

西安交通大学

数字图像处理作业报告

作业 1：BMP 格式与基本变换

摘要：本次报告首先简要介绍了 BMP 图像格式，并使用 UltraEdit 打开 7.bmp，对其内容进行了详细分析。然后通过 Matlab 编程实现了以下图像的基本变换与计算：（1）将图像按灰度级逐级递显示；（2）计算图像的均值方差；（3）使用近邻、双线性性和双三次插值法将 512×512 的图像缩放到 2048×2048 ；（4）将两幅图像分别进行水平错切和旋转 30 度，并采用用近邻、双线性和双三次插值法缩放到 2048×2048 。

关键词：BMP，UltraEdit，数字图像基本变换，MATLAB

姓 名：胡 欣 盈

班 级：自 动 化 9 4

学 号：2 1 9 4 3 2 3 1 7 6

提交日期：2022 年 3 月 12 日

目 录

一、BMP 图像格式简介.....	3
(1) BMP 简介.....	3
(2) BMP 的组成成分.....	3
(3) 以 7. bmp 为例具体分析.....	4
二、灰度级变换.....	5
三、计算图像的均值方差.....	6
四、图像缩放.....	7
五、图像的水平错切、旋转和缩放.....	8
附录.....	12
附录 1: 参考文献.....	12
附录 2: 源代码.....	13

一、BMP 图像格式简介

(1) BMP 简介

BMP (全称 Bitmap) 是 Windows 操作系统中的标准图像文件格式, 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。

(2) BMP 的组成成分

位图文件可看成由 4 个部分组成:

1、位图文件头 (bitmap-file header): 它占到 14 字节, 包含 BMP 图像文件的类型、显示内容等信息

2、位图信息头 (bitmap-information header): 它占到 40 字节, 包含有 BMP 图像的宽、高、压缩方法, 以及定义颜色等信息

3、调色板 (color table): 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

4、定义位图的字节, 位图数据, 即图像数据 (Data Bits 或 Data Body) 阵列: 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

（3）以 7.bmp 为例具体分析

使用 UltraEdit 打开文件 7.bmp，如图 1-1 所示。由于每个字段是小端描述，故需从后往前读。

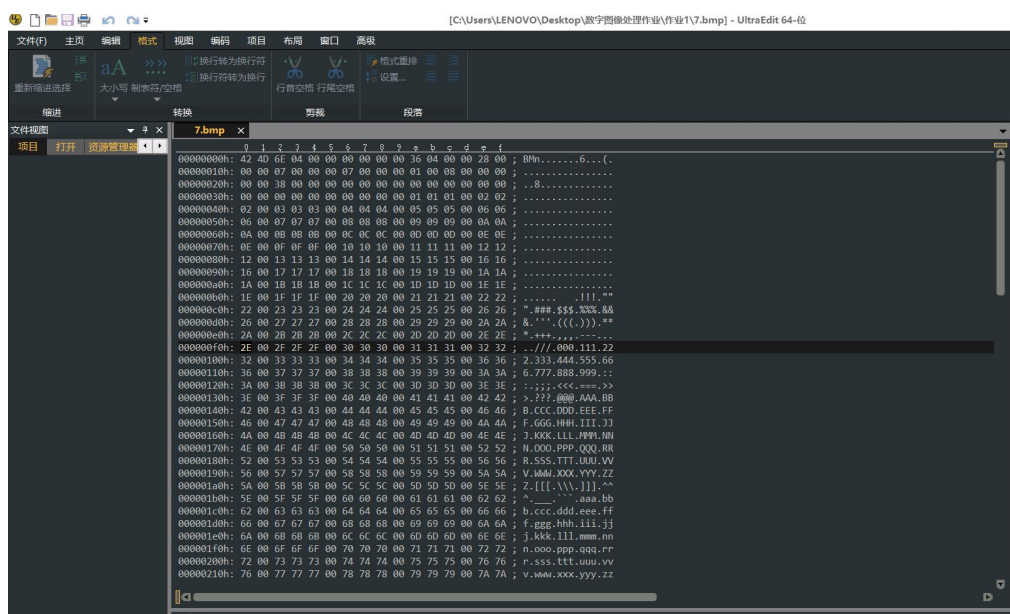


图 1-1 7.bmp 内容

将文件 7.bmp 的内容与 BMP 文件格式对应，可得到下表

字段	字节长度	意义	7.bmp
bfType	2	一定为 19778，其转化为十六进制为 0x4d42，对应的字符串为 BM	4D 42
bfSize	4	文件大小	00 00 04 6E (1134 Bytes)
bfReserved1	2	保留字 1，一般为 0	00 00
bfReserved2	2	保留字 2，一般为 0	00 00
bfOffBits	4	位图数据部分相对于文件首的起始偏移量	00 00 04 36 (1078 Bytes)
biSize	4	当前结构体的大小	00 00 00 28 (40 Bytes)
biWidth	4	图像宽度（像素）	00 00 00 07 (7pixels)
biHeight	4	图像高度（像素）	00 00 00 07 (7pixels)
biPlanes	2	图像的帧数，一般为 1	00 01
biBitCount	2	比特数，一像素所占的位数	00 08 (8 Bytes/pixel)
biCompression	4	图像数据压缩类型	00 00 00 00

biSizeImage	4	图像大小（字节）	00 00 00 38 (56 Bytes)
biXPelsPerMeter	4	水平分辨率	00 00 00 00
biYPelsPerMeter	4	垂直分辨率	00 00 00 00
biClrUsed	4	实际颜色索引数	00 00 00 00
biClrImportant	4	重要的颜色索引数	00 00 00 00

二、灰度级变换

本题要求把 lena 512*512 图像灰度级逐级递减 8-1 显示，不妨对图像进行的操作是直接灰度变换。直接灰度变化可以借助图像的位面表示进行。对一幅用多个比特表示其灰度值的图像来说，其中的每个比特可看做表示了一个二值的平面，也称位面，如 2-1 所示。一幅其灰度级用 8bit 表示的图像有 8 个位面，一般用位面 0 表示最低位面，位面 7 表示最高位面。借助图像的位面表示形式可通过对图像特定位面的操作来达到对图像的增强效果。

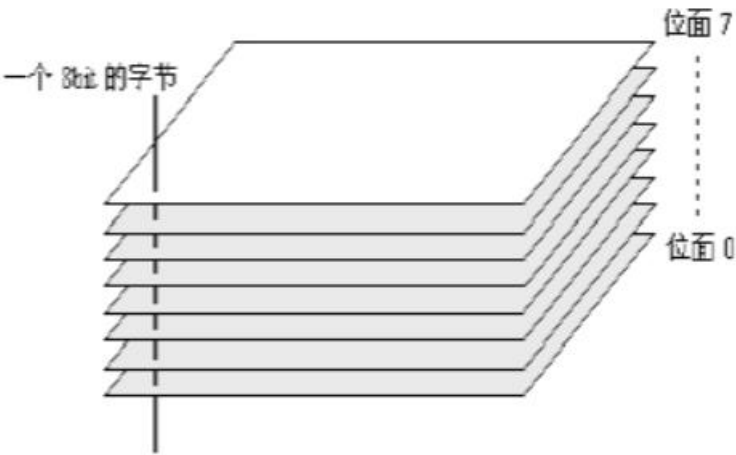


图 2-1 图像的位面

通过阅读图片属性得知，该图片位深度为 8，因此深度范围为 0~255。只要依次将深度除以 2，即可得到灰度逐级递减的图片结果如图 2-2 所示。

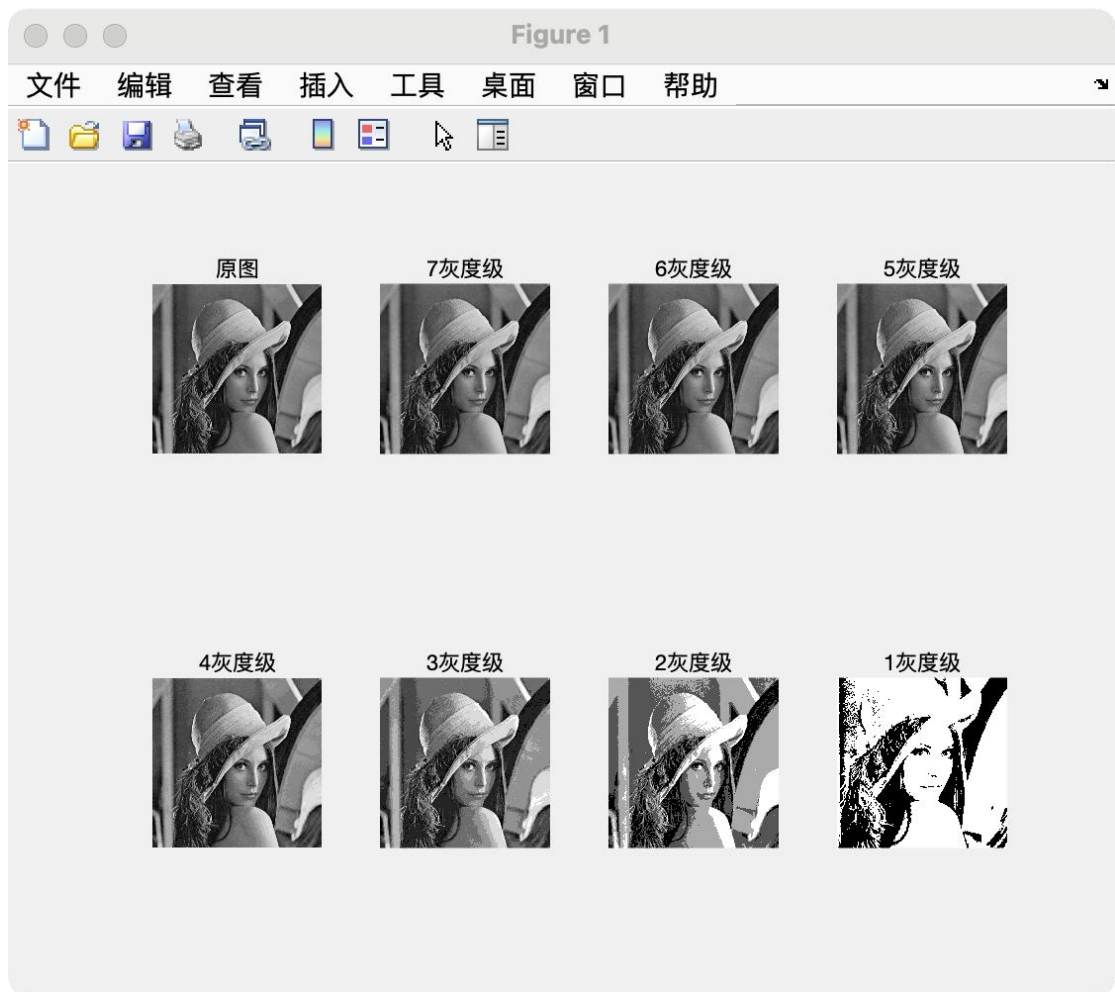


图 2-2 图像 lena.bmp 灰度级逐级递减

三、计算图像的均值方差

本题要求计算 lena.bmp 图像的均值方差, 在 MATLAB 中, 用 `imread()` 将图片读入后, 直接用 `mean2()`, 和 `std2()` 函数对图片矩阵进行操作即可得到图像的均值和标准差, 在对标准差求平方即可得到方差。结果如下图所示, $\text{mean} = 99.0512$ $\text{var} = 2.7960\text{e}+03$ 。

```

命令窗口
mean =
    99.0512

var =
    2.7960e+03

```

图 3-1 图像 lena.bmp 的均值方差

四、图像缩放

本题要求把 512×512 的 `lena.bmp` 图像用近邻、双线性性和双三次插值法放大到 2048×2048 。在用 MATLAB 读入图片后，通过 `imresize` 函数将图片进行放大，函数格式为 `imresize(name, [x, y], 'pattern')`，可通过将模式 'pattern' 分别设置为：最近邻 `nearest`，双线性 `bilinear`，双三次 `bicubic` 实现不同方法的插值处理。

考虑到 MATLAB 无法显示 2048×2048 大小的图片，会对其进行缩放，为方便以原图大小观察，使用 `imtool()` 函数来显示图像。如图 4-1~4-3 所示。由于难以直接对比出插值效果，我们选取图像中的细节部分进行放大比对，如图 4-4~4-6 所示，可以明显看出，双线性和双三次插值的图片线条更平滑，效果要优于最近邻插值。

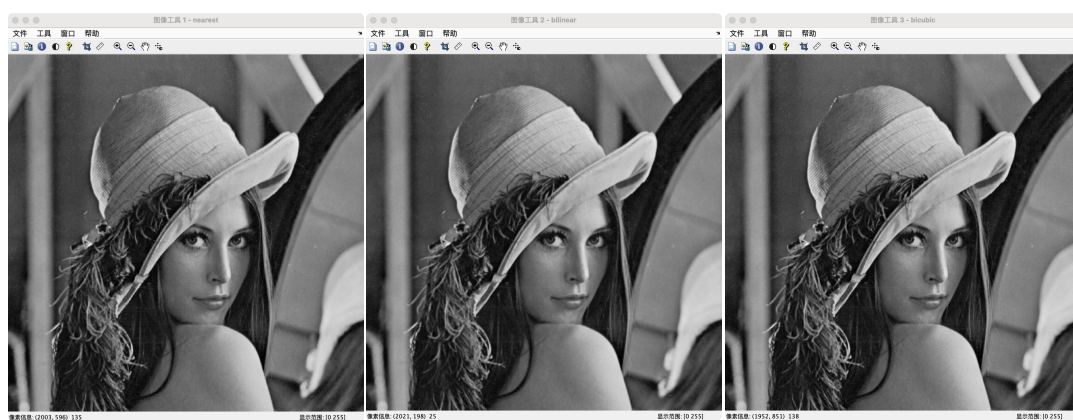


图 4-1~4-3 近邻、双线性性和双三次插值法放大后的 `lena.bmp`



4-4~4-6 近邻、双线性性和双三次插值法放大后的 `lena.bmp` 局部图

五、图像的水平错切、旋转和缩放

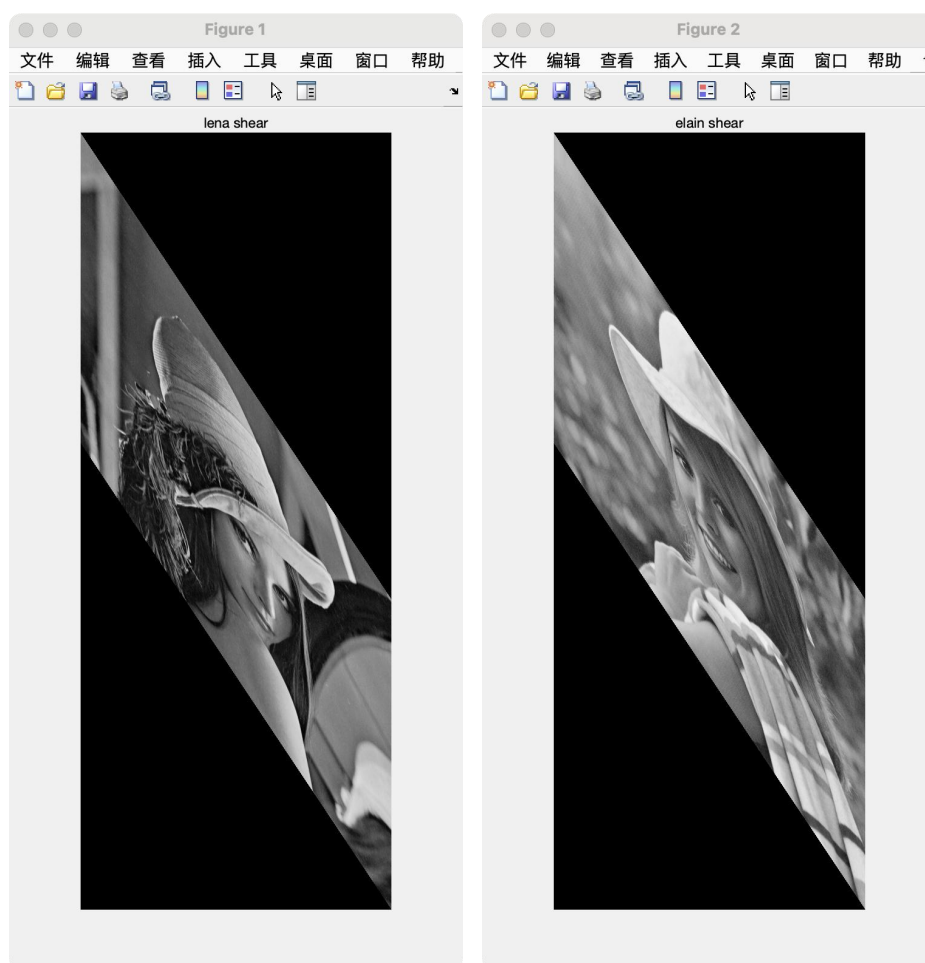
本题要求把 lena.bmp 和 elain.bmp 图像分别进行参数为 1.5 的水平错切、旋转 30 度的操作，并采用用近邻、双线性和双三次插值法缩放到 2048*2048。

(1) 水平错切与旋转

首先通过构建仿射矩阵的方法，利用 affine2d 和 imwarp 函数进行二维仿射几何变换，从而完成水平错切与旋转变换。

进行水平错切的仿射矩阵为：
$$T_{\text{shear}} = \begin{bmatrix} 1 & 1.5 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

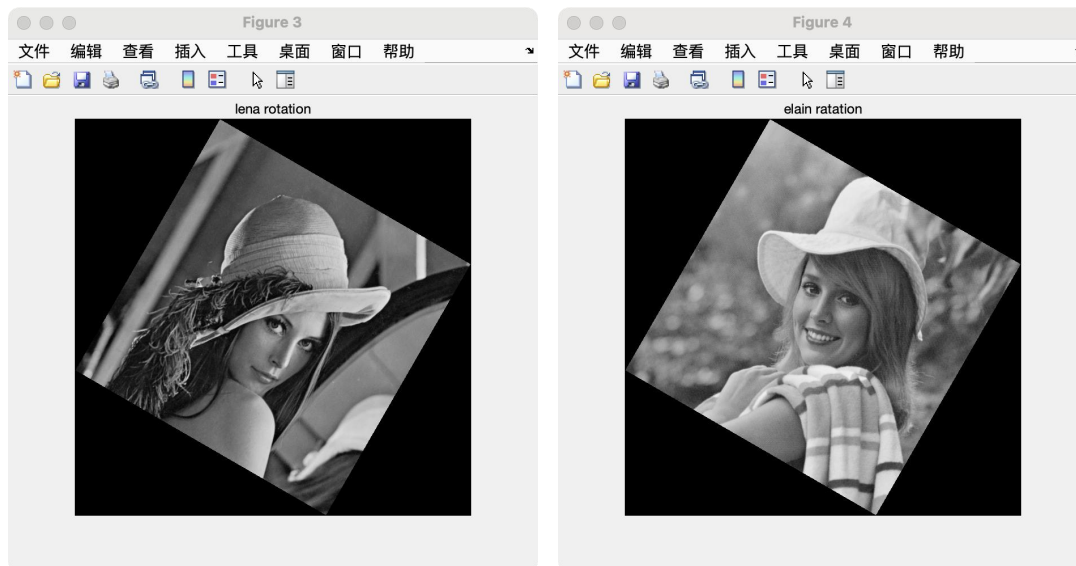
错切后的图像如图 5-1、5-2 所示。



5-1~5-2 lena.bmp 和 elain.bmp 进行参数为 1.5 的水平错切

旋转 30° 的仿射矩阵为: $T_{\text{rotation}} = \begin{bmatrix} \cos 30^\circ & \sin 30^\circ & 0 \\ -\sin 30^\circ & \cos 30^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix}$

旋转 30° 后的图像如图 5-3、5-4 所示。



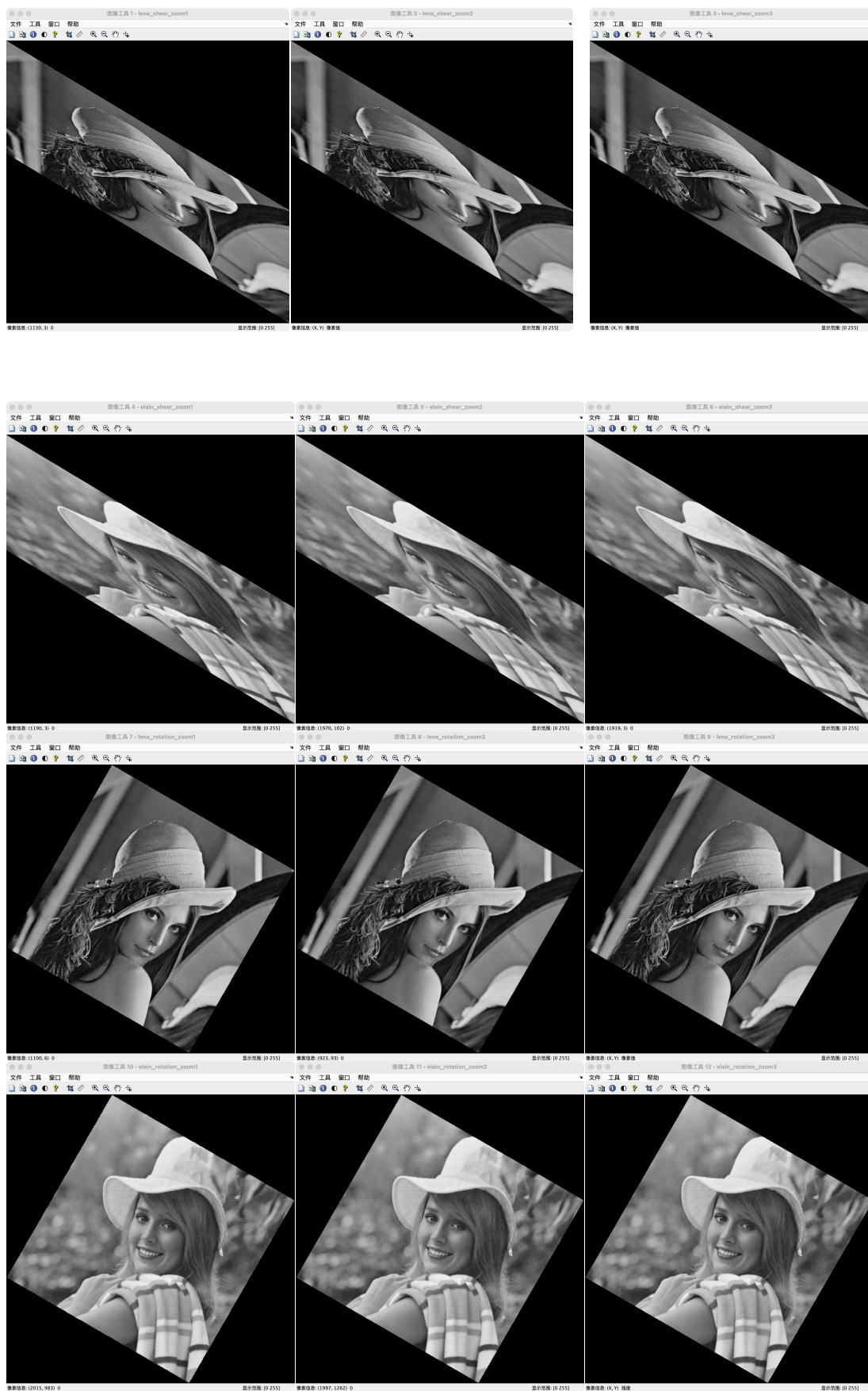
5-3~5-4 lena.bmp 和 elain.bmp 进行 30° 旋转后

(2) 图像缩放

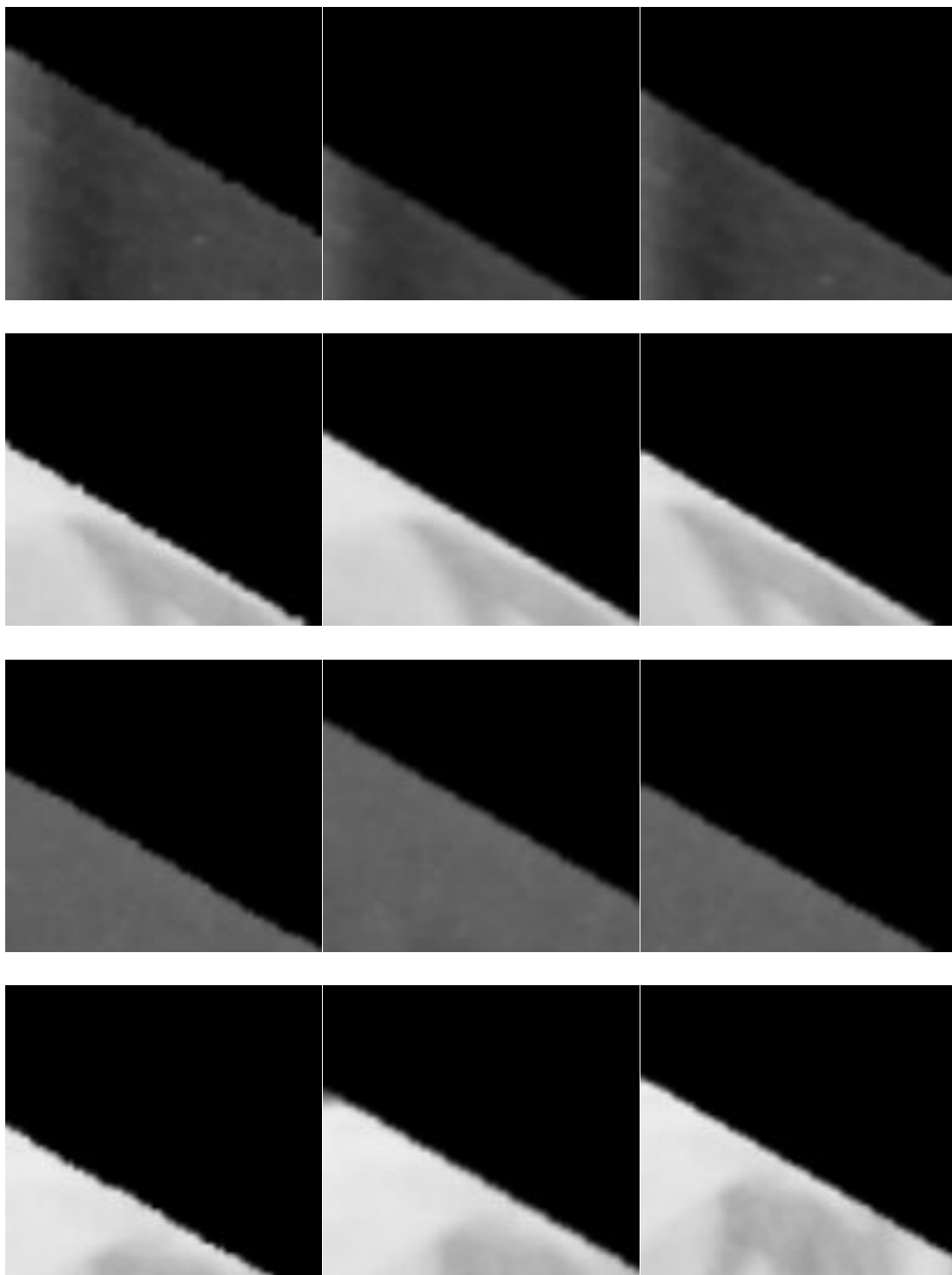
具体缩放步骤与第四大点基本一致, 结果如图 5-5~5-16 所示。

由于难以直接对比出插值效果, 我们仍旧选取图像中的细节部分进行放大比对, 如图 5-17~5-28 所示。根据图片细节部分的对比, 我们不难得出和之前一样的结论, 双线性插值和双三次插值得到的线条要比最近邻插值得到的更加平滑。

虽然双三次差值是对双线性插值的改善, 但从本次实验结果并不能非常明显地体现出双三次插值的优势。



5-5~5-16 近邻、双线性插值和双三次插值法缩放
水平错切与旋转后的 lena.bmp 与 elain.bmp



5-5~5-16 近邻、双线性 and 双三次插值法缩放

水平错切与旋转后的 lena.bmp 与 elain.bmp 的局部图

附录

附录 1：参考文献

- [1] 阮秋琦, 阮宇智等. 数字图像处理(第三版)(美)冈萨雷斯[M], 2003, 电子工业出版社
- [2] 门敬文•数字图像处理 MATLAB 版[M]. 2007. 2, 国防工业出版社.
- [3] 数字图像处理——虚宇宸轩-CSDN 博客

附录 2：源代码

```
1.  clear;
2.  clc;
3.
4.  %Task1
5.  lena=imread('Users/macbookpro/Desktop/lena.bmp');
6.  lena7=lena/2;  lena6=lena/4;  lena5=lena/8;
7.  lena4=lena/16;  lena3=lena/32;  lena2=lena/64;
8.  lena1=lena/128;
9.  subplot(2,4,1);  imshow(lena,[0,256]);title('原图');
10. subplot(2,4,2);  imshow(lena7,[0,127]);title('7 灰度级');
11. subplot(2,4,3);  imshow(lena6,[0,63]);title('6 灰度级');
12. subplot(2,4,4);  imshow(lena5,[0,31]);title('5 灰度级');
13. subplot(2,4,5);  imshow(lena4,[0,15]);title('4 灰度级');
14. subplot(2,4,6);  imshow(lena3,[0,7]);title('3 灰度级');
15. subplot(2,4,7);  imshow(lena2,[0,3]);title('2 灰度级');
16. subplot(2,4,8);  imshow(lena1,[0,1]);title('1 灰度级');
17.
18. %Task2
19. mean = mean2(lena)
20. std = std2(lena);
21. var = std^2
22.
23. %Task3
24. nearest=imresize(lena,[2048,2048],'nearest');
25. bilinear=imresize(lena,[2048,2048],'bilinear');
26. bicubic=imresize(lena,[2048,2048],'bicubic');
27. imtool(nearest);
28. imtool(bilinear);
29. imtool(bicubic);
30.
31. %Task4
32. elain=imread('Users/macbookpro/Desktop/elain1.bmp');
33. %构建仿射矩阵:
34. T_shear=[1,1.5,0;0,1,0;0,0,1];
35. T_rotation=[cos(pi/6),sin(pi/6),0;-sin(pi/6),cos(pi/6),0;0,0,1];
36.
37. shear=affine2d(T_shear);
38. rotation=affine2d(T_rotation);
39.
40. lena_shear=imwarp(lena,shear);
41. elain_shear=imwarp(elain,shear);
```

```
42. lena_rotation=imwarp(lena,rotation);
43. elain_rotation=imwarp(elain,rotation);
44.
45. figure(1);imshow(lena_shear);title("lena shear");
46. figure(2);imshow(elain_shear);title("elain shear");
47. figure(3);imshow(lena_rotation);title("lena rotation");
48. figure(4);imshow(elain_rotation);title("elain rotation");
49.
50. lena_shear_zoom1=imresize(lena_shear,[2048,2048],'nearest');
51. lena_shear_zoom2=imresize(lena_shear,[2048,2048],'bilinear');
52. lena_shear_zoom3=imresize(lena_shear,[2048,2048],'bicubic');
53. elain_shear_zoom1=imresize(elain_shear,[2048,2048],'nearest');
54. elain_shear_zoom2=imresize(elain_shear,[2048,2048],'bilinear');
55. elain_shear_zoom3=imresize(elain_shear,[2048,2048],'bicubic');
56.
57. lena_rotation_zoom1=imresize(lena_rotation,[2048,2048],'nearest');
58. lena_rotation_zoom2=imresize(lena_rotation,[2048,2048],'bilinear');
59. lena_rotation_zoom3=imresize(lena_rotation,[2048,2048],'bicubic');
60. elain_rotation_zoom1=imresize(elain_rotation,[2048,2048],'nearest');
61. elain_rotation_zoom2=imresize(elain_rotation,[2048,2048],'bilinear');
62. elain_rotation_zoom3=imresize(elain_rotation,[2048,2048],'bicubic');
63.
64. imtool(lena_shear_zoom1);
65. imtool(lena_shear_zoom2);
66. imtool(lena_shear_zoom3);
67. imtool(elain_shear_zoom1);
68. imtool(elain_shear_zoom2);
69. imtool(elain_shear_zoom3);
70. imtool(lena_rotation_zoom1);
71. imtool(lena_rotation_zoom2);
72. imtool(lena_rotation_zoom3);
73. imtool(elain_rotation_zoom1);
74. imtool(elain_rotation_zoom2);
75. imtool(elain_rotation_zoom3);
```