



南京理工大学
NANJING UNIVERSITY OF SCIENCE & TECHNOLOGY

数字图像处理 实验报告

课程	:	数字图像处理	
教师	:	张玉珍	
组号	:	第9组	
组员	:	许婷	学号 : 9161040G0508
组员	:	孙宏寰	学号 : 9161040G0707
组员	:	董建博	学号 : 9161040G0718
组员	:	许晓明	学号 : 9161040G0734
组员	:	周茂源	学号 : 9161040G0809

2019 年 4月

目录

1	数字图像处理线性增强	1
1.1	实验目的	1
1.2	实验内容	1
1.3	实验过程	1
1.4	实验代码	1
1.5	实验结果	2
1.6	实验分析	2
2	数字图像处理中值滤波	3
2.1	实验目的	3
2.2	实验内容	3
2.3	实验原理	3
2.4	实验步骤	3
2.5	实验代码及解释	4
2.6	实验结果	6
2.7	实验分析	6
3	数字图像直方图均衡化处理	8
3.1	实验目的	8
3.2	实验内容	8
3.3	实验原理	8
3.4	实验步骤	8
3.5	实验代码	8
3.6	实验结果	10
3.7	实验分析	10
4	数字图像处理空间平滑	13
4.1	实验目的	13
4.2	实验内容	13
4.3	实验步骤	13
4.4	实验原理	13

4.5 实验程序	13
4.6 实验结果	14
分工情况	16

实验1 数字图像处理线性增强

1.1 实验目的

编写线性增强的C或C++或MATLAB语言程序及相应显示程序。

1.2 实验内容

利用MATLAB实现图像线性增强。

1.3 实验过程

1. 首先通过imread读取一张彩色图片。
2. 将彩色图像进行线性变换
3. 通过matlab的内置函数将彩色图转换成灰度图。
4. 判断图像X的大小，再通过for循环遍历图片中的每一个像素，通过函数进行线性变换

1.4 实验代码

```
1  clc;
2  R=imread('蛋糕.jpg');
3  H=2+1.35*(R-2);%线性变换
4  X=rgb2gray(R);%将真彩色RGB图像转换成灰度图像
5  [a,b]=size(X);
6  for i=1:a-1
7      for j=1:b-1
8          I(i,j)=2+1.35*(X(i,j)-2);
9      end
10 End
11 subplot(231);imshow(R);title('原始图像');
12 subplot(232);imshow(uint8(H));title('增强图像');
13 subplot(233);imshow(X);title('灰度图像');
14 subplot(234);imhist(X,256);title('原图直方图');
15 subplot(235);imhist(I,256);title('增强直方图');
16 hold on;
17 x=[0:5];
18 y=2+1.35*(x-2);
19 subplot(236);plot(x,y);title('函数图像');
```

1.5 实验结果

实验结果见图1.1。

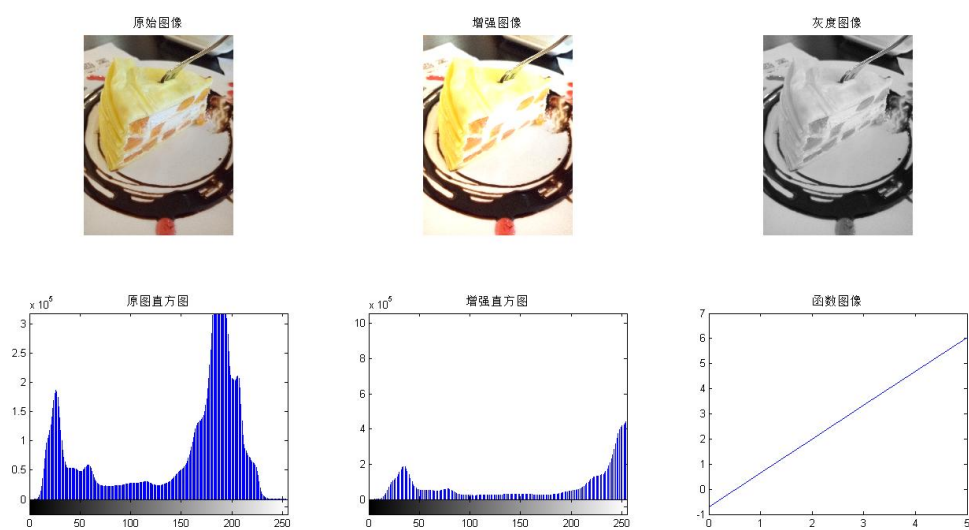


图 1.1

1.6 实验分析

该实验属于图像增强的线性变换部分。通过变换使图像的动态范围增大，图像对比度扩展，图像变清晰，特征明显。通过直方图看出，图像在变换前灰度过于集中在一个部分，变换后对图像的每一像素进行了拉伸，灰度值分布均匀，有效的改善了图像视觉效果。

实验2 数字图像处理中值滤波

2.1 实验目的

掌握中值滤波的基本原理, 熟悉中值滤波的方法。

2.2 实验内容

编写中值滤波及图像平滑化的matlab语言程序以及相应的显示程序和原图比较, 分析四点平滑滤波图像平滑化后的效果。

2.3 实验原理

中值滤波是基于排序统计理论的一种能有效抑制噪声的非线性信号处理技术, 中值滤波的基本原理是把数字图像或数字序列中一点的值用该点的一个邻域中各点值的中值代替, 让周围的像素值接近的真实值, 从而消除孤立的噪声点。方法是用某种结构的二维滑动模板, 将板内像素按照像素值的大小进行排序, 生成单调上升(或下降)的为二维数据序列。二维中值滤波输出为

$$g(x, y) = \text{med}f(x - k, y - l), (k, l \in W)$$

其中, $f(x, y)$, $g(x, y)$ 分别为原始图像和处理后图像。W为二维模板, 通常为 3×3 , 5×5 区域, 也可以是不同的形状, 如线状, 圆形, 十字形, 圆环形等。

具体流程如下:

1. 通过从图像中的某个采样窗口取出奇数个数据进行排序
2. 用排序后的中值取代要处理的数据即可

2.4 实验步骤

1. 新建 matlab 工程;
2. 分析程序流程;
3. 编写程序;
4. 运行调试;
5. 显示结果;
6. 分析结果。

2.5 实验代码及解释

中值滤波程序

```
1 A=imread('beijing.jpg'); %读取北京图片
2 J=rgb2gray(A); %将RGB彩色图像转化为灰度图，减少运算量，提高运算速度
3 subplot(1,2,1);
4 P1=imnoise(J,'salt&pepper',0.03) %对图像生成椒盐噪声
5 imshow(P1)
6 title('原图像')
7 [m,n]=size(J); %取得原图矩阵尺寸
8 B=zeros(9); %生成9*9的全零方阵
9 for i=2:m-1 %从第二个开始到倒数第二个结束
10 for j=2:n-1
11 b=1;
12 for p=(i-1):(i+1) %取小框3行
13 for q=(j-1):(j+1) %取小框3列
14 B(b)=J(p,q); %将选中3*3的值赋给B
15 b=b+1;
16 end
17 end
18 for d=1:8 %开始冒泡程序
19 if(B(d)>B(d+1)) %由小到大排序，大则交换，否则不交换
20 t=B(d);
21 B(d)=B(d+1);
22 B(d+1)=t;
23 end
24 end
25 J(i,j)=B(5,1); %将排序后的像素值赋给J
26 end
27 end
28 subplot(1,2,2);
29 imshow(J)
30 title('中值滤波后')
```

为对比中值滤波对椒盐噪声和高斯噪声的作用效果，改进程序如下

```
1 clc
2 A=imread('beijing.jpg');
3 J=rgb2gray(A);
4 E=J;
5 subplot(3,2,5);
6 imshow(E);
7 title('原图像')
```

```

8 P1=imnoise(J,'gaussian',0.2);%对图像生成高斯噪声
9 subplot(3,2,2);
10 imshow(P1)
11 title('高斯原图像 ')
12 [m,n]=size(J) ;
13 B=zeros(9);
14 for i=2:m-1
15     for j=2:n-1
16         b=1;
17         for p=(i-1):(i+1)
18             for q=(j-1):(j+1)
19                 B(b)=J(p,q);
20                 b=b+1;
21             end
22         end
23         d=1:8
24         if(B(d)>B(d+1))
25             t=B(d);
26             B(d)=B(d+1);
27             B(d+1)=t;
28         end
29     end
30     J(i,j)=B(5,1);
31 end
32 end
33 subplot(3,2,4);
34 imshow(J)
35 title('高斯噪声中值滤波后 ')
36 C=rgb2gray(A);
37 subplot(3,2,1);
38 P2=imnoise(C,'salt&pepper',0.2);%对图像生成椒盐噪声
39 imshow(P2)
40 title('椒盐原图像')
41 [x,y]=size(C) ;
42 D=zeros(9);
43 for i=2:x-1
44     for j=2:y-1
45         b=1;
46         for p=(i-1):(i+1)
47             for q=(j-1):(j+1)
48                 D(b)=C(p,q);
49                 b=b+1;
50             end
51         end
52     for d=1:8

```



```

53 if (D(d)>D(d+1))
54     t=D(d);
55     D(d)=D(d+1);
56     D(d+1)=t;
57 end
58 end
59 C(i,j)=D(5,1);
60 end
61 end
62 subplot(3,2,3);
63 imshow(C)
64 title('椒盐噪声中值滤波后')

```

2.6 实验结果

实验结果见图2.1~图2.2。



图 2.1

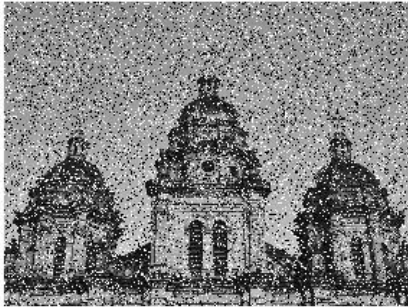
2.7 实验分析

因为中值滤波的原理是取合理的邻近像素值来代替噪声点，所以只适用于椒盐噪声的去除。

但适当选取模板的大小（或结构形状）也很重要，将上述程序的 3×3 模板改变扩大为 7×7 或更大，图像的清晰度就遭到了破坏。

理论上来说，中值滤波对于椒盐噪声的效果应该好于高斯噪声。

椒盐原图像



高斯原图像



椒盐中值滤波后



高斯中值滤波后



图 2.2

实验3 数字图像直方图均衡化处理

3.1 实验目的

通过了解直方图的绘制、均衡化的基本原理与实现方法，熟悉在计算机上进行直方图均衡化、线性变换的方法，来掌握简单的直方图均衡化和线性变换程序设计编写。

3.2 实验内容

编写直方图的显示、均衡化的matlab语言程序，同时显示变换后的直方图。

3.3 实验原理

直方图均衡化处理的中心思想是把原始图像的灰度直方图从比较集中的某个灰度区间变成在全部灰度范围内的均匀分布。通过对图像进行非线性拉伸，重新分配图像像素值，使一定灰度范围内的像素数量大致相同，把给定图像的直方图分布改变成“均匀”分布直方图分布。

利用累积分布函数，完成原图像直方图均匀分布的效果。

3.4 实验步骤

1. 新建工程；
2. 分析程序流程；
3. 编写程序；
4. 运行调试；
5. 显示结果；
6. 分析结果。

3.5 实验代码

```
1 close all;clear all;clc;
2 A= imread('E:\baogao数字图像处理\picture\IMG_20190314_194136.jpg');
3 original=rgb2gray(A);%转灰度
4 [h w] = size(original); %获取待处理图片尺寸
5 p=zeros(6,256);
6 f=zeros(1,256);%统计每个像素值出现次数
7 for i=1:1:h
8 for j=1:1:w
```

```

9  for k=0:1:255
10 if original(i,j)==k
11 p(1,k+1)=p(1,k+1)+1;
12 end
13 end
14 end
15 end%计算概率直方图
16 for i=1:1:256
17 p(2,i)=p(1,i)/(h*w);
18 end %求累计概率，得到累计直方图
19 p(3,1)=p(2,1);
20 for i = 2 : 256
21 p(3, i) = p(3, i - 1) + p(2, i);
22 end%用p3数组实现灰度值[0, 255]的映射
23 for i = 1:1:256
24 f(i) = round(p(3, i) * 255);
25 end %直方图均衡化
26 equalize=uint8(zeros(h,w));
27 for i=1:1:h
28 for j=1:1:w
29 for k=0:1:255
30 if original(i,j)==k
31 equalize(i,j)=f(k+1) ;
32 end
33 end
34 end
35 end%统计每个像素值出现次数
36 for i=1:1:h
37 for j=1:1:w
38 for k=0:1:255
39 if equalize(i,j)==k
40 p(4,k+1)=p(4,k+1)+1;
41 end
42 end
43 end
44 end%计算新的概率直方图
45 for i=1:1:256
46 p(5,i)=p(4,i)/(h*w);
47 end%求累计概率，得到新的累计直方图
48 p(6,1)=p(5,1);
49 for i = 2 : 256
50 p(6, i) = p(6, i - 1) + p(5, i);
51 end
52 subplot(231)
53 imshow(original)

```

```

54 title('原图');
55 subplot(232)
56 bar(p(2,:))
57 xlim([0,256]);
58 set(gca,'XGrid','on');
59 set(gca,'YGrid','on');
60 title('原概率直方图');
61 subplot(233)
62 bar(p(3,:))
63 xlim([0,256]);
64 set(gca,'XGrid','on');
65 set(gca,'YGrid','on');
66 title('原累计直方图');
67 subplot(234)
68 imshow(equalize)
69 title('直方图均衡化');
70 subplot(235)
71 bar(p(5,:))
72 xlim([0,256]);
73 set(gca,'XGrid','on');
74 set(gca,'YGrid','on');
75 title('新的概率直方图');
76 subplot(236)
77 bar(p(6,:))
78 xlim([0,256]);
79 set(gca,'XGrid','on');
80 set(gca,'YGrid','on');
81 title('新的累计直方图');

```

3.6 实验结果

实验结果见图3.1~ 图3.4。

3.7 实验分析

如图3.1 所示，直方图均衡化对于像素灰度集中在低值区（图片偏灰暗）的图片改善效果较强，但对于如图3.2 的灰度值集中在高值区（图像过曝）的图片，则改善效果不佳。而对于如图3.3,3.4 的灰度值过度集中在某一区域的图片，由于直方图均衡化变换后图像的灰度级减少，某些细节消失。尤其是如图3.3，直方图有高峰，经处理后对比度不自然的过分增强，产生马赫带效应。

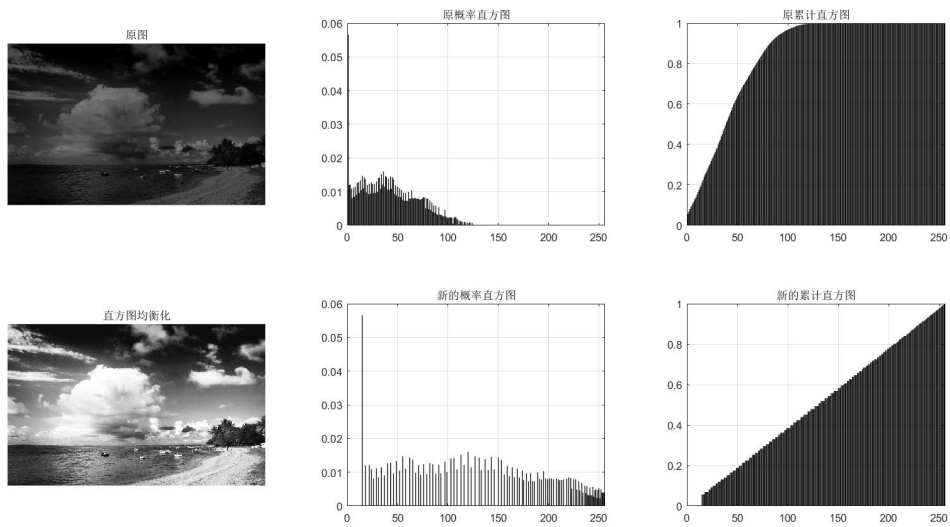


图 3.1

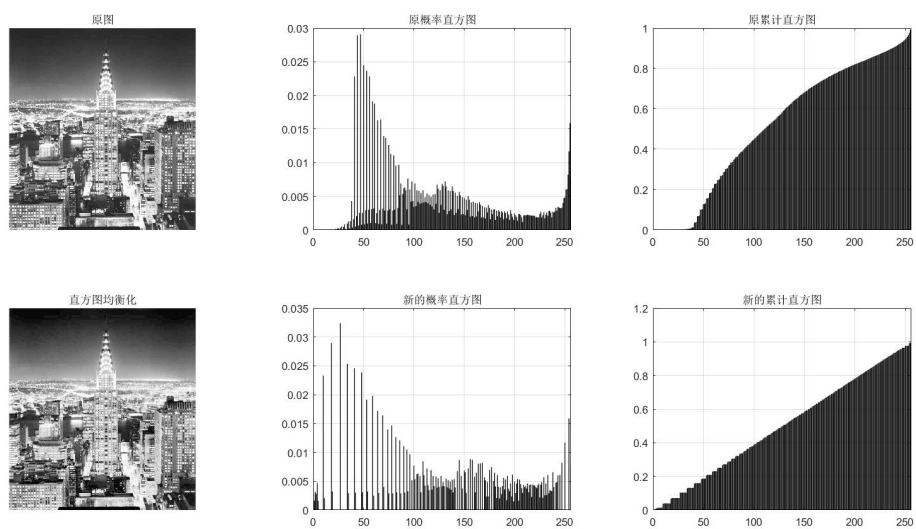


图 3.2

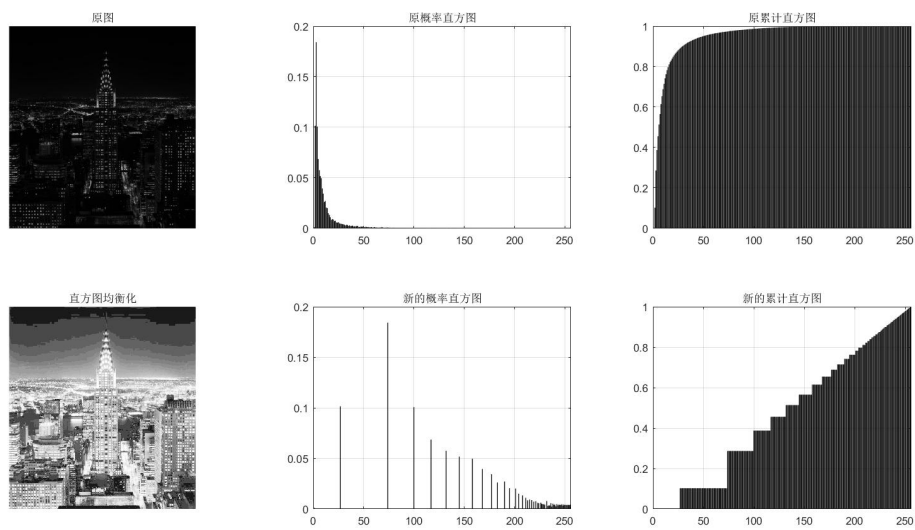


图 3.3

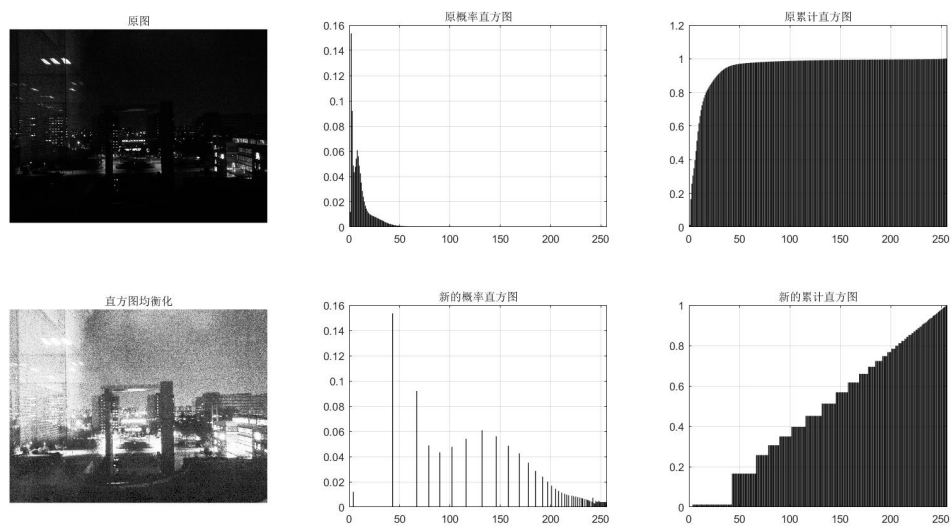


图 3.4

实验4 数字图像处理空间平滑

4.1 实验目的

掌握图像平滑基本原理，掌握4点平滑，8点平滑，中值滤波平滑基本原理，掌握计算机软件处理图像的方法。

4.2 实验内容

编写4点平滑，8点平滑，中值滤波的C或C++或matlab语言程序以及相应的显示程序。

4.3 实验步骤

1. 原理过程分析；
2. 程序流程分析；
3. 编写程序；
4. 运行调试；
5. 结果展示；

4.4 实验原理

用像素值邻域（4联通或8联通）的平均值代替该店像素值，实现平滑效果，能有效的去除噪声。

数字图像处理中简化为下列公式：

$$B(i, j) = A(i, j - 1)/5 + A(i, j)/5 + A(i, j + 1)/5 + A(i - 1, j)/5 + A(i + 1, j)/5$$

$$B(i, j) = A(i, j - 1)/9 + A(i, j)/9 + A(i, j + 1)/9 + A(i - 1, j)/9 + A(i + 1, j)/9 + A(i - 1, j - 1)/9 + A(i + 1, j - 1)/9 + A(i - 1, j + 1)/9 + A(i + 1, j + 1)/9$$

4.5 实验程序

4点平滑实验代码

```
1 A=imread('C:\Users\loveless\Documents\MATLAB\reflection.jpg');% 读入图像
2 I=imnoise(A, 'salt & pepper', 0.025);% 加入椒盐噪声
3 subplot(1,2,1);imshow(I);title('原图像');% 显示原图像
```



```

4  [a,b]=size(I); % 读取图像尺寸大小
5  J=double(I);
6
7  for i=2:a-1% 处理图像矩阵
8  for j=2:b-1
9  J(i,j)=(J(i-1,j)+J(i,j+1)+J(i+1,j)+J(i,j-1))/4; %4点均值
10 end
11 end
12 J=uint8(J);
13 subplot(1,2,2);imshow(J);title('点均值平滑后的图像4');%显示处理后的图像

```

8点平滑实验代码

```

1  A=imread('C:\Users\loveless\Documents\MATLAB\reflection.jpg');%读入图像
2  I=imnoise(A,'salt & pepper',0.025);%加入椒盐噪声
3  [a,b]=size(I); %读取图像尺寸大小
4  J=double(I);
5  for i=2:a-1
6  for j=2:b-1
7  J(i,j)=(J(i-1,j-1)+J(i-1,j)+J(i-1,j+1)+J(i,j-1)+...
8  J(i,j+1)+J(i+1,j-1)+J(i+1,j)+J(i+1,j+1))/8;%8点均值
9  end
10 end
11 J=uint8(J);
12 subplot(1,2,1);imshow(I);title('原图像');%显示加噪声后的图像
13 subplot(1,2,2);imshow(J);title('点均值平滑后的图像8');%显示处理后的图像

```

4.6 实验结果

4点平滑显示结果见图4.1，8点平滑显示结果见图4.2。

原图像



4点均值平滑后的图像

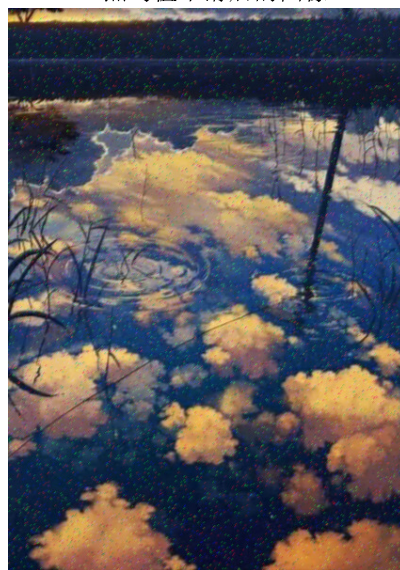
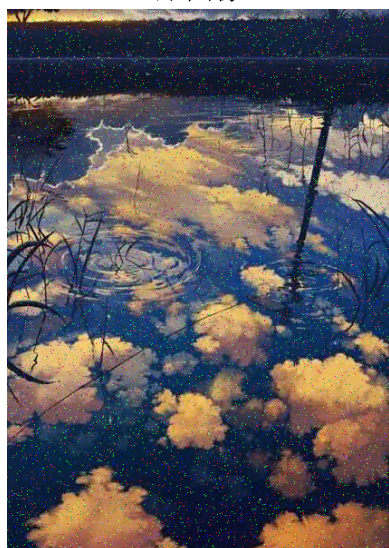


图 4.1

原图像



8点均值平滑后的图像

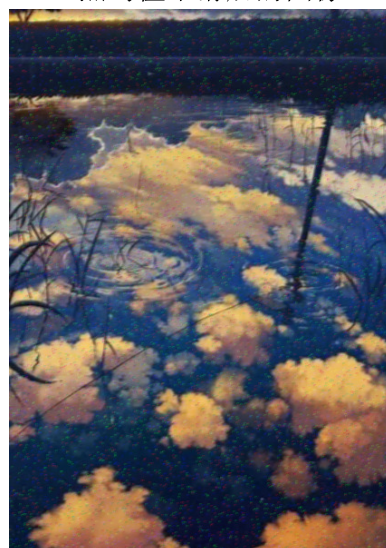


图 4.2

分工情况

表 4.1 第9组实验分工表

组员	分工
许婷	数字图像处理线性增强
孙宏寰	数字图像处理空间平滑
董建博	报告撰写
许晓明	数字图像直方图均衡化处理
周茂源	数字图像处理中值滤波