

数字图像处理第一次作业报告

学号：2160504063

姓名：陈梹捷频

班级：自动化 63

提交日期：2019/3/4

摘要

本次作业首先简单阐述了 BMP 图像格式及其相关数据结构，随后主要完成了作业要求中关于图像处理与计算的各项任务。本次作业以 Matlab 2018 为平台，通过对 lena.bmp, elain.bmp 图像文件的编程处理，分别得到了 lena.bmp 图像的 8 到 1 级灰度逐级递减显示，lena.bmp 图像的均值和方差，通过近邻、双线性和双三次插值法对 lena.bmp 进行 4 倍放大后得到的 2048×2048 尺寸图像，和对 lena.bmp、elain.bmp 图像分别进行水平偏移变换和旋转变换后的图像及其 4 倍插值放大图像。

1、Bmp 图像格式简介,以 7.bmp 为例说明;

典型的 BMP 图像文件由四部分组成:

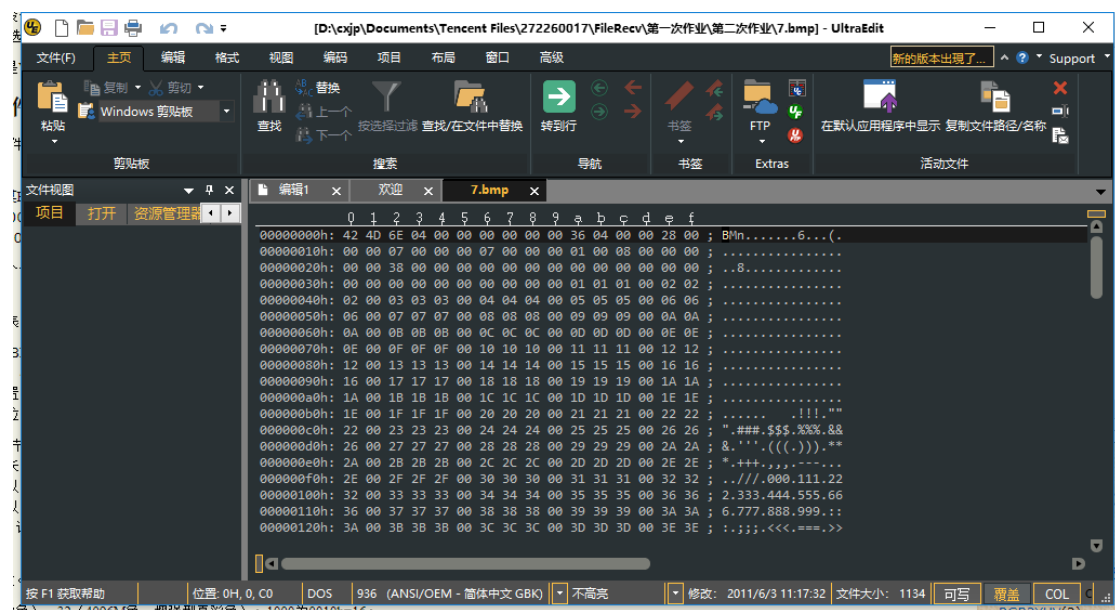
1: 位图头文件数据结构,它包含 BMP 图像文件的类型、显示内容等信息;

2: [位图](#)信息数据结构,它包含有 BMP 图像的宽、高、压缩方法,以及定义颜色等信息;

3: [调色板](#),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: 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。

用 ultraedit 打开 7.bmp 得到其数据如下:



一、bmp 文件头

1-2 : 424dh = 'BM',表示这是 Windows 支持的位图格式。有很多声称开头两个字节必须为'BM'才是位图文件,从上表来看应为开头两个字节必须为'BM'才是 Windows 位图文件。

3-5 : 0000046eh = 1126 B = 1.10 kB, 通过查询文件属性发现一致。

6-9 : 这是两个保留段,为 0。

A-D: 00000436h = 1078。即从文件头到位图数据需偏移 1078 字节。

共有 14 个字节。

二、位图信息头

0E-11: 00000028h = 40,这就是说我这个位图信息头的大小为 40 个字节。

12-15: 00000007h = 7, 图像宽为 7 像素, 与文件属性一致。

16-19: 00000007h = 7, 图像高为 7 像素, 与文件属性一致。这是一个正数, 说明图像数据是从图像左下角到右上角排列的。

1A-1B: 0001h, 该值总为 1。

1C-1D: 0008h = 8, 表示每个像素占 8 个比特, 即该图像共有 256 种颜色。

1E-21: 00000000h, BI_RGB, 说明本图像不压缩。

22-25: 00000000h, 图像的大小, 因为使用 BI_RGB, 所以设置为 0。

26-29: 00000000h, 水平分辨率, 缺省。

2A-2D: 00000000h, 垂直分辨率, 缺省。

2E-31: 00000007h = 7,说明本位图实际使用的颜色索引数为 7, 与 1C-ID 得到的结论一致。

32-35: 00000007h = 7,说明本位图重要的颜色索引数为 7, 与前面得到的结论一致。

三、调色板

前面也已经提过, 调色板其实是一张映射表, 标识颜色索引号与其代表的颜色的对应关系。它在文件中的布局就像一个二维数组 `palette[N][4]`, 其中 `N` 表示总的颜色索引数, 每行的四个元素分别表示该索引对应的 B、G、R 和 Alpha 的值, 每个分量占一个字节。如不设透明通道时, Alpha 为 0。因为前面知道, 本图有 256 个颜色索引, 因此 `N = 256`。索引号就是所在行的行号, 对应的颜色就是所在行的四个元素。一共有 256 种颜色, 每个颜色占用 4 个字节, 就是一共 1024 个字节, 再加上前面的文件信息头和位图信息头的 54 个字节加起来一共是 1078 个字节。也就是说在位图数据出现之前一共有 1078 个字节, 与我们在文件信息头得到的信息: 文件头到文图数据区的偏移为 1078 个字节一致!

四、位图数据

每个像素占一个字节, 取得这个字节后, 以该字节为索引查询相应的颜色, 并显示到相应的显示设备上就可以了。注意: 由于位图信息头中的图像高度是正数, 所以位图数据在文件中的排列顺序是从左下角到右上角, 以行为主序排列的。也即我们见到的第一个像素 60 是图像最左下角的数据, 第二个人像素 60 为图像最后一行第二列的数据, ...一直到最后一行的最后一列数据, 后面紧接的是倒数第二行的第一列的数据, 依此类推。

2、把 lena 512*512 图像灰度级逐级递减 8-1 显示;

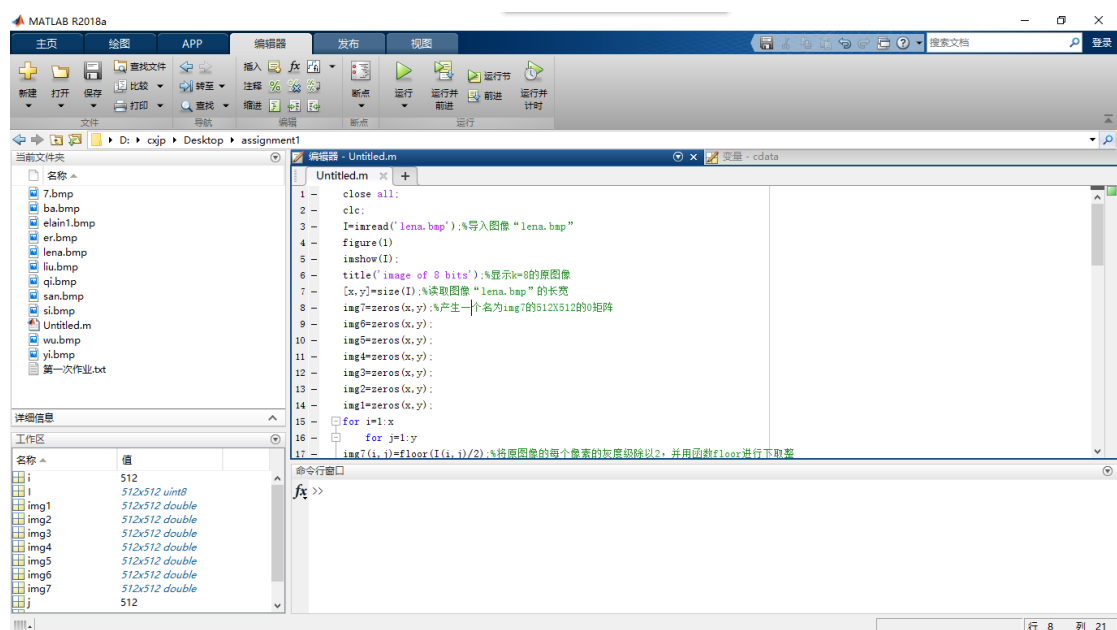
源代码:

```
close all;clear all;clc;
I=imread('lena.bmp');%导入图像“lena.bmp”
figure(1) imshow(I);
title('image of 8 bits');%显示 k=8 的原图像
[x,y]=size(I);%读取图像“lena.bmp”的长宽
img7=zeros(x,y);%产生一个名为 img7 的 512X512 的 0 矩阵
img6=zeros(x,y);
img5=zeros(x,y); img4=zeros(x,y); img3=zeros(x,y); img2=zeros(x,y); img1=zeros(x,y);
```

```

    for i=1:x for j=1:y
        img7(i,j)=floor(I(i,j)/2);%将原图像的每个像素的灰度级除以 2，并用函数 floor 进行下取整
    end end figure(2)
imshow((img7),[0,127]); title('image of 7 bits'); for i=1:x for j=1:y
img6(i,j)=floor(img7(i,j)/2);%将原图像的每个像素的灰度级除以 2，并用函数 floor 进行下取整
    end end figure(3)
imshow((img6),[0,63]); title('image of 6 bits'); for i=1:x for j=1:y
img5(i,j)=floor(img6(i,j)/2);%将原图像的每个像素的灰度级除以 2，并用函数 floor 进行下取整
    end end figure(4)
imshow((img5),[0,31]); title('image of 5 bits'); for i=1:x for j=1:y
img4(i,j)=floor(img5(i,j)/2);%将原图像的每个像素的灰度级除以 2，并用函数 floor 进行下取整
    end end figure(5)
imshow((img4),[0,15]); title('image of 4 bits'); for i=1:x for j=1:y
img3(i,j)=floor(img4(i,j)/2);%将原图像的每个像素的灰度级除以 2，并用函数 floor 进行下取整
    end end figure(6)
imshow((img3),[0,7]); title('image of 3 bits'); for i=1:x for j=1:y
img2(i,j)=floor(img3(i,j)/2);%将原图像的每个像素的灰度级除以 2，并用函数 floor 进行下取整
    end end figure(7)
imshow((img2),[0,3]); title('image of 2 bits'); for i=1:x for j=1:y
img1(i,j)=floor(img2(i,j)/2);%将原图像的每个像素的灰度级除以 2，并用函数 floor 进行下取整
    end end figure(8)
imshow((img1),[0,1]); title('image of 1 bits');

```



以下是实验结果（8~1）

image?of?8?bits



image?of?7?bits



image?of?6?bits



image?of?5?bits



image?of?4?bits



image?of?3?bits



image?of?2?bits



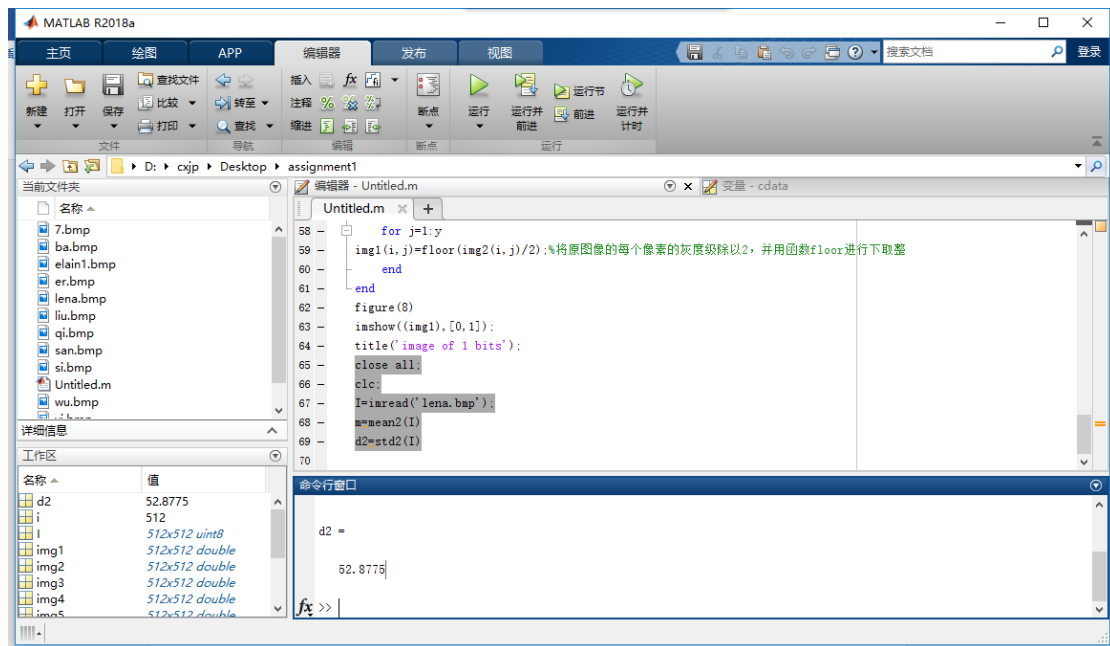


3、计算 lena 图像的均值方差;

源代码:

```
close all;  
clc;  
I=imread('lena.bmp');  
m=mean2(I)  
d2=std2(I)
```

以下是实验结果:



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

源代码:

```

a=imread('lena.bmp');
figure(1);
imshow(a);
title('ystdx');
b1=imresize(a,[2048 2048],'nearest');
figure(2);
imshow(b1);
title('zjlczf');
b2=imresize(a,[2048 2048],'bilinear');
figure(3);
imshow(b2);
title('sxxczf');
b3=imresize(a,[2048 2048],'bicubic');
figure(4);
imshow(b3);

```

```
title('sscczf');
```

以下是实验结果：

原始图像：

ystx



最近邻插值法：

zjlczf



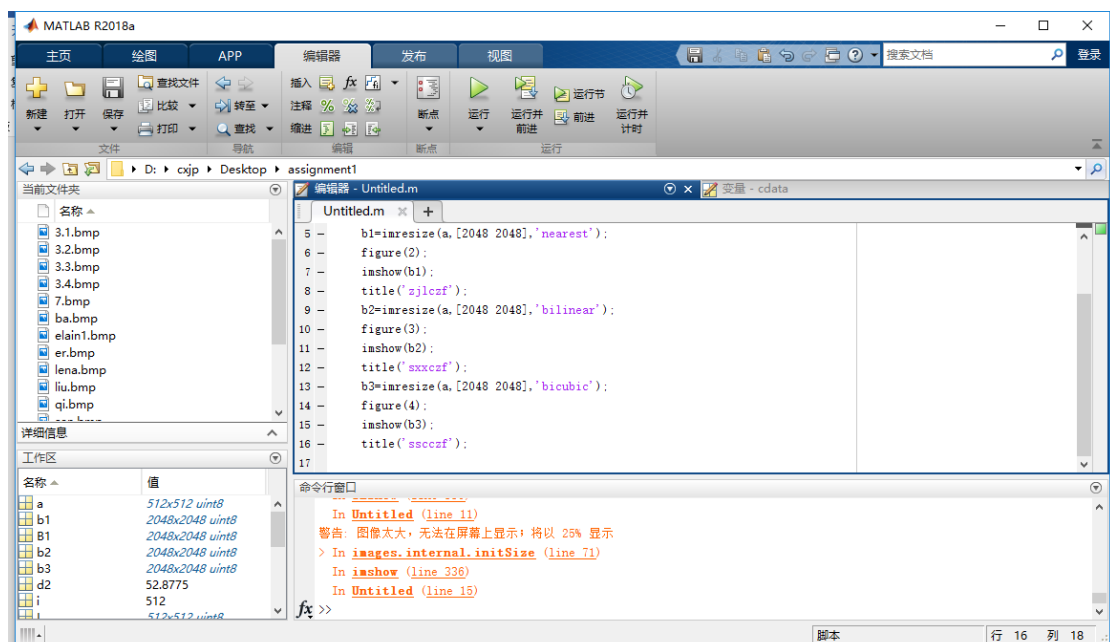
双线性插值法：

sxxczf



双三次插值法：

sscczf



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；

Lena 的水平源代码：

transformtype='affine';

```

transformmatrix=[1 1.5 0;0 1 0;0 0 1];
T=maketform(transformtype,transformmatrix);
I=imread('lena.bmp');
nI=imtransform(I,T);
figure(1);
imshow(I);
title('ystx');
b1=imresize(nI,[2048 2048],'nearest');
subplot(1,3,1);
imshow(b1);
title('sppyzjinc');
b2=imresize(nI,[2048 2048],'bilinear');
subplot(1,3,2);
imshow(b2);
title('sppysxxnc');
b3=imresize(nI,[2048 2048],'bicubic');
subplot(1,3,3);
imshow(b3);
title('sppysscnc');

```

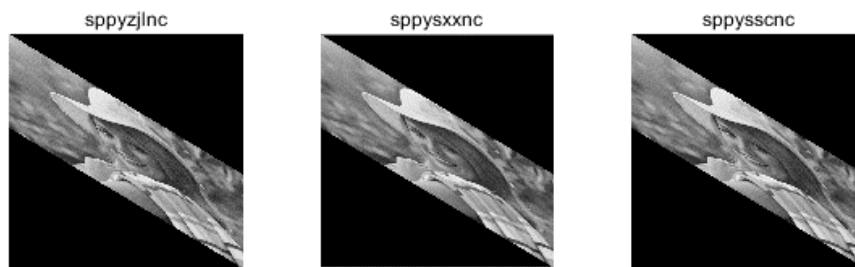
下面是实验结果:



Elain 的水平源代码:

```
transformtype='affine';
transformmatrix=[1 1.5 0;0 1 0;0 0 1];
T=maketform(transformtype,transformmatrix);
I=imread('elain1.bmp');
nI=imtransform(I,T);
figure(5);
imshow(I);
title('ystx');
b1=imresize(nI,[2048 2048],'nearest');
subplot(1,3,1);
imshow(b1);
title('sppyzjlnc');
b2=imresize(nI,[2048 2048],'bilinear');
subplot(1,3,2);
imshow(b2);
title('sppysxxnc');
b3=imresize(nI,[2048 2048],'bicubic');
subplot(1,3,3);
imshow(b3);
    title('sppysscnc');
```

下面是实验结果:



Lena 的旋转源代码:

```
I=imread('lena.bmp');
figure(1);
imshow(I);
title('ystx');
I1=imrotate(I,30,'nearest');
b1=imresize(I1,[2048 2048],'nearest');
subplot(1,3,1);
imshow(b1);
title(' 30 zjnc');
I2=imrotate(I,30,'bilinear');
b2=imresize(I2,[2048 2048],'bilinear');
subplot(1,3,2);
imshow(b2);
title(' 30 sxxnc');
I3=imrotate(I,30,'bicubic');
b3=imresize(I3,[2048 2048],'bicubic');
subplot(1,3,3);
imshow(b3);
title(' 30 sscnc');
```

以下是实验结果：

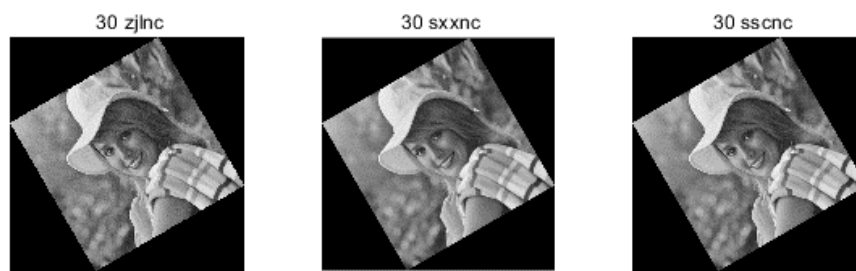


Elain 的旋转源代码：

```
L=imread('elain1.bmp');
figure(5);
imshow(L);
title('ystx');
L1=imrotate(L,30,'nearest');
c1=imresize(L1,[2048 2048],'nearest');
subplot(1,3,1);
imshow(c1);
title(' 30 zjnc');
L2=imrotate(L,30,'bilinear');
c2=imresize(L2,[2048 2048],'bilinear');
subplot(1,3,2);
imshow(c2);
title(' 30 sxxnc');
L3=imrotate(L,30,'bicubic');
c3=imresize(L3,[2048 2048],'bicubic');
subplot(1,3,3);
imshow(c3);
```

```
title(' 30 sscnc');
```

以下是实验结果：



附录

【参考文献】

[1] 冈萨雷斯. 数字图像处理（第三版）北京：电子工业出版社，2011

源代码见单独 txt 文件