

中国科学技术大学计算机学院

《数字图像处理与分析》实验报告

2021.06.15



实验题目： 图像处理实验

学生姓名： 胡毅翔

学生学号： PB18000290

计算机实验教学中心制
2019 年 9 月

目录

| | |
|-----------------|----|
| 1 实验目的 | 3 |
| 2 实验环境 | 3 |
| 3 实验内容 | 3 |
| 4 实验一 图像几何变换 | 3 |
| 4.1 图像的平移 | 3 |
| 4.1.1 实验原理 | 3 |
| 4.1.2 实验内容 | 4 |
| 4.2 图像的旋转 | 4 |
| 4.2.1 实验原理 | 4 |
| 4.2.2 实验内容 | 4 |
| 4.3 图像的缩放 | 5 |
| 4.3.1 实验原理 | 5 |
| 4.3.2 实验内容 | 5 |
| 4.4 图像几何失真校正 | 6 |
| 4.4.1 实验原理 | 6 |
| 4.4.2 实验内容 | 8 |
| 5 实验二 图像点处理增强 | 9 |
| 5.1 灰度的线性变换 | 9 |
| 5.1.1 实验原理 | 9 |
| 5.1.2 实验内容 | 9 |
| 5.2 灰度拉伸 | 10 |
| 5.2.1 实验原理 | 10 |
| 5.2.2 实验内容 | 11 |
| 5.3 灰度直方图 | 12 |
| 5.3.1 实验原理 | 12 |
| 5.3.2 实验内容 | 12 |
| 5.4 直方图均衡 | 13 |
| 5.4.1 实验原理 | 13 |
| 5.4.2 实验内容 | 13 |
| 6 实验三 图像空间域滤波增强 | 14 |
| 6.1 邻域平均法 | 14 |

| | |
|--------------------------|-----------|
| 6.1.1 实验原理 | 14 |
| 6.1.2 实验内容 | 14 |
| 6.2 超限邻域平均法 | 15 |
| 6.2.1 实验原理 | 15 |
| 6.2.2 实验内容 | 15 |
| 6.3 中值滤波器 | 15 |
| 6.3.1 实验原理 | 15 |
| 6.3.2 实验内容 | 15 |
| 6.4 超限中值滤波器 | 16 |
| 6.4.1 实验原理 | 16 |
| 6.4.2 实验内容 | 16 |
| 6.5 分析比较 | 16 |
| 6.5.1 实验内容 | 16 |
| 6.6 边缘检测 | 17 |
| 6.6.1 实验原理 | 17 |
| 6.6.2 实验内容 | 18 |
| 7 实验四 图像变换及频域滤波增强 | 20 |
| 7.1 Fourier 变换 | 20 |
| 7.1.1 实验内容 | 20 |
| 7.2 低通滤波器 | 21 |
| 7.2.1 实验内容 | 21 |
| 7.3 低通滤波器去噪 | 23 |
| 7.3.1 实验内容 | 23 |
| 7.4 高通滤波器 | 24 |
| 7.5 低通滤波器去噪 | 25 |
| 7.5.1 高频增强滤波 | 25 |
| 8 实验五 图像恢复与图像分割 | 26 |
| 8.1 逆滤波与维纳滤波 | 26 |
| 8.1.1 实验内容 | 26 |
| 8.2 大津法 | 27 |
| 8.2.1 实验原理 | 27 |
| 8.2.2 实验内容 | 28 |
| 8.3 区域分割 | 28 |
| 8.3.1 实验内容 | 28 |

1 实验目的

本实验的目的是通过实验进一步理解和掌握数字图像处理与分析的原理和方法。通过分析、实现现有的图像处理算法，学习和掌握常用的图像处理与分析技术。

2 实验环境

1. PC 一台
2. Windows 10 操作系统
3. Matlab 9.5.0.1033004 (R2018b) Update 2
4. Visual Studio Code 1.56.2

3 实验内容

数字图像处理的实验内容主要包括五个方面：

1. 图像几何变换。
2. 对图像进行空间域滤波，提高图像视觉质量，以便于人眼观察、理解或用计算机对其进一步处理。
3. 对图像作频域变换，进行频率域滤波增强处理。
4. 在空间域和频域提取、描述和分析图像中所包含的特征，便于计算机对图像作进一步的分析和理解，经常作为模式识别和计算机视觉的预处理。这些特征包括很多方面，如图像的频域特性、边界特征等。
5. 了解常见的图像退化模型和相应的图像恢复算法，从本质上改善图像质量；对图像进行分析，采用阈值法、区域分裂合并法等分割算法，获取图像中感兴趣目标区域。

4 实验一 图像几何变换

4.1 图像的平移

4.1.1 实验原理

图像平移就是将图像中所有的点都按照指定的平移量水平、垂直移动。如：设 (x_0, y_0) 为原图像的一点，图像水平平移量为 tx ，垂直平移量为 ty ，则平移后坐标变为 (x_1, y_1) ，显然， (x_0, y_0) 和 (x_1, y_1) 有如下关系：

$$\begin{cases} x_1 = x_0 + tx \\ y_1 = y_0 + ty \end{cases}$$

用矩阵表示如下:

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & tx \\ 0 & 1 & ty \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix}$$

4.1.2 实验内容

输入一幅图像，根据输入的水平和垂直平移量，显示平移后的图像。

输入水平偏移量 100，垂直偏移量 100，得到的结果为：

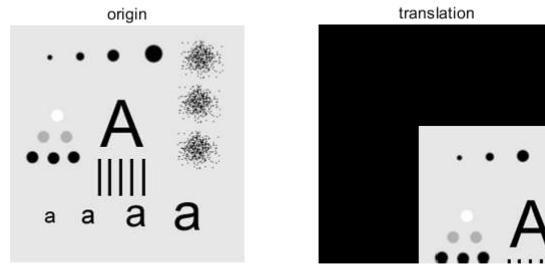


图 1: 图像的平移

4.2 图像的旋转

4.2.1 实验原理

图像绕中心点（原点）旋转的公式如下:

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix}$$

图像如果绕一个指定点 (a, b) 旋转，则先要将坐标系平移到该点，再进行旋转，然后平移回新的坐标原点。则旋转变换表达式为:

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & a \\ 0 & 1 & b \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -a \\ 0 & 1 & -b \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix}$$

4.2.2 实验内容

输入一幅图像，根据输入的旋转角度参数，绕图像中心点旋转，分别用最近邻插值和双线性插值显示旋转后的图像。

输入旋转角度参数 60, 得到的结果为:

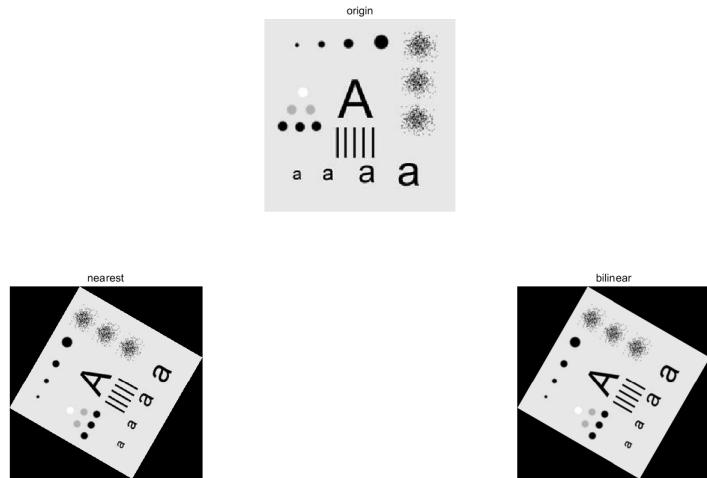


图 2: 图像的旋转

4.3 图像的缩放

4.3.1 实验原理

假设图像 x 轴方向缩放比率为 c, y 轴方向缩放比率为 d, 那么原图中, 点 (x_0, y_0) 对应于新图中的点 (x_1, y_1) 的转换矩阵为:

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} c & 0 & 0 \\ 0 & d & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix}$$

4.3.2 实验内容

输入一幅图像, 根据输入的水平和垂直缩放量, 分别用最近邻插值和双线性插值, 显示缩放后的图像。

水平方向缩放为原来的 $1/3$, 垂直方向缩放为原来的 $1/2$, 得到的结果为:

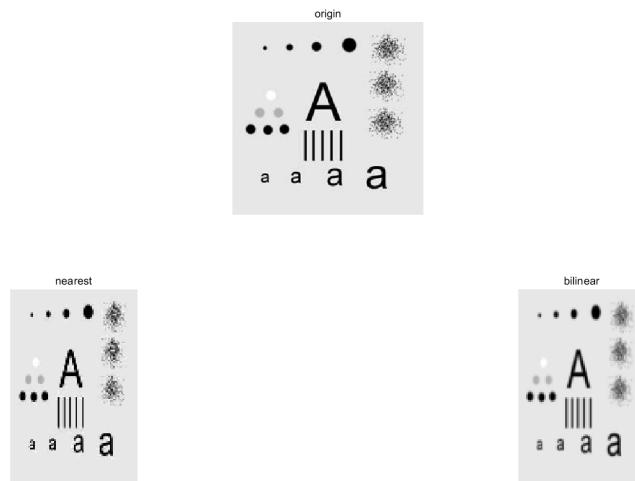


图 3: 图像的缩放

4.4 图像几何失真校正

4.4.1 实验原理

设原图像: $f(x, y)$ 失真图像: $g(x', y')$

$$x' = s(x, y)$$

$$y' = t(x, y)$$

通过选取控制点对, 求出 (x', y') 与 (x, y) 坐标之间的关系

1. 线性失真:

$$x' = a_1x + a_2y + a_3$$

$$y' = b_1x + b_2y + b_3$$

6 个系数要 3 个控制点对为减少误差, 选取 $n > 3$ 个控制点对

$$\begin{bmatrix} x'_1 \\ x'_2 \\ \vdots \\ x'_n \end{bmatrix} = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

$$M = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \ddots & \ddots \\ \vdots & \ddots & \ddots \\ x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = (M^T M)^{-1} M^T \begin{bmatrix} x'_1 \\ x'_2 \\ \vdots \\ x'_n \end{bmatrix} = (M^T M)^{-1} M^T \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix} = (M^T M)^{-1} M^T \begin{bmatrix} y'_1 \\ y'_2 \\ \vdots \\ y'_n \end{bmatrix}$$

2. 双线性失真：

$$\begin{cases} x' = a_1xy + a_2x + a_3y + a_4 \\ y' = b_1xy + b_2x + b_3y + b_4 \end{cases}$$

8个系数要4个控制点对为减少误差,选取n>4个控制点对

$$M = \begin{bmatrix} x_1y_1 & x_1 & y_1 & 1 \\ x_2y_2 & x_2 & y_2 & 1 \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ x_ny_n & x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = (M^T M)^{-1} M^T \begin{bmatrix} x'_1 \\ x'_2 \\ \cdot \\ \cdot \\ \cdot \\ \vdots \\ \cdot_n \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} = (M^T M)^{-1} M^T \begin{bmatrix} y'_1 \\ y'_2 \\ \cdot \\ \cdot \\ \cdot \\ \vdots \\ y'_n \end{bmatrix}$$

3. 二次校正式：

$$x' = a_1x^2 + a_2y^2 + a_3xy + a_4x + a_5y + a_6$$

$$y' = b_1x^2 + b_2y^2 + b_3xy + b_4x + b_5y + b_6$$

12 个系数要 6 个控制点对为减少误差, 选取 $n > 6$ 个控制点对

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{bmatrix} = (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \begin{bmatrix} x'_1 \\ x'_2 \\ \vdots \\ \vdots \\ \dot{x}'_n \end{bmatrix}$$

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{bmatrix} = (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \begin{bmatrix} y'_1 \\ y'_2 \\ \vdots \\ y'_n \end{bmatrix}$$

$$\mathbf{M} = \begin{bmatrix} x_1^2 & y_1^2 & x_1y_1 & x_1 & y_1 & 1 \\ x_2^2 & y_2^2 & x_2y_2 & x_2 & y_2 & 1 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots \\ x_n^2 & y_n^2 & x_ny_n & x_n & y_n & 1 \end{bmatrix}$$

4.4.2 实验内容

输入图像 alphabet1.jpg 及几何失真图像 alphabet2.jpg, 设置控制点进行几何失真校正, 显示校正后的图像。

选取 4 个控制点:

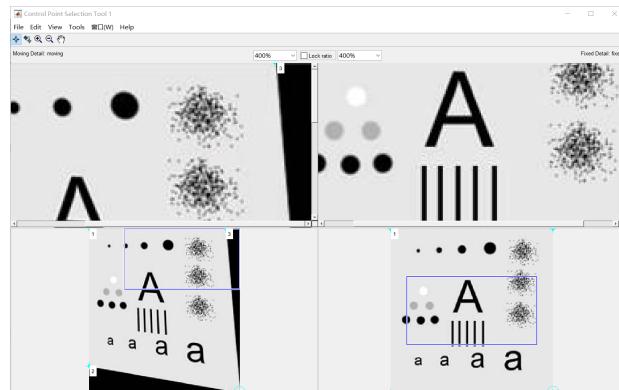


图 4: 控制点的选取

得到的校正结果:

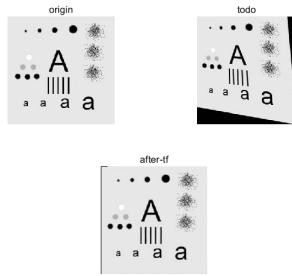


图 5: 图像几何失真校正

5 实验二 图像点处理增强

5.1 灰度的线性变换

5.1.1 实验原理

灰度的线性变换就是将图像中所有的点的灰度按照线性灰度变换函数进行变换。该线性灰度变换函数是一个一维线性函数:

$$f(x) = f_A \cdot x + f_B$$

灰度变换方程为:

$$D_B = f(D_A) = f_A \cdot D_A + f_B$$

其中参数 f_A 为线性函数的斜率, f_B 为线性函数的在 y 轴的截距, D_A 表示输入图像的灰度, D_B 表示输出图像的灰度。

5.1.2 实验内容

输入一幅图像, 根据输入的斜率和截距进行线性变换, 并显示。

关键函数为:

```

1 function [new] = LinearTransformFunc(original, k, d)
2     new = original;
3     [a, b] = size(original);
4
5     for i = 1:a
6
7         for j = 1:b
8             tmp = original(i, j) * k + d;
9

```

```

10         if tmp > 255
11             tmp = 255;
12         elseif tmp < 0
13             tmp = 0;
14         end
15
16     new(i, j) = tmp;
17 end
18
19 end
20
21 end

```

输入的斜率为 3，截距为 -44，结果为：

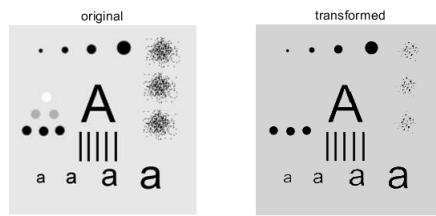


图 6: 灰度的线性变换

5.2 灰度拉伸

5.2.1 实验原理

灰度拉伸和灰度线性变换相似。不同之处在于它是分段线性变换。表达式如下：

$$f(x) = \frac{y_1}{x_1}x; \quad x < x_1$$

$$f(x) = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1) + y_1; \quad x_1 \leq x \leq x_2$$

$$f(x) = \frac{255 - y_2}{255 - x_2}(x - x_2) + y_2; \quad x > x_2$$

其中， (x_1, y_1) 和 (x_2, y_2) 是分段函数的转折点。

5.2.2 实验内容

输入一幅图像，根据选择的转折点，进行灰度拉伸，显示变换后的图像。

关键函数为：

```
1 function [new] = StretchFunc(original, x1, y1, x2, y2)
2     new = original;
3
4     w = size(new, 1);
5     h = size(new, 2);
6
7     k1 = y1 / x1;
8
9     dk1 = (y2 - y1) / (x2 - x1);
10    dk2 = (255 - y2) / (255 - x2);
11
12    for i = 1:w
13
14        for j = 1:h
15            x = new(i, j);
16
17            if x < x1
18                new(i, j) = k1 * x;
19            elseif x < x2
20                new(i, j) = dk1 * (x - x1) + y1;
21            else
22                new(i, j) = dk2 * (x - x2) + y2;
23            end
24
25            if new(i, j) > 255
26                new(i, j) = 255;
27            elseif new(i, j) < 0
28                new(i, j) = 0;
29            end
30
31        end
32    end
```

```
33     end  
34  
35 end
```

参数 x_1, y_1, x_2, y_2 分别为 30, 10, 200, 250, 得到的结果为:



图 7: 灰度拉伸

5.3 灰度直方图

5.3.1 实验原理

灰度直方图是灰度值的函数，描述的是图像中具有该灰度值的像素的个数，其横坐标表示像素的灰度级别，纵坐标表示该灰度出现的频率 (像素的个数)。

5.3.2 实验内容

输入一幅图像，显示它的灰度直方图，可以根据输入的参数（上限、下限）显示特定范围的灰度直方图。

参数上限、下限分别为 100, 255，得到的结果为:

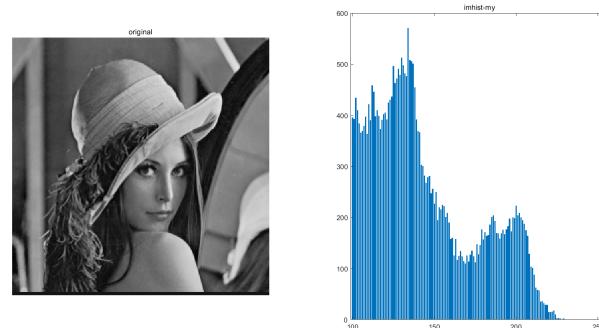


图 8: 灰度直方图

5.4 直方图均衡

5.4.1 实验原理

1. 统计图像中各灰度级像素个数 n_k ;
2. 计算直方图: $p_r(r_k) = \frac{n_k}{n}$;
3. 计算累计直方图: $s_k = T(r_k) = \sum_{j=0}^k p_r(r_j) = \sum_{j=0}^k \frac{n_j}{n}$;
4. 取整 $S_k = \text{int}[(L - 1)s_k + 0.5]$;
5. 确定映射对应关系: $k \rightarrow S_k$;
6. 对图像进行增强变换 ($k \rightarrow S_k$)。

其中 L 是灰度等级, n 是图像总像素数。

5.4.2 实验内容

1. 显示一幅图像 pout.bmp 的直方图;
2. 用直方图均衡对图像 pout.bmp 进行增强;
3. 显示增强后的图像及其直方图;
4. 用原始图像 pout.bmp 进行直方图规定化处理, 将直方图规定化为高斯分布;
5. 显示规定化后的图像及其直方图。

得到的结果为:

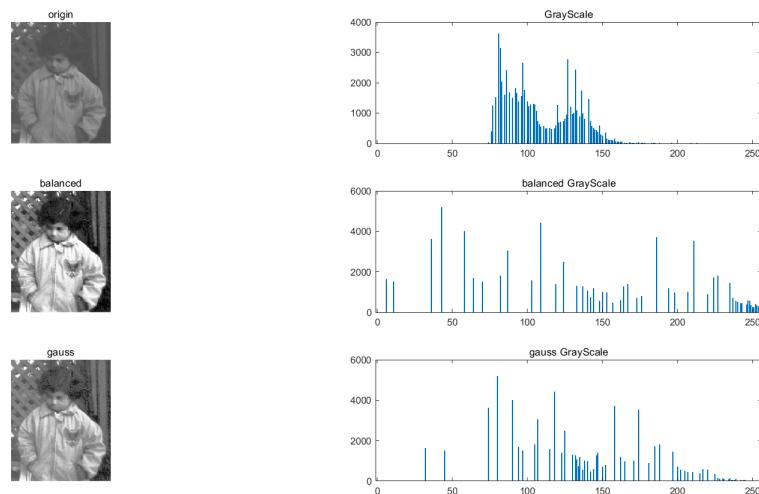


图 9: 直方图均衡

6 实验三 图像空间域滤波增强

6.1 邻域平均法

6.1.1 实验原理

用均值滤波器去除图像中的噪声（选 3×3 窗口）

$$f(x_0, y_0) = \frac{1}{N \times N} \sum f(x, y)$$

6.1.2 实验内容

用原始图像 lena.bmp 或 cameraman.bmp 分别加产生的 3% 椒盐噪声、高斯噪声、随机噪声合成有噪声的图像并显示以及邻域平均法去噪后的结果。

得到的结果为：



图 10: 有噪声的图像

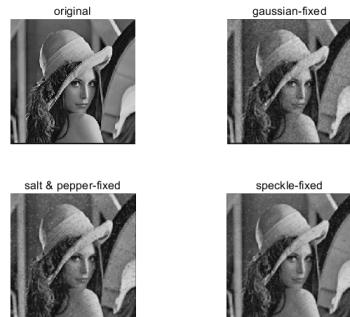


图 11: 邻域平均法

6.2 超限邻域平均法

6.2.1 实验原理

用超限邻域平均法去除图像中的噪声：如果某个像素的灰度值大于其邻域像素的平均值，且达到了一定水平，则判断该像素为噪声，继而用邻域像素的均值取代这一像素值，

$$g(i, j) = \begin{cases} \frac{1}{N \times N} \sum_{(x,y) \in A} f(x, y), & \left| f(i, j) - \frac{1}{N \times N} \sum_{(x,y) \in A} f(x, y) \right| > T \\ f(i, j), & \text{其它} \end{cases}$$

T 为某一阈值。

6.2.2 实验内容

用原始图像 lena.bmp 或 cameraman.bmp 分别加产生的 3% 椒盐噪声、高斯噪声、随机噪声合成有噪声的图像并显示以及超限邻域平均法去噪后的结果。

设置阈值为 50，得到的结果为：



图 12: 超限邻域平均法

6.3 中值滤波器

6.3.1 实验原理

用中值滤波器去除图像中的噪声（选 3x3 窗口做中值滤波）；

$$f(x_0, y_0) = \text{Med} \{f(x, y) \mid x \in [x_0 - N, x_0 + N], y \in [y_0 - N, y_0 + N]\}$$

6.3.2 实验内容

用原始图像 lena.bmp 或 cameraman.bmp 分别加产生的 3% 椒盐噪声、高斯噪声、随机噪声合成有噪声的图像并显示以及中值滤波器去噪后的结果。

得到的结果为：



图 13: 中值滤波器

6.4 超限中值滤波器

6.4.1 实验原理

用超限中值滤波器去除图像中的噪声：当某个像素的灰度值超过窗口中像素灰度值排序中间的那个值，且达到一定水平时，则判断该点为噪声，用灰度值排序中间的那个值来代替；否则还是保持原来的灰度值。

6.4.2 实验内容

用原始图像 lena.bmp 或 cameraman.bmp 分别加产生的 3% 椒盐噪声、高斯噪声、随机噪声合成有噪声的图像并显示以及超限中值滤波器去噪后的结果。

设置阈值为 50，得到的结果为：



图 14: 超限中值滤波器

6.5 分析比较

6.5.1 实验内容

将四种处理方法的结果与原图比较，注意不同处理方法对边缘的影响。

根据实验结果，可以看出，邻域平均法虽然可以平滑噪声，但使得图像变得模糊，尤其是边缘变得不清晰。超限邻域平均法则效果较好，在灰度较为连续的区域没有出现模糊的情况，边缘也得到较好的处理，去除噪声的效果更明显。中值滤波器则对高斯噪声的处理效果较差，去除得不够彻底，在处理椒盐噪声和随机噪声时，处理效果好，边缘也没有模糊的情况。超限中值滤波器，进一步提高了去噪效果，在高斯噪声上也得到了较好的效果，边缘等细节也保留得较好。

6.6 边缘检测

6.6.1 实验原理

边缘检测主要有以下几种常用的算子：

1. Roberts 算子

它由下式给出：

$$G[F(x, y)] \approx |F(x, y) - F(x + 1, y + 1)| + |F(x + 1, y) - F(x, y + 1)|$$

其中 $F(x, y)$ 是具有整数像素坐标的输入图像。

2. Sobel 算子

图像中的每个点都用下面的两个模板做卷积，一个对通常的垂直边缘响应最大，另一个对水平边缘响应最大，两个卷积的最大值作为该点的输出位。

| | | |
|----|----|----|
| -1 | -2 | -1 |
| 0 | 0 | 0 |
| 1 | 2 | 1 |

| | | |
|----|---|---|
| -1 | 0 | 1 |
| -2 | 0 | 2 |
| -1 | 0 | 1 |

3. Prewitt 算子

下面的两个模板形成了 Prewitt 算子，和使用 Sobel 算子的方法一样。

| | | |
|----|----|----|
| -1 | -1 | -1 |
| 0 | 0 | 0 |
| 1 | 1 | 1 |

| | | |
|---|---|----|
| 1 | 0 | -1 |
| 1 | 0 | -1 |
| 1 | 0 | -1 |

4. 拉普拉斯算子

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = f(i-1, j) + f(i+1, j) + f(i, j+1) + f(i, j-1) - 4f(i, j)$$

分别用模板:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

和

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

进行边缘检测（用拉普拉斯算子：将输出图灰度值范围通过尺度变换变回到：0-255 范围；）

5. Canny 算子

- (a) 用高斯滤波器平滑图像;
- (b) 用一阶偏导的有限差分来计算梯度的幅值和方向;
- (c) 对梯度幅值进行非极大值抑制;
- (d) 用双阈值算法检测和连接边缘。

6.6.2 实验内容

要求对 blood.bmp、lena.bmp，分别用前面所述的算子进行边缘检测，显示边缘检测结果图像。

得到的结果为：

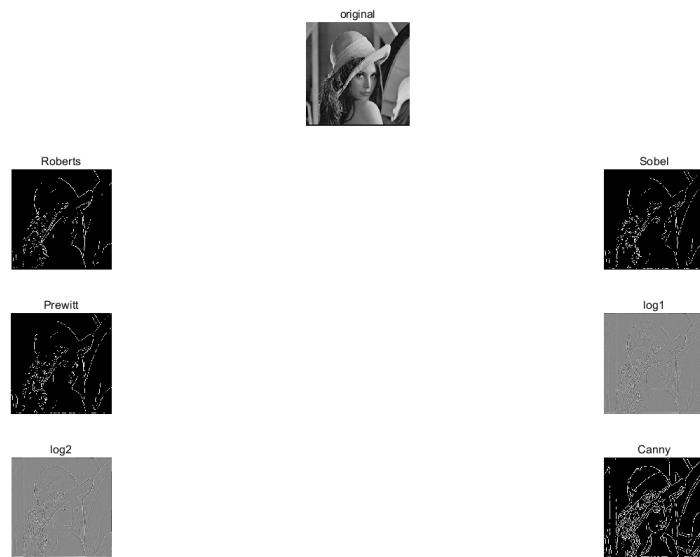


图 15: 边缘检测 (lena.bmp)

得到的结果为：

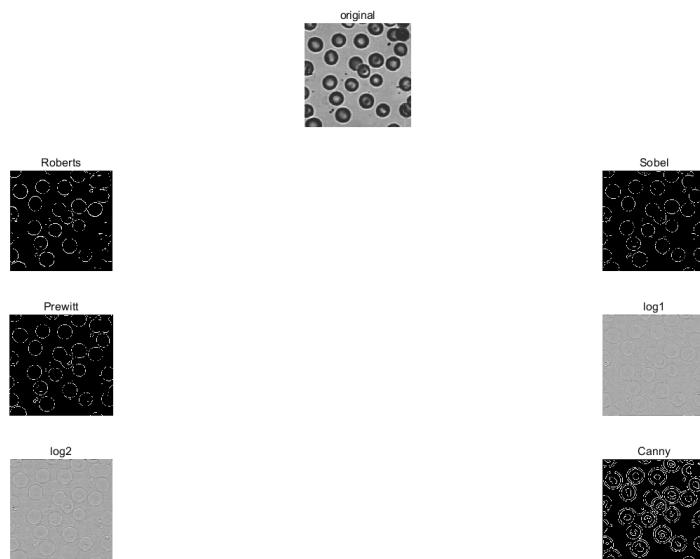


图 16: 边缘检测 (blood.bmp)

7 实验四 图像变换及频域滤波增强

7.1 Fourier 变换

7.1.1 实验内容

用 Fourier 变换算法，对 rect1.bmp 和 rect2.bmp 图像作二维 Fourier 变换；并显示其频谱。要求对幅度作变换（由于高、低频幅度相差很大），将低频移到中心点。

得到的结果为：

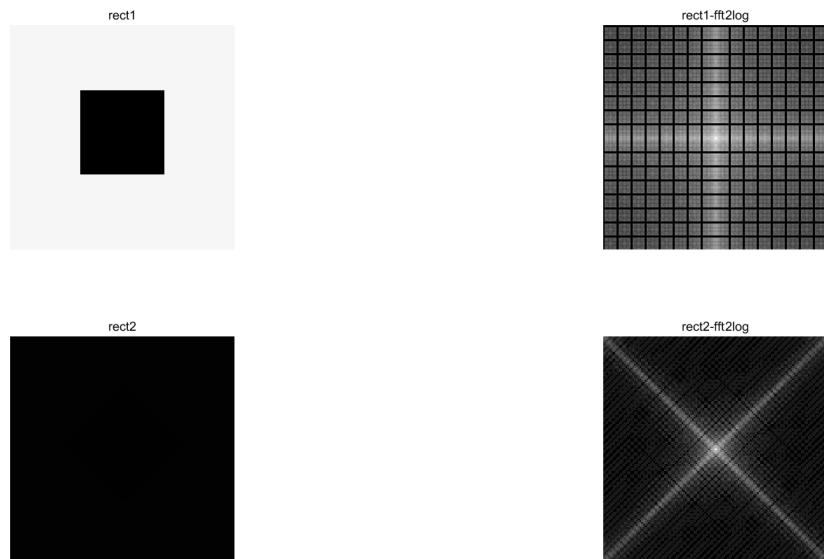


图 17: Fourier 变换 1

用 Fourier 系数的幅度进行 Fourier 反变换，并显示其图像；用 Fourier 系数的相位进行 Fourier 反变换，并显示其图像；比较 3、4 的结果，评价人眼对图像幅频特性和相频特性的敏感度。将图像的 Fourier 变换置为其共轭后进行反变换，显示其图像，并与原始图像比较其差异。

得到的结果为：

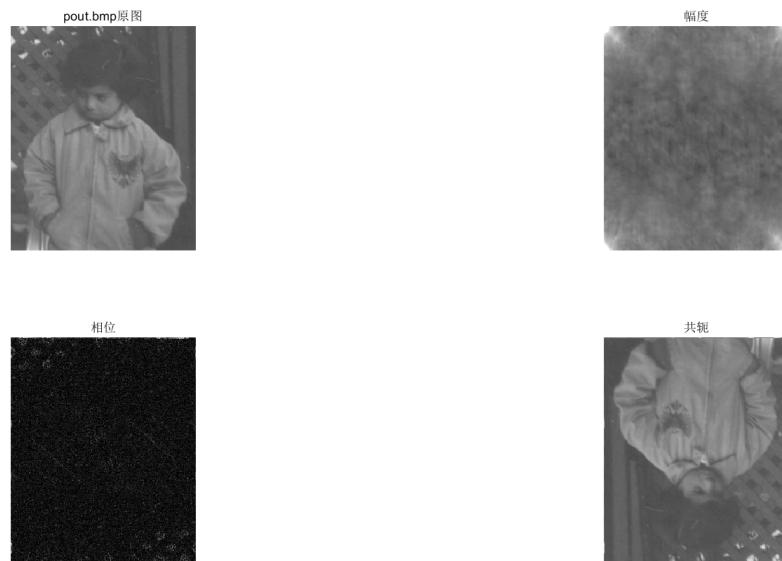


图 18: Fourier 变换 2

可以看出人眼对图像的幅频特性较为敏感，对相频特性则较不敏感。在该例子中，图像的 Fourier 变换共轭后再进行逆变换得到的图像为原图旋转 180° 的结果。

7.2 低通滤波器

7.2.1 实验内容

对图像 pout.bmp、Girl.bmp 分别采用理想低通滤波器、巴特沃斯低通滤波器和高斯低通滤波器（截止频率自选），再做反变换，观察不同截止频率下采用不同低通滤波器得到的图像与原图像的区别，特别注意振铃效应。

选择 $d=30$ ，得到的结果为：



图 19: 低通滤波器 (pout.bmp)

选择 $d=30$, 得到的结果为:



图 20: 低通滤波器 (Girl.bmp)

7.3 低通滤波器去噪

7.3.1 实验内容

对原始图像 Girl.bmp 分别加椒盐噪声、高斯噪声，产生有噪声图像，利用上述低通滤波器进行去噪，对比去噪效果。

选择 $d=30$ ，得到的结果为：

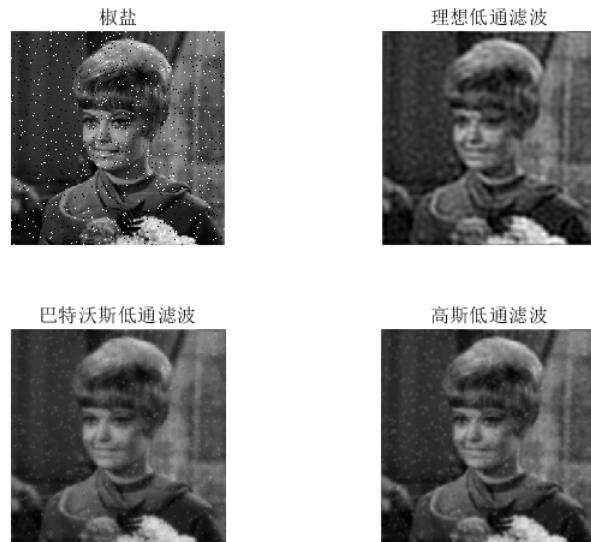


图 21: 低通滤波器去噪 (椒盐噪声)

选择 $d=30$ ，得到的结果为：



图 22: 低通滤波器去噪 (高斯噪声)

7.4 高通滤波器

对图像 pout.bmp、Girl.bmp 分别采用高通滤波器、巴特沃斯高通滤波器和高斯高通滤波器（截止频率自选），再做反变换，观察不同截止频率下采用不同高通滤波器得到的图像与原图像的区别，特别注意振铃效应。

选择 $d=5$ ，得到的结果为：



图 23: 高通滤波器 (pout.bmp)

选择 $d=5$ ，得到的结果为：



图 24: 高通滤波器 (Girl.bmp)

7.5 低通滤波器去噪

7.5.1 高频增强滤波

对图像 pout.bmp 经过高频增强滤波，再进行直方图均衡化，显示结果图像；对图像 pout.bmp 先进行直方图均衡化，再经过高频增强滤波，显示结果图像；观察对比不同处理顺序对结果图像的影响。

得到的结果为：



图 25: 高频增强滤波

8 实验五 图像恢复与图像分割

8.1 逆滤波与维纳滤波

8.1.1 实验内容

对图像 flower2.jpg 加高斯噪声，产生有噪声图像，分别对其采用逆滤波和维纳滤波进行恢复，显示、对比分析恢复结果图像。

得到的结果为：

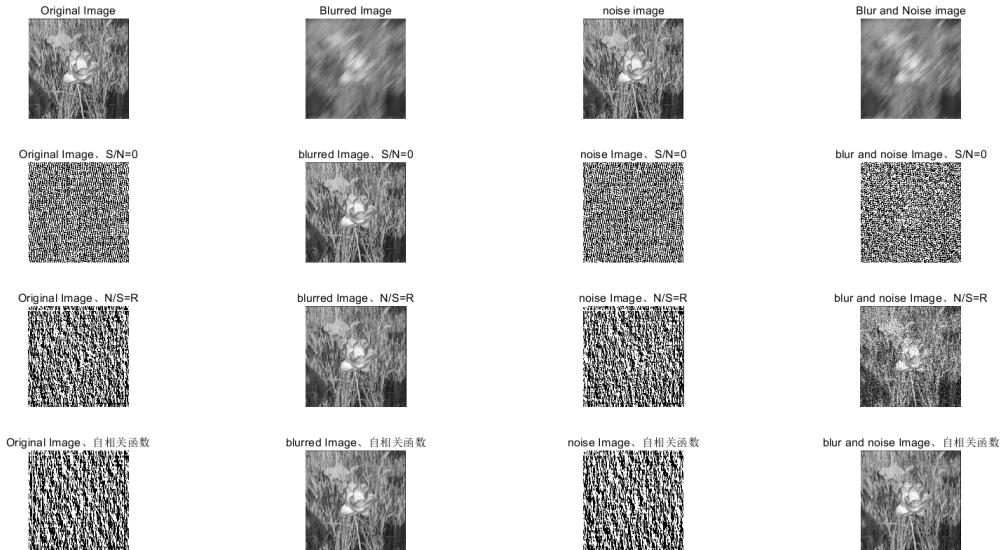


图 26: 逆滤波与维纳滤波

说明: 第一列为原图, 第二列为运动模糊, 第三列为加高斯噪声, 第四列为运动模糊加高斯噪声。第一行为滤波前, 第二行为逆滤波, 第三行为维纳滤波, 第四行为使用自相关函数作为参数进行滤波。

8.2 大津法

8.2.1 实验原理

OTSU 是基于最大类间方差的自适应间值选取法, Matlab 工具箱提供的 graythresh 函数求取间值采用的正是 Ostu 法。OTSU 算法步骤如下:

1. 统计灰度级中每个像素在整幅图像中的个数。
2. 计算每个像素在整幅图像的概率分布。
3. 对灰度级进行遍历枚举, 计算当前灰度值下前景背景类间概率。
4. 通过目标函数计算出类内与类间方差下对应的间值。

对于图像 $I(x, y)$, 前景(即目标)和背景的分割间值记作 T , 属于前景的像素点数占整幅图像的比例记为 ω_0 , 其平均灰度 μ_0 ; 背景像素点数占整幅图像的比例为 ω_1 , 其平均灰度为 μ_1 。图像的总平均灰度记为 μ , 类间方差记为 g 。假设图像的背景较暗, 并且图像的大小为 $M \times N$, 图像中像素的灰度值小于间值 T 的

像素个数记作 N_0 ，像素灰度大于阈值 T 的像素个数记作 N_1 ，则有：

$$\omega_0 = N_0 / (M \times N)$$

$$\omega_1 = N_1 / (M \times N)$$

$$N_0 + N_1 = M \times N$$

$$\omega_0 + \omega_1 = 1$$

$$\mu = \omega_0^* \mu_0 + \omega_1^* \mu_1$$

$$g = \omega_0^* (\mu_0 - \mu)^2 + \omega_1^* (\mu_1 - \mu)^2$$

将式 (5) 代入式 (6)，得到等价公式： $g = \omega_0^* \omega_1^* (\mu_0 - \mu_1)^2$ 采用遍历的方法可以得到类间方差最大的阈值 T 。

8.2.2 实验内容

对图像 lena.bmp 采用大津法 (OTSU) 自动选取阈值进行分割，显示分割二值化结果图像。

得到的结果为：



图 27: 大津法

8.3 区域分割

8.3.1 实验内容

对图像 cameraman.bmp 采用四叉树表达的迭代区域分裂合并算法进行分割。显示分割结果图像。

得到的结果为：



图 28: 区域分裂合并