：2160300167

提交日期：2019.3.3

摘要：

本次实验利用 `matlab` 进行，主要进行 `bmp` 图象的基础处理，例如图像的插值，`shear`，旋转等操作，通过本次实验，我得以了解图象在计算机中的表示方式，不同插值算法的原理和区别，以及各自的优缺点。还可以了解图象旋转等变换的数学原理和相关函数，从而使得今后的图像处理更加快捷，为今后更加复杂的图像处理打下基础。

### 1.Bmp 图像格式简介,以 7.bmp 为例说明

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 1bit、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。

将 7.bmp 用 matlab 读取，可得到如下表格：

82	82	73	59	55	80	90
97	89	90	95	71	40	69
104	71	63	105	93	76	42
88	75	85	101	90	91	70
97	92	91	99	72	71	82
98	101	102	86	69	71	95
103	99	100	84	86	98	98

可知，黑白 bmp 图片由灰度矩阵表示，矩阵中每个数字对应该位置的灰度，默认在 0-255 之间，其中 0 为黑色，255 为白色。

### 2.把 lena 512\*512 图像灰度级逐级递减 8-1 显示

实现方法：

使用 `imread` 函数读取图像信息，得到 512X512 矩阵，对于矩阵中信息，依次除以二，得到灰度位数减少后的矩阵，分别对应相应的灰度位数使用 `imshow` 函数打印。

实验结果：

原	始	图	片
---	---	---	---



灰度等级 7:



灰度等级 6:



灰度等级 5:



灰度等级 4:



灰度等级 3:



灰度等级 2:





灰度等级 1:



### 3、计算 lena 图像的均值方差

实现方法:

使用 `imread` 读取图像，得到矩阵，对于矩阵，使用 `mean` 函数求平均值，使用 `std2` 函数求得标准差，平方后得到方差。

结果:

平均值 99.0512

标准差 52.8776

方差 2.7960e+03

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

实现方法:

#### 1.近邻插值法:

由近邻插值法的定义可知就是令变换后像素的灰度值等于距它最近的输入像素的灰度值  
设原图宽  $W$  高  $H$ ，变化后的图宽  $w$  高  $h$ ，则目标图像像素点坐标  $(x,y)$  对应的原图坐标  $(X,Y)$  的关系，按照比例一致 可知道

$$Y = W / w * y;$$

$$X = H / h * x;$$

则目标图  $(x, y)$  处的灰度值与原图  $(X, Y)$  处的灰度值一致:  $g(x, y) = g(X, Y)$ 。

#### 2.双线性插值法:

方法一: 根据几何关系，从变换后  $B$  图像中的坐标  $(x,y)$  得到原图  $A$  图像中的坐标  $(x/4,y/4)$ ，但是，映射得到的这个坐标  $(x/4,y/4)$  并没有刚好位于  $A$  图像中的整数坐标上，而是映射到了

四个像素坐标 $(a,b)$ 、 $(a+1,b)$ 、 $(a,b+1)$ 、 $(a+1,b+1)$ 所围成的矩形之间，其中， $a$ 、 $b$  是  $A$  图像的整数坐标。现在的问题就是如何根据  $A(a,b)$ 、 $A(a+1,b)$ 、 $A(a,b+1)$ 、 $A(a+1,b+1)$  这四个点上的灰度级求出  $A(x/4,y/4)$  处的灰度级。双线性插值技术采用的方法是：假设  $A$  图像的灰度级变化在纵向方向上是线性变化的，这样根据直线方程或者几何比例关系就能够求得  $(a,y/4)$  和  $(a+1,y/4)$  坐标处的灰度级  $A(a,y/4)$  和  $A(a+1,y/4)$ 。然后，再假设在  $((a,y/4),A(a,y/4))$  和  $((a+1,y/4),A(a+1,y/4))$  这两点所确定的直线上，灰度级仍然是线性变化的。求出直线方程，于是就可以求得  $(x/4,y/4)$  处的灰度级  $A(x/4,y/4)$ 。

由此可写出程序实现。

方法二：

直接使用 matlab 函数 `imresize`，设置参数为 'bicubic'，可简便的进行双线性插值。

#### 1. 双三次插值法

在这种方法中，函数  $f$  在点  $(x,y)$  的值可以通过矩形网格中最近的十六个采样点的加权平均得到，在这里需要使用两个多项式插值三次函数，每个方向使用一个。

直接使用 matlab 函数 `imresize`，设置参数为 'bilinear'，可简便的进行双三次插值。

实验结果：



最临近插值（matlab 警告：图像太大，无法在屏幕上显示；将以 33% 显示，后同）



双线性插值



### 双三次插值

观察人物帽檐对比可发现，最邻近插值有明显的锯齿感，双线性插值则几乎看不出锯齿感，双三次插值效果最好，边缘最为柔和。

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；

实现方式：

#### 1.水平 shear

由水平 shear 公式可知，设原图的坐标为(x,y)，变换后图形坐标为 (X,Y)，则可得

$X=x;$

$Y+X*\text{参数}=y;$

由此公式可依次对像素点进行变换，得到最终图像。

#### 2. 旋转 30 度

方法一：

给定角度，则可以旋转矩阵如下所示

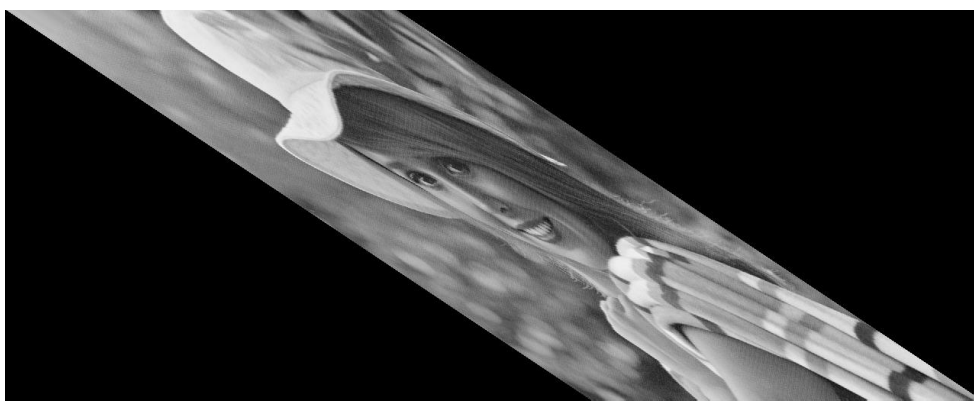
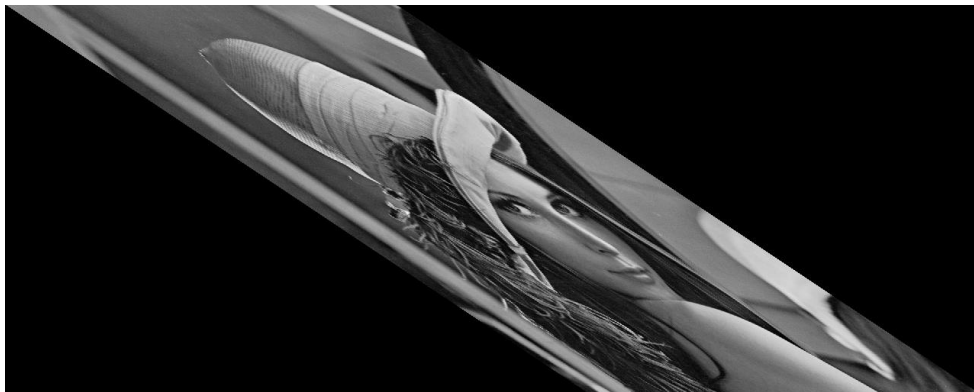
$$\begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

将原图形坐标矩阵乘以变换矩阵则可得到旋转后图形的坐标。

方法二：

使用 imrotate 函数可直接对图像进行旋转。

结果：



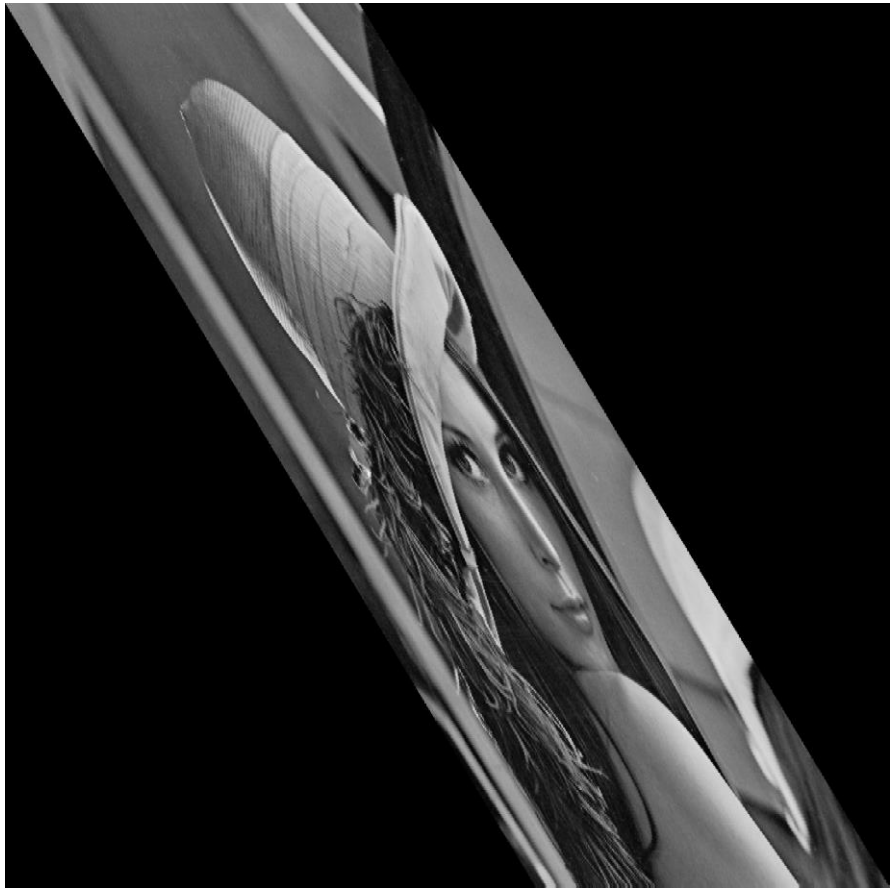
水平 shear (参数为 1.5)

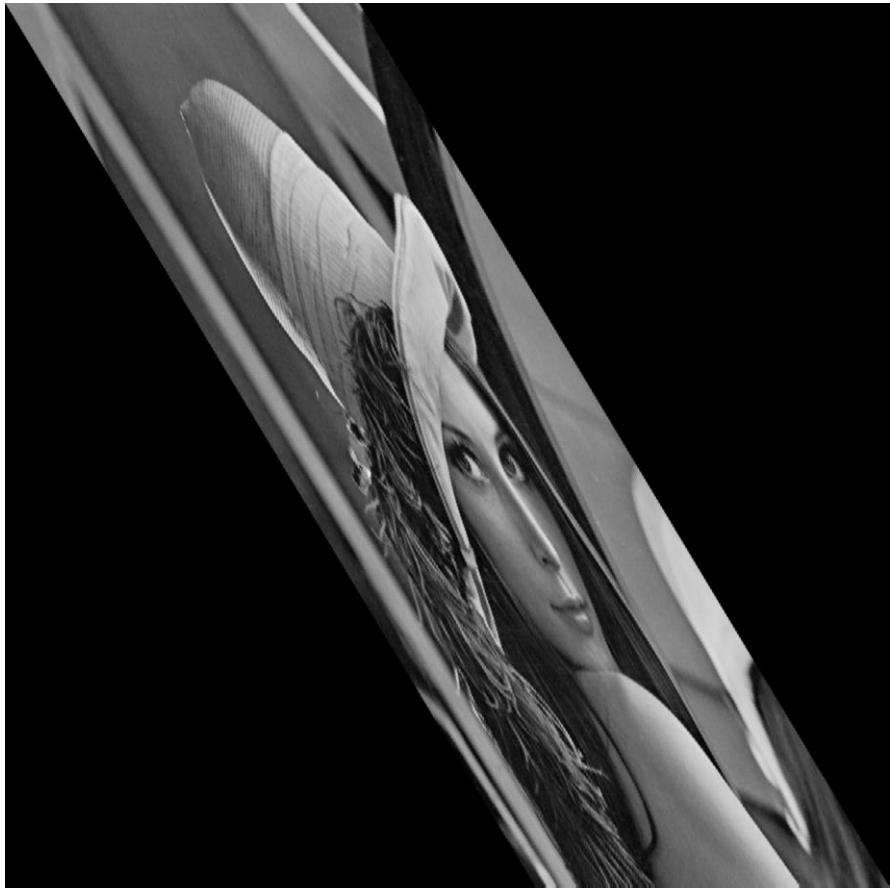


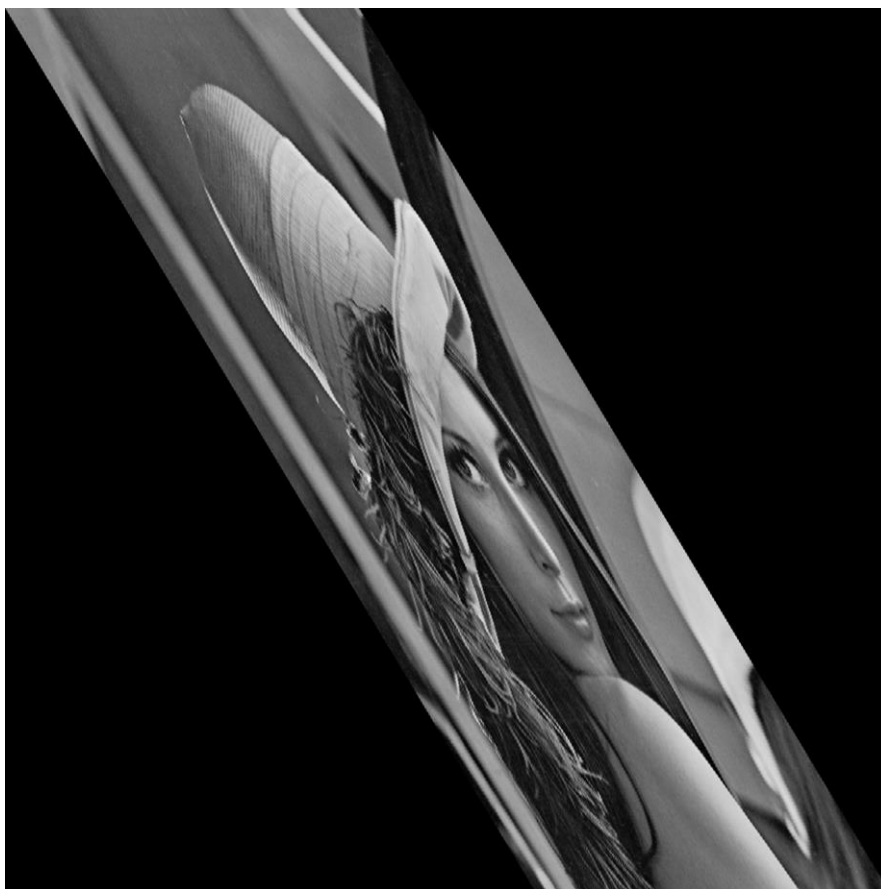


旋转 30 度

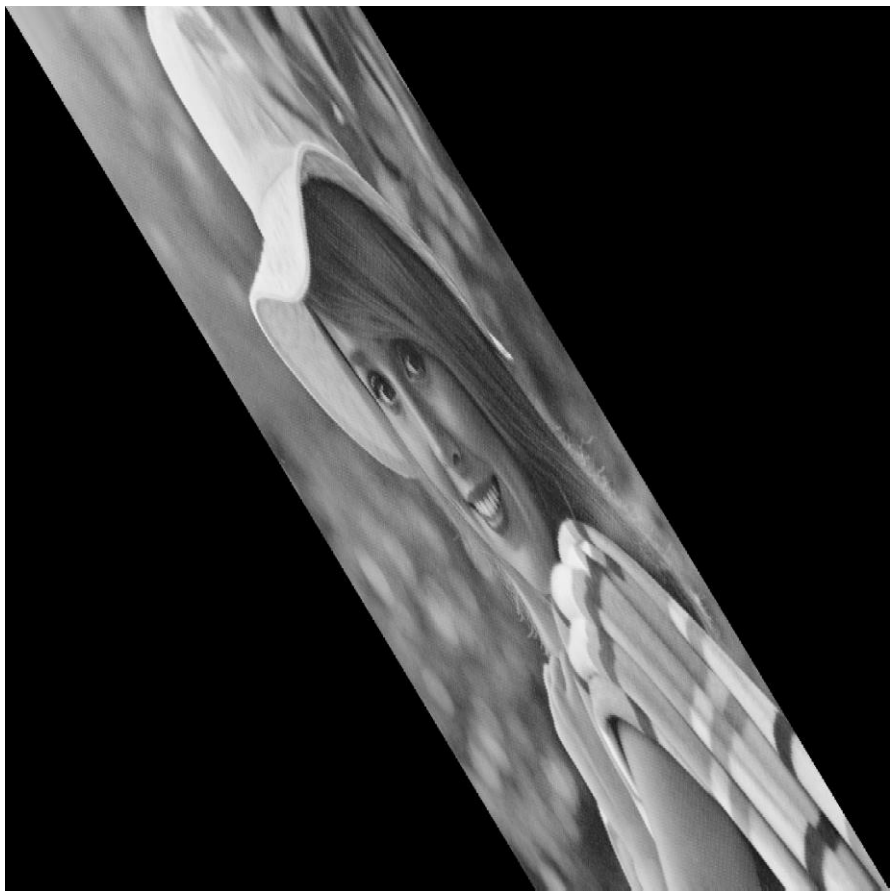


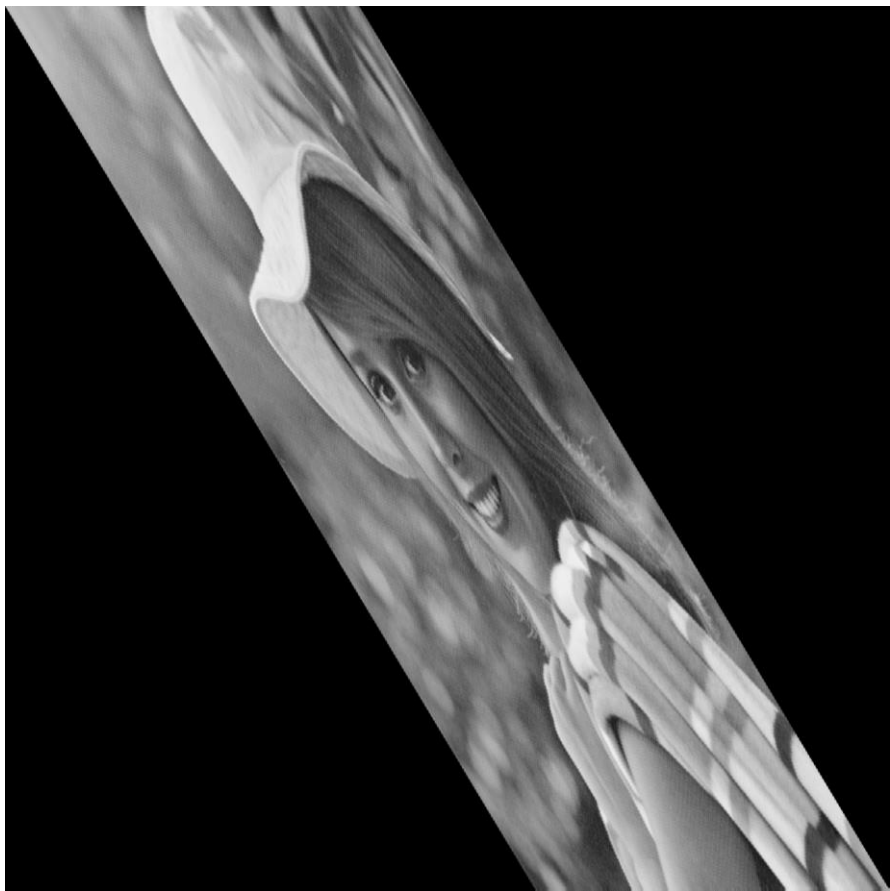


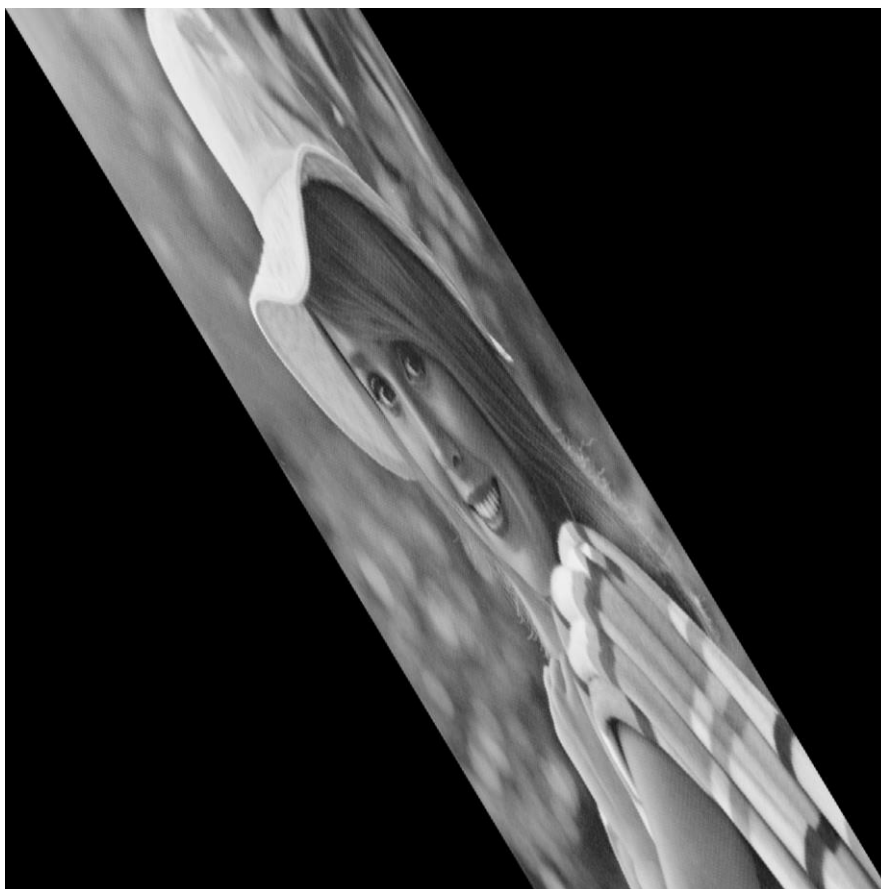




lena 水平 shear 后 ZOOM (分别使用最邻近, 双线性, 双三次)







Elain1 水平 shear 后 ZOOM (分别使用最邻近, 双线性, 双三次)









Lena 旋转 30 度后 ZOOM（分别使用最邻近，双线性，双三次插值）







Elaine1 旋转 30 度后 ZOOM（分别使用最邻近，双线性，双三次插值）

参考文献：

百度百科

CSDN

博客园

GitHub