



哈爾濱工業大學

计算建模实验

实验题目 噪声模型和椒盐去噪

学 号 1190200122

姓 名 袁野

指导教师 刘绍辉

日 期 2021 年 11 月 27 日

一、实验目的

了解椒盐噪声的基本含义和实际应用，掌握如何在图像中增加椒盐噪声，如何滤除图像中的椒盐噪声，了解如何利用 PSNR 和 SSIM 对于图像的去噪效果进行评估。

掌握如何对于图像进行二值化处理，包括直接设定阈值和半色调 (halftone) 方法。

二、实验内容

1、椒盐噪声

编程读取一幅灰度 bitmap 图像，并读取图像中任意位置的像素值，理解基本的位图结构，并显示图像。
理解图像数据。

思考：如果是 RGB 数据会怎么样？如果是 JPEG，PNG 格式，最后在显示的时候如何显示？

2、图像中增加椒盐噪声

噪声函数基本参数：

椒盐噪声的比例 $psn \in (0, 100)$ ：表示将图像中的像素随机修改为椒盐噪声的比率。

椒盐噪声中椒和盐的比例 $p, s: p+s=1$ ，一般可以设置为 0.5, 0.5。

椒盐噪声中椒、盐的值， $pxlow, pxhigh$ ，通常可以设置为图像允许值的最小值和最大值，例如 8 位灰度图像，设置为 0, 255。

给定一幅 8-比特灰度图像，调用噪声函数添加椒盐噪声。

3、椒盐噪声滤除

给定带椒盐噪声的图像，输出恢复的图像：

基本方法：利用非椒盐位置的真实像素信息（图像信号的先验信息，或者马尔可夫场的局部势能函数），来推导椒盐位置的真实像素值。

根据椒盐量的多少，可以利用的邻域可大可小，也就是马尔科夫随机场中的团的大小，你可以自己定，例如一阶邻域，二阶邻域，...，确认多大的噪声量，用多大的邻域最合适，效果最好。

思考：如果原来图像又含有高斯噪声，这时候怎么办？如果只含有高斯噪声，如何去噪效果会好？（鼓励实现添加高斯噪声的函数，并尝试去除高斯噪声）

4、实验效果对比

对椒盐噪声量从 5%-95% 做实验进行对比，计算 SNR 或者 PSNR 值，和 SSIM 值。

5、二值化

直接设定阈值，然后用像素值与阈值比较：

$$I(i, j) > Th, BI(i, j) = 255; I(i, j) \leq Th, BI(i, j) = 0$$

如何选取阈值，采用 OTSU 方法（大津阈值）。采用半色调方法

(halftone): 例如用误差扩散的方式来进行二值化。

三、 实验过程

1、 椒盐噪声

Bitmap 图像是一种没经过压缩的图像储存格式，其中每个像素点的数值范围为 0-255，占位 1 字节。

我们使用 cv2 中的 imread 函数将灰度图读入后将其输出，发现其是一个取值为 0-255 的大矩阵。

以上代码为 4.1.py

而 RGB 图像的每个像素由独立的 R,G,B 三个分量组成，每个分量各占 1 字节，范围也是 0-255。

JPEG 是对于图片的无损压缩，PNG 是对于图片的有损压缩，会丢失一部分信息，但是显示之后肉眼难以分辨

2、 图像中增加椒盐噪声

我们对于每一个像素点生成一个取值范围为 0-1 的随机实数，若其取值为 $(0, \text{psn}/100)$ ，那么我们将其修改为椒盐噪声，修改椒盐噪声时采用同样的方法，我们生成一个取值范围为 0-1 的随机实数，若其取值为 $(0, p/100)$ ，则将其修改为椒，否则修改为盐。我们可以通过传进不同的 psn 值来产生不同的椒盐噪声比例

3、 椒盐噪声滤除

我们采用中值滤波的方式，将每个像素点的像素值改为其周围一定区域的矩形的像素值的中位数，这样采用通过周围像素值来更新当前像素值的方式可以将图片的噪声达到一定效率的消除。

在 cv2 中的 medianBlur 函数可以自动对图片进行中值滤波。

4、 计算 SNR、PSNR、SSIM

SNR 为信噪比，计算公式为

$$\text{SNR (dB)} = 10 \cdot \log_{10} \left[\frac{\sum_{x=1}^{N_x} \sum_{y=1}^{N_y} (f(x,y))^2}{\sum_{x=1}^{N_x} \sum_{y=1}^{N_y} (f(x,y) - \hat{f}(x,y))^2} \right]$$

PSNR 为峰值信噪比，是图像压缩等领域信号重建质量的测定方法，计算公式为

$$\begin{aligned}\text{PSNR (dB)} &= 10 \cdot \log_{10} \left[\frac{\sum_{x=1}^{N_x} \sum_{y=1}^{N_y} (255)^2}{\sum_{x=1}^{N_x} \sum_{y=1}^{N_y} (f(x,y) - \hat{f}(x,y))^2} \right] \\ &= 10 \cdot \log_{10} \left[\frac{255^2}{MSE} \right]\end{aligned}$$

SSIM 为结构相似性，是一种衡量两幅图像相似度的指标，计算公式为

$$l(x, y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1} \quad c(x, y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \quad s(x, y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}$$

一般取 $c_3 = c_2/2$ 。

- μ_x 为 x 的均值
- μ_y 为 y 的均值
- σ_x^2 为 x 的方差
- σ_y^2 为 y 的方差
- σ_{xy} 为 x 和 y 的协方差
- $c_1 = (k_1 L)^2, c_2 = (k_2 L)^2$ 为两个常数，避免除零
- L 为像素值的范围， $2^B - 1$
- $k_1 = 0.01, k_2 = 0.03$ 为默认值

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

根据上述三个公式我们可以对其进行计算不同 psn 下这三个数值。

以上代码为 4.2_3_4.py

5、二值化

二值化就是将图像中所有的像素点分类成黑点或白点的过程。这样了以凸显图像的轮廓。

这里我们介绍两种方法：

5.1 OTSU

一种实现二值化的方法是直接设定阈值，然后用像素值与阈值比较，需要解决的就是如何选取阈值的问题，采用 OTSU 方法。OTSU 的中心思想是阈值 T

应该使目标与背景两类的类之间方差最大。对于一幅图像，设当前景与背景的分割阈值为 t 时，前景点占图像的比例为 w_0 ，均值为 u_0 ，背景点占比图像的比例为 w_1 ，均值为 u_1 ，则整个图像的均值为 $u=w_0*u_0+w_1*u_1$ ，建立目标函数 $g(t)=w_0*(u_0-u)^2+w_1*(u_1-u)^2$ ， $g(t)$ 就是当分割阈值为 t 时类间方差表达式。OTSU 算法使得 $g(t)$ 取得全局最大值，当 $g(t)$ 为最大值时所对应的 t 为最佳阈值。OTSU 算法又称为最大类间方差算法。

5.2 半色调方法

利用 halftone 实现二值化，我们将图像变为 0 或 255 之后，其像素点会与真实像素值产生偏差从而导致失真，这样的话我们需要将误差值按照一定的比例系数分配给周围的像素点，比例系数如下：

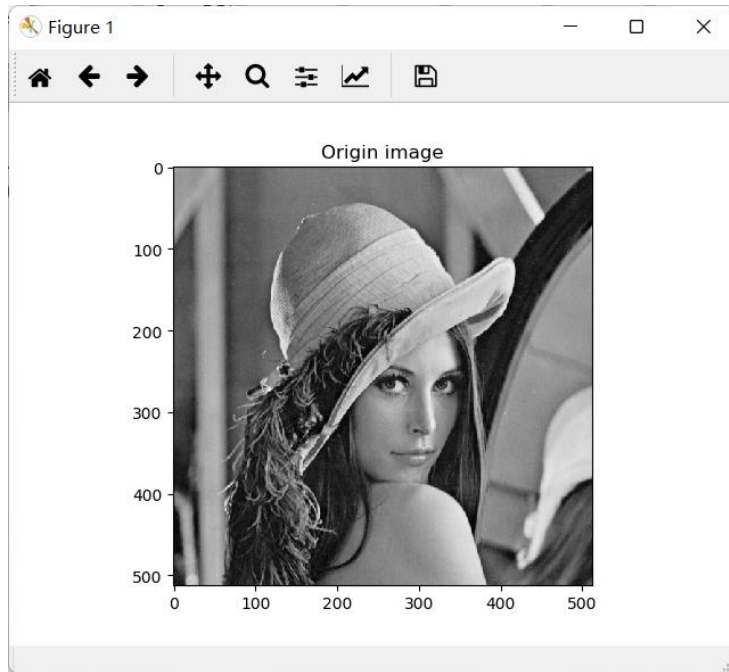
$$K = \frac{1}{48} \times \begin{bmatrix} & & \Delta & 7 & 5 \\ 3 & 5 & 7 & 5 & 3 \\ 1 & 3 & 5 & 3 & 1 \end{bmatrix}$$

以上代码为 4.5.py

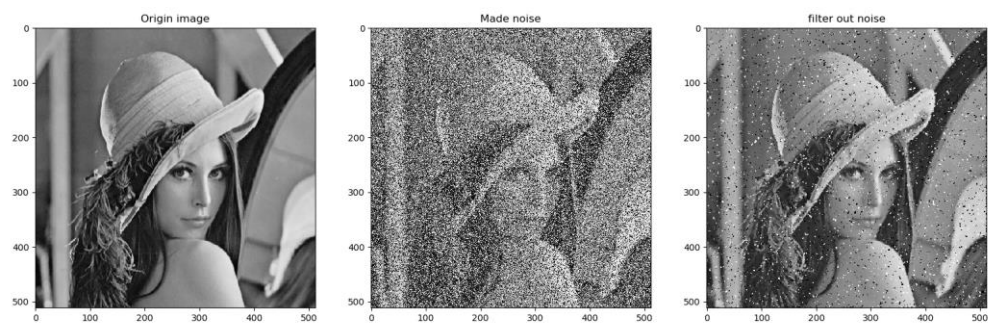
四、实验结果

1、椒盐噪声

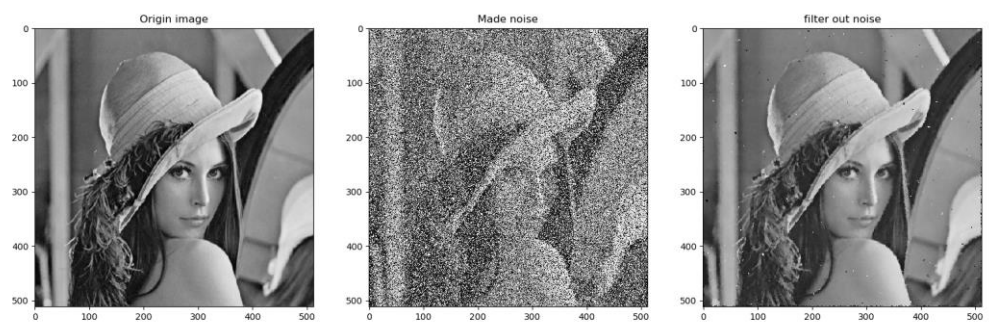
```
D:\soft\anaconda3\python.exe D:/github/School_Pro/计算建模/Lab4/4.1.py
[[161 161 161 ... 169 154 127]
 [161 161 161 ... 169 154 127]
 [161 161 161 ... 169 154 127]
 ...
 [ 40  40  48 ... 102  98  96]
 [ 41  41  52 ... 102 103 106]
 [ 41  41  52 ... 102 103 106]]
```



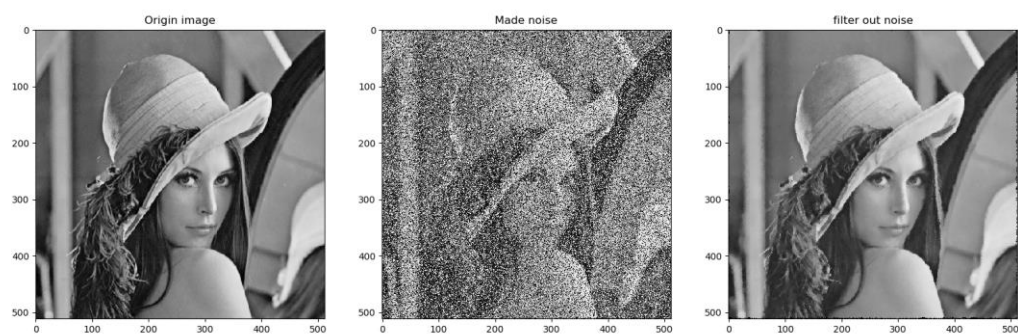
- 2、 图像中增加椒盐噪声
我们将 psn 设置为 45
中值滤波时，窗口为 3*3:



窗口为 5*5:



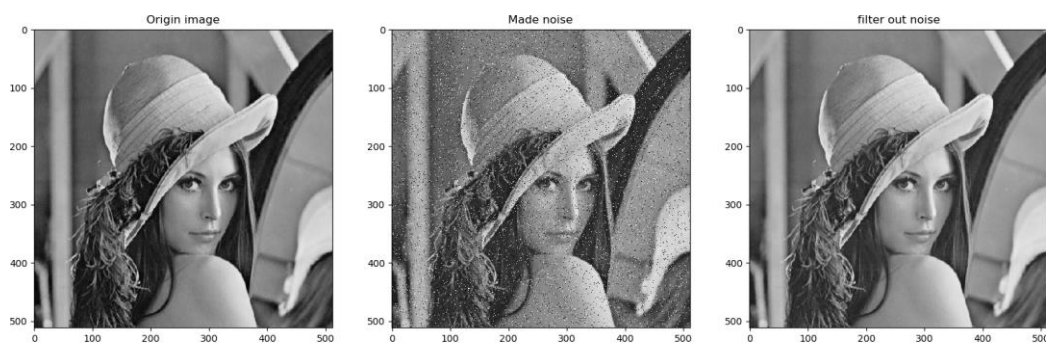
窗口为 7*7:



我们将 psn 设置为 5 到 95，窗口大小设置为 3*3:

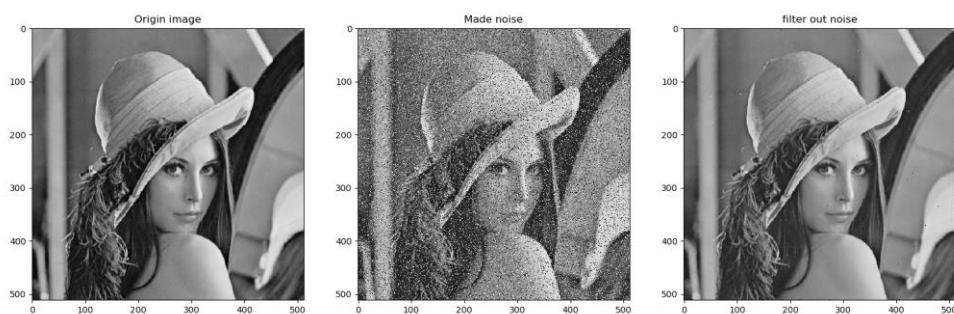
psn=5:

```
Now rate is 5%.  
SNR= 38.20964550863577  
PSNR= 43.947912610223476  
SSIM= 0.999451945920025
```



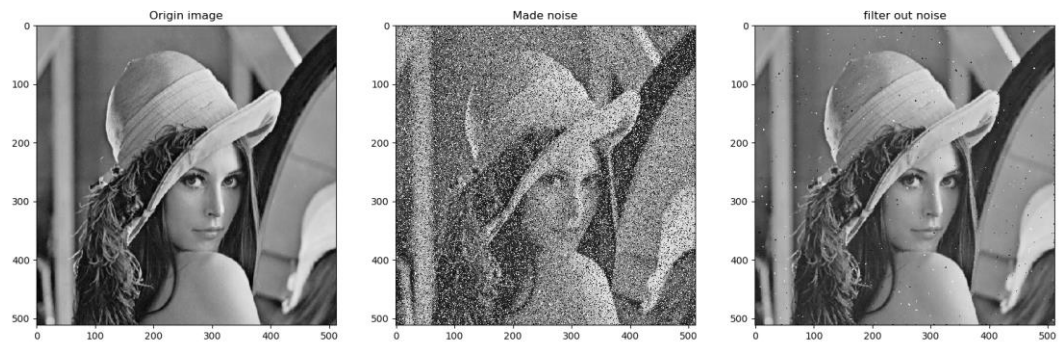
psn=15:

```
Now rate is 15%.  
SNR= 29.810964886939757  
PSNR= 35.549231988527474  
SSIM= 0.9961904221812948
```



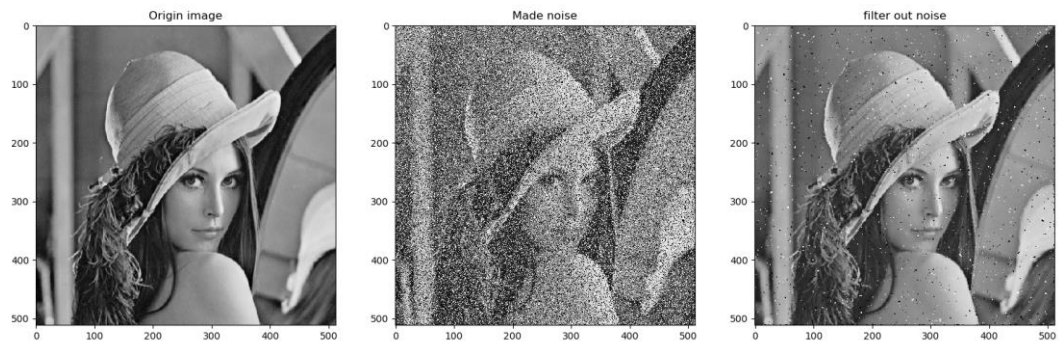
psn=25:

```
Now rate is 25%.  
SNR= 23.038558101883275  
PSNR= 28.776825203470988  
SSIM= 0.9820273547457821
```



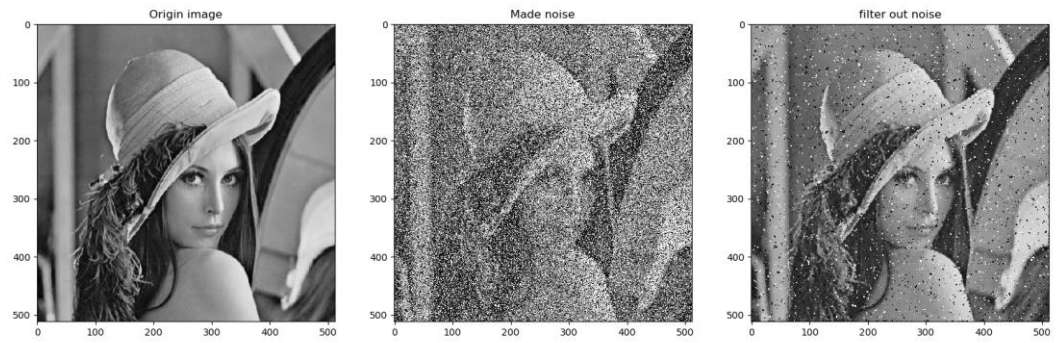
psn=35:

```
Now rate is 35%.  
SNR= 17.386737795516424  
PSNR= 23.125004897104134  
SSIM= 0.9361326874375293
```



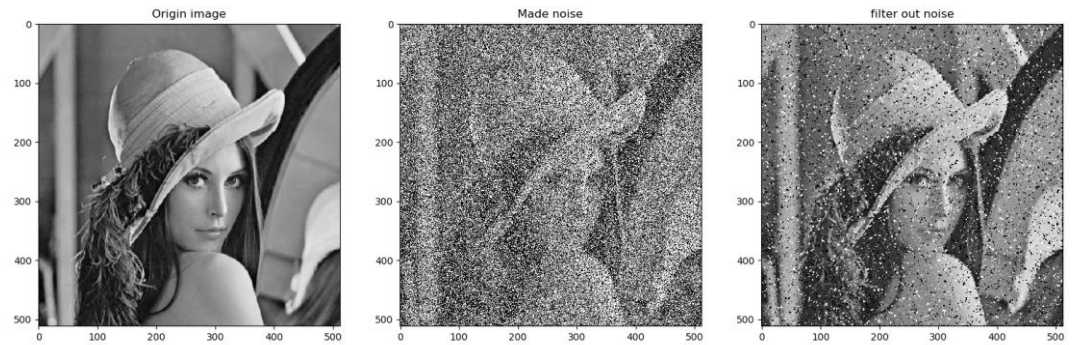
psn=45:

```
Now rate is 45%.  
SNR= 12.734301553471088  
PSNR= 18.4725686550588  
SSIM= 0.8286357731003282
```

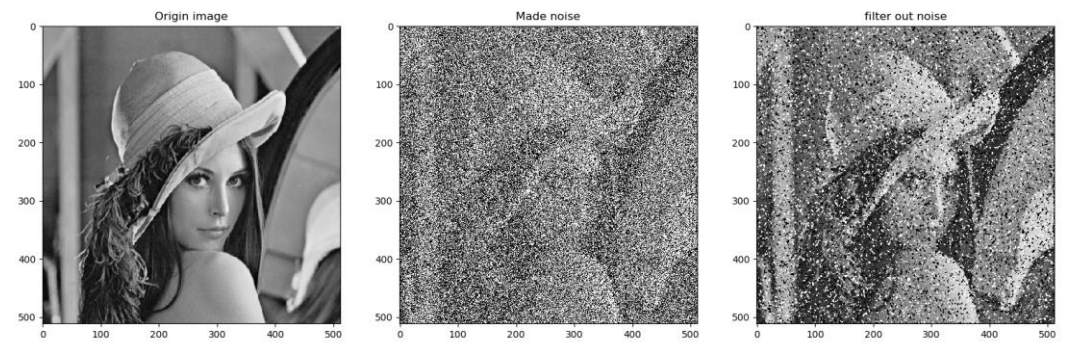
psn=55:

```
Now rate is 55%.
SNR= 9.321932998670237
PSNR= 15.06020010025795
SSIM= 0.6744975155108442
```



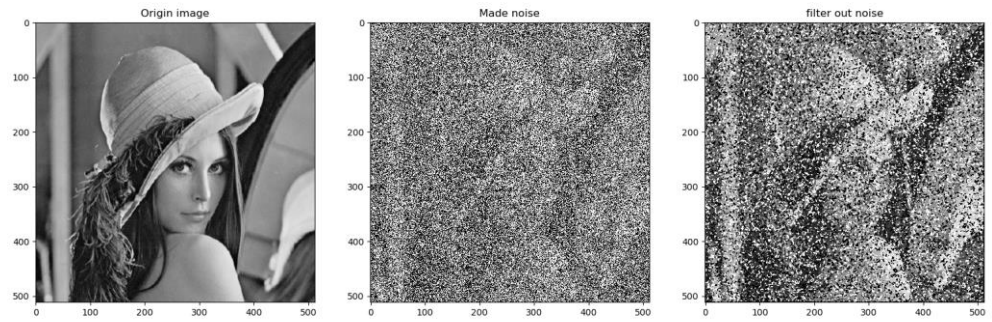
psn=65:

```
Now rate is 65%.
SNR= 6.361973731647778
PSNR= 12.100240833235489
SSIM= 0.47840095330073407
```



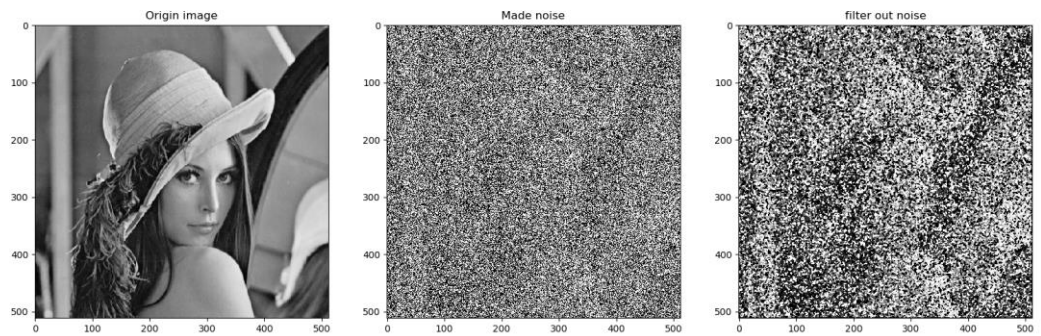
psn=75:

```
Now rate is 75%.  
SNR= 4.049243030102167  
PSNR= 9.78751013168988  
SSIM= 0.3072836339151222
```



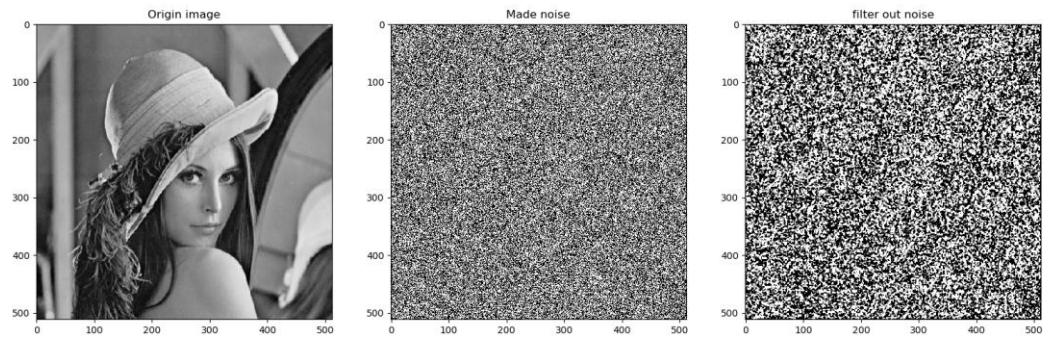
psn=85:

```
Now rate is 85%.  
SNR= 2.0668499384436516  
PSNR= 7.805117040031364  
SSIM= 0.1570914061208699
```

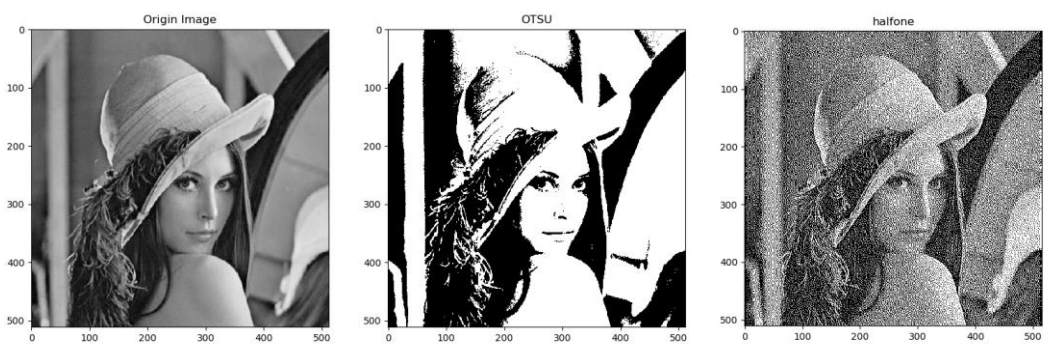


psn=95:

```
Now rate is 95%.  
SNR= 0.3835833467409222  
PSNR= 6.121850448328635  
SSIM= 0.04154673461028762
```



3、 二值化



五、 心得体会

对椒盐噪声有了更直观的认识，同时对对椒盐噪声的去除方式有了更加深入的了解，学会了使用 SNR、PSNR、SSIM 对去噪后的图进行评价，学会了图像的二值化处理。