数字图像处理

第六讲　图像复原（上）

王伟强

中国科学院大学计算机科学与技术学院

内容概要

- 图像降质/复原过程模型
- 噪声模型
- 仅存在噪声的空域滤波图像复原
- 频域滤波周期噪声
- 估计退化函数
- 直接逆滤波
- 维纳过滤
- 约束最小二乘滤波

图像退化与复原

- 图像复原

  以某种预定义的方式改善给定图像。

- 图像复原和图像增强之间的区别

- 图像退化与复原的数学模型

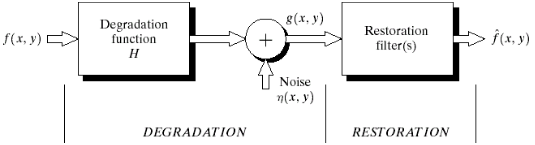


FIGURE 5.1 A model of the image degradation/restoration process.

$$g(x,y) = H[f(x,y)] + \eta(x,y)$$
$$g(x,y) = h(x,y) * f(x,y) + \eta(x,y)$$
$$G(u,v) = H(u,v)F(u,v) + N(u,v)$$

噪声模型

- 使用函数imnoise添加噪声
- 具有指定分布的空间随机噪声（独立同分布的空间随机噪声）
- 周期噪声
- 估算噪声参数

用imnoise函数添加噪声

- g = imnoise(f, type, parameters)

  f：输入图像，类型和参数将在后面解释。

  [注意]：在向其添加噪声之前，将输入图像转换为[0,1]范围内double类图像

例如

G=imnoise(f,'gaussian',m,var)

G=imnoise(f,'localvar',V)

G=imnoise(f,'localvar',image intensity,var)

G=imnoise(f,'salt&pepper',d))

G=imnoise(f, 'speckle', var)

G=imnoise(f, 'poisson')

产生具有指定分布的随机噪声

- 一个著名的结果：如果w是区间(0, 1)上的一个均匀分布的随机变量，那么我们可以通过求解方程得到一个具有指定累积分布函数(CDF) F_z 的随机变量 z

$$z = F_z^{-1}(w)$$

- 一个例子：用瑞利CDF生成随机数z

$$F_z(z) = \begin{cases} 1 - e^{-(z-a)^2/b} & if\ z \geq a \\ 0 & if\ z < a \end{cases}$$

我们可以通过求解以下等式来获得

$$1 - e^{-(z-a)^2/b} = w$$

得到

$$z = a + \sqrt{-b\ln(1-w)}$$

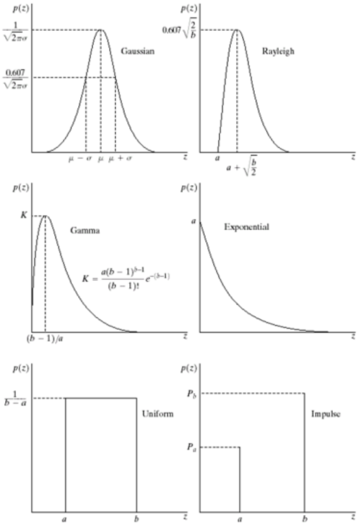
产生具有指定分布的随机噪声

TABLE 5.1 Generation of random variables.

Name	PDF	Mean and Variance	CDF	Generator†
Uniform	$p_z(z) = \begin{cases} \frac{1}{b-a} & \text{if } a \le z \le b \\ 0 & \text{otherwise} \end{cases}$	$m = \frac{a+b}{2},\quad \sigma^2 = \frac{(b-a)^2}{12}$	$F_z(z) = \begin{cases} 0 & z < a \\ \frac{z-a}{b-a} & a \le z \le b \\ 1 & z > b \end{cases}$	MATLAB function `rand`
Gaussian	$p_z(z) = \frac{1}{\sqrt{2\pi}b} e^{-(z-a)^2/2b^2}$ $-\infty < z < \infty$	$m = a,\quad \sigma^2 = b^2$	$F_z(z) = \int_{-\infty}^{z} p_z(v)\,dv$	MATLAB function `randn`
Salt & Pepper ??	$p_z(z) = \begin{cases} P_a & \text{for } z = a \\ P_b & \text{for } z = b \\ 0 & \text{otherwise} \end{cases}$ $b > a$	$m = aP_a + bP_b$ $\sigma^2 = (a-m)^2P_a + (b-m)^2P_b$	$F_z(z) = \begin{cases} 0 & \text{for } z < a \\ P_a & \text{for } a \le z < b \\ P_a + P_b & \text{for } b \le z \end{cases}$	MATLAB function `rand` with some additional logic
Lognormal	$p_z(z) = \frac{1}{\sqrt{2\pi}bz} e^{-[\ln(z)-a]^2/2b^2}$?? 0	$m = e^{a+(b^2/2)},\ \sigma^2 = [e^{b^2} - 1]e^{2a+b^2}$	$F_z(z) = \int_{0}^{z} p_z(v)\,dv$	$z = ae^{bN(0,1)}$??
Rayleigh	$p_z(z) = \begin{cases} \frac{2}{b}(z-a)e^{-(z-a)^2/b} & z \ge a \\ 0 & z < a \end{cases}$	$m = a + \sqrt{\pi b/4},\ \sigma^2 = \frac{b(4-\pi)}{4}$	$F_z(z) = \begin{cases} 1 - e^{-(z-a)^2/b} & z \ge a \\ 0 & z < a \end{cases}$	$z = a + \sqrt{b\ln[1 - U(0,1)]}$
Exponential	$p_z(z) = \begin{cases} ae^{-az} & z \ge 0 \\ 0 & z < 0 \end{cases}$	$m = \frac{1}{a},\quad \sigma^2 = \frac{1}{a^2}$	$F_z(z) = \begin{cases} 1 - e^{-az} & z \ge 0 \\ 0 & z < 0 \end{cases}$	$z = -\frac{1}{a}\ln[1 - U(0,1)]$
Erlang	$p_z(z) = \frac{a^b z^{b-1}}{(b-1)!} e^{-az}$ $z \ge 0$	$m = \frac{b}{a},\quad \sigma^2 = \frac{b}{a^2}$	$F_z(z) = \left[1 - e^{-az}\sum_{n=0}^{b-1}\frac{(az)^n}{n!}\right]$ $z \ge 0$	$z = E_1 + E_2 + \cdots + E_b$ (The E's are exponential random numbers with parameter a.)

† $N(0,1)$ denotes normal (Gaussian) random numbers with mean 0 and a variance of 1. $U(0,1)$ denotes uniform random numbers in the range $(0,1)$.

一些重要的概率密度函数的形状

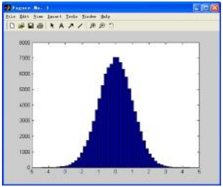


a	b
c	d
e	f

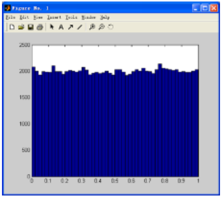
FIGURE 5.2 Some important probability density functions.

产生具有指定分布的随机噪声

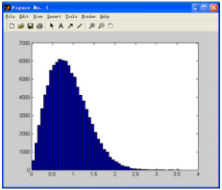
```
r=imnoise2('gaussian',100000,1,0,1);
hist(r,50);
```



```
ru=imnoise2('uniform',100000,1,0,1);
hist(ru,50);
```



```
rr=imnoise2('rayleigh',100000,1,0,1);
hist(rr,50);
```



产生具有指定分布的随机噪声

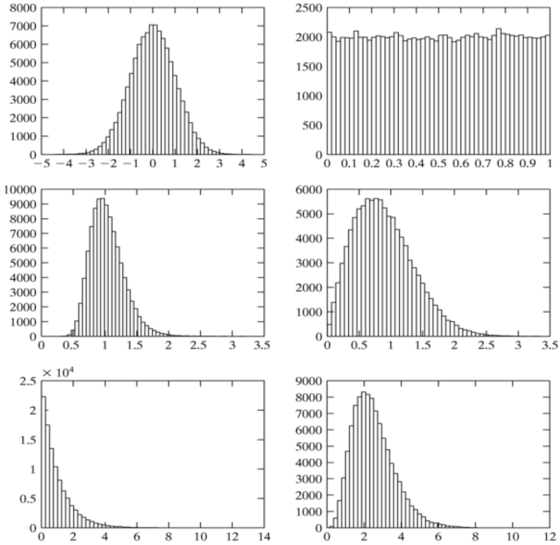


FIGURE 5.2 Histograms of random numbers: (a) Gaussian, (b) uniform, (c) lognormal, (d) Rayleigh, (e) exponential, and (f) Erlang. In each case the default parameters listed in the explanation of function `imnoise2` were used.

周期噪声

- 周期性噪声信号

$$r(x,y) = A\sin(2\pi u_0(x+B_x)/M + 2\pi v_0(y+B_y)/N)$$

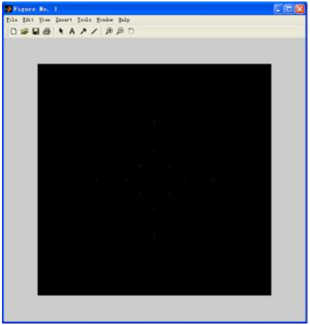
- 它的对应傅立叶变换

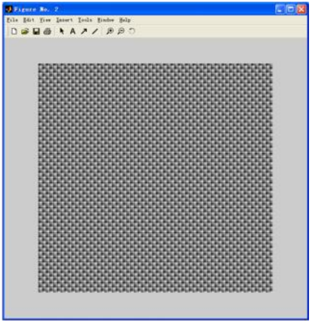
$$R(u,v) = j\frac{A}{2}[\exp(j2\pi u_0 B_x/M)\delta(u+u_0, v+v_0) - \exp(j2\pi v_0 B_y/N)\delta(u-u_0, v-v_0)]$$

??

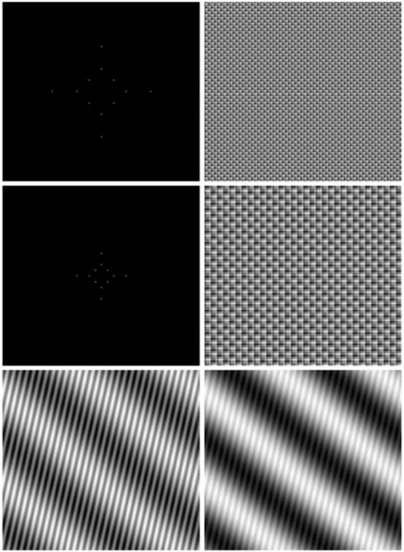
周期噪声

```
C=[0 64;0 128;32 32;64 0;128 0;-32 32];
[r,R,S]=imnoise3(512,512,C);
imshow(S,[]);
figure,imshow(r,[]);
```





周期噪声



a b
c d
e f

FIGURE 5.3 (a) Spectrum of specified impulses. (b) Corresponding sine noise pattern. (c) and (d) A similar sequence. (e) and (f) Two other noise patterns. The dots in (a) and (c) were enlarged to make them easier to see.

例子:图像被独立噪声污染

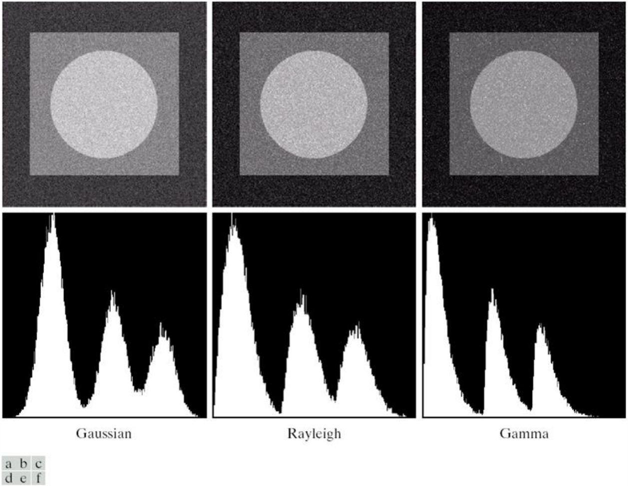


FIGURE 5.4 Images and histograms resulting from adding Gaussian, Rayleigh, and gamma noise to the image in Fig. 5.3.

例子:图像被独立噪声损坏

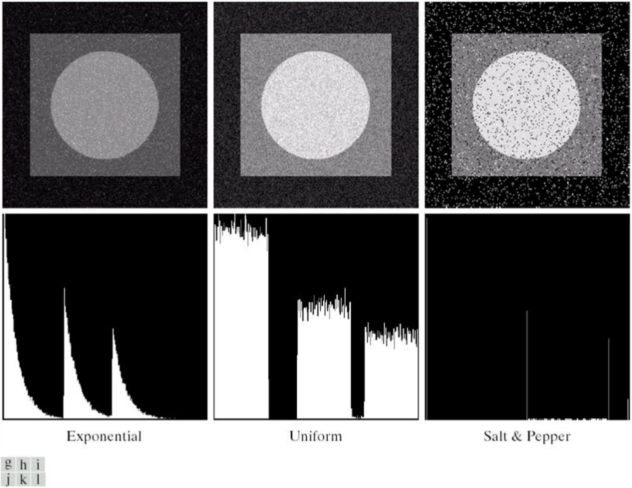
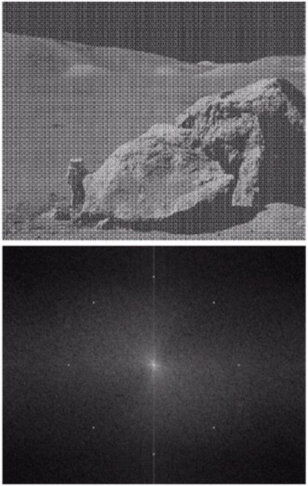


FIGURE 5.4 *(Continued)* Images and histograms resulting from adding exponential, uniform, and impulse noise to the image in Fig. 5.3.

例子:图像被周期性噪声污染



a
b

FIGURE 5.5
(a) Image corrupted by sinusoidal noise. (b) Spectrum (each pair of conjugate impulses corresponds to one sine wave). (Original image courtesy of NASA.)

估计噪声参数

- 通过对图像的傅里叶谱的审视来估计周期性噪声的的参数，例如通过肉眼去发现频谱中的频率尖峰。
- 尝试直接从图像推断噪声分量的周期性通常仅在简单的情况下是可行的。
- 在噪声尖峰非常明显，或者具有对干扰的频率分量一般位置的先验知识的情况下，才可以进行自动分析。
- 噪声PDF的参数可以部分从传感器规范中得知，但通常需要针对特定的成像环境估计它们，例如，一组“平坦”环境。
- 当仅有生成的图像可用时，通常可以从相对恒定灰度级的图像小区域中估计噪声PDF的参数。

估计噪声参数

- 最简单的方法就是利用图像中的采样数据来估计噪声的均值和方差。
- 通过直方图的形状来辨识最接近PDF匹配。
- 如果形状近似为高斯分布，均值和方差就是我们需要的全部了。对于5.2.2节中讨论的其他形状，我们可使用均值和方差来求解参数 ***a*** 和 ***b***。
- 统计矩与中心矩

$$\mu_n = \sum_{i=0}^{L-1}(z_i - m)^n p(z_i) \quad \text{where } m = \sum_{i=0}^{L-1} z_i p(z_i)$$

$$\text{when } n = 0 \qquad \mu_0 = \sum_{i=0}^{L-1}(z_i - m)^0 p(z_i) = 1$$

$$\text{when } n = 1 \qquad \mu_1 = \sum_{i=0}^{L-1}(z_i - m) p(z_i)$$

$$= \sum_{i=0}^{L-1} z_i p(z_i) - m \sum_{i=0}^{L-1} p(z_i) = 0$$

$$\text{when } n = 2 \qquad \mu_2 = \sum_{i=0}^{L-1}(z_i - m)^2 p(z_i) = Var(z)$$

估计噪声参数

- 均值和中心矩

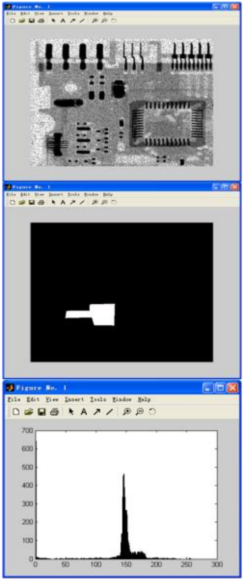
  [v,unv]=statmoments(p,n)

- 选择感兴趣的地区

  B=roipoly(f,c,r)

```
f=imread('Fig0505(a)(ckt pepper only).tif');
imshow(f)
[B,c,r]=roipoly(f);   %鼠标选择
imshow(B)

[p,npix]=histroi(f,c,r);
bar(p,1)
```



仅有噪音情况下的图像复原 – 空域滤波

- 如果只有噪音，图像退化模型可表示为

$$g(x,y) = f(x,y) + \eta(x,y)$$
$$G(u,v) = F(u,v) + N(u,v)$$

- 空域噪声滤波器
- 自适应空空域滤器

均值滤波

- 算术平均滤波

$$\hat{f}(x,y) = \frac{1}{mn}\sum_{(s,t)\in S_{xy}} g(s,t)$$

- 几何均值滤波器

$$\hat{f}(x,y) = \left[\prod_{(s,t)\in S_{xy}} g(s,t)\right]^{\frac{1}{mn}}$$

- 调和均值滤波器

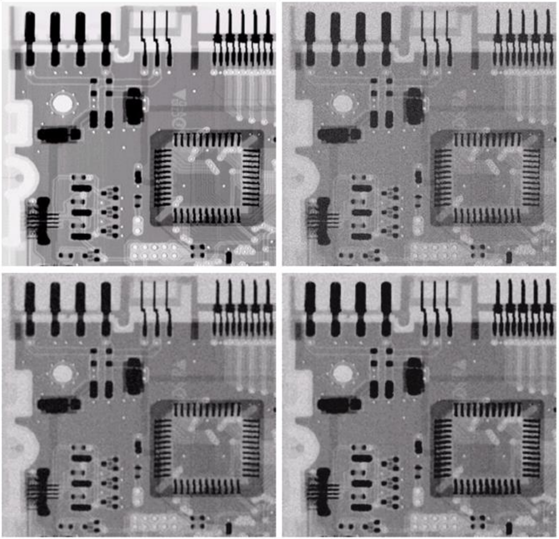
$$\hat{f}(x,y) = \frac{mn}{\sum_{(s,t)\in S_{xy}} \frac{1}{g(s,t)}}$$

- 反调和均值滤波器

$$\hat{f}(x,y) = \frac{\sum_{(s,t)\in S_{xy}} g(s,t)^{Q+1}}{\sum_{(s,t)\in S_{xy}} g(s,t)^{Q}}$$

反调和均值滤波器非常适合消除盐和胡椒噪声的影响。当 Q > 0 时，该滤波器可消除椒噪声；当 Q < 0 时，它适合消除盐噪声。它不能同时消除椒盐噪声。

算术与几何均值滤波器的滤波效果



a b
c d

FIGURE 5.7 (a) X-ray image. (b) Image corrupted by additive Gaussian noise. (c) Result of filtering with an arithmetic mean filter of size 3×3. (d) Result of filtering with a geometric mean filter of the same size. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

调和均值与反调和均值滤波器

- 调和均值滤波器能很好地去除盐噪声，但不能滤除椒噪声。对于高斯噪声很好

$$\hat{f}(x,y)=\frac{mn}{\sum_{(s,t)\in S_{xy}}\frac{1}{g(s,t)}}$$

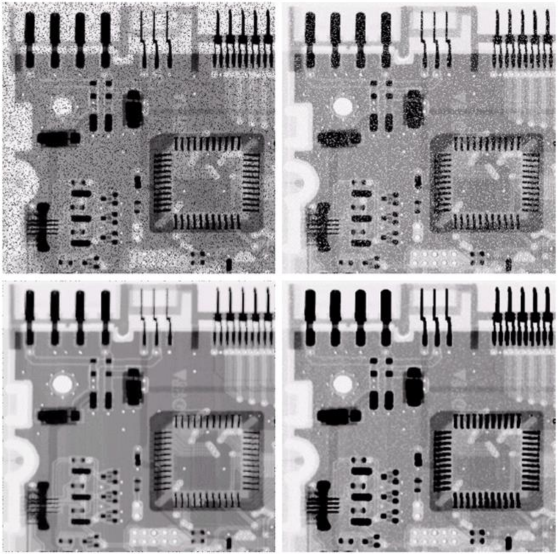
- 反调和均值滤波器能有效地消除的盐噪声（当Q为负值时）；当Q为正值时，对椒噪声也有很好的效果。但它不能有效处理两种噪声同时存在的情况。

$$\hat{f}(x,y)=\frac{\sum_{(s,t)\in S_{xy}}g(s,t)^{Q+1}}{\sum_{(s,t)\in S_{xy}}g(s,t)^{Q}}$$

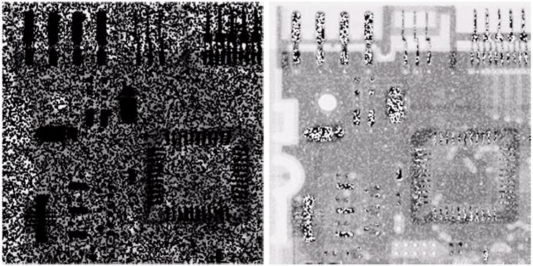
调和均值与反调和均值滤波器的滤波示例

a	b
c	d

FIGURE 5.8 (a) Image corrupted by pepper noise with a probability of 0.1. (b) Image corrupted by salt noise with the same probability. (c) Result of filtering (a) with a 3×3 contraharmonic filter of order 1.5. (d) Result of filtering (b) with $Q = -1.5$.



调和均值与反调和均值滤波器的滤波示例



a b

FIGURE 5.9 Results of selecting the wrong sign in contraharmonic filtering. (a) Result of filtering Fig. 5.8(a) with a contraharmonic filter of size 3 × 3 and $Q = -1.5$. (b) Result of filtering 5.8(b) with $Q = 1.5$.

统计排序滤波器

- 中值滤波器

$$\hat{f}(x,y) = \underset{(s,t)\in S_{xy}}{median}\ g(s,t)$$

- 最大、最小滤波器

$$\hat{f}(x,y) = \max_{(s,t)\in S_{xy}} g(s,t)$$

$$\hat{f}(x,y) = \min_{(s,t)\in S_{xy}} g(s,t)$$

- 中点滤波器

$$\hat{f}(x,y) = \frac{1}{2}[\max_{(s,t)\in S_{xy}} g(s,t) + \min_{(s,t)\in S_{xy}} g(s,t)]$$

- 阿尔法均值滤波

$$\hat{f}(x,y) = \frac{1}{mn-d}\sum_{(s,t)\in S_{xy}} g_r(s,t)$$

中值滤波器对椒盐噪声的滤波效果

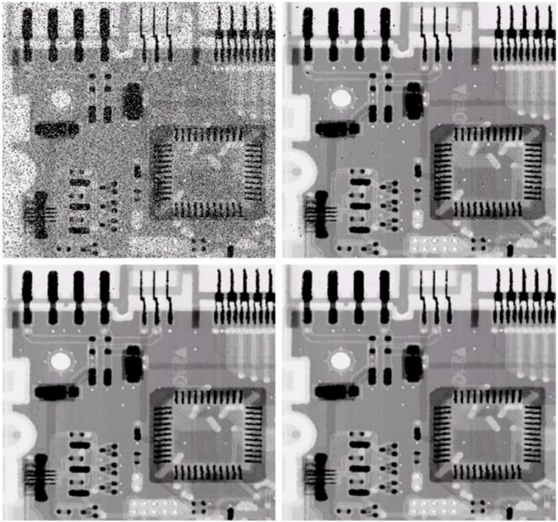
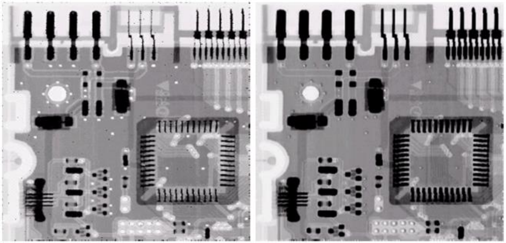


FIGURE 5.10 (a) Image corrupted by salt-and-pepper noise with probabilities $P_a = P_b = 0.1$. (b) Result of one pass with a median filter of size 3×3. (c) Result of processing (b) with this filter. (d) Result of processing (c) with the same filter.

最大、最小值滤波器对椒盐噪声的滤波效果



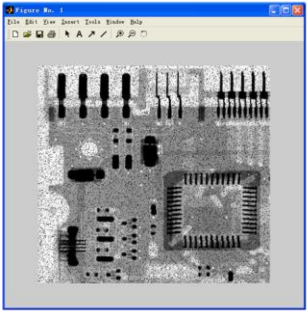
a b

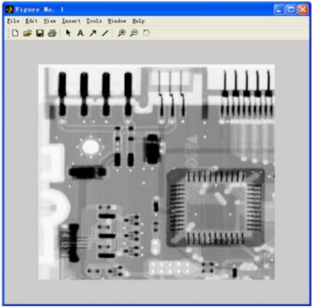
FIGURE 5.11 (a) Result of filtering Fig. 5.8(a) with a max filter of size 3 × 3. (b) Result of filtering 5.8(b) with a min filter of the same size.

空域噪声滤波器滤波实验

% 图像仅受椒噪声污染，污染概率为0.1
% 过滤椒噪声采用一个Q为正值的反谐波滤波器

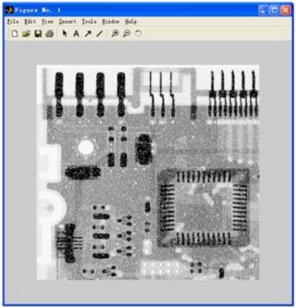
```
gp=imread('Fig0505(a)(ckt pepper only).tif');
fp=spfilt(gp,'chmean',3,3,1.5);
imshow(gp),figure,imshow(fp)
```

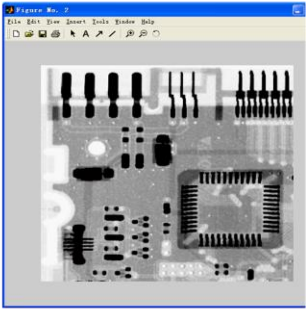




空域噪声滤波器滤波实验

```
% 过滤盐噪声采用一个Q为负值的反调和滤波器
gs=imread('Fig0505(b)(ckt_salt_only).tif');
fs=spfilt(gs,'chmean',3,3,-1.5);
imshow(gs),figure,imshow(fs)
```





自适应空域中值滤波器

- 符号:

Z_{min} = *minimum intensity value in* S_{xy}

Z_{max} = *maximum intensity value in* S_{xy}

Z_{med} = *median intensity value in* S_{xy}

Z_{xy} = *intensity value at coordinates* x,y

- 自适应中值滤波算法

Level A:

If $Z_{min} < Z_{med} < Z_{max}$, go to level B

else increase the window size

if window size $\leqslant S_{max}$, repeat level A

else output Z_{med}

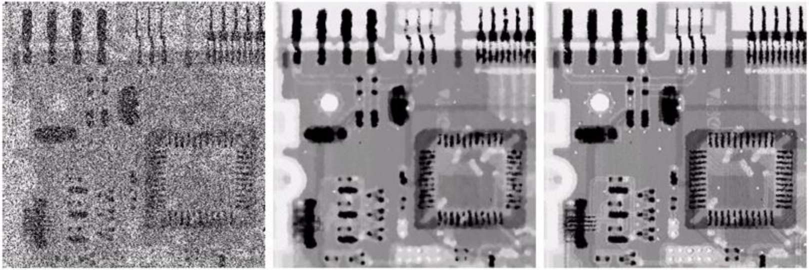
Level B:

If $Z_{min} < Z_{xy} < Z_{max}$, output Z_{xy}

else output Z_{med}

自适应空域中值滤波器的滤波效果

f=adpmedian(g,Smax)



a b c

FIGURE 5.14 (a) Image corrupted by salt-and-pepper noise with probabilities $P_a = P_b = 0.25$. (b) Result of filtering with a 7×7 median filter. (c) Result of adaptive median filtering with $S_{max} = 7$.

用频域滤波消除周期噪声

➢ 带阻滤波器

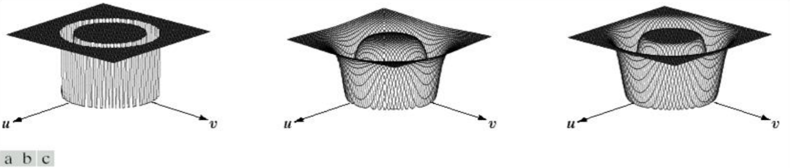
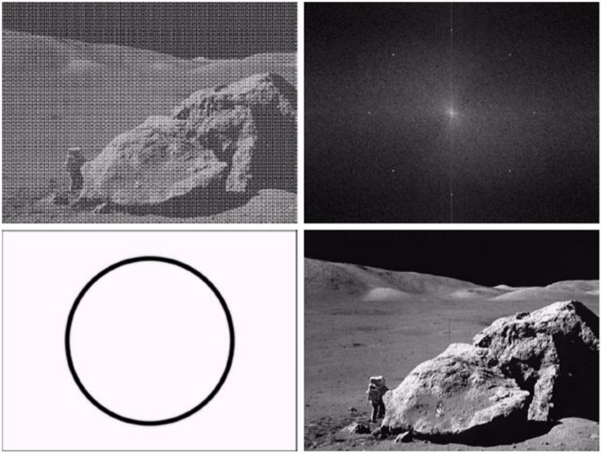


FIGURE 5.15 From left to right, perspective plots of ideal, Butterworth (of order 1), and Gaussian bandreject filters.

- $H(u,v) =$
$$\begin{cases} 1 & D(u,v) < D_0 - \frac{W}{2} \\ 0 & D_0 - \frac{W}{2} < D(u,v) < D_0 + \frac{W}{2} \\ 1 & D(u,v) > D_0 + \frac{W}{2} \end{cases}$$
- $H(u,v) =$
$$\frac{1}{1+\left[\frac{D(u,v)W}{D^2(u,v)-D_0^2}\right]^{2n}}$$
- $H(u,v) =$
$$1 - exp\left[-\frac{1}{2}\left(\frac{D^2(u,v)-D_0^2}{D(u,v)W}\right)^2\right]$$

用频域滤波消除周期噪声

- 带阻滤波器的应用



a b
c d

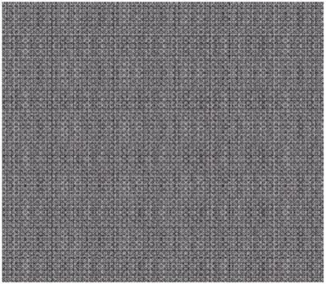
FIGURE 5.16 (a) Image corrupted by sinusoidal noise. (b) Spectrum of (a). (c) Butterworth bandreject filter (white represents 1). (d) Result of filtering. (Original image courtesy of NASA.)

用频域滤波消除周期噪声

- 带通阻滤波器

$$H_{bp}(u,v) = 1 - H_{br}(u,v)$$

FIGURE 5.17 Noise pattern of the image in Fig. 5.16(a) obtained by bandpass filtering.



用频域滤波消除周期噪声

- 陷波滤波器

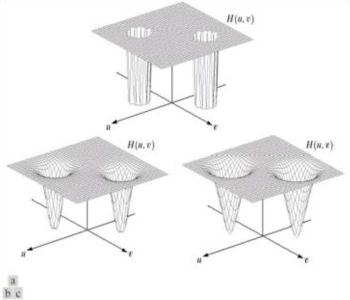


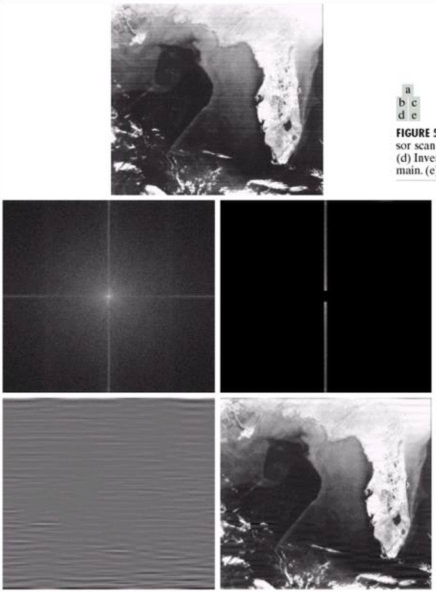
FIGURE 5.18 Perspective plots of (a) ideal, (b) Butterworth (of order 2), and (c) Gaussian notch (reject) filters.

- $H(u,v)=\begin{cases} 0 & D_1(u,v)\le D_0 \ or \ D_2(u,v)\le D_0 \\ 1 & else \end{cases}$
- $H(u,v)=\frac{1}{1+[\frac{D_0^2}{D_1(u,v)D_2(u,v)}]^n}$
- $H(u,v)=1-exp[-\frac{1}{2}(\frac{D_1(u,v)D_2(u,v)}{D_0^2})]$

其中

$$D_1(u,v)=[(u-\frac{M}{2}-u_0)^2+(v-\frac{N}{2}-v_0)^2]^{0.5}$$
$$D_2(u,v)=[(u-\frac{M}{2}+u_0)^2+(v-\frac{N}{2}+v_0)^2]^{0.5}$$

用陷波滤波器去除周期性噪声



a
b c
d e

FIGURE 5.19 (a) Satellite image of Florida and the Gulf of Mexico (note horizontal sensor scan lines). (b) Spectrum of (a). (c) Notch pass filter shown superimposed on (b). (d) Inverse Fourier transform of filtered image, showing noise pattern in the spatial domain. (e) Result of notch reject filtering. (Original image courtesy of NOAA.)

最佳陷波滤波算法

- 当存在多个干扰分量时，由于滤波过程中去除的图像信息过多，上述方法的滤波效果并不总是可以接受的。
- 这里讨论的方法的最优是在最小化被复原图像局部方差意义下的。
- 首先，通过如下计算获得对噪声的初始估计

$$N(u,v) = F_N(u,v)G(u,v)$$
$$\eta(x,y) = \Im^{-1}(F_N(u,v)G(u,v))$$

其中 $F_N(u,v)$ 是获取噪声干扰模式的陷波滤波器。

- 令 $$\hat{f}(x,y) = g(x,y) - w(x,y)\eta(x,y)$$

我们将寻找确定调制函数 $w(x,y)$，来最小化 $\hat{f}(x,y)$ 的局部方差

最佳陷波滤波算法

- 目标函数: $\min \sigma^2 = \frac{1}{(2a+1)(2b+1)} \sum_{s=-a}^{a} \sum_{t=-b}^{b} [\hat{f}(x+s, y+t) - \bar{f}]^2$

  where $\bar{f} = \frac{1}{(2a+1)(2b+1)} \sum_{s=-a}^{a} \sum_{t=-b}^{b} \hat{f}(x+s, y+t)$

- 进一步可得

$$\sigma^2 = \frac{1}{(2a+1)(2b+1)} \sum_{s=-a}^{a} \sum_{t=-b}^{b} \Big([g(x+s, y+t) - w(x+s, y+t) \cdot \eta(x+s, y+t)] - [\overline{g(x,y)} - \overline{w(x,y)\eta(x,y)}]\Big)^2$$

  - 若令 $w(x+s, y+t) = w(x, y)$
  - 我们有

$$\sigma^2 = \frac{1}{(2a+1)(2b+1)} \sum_{s=-a}^{a} \sum_{t=-b}^{b} \Big([g(x+s, y+t) - w(x, y) \cdot \eta(x+s, y+t)] - [\overline{g(x,y)} - w(x,y)\overline{\eta(x,y)}]\Big)^2$$

最佳陷波滤波算法

- 为了最小化 σ^2，我们要求解方程

$$\frac{\partial \sigma^2}{\partial w(x,y)} = 0$$

- 求得的结果为：

$$w(x,y) = \frac{\overline{g(x,y)\eta(x,y)} - \bar{g}(x,y)\bar{\eta}(x,y)}{\overline{\eta^2}(x,y) - \bar{\eta}^2(x,y)}$$

最佳陷波滤波算法的计算效果

a b

FIGURE 5.20 (a) Image of the Martian terrain taken by *Mariner 6*. (b) Fourier spectrum showing periodic interference. (Courtesy of NASA.)

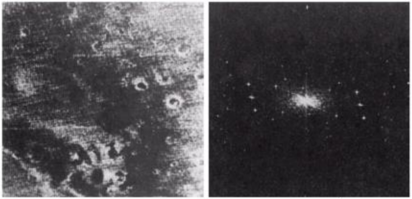
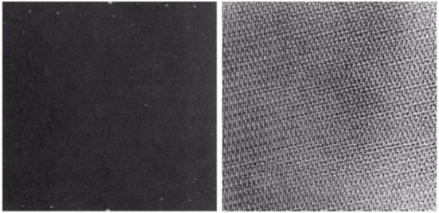
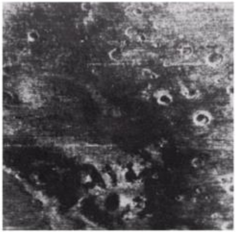


FIGURE 5.21 Fourier spectrum (without shifting) of the image shown in Fig. 5.20(a). (Courtesy of NASA.)



a b

FIGURE 5.22 (a) Fourier spectrum of $N(u, v)$, and (b) corresponding noise interference pattern $\eta(x, y)$. (Courtesy of NASA.)



E 5.23 Processed image. (Courtesy of NASA.)

作业

1. 推导对数高斯分布的随机数发生方程，并用matlab实现对数高斯分布的随机数发生器，然后绘制相关分布的图形。

2. 对公式

$$\hat{f}(x,y) = \frac{\sum_{(s,t)\in S_{xy}} g(s,t)^{Q+1}}{\sum_{(s,t)\in S_{xy}} g(s,t)^{Q}}$$

给出的反调和均值滤波器回答下列问题：

（a）解释为什么当 Q 是正值时滤波对去除“胡椒”噪声有效？

（b）解释为什么当 Q 是负值时滤波对去除“盐”噪声有效？

3. （选做题）研究探索自适应空域中值滤波算法，并提出你的改进想法，分析你的理由。