第3章 空间域图像增强

§ 3.1 概述

§ 3.2 基本灰度变换增强

§ 3.3 直方图处理

§ 3.4 图像平滑

§ 3.5 空间滤波基础

§ 3.6 平滑空间滤波器

§ 3.7 锐化空间滤波器

§ 3.8 混合空间增强法

§3.1 概述

图像增强是数字图像处理的基本技术之一；

图像增强含义：对图像进行加工，改善视觉效果。

图像增强分类：根据处理进行的空间不同，分为二大类：

(1) 基于空间域的方法

(2) 基于频率域的方法

一、基于空间域的方法

在图像处理中，空域是指由像素组成的空间，亦称为图像域法；以点运算(像素)和模板运算(小区域块)为主。

空域增强方法可表示为：

$$g(x,y)=\mathbf{EH}[\,f(x,y)]$$

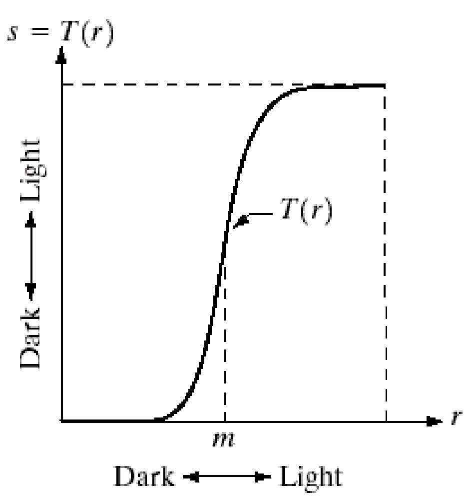
***f*(*x*,*y*)表示增强前的图像，*g*(*x*,*y*)表示增强后的图像；EH表示增强操作，具体可以是逐点操作、模板操作或逐图像操作：**

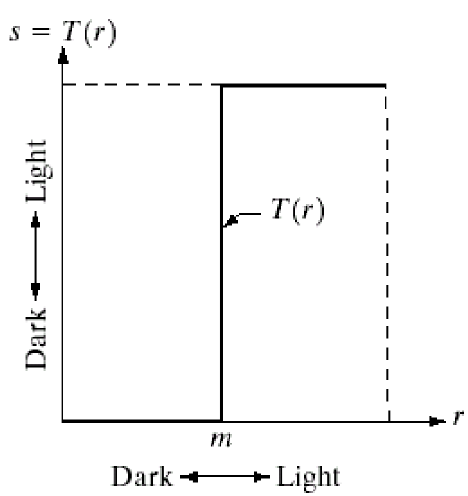
1. 若EH是定义在每个点(*x*,*y*)上的操作， 称EH是点操作。

$$f(x,y)=r \longrightarrow \boxed{T} \longrightarrow g(x,y)=s$$

T **表示增强函数EH**

$$g(x,y)=T[r]$$





***m*为阈值**

图像的点处理操作关键在于设计合适的映射函数(直线或曲线)，映射函数的设计有两类方法：

(1)根据图像特点和处理工作需求，人为设计映射函数，试探其处理效果，如直接灰度变换方法；

(2)从改变图像整体的灰度分布出发，设计一种映射函数，使变换后图像灰度直方图达到或接近预定的形状，如直方图修正方法；

2. 若EH是定义在(x,y)的某个邻域上(常用正方形邻域)，EH称为模板操作或邻域操作。

3. EH既可以操作于一幅图像，还可以作用于一系列图像。

二、基于变换域的方法

主要为频率域法，变换空间采用频域空间，频域空间的增强方法需要进行数学正变换T 和反变换T^{-1} 。

$$g(x,y) = \mathbf{T}^{-1}\{\ \mathbf{EH}\ [\ \mathbf{T}[\ f\,(x,y)]\ ]\ \};$$

正变换T： 把图像从图像空间转换到频率空间；

增强处理：EH对频率空间的数据进行操作；

反变换T^{-1}：把图像从频率空间转换回图像空间；

频域图像增强在第四章介绍。

其它有关图像增强分类的方法：

如若按处理策略分类，图像增强可以分为全局处理和局部处理两类。

如按处理对象分类，图像增强可以分为灰度图像增强和彩色图像增强两类。

§ 3.2 基本灰度变换增强

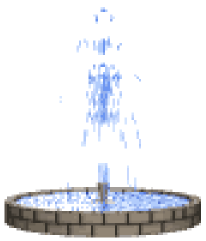
灰度变换是指将一个灰度区间映射到另一个灰度区间的变换过程，是图像增强的重要手段之一。

通过灰度变换可使图像动态范围加大(拉伸)，图像对比度扩展，使图像更清晰，特征更明显。

灰度变换方法的分类：

(1)线性灰度变换方法；

(2)非线性灰度变换方法；



灰度变换增强的一般形式：

(1) $s = \mathrm{EH}(r)$； 或 $s = T(r)$；

r, s分别表示增强前后在像素(x,y)处的灰度值；属于点处理。

(2) $s = \mathrm{EH}[r, n(r)]$；

$n(r)$表示原图像$f(\bullet)$在$r(x,y)$的邻域内像素的灰度值；

采用模板(mask)与图像卷积来实现邻域内的增强操作，属于邻域处理。

线性灰度变换

对输入图像灰度作线性扩张或收缩，映射函数为一个直线方程，其表达式如下：

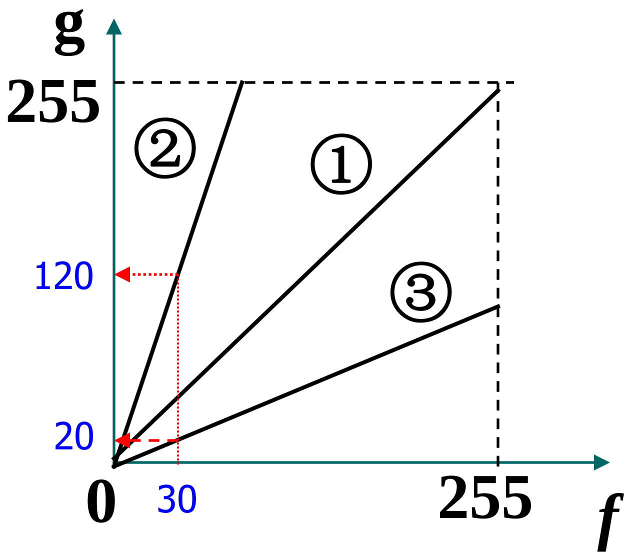
$$g(x,y)=af(x,y)+b$$

其中：a 相当于变换直线的斜率， b 相当于截距；

$$b=0;\quad \begin{cases} a>1\text{——对比度扩张} \\ a<1\text{——对比度收缩} \\ a=1\text{——相当于复制} \end{cases}$$

$$b\neq 0;\quad \text{灰度偏置，图像整体变亮或变暗}b\text{值；}$$

以下是一些灰度变换的例子。 g=EH(*f*)。

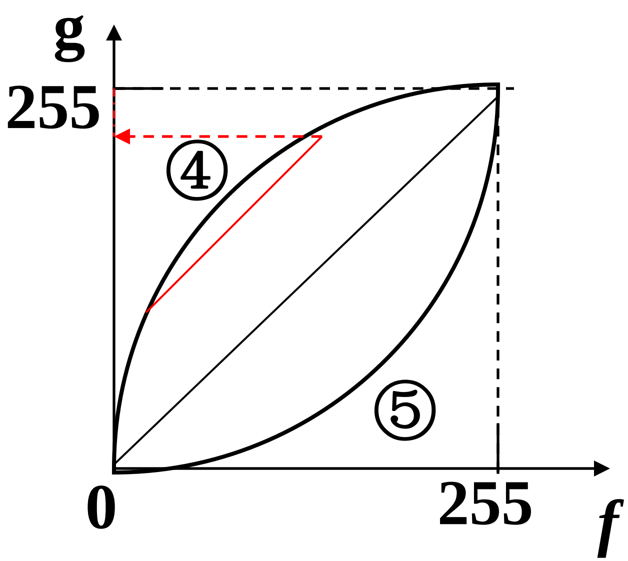


① 表示g= *f*，灰度不变。(a=1)

② 表示灰度均匀变亮。(a>1)

③ 表示灰度均匀变暗。 (a<1)

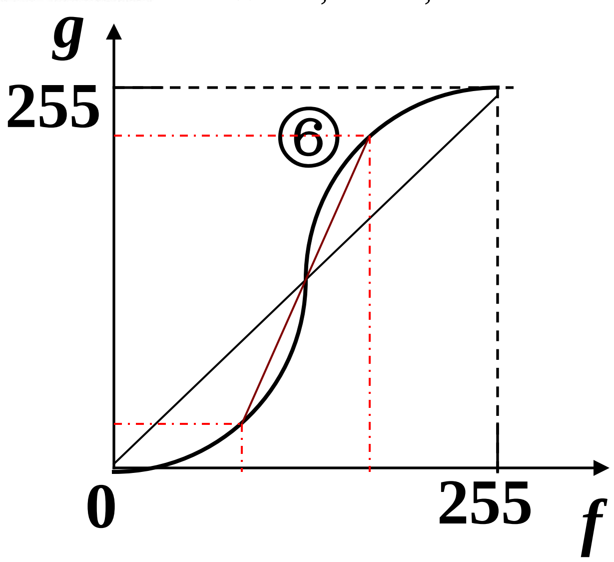
① ②和③为线性变换。



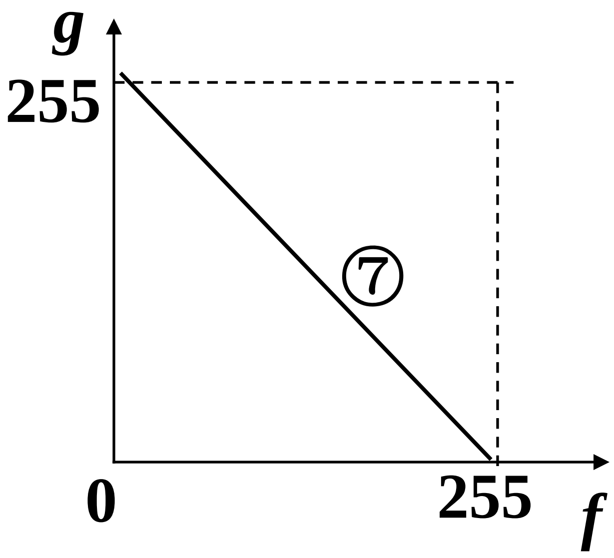
④ 暗区灰度扩张，亮区压缩。

⑤ 暗区压缩，亮区扩张。

④⑤和⑥为非线性变换。



⑥ 中间灰度拉伸，其余压缩。



⑦ 灰度求反，生成负片。

§ 3.2.1 图像反转

图像反转

黑白倒置，阴图(正片) --> 阳图(负片)；

变换函数： **g(x,y) = - f (x,y) + 255；a = -1; b=255;**

实例：

§ 3.2.2 对数变换——动态范围伸缩

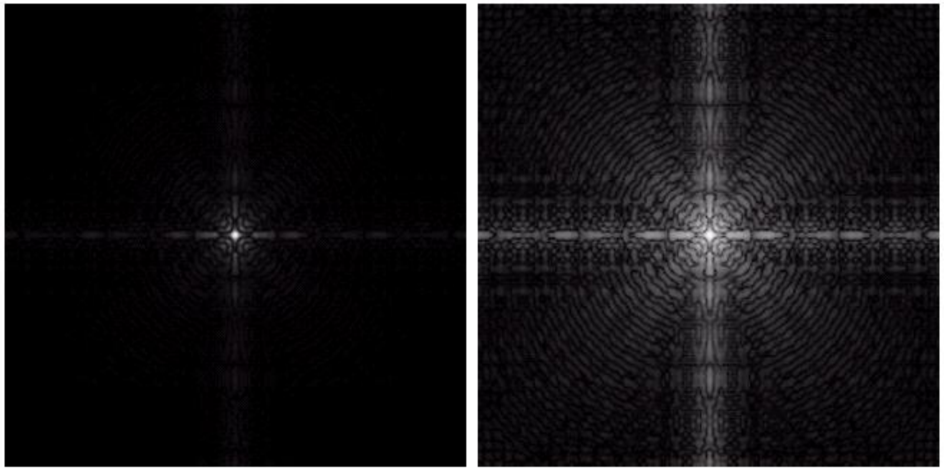
对数变换增强属于非线性灰度变换增强(视觉模型)。

对数形式的EH： $g(x,y)= c * \log(1+f(x,y))$， $f(x,y)$为像素点的灰度；依据图像模型，这种增强符合人的视觉特性。

一般形式：

$$g(x,y)=\frac{b'-a'}{\log b-\log a}(\log f(x,y)-\log a)+a'$$

对数变换增强使低灰度范围得到扩展，高灰度范围得到压缩。



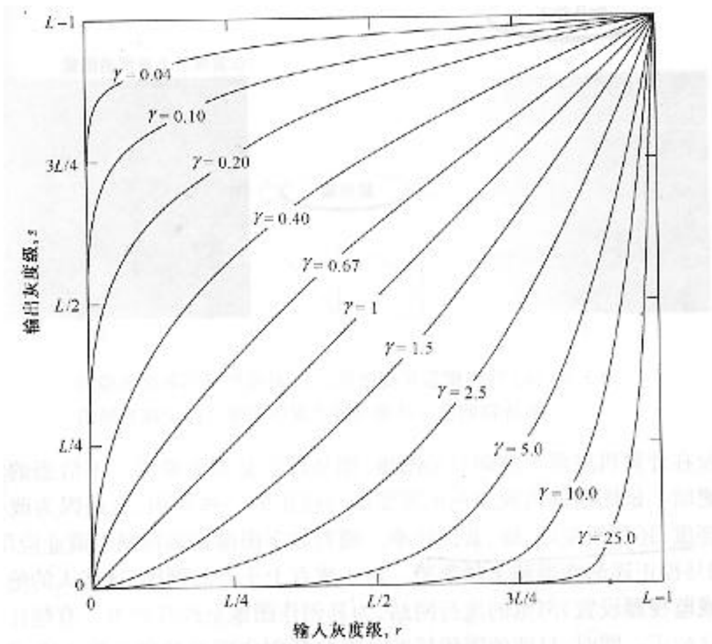
a', b', a, b分别是变换前后的灰度范围

§ 3.2.3 幂次变换

幂次变换增强属于非线性灰度变换增强。

指数形式的EH： $g(x,y)=c\,f(x,y)^{\gamma}$；也称为伽马校正。

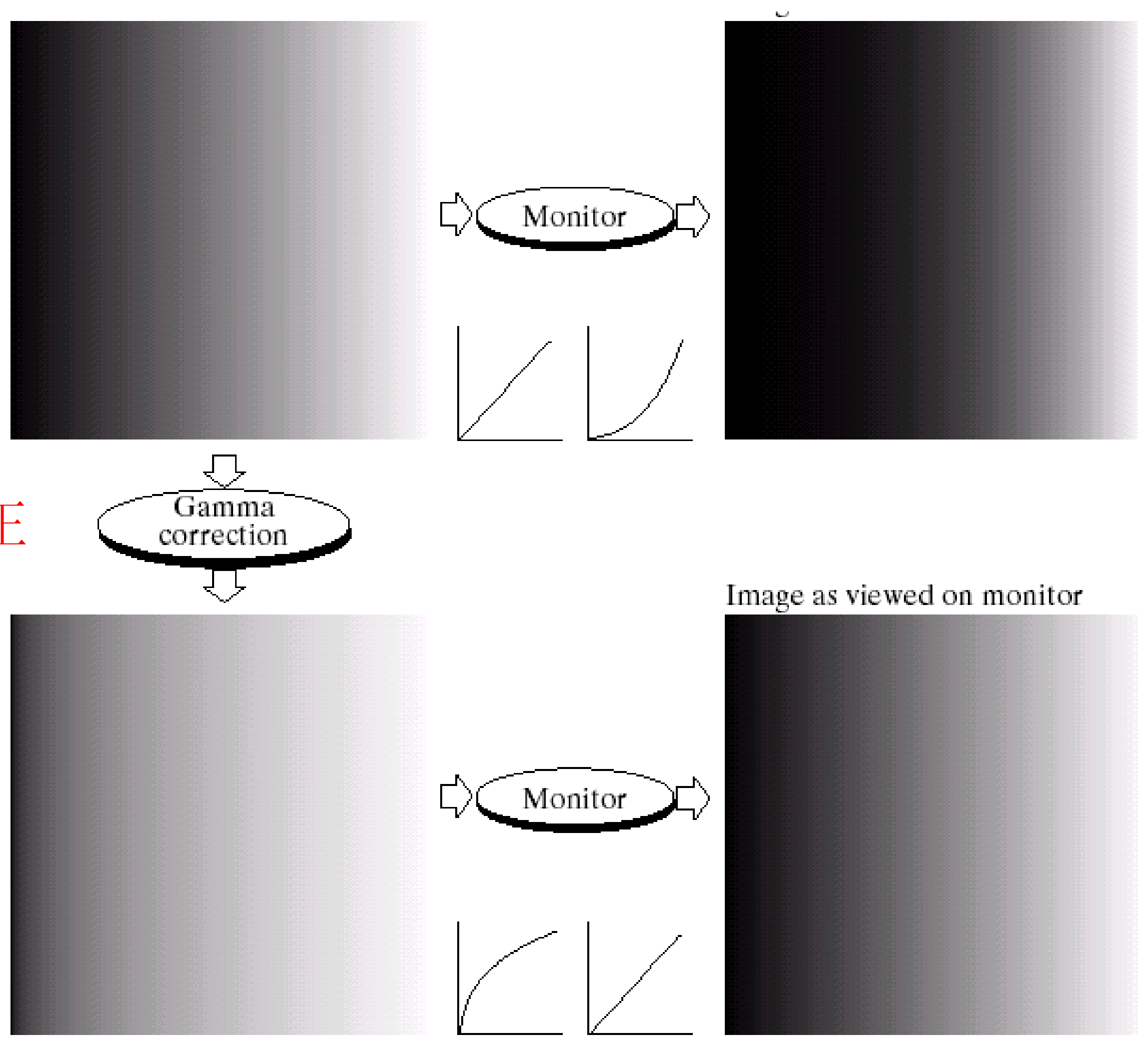
$\gamma < 1$时低灰度范围得到扩展， $\gamma > 1$时效果相反。



幂次变换增强实例：显示器灰度渐变图像校正。

CRT是电压-强度响应装置，属指数变化为2.5的幂函数。

伽马校正



图像输入到CRT前先校正，进行1/2.5=0.4的幂变换。

§ 3.2.4 分段线性变换函数

1. 线性变换： $g(x,y)=af(x,y)+b$ **；**

2. 分段线性变换：突出(拉伸)感兴趣的灰度区间，相对抑制(压缩)不感兴趣的灰度区间 (Contrast stretching)；

分段线性变换公式为：

$$g = g_a + \frac{g_b - g_a}{f_b - f_a} * (f - f_a)$$

f_a、f_b **和** g_a、g_b **分别为变换前后线段的端点处灰度值；**

f **表示原像素灰度，** ***g*** **表示增强后像素灰度。**

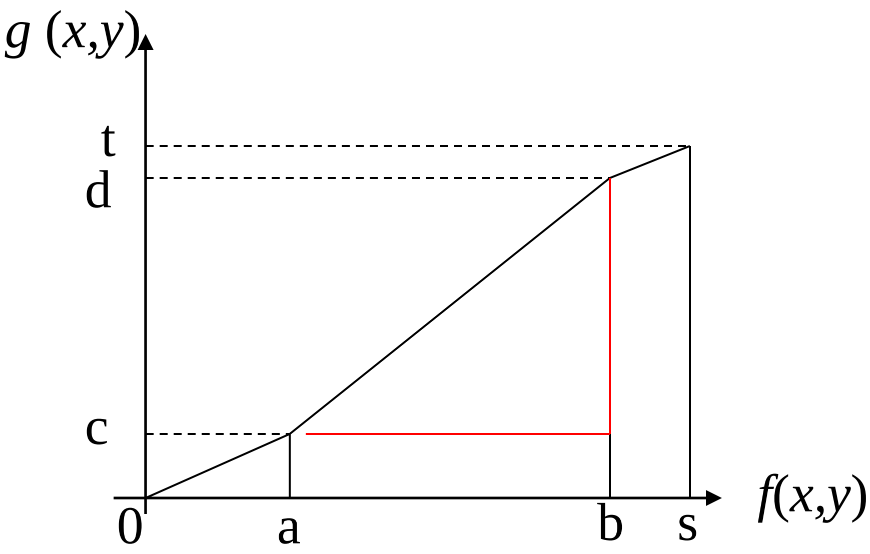
若图像中大部分像素的灰度级在[*a*, *b*]范围内，少部分像素分布在小于*a*和大于*b*的区间内。此时可用下式作变换：(窗口变换，特殊的分段线性变换)

原始图像：*f* (*x*,*y*)，灰度范围[*a*, *b*]，

变换后图像：*g* (*x*,*y*)，灰度范围变为[*a'*, *b'*]，

$$g(x,y)=\begin{cases} a';\ \underline{f(x,y)<a} \\ a'+\dfrac{b'-a'}{b-a}(f(x,y)-a);\ \underline{a\le f(x,y)<b} \\ b';\ \underline{f(x,y)\ge b} \end{cases}$$

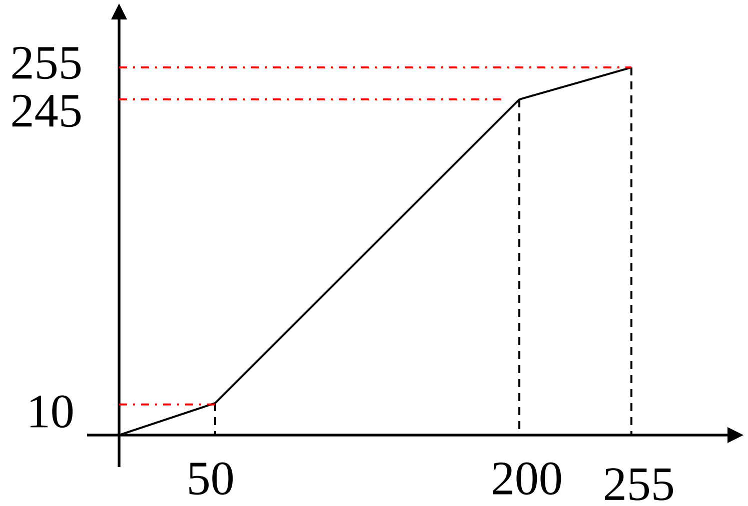
3 . 三段线性变换图示与公式（三段线性变换方法较常用）



计算公式：

$$g(x,y)=\begin{cases}(c/a)*f(x,y)\\ c+\dfrac{(d-c)}{(b-a)}*(f(x,y)-a)\\ d+\dfrac{(t-d)}{(s-b)}*(f(x,y)-b)\end{cases}$$

分段线性变换举例：在对比度增强中，若要把灰度范围[0,50]压缩成[0,10]，把灰度[50,200]扩展成[10,245]，把灰度[200,255]压缩成[245,255]，试写出其分段线性变换方程。



解：按分段式公式可得：

当$0 \le f(x,y) < 50$，

$$g_1(x,y) = 10/50*f(x,y) = 1/5*f(x,y) = \underline{0.2*f(x,y)};$$

当$50 \le f(x,y) \le 200$，

$$g_2(x,y) = (245-10)/(200-50)*[f(x,y) - 50]+10;$$

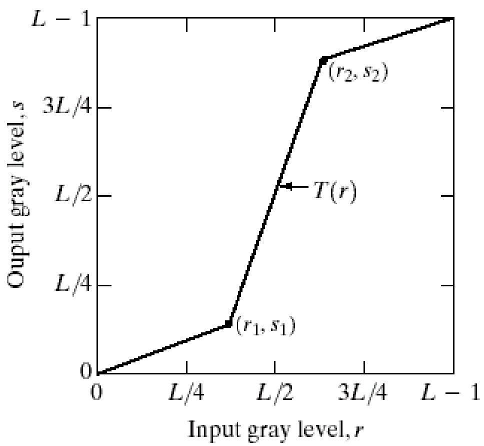
$$= (47/30)*f(x,y) - 205/3 = \underline{1.57*f(x,y) - 68.3};$$

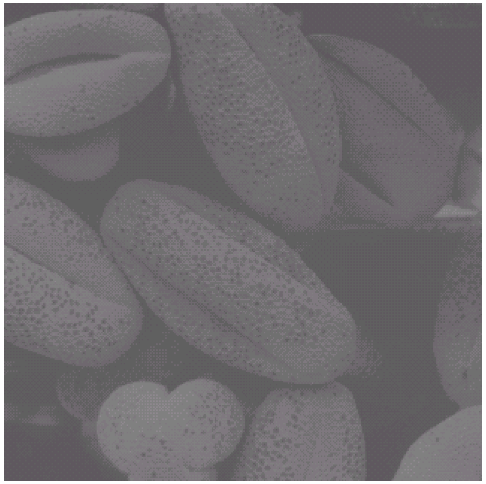
当$200 \le f(x,y) \le 255$，

$$g_3(x,y) = (255-245)/(255-200)*[f(x,y)- 200] + 245;$$

$$= (2/11)*f(x,y) + 2295/11 = \underline{0.182*f(x,y) + 208.6};$$

三段式增强实例

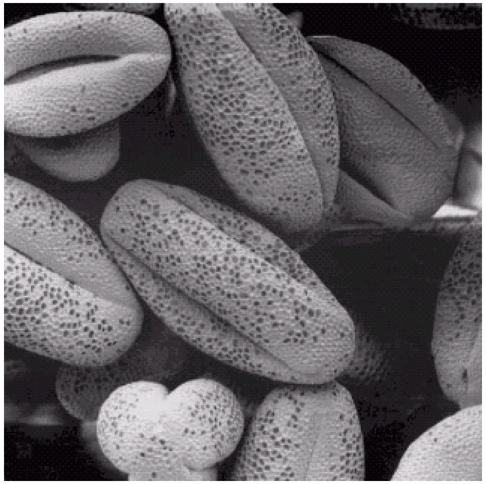




源图像

对比度低

增强图像



二值图像

§ 3.3 直方图处理

直方图Histogram的概念。

直方图修正以概率统计为基础进行增强处理。

直方图的定义：设n表示图像f 的总像素数；

图像的灰度统计直方图定义为 $P(r_k)=\frac{n_k}{n}$ ；

$$\sum_{k=0}^{L-1} P(r_k)=1$$

式中，r_k表示图像第k个灰度级， n_k表示图像中具有灰度值r_k的像素数；L为灰度级个数；直方图是1D的离散函数。

即$p(r_k)$表示该灰度级的相对频数；

图像直方图描述了图像中各灰度级出现的相对频率。

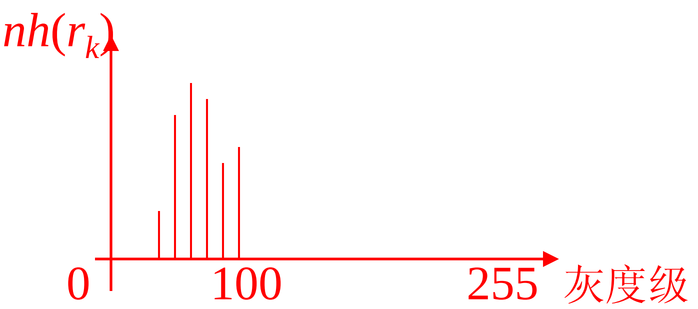
直方图画图：以横轴表示直方图的灰度级，纵轴表示对应灰度级的像素个数，逐个灰度级画出就可得到图像的直方图；

直方图提供了原图的灰度值分布情况和整体描述；

通过改变直方图的形状可以达到增强图像对比度的效果。

计算直方图的C程序段代码： int nh[255];//计数器数组

```
for(x=0; x<N-1; x++)
    for(y=0; y<N-1; y++)
        ++nh[f(x,y)];
```



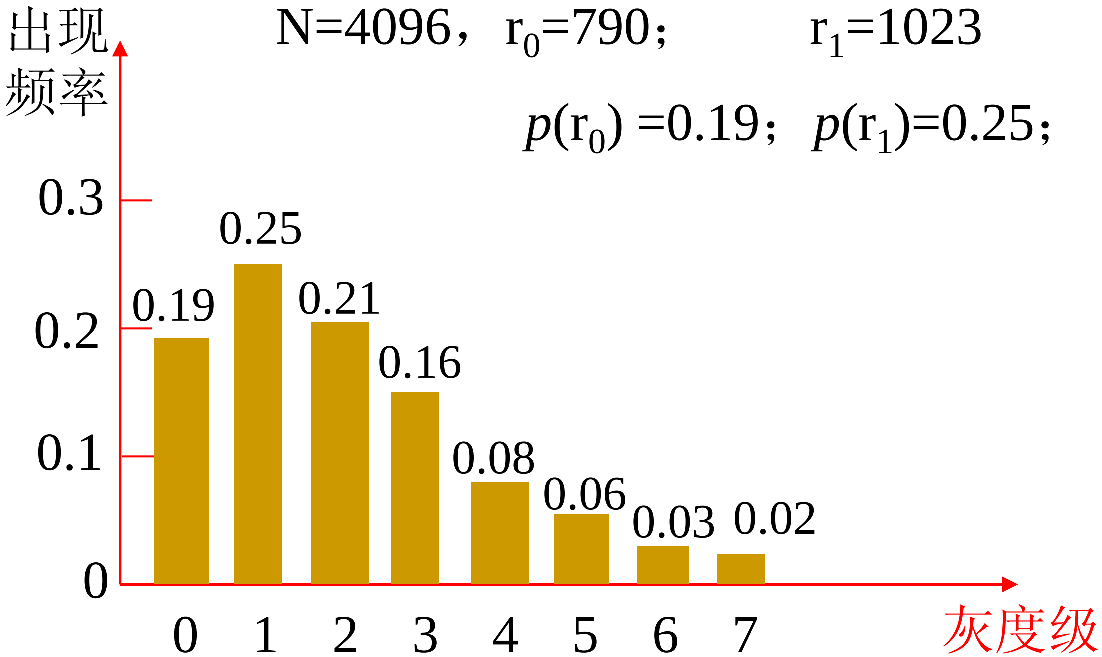
Matlab中函数imhist(*f*)；

直方图举例：

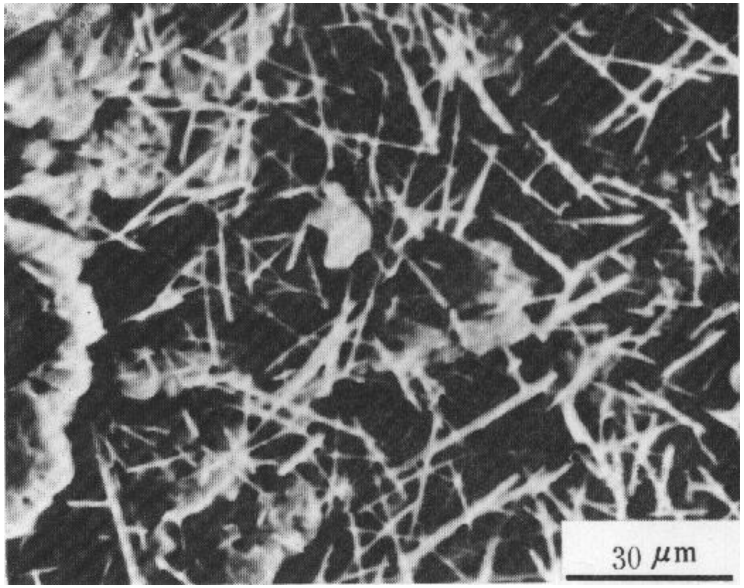
设有1幅64×64，8个灰度级的图像，其直方图如下：

N=4096，r_0=790； r_1=1023

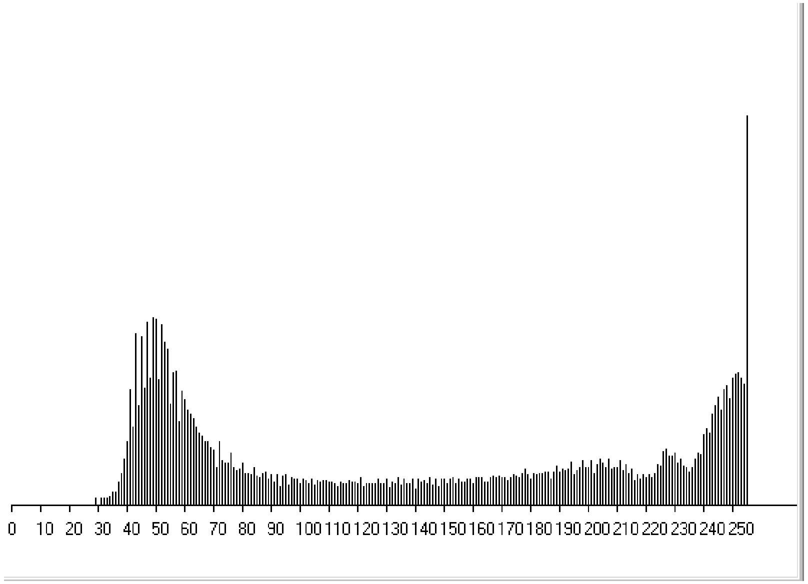
$p(r_0)$ =0.19；$p(r_1)$=0.25；



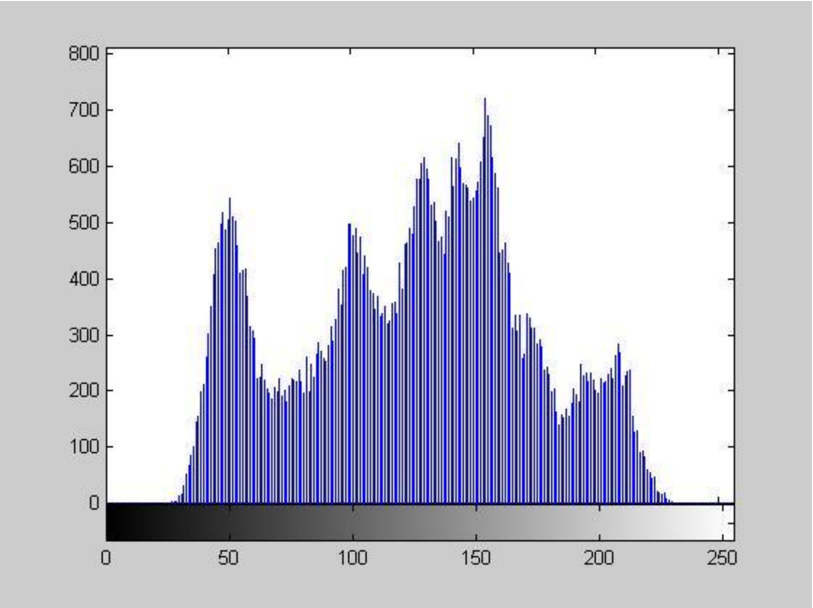
直方图



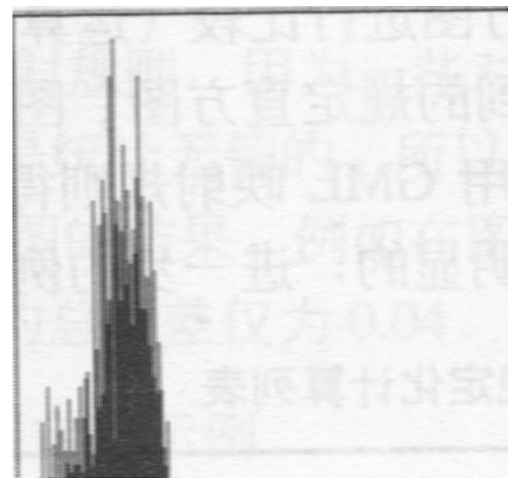
微观结构图



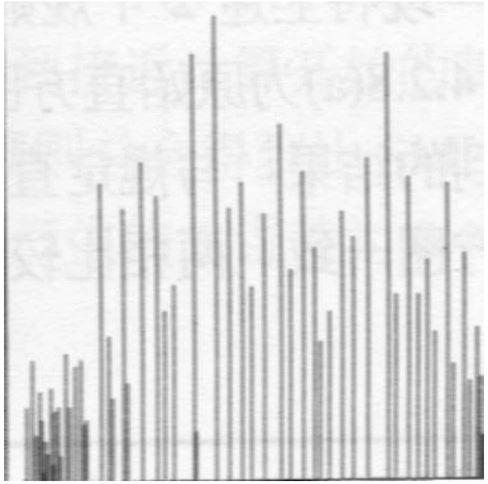
左图对应的直方图


800
700
600
500
400
300
200
100
0
0
50
100
150
200
250

基于直方图的灰度变换, 是调整图像直方图到一个预定的形状。例如, 一些图像由于其灰度分布集中在较窄的区间，对比度很弱, 图像细节看不清楚。



此时, 可采用图像灰度直方图均衡化处理, 使得图像的灰度分布趋向均匀, 图像所占有的像素灰度间距拉开, 加大图像反差, 改善视觉效果, 达到增强目的。



直方图均衡处理后，直方图中灰度占据了整个图像灰度值的允许范围，增加了图像灰度的动态范围，细节清楚了。

§3.3 直方图处理

§3.3.1 直方图均匀化

直方图修正常用的方法有:

1)直方图均匀化、2)直方图规定化;

直方图均匀化的含义：把原始图像的直方图变换为均匀分布的形式，让各灰度级出现的概率相同，由不均匀变得均匀；

一、直方图均匀化处理的思路

将原始图像灰度级 r 归一化在 $0 \sim 1$ 之间，即

$$0 \le r \le 1$$

设 $p_r(r)$ 为原始图像灰度分布的概率密度函数；

直方图均匀化处理实际上就是寻找一个灰度变换函数 $\boldsymbol{T}$，使变化后的灰度值 $s=T(r)$ ，且归一化为 $0\le s\le 1$ ；

再建立原像素 r 与新像素 s 之间的映射关系；

主要的要求是：处理后图像灰度分布的概率密度函数均为 $p_s(s)=1$ ，即期望处理后所有灰度级出现的概率基本相同。

二、过程分析：先考虑连续情况

设非均匀概率密度函数为$p_r(r)$、变换函数为$T(r)$ (单值单增)；均匀概率密度函数为$p_s(s)$，r, s分别为变换前后的灰度值，并归一化为$0 \le r,s \le 1$；寻求变换函数$S(s)$。

在灰度变换的dr和ds区间内，像素点个数是不变的，因此有：

$$\int_{r_j}^{r_j+dr} p_r(r)dr = \int_{s_j}^{s_j+ds} p_s(s)ds$$

等式成立

去掉下标j，当 $dr \to 0$ 、$ds \to 0$ 时，$\frac{ds}{dr} = \frac{p_r(r)}{p_s(s)}$ ，

由于希望 $p_s(s) = 1$ ，

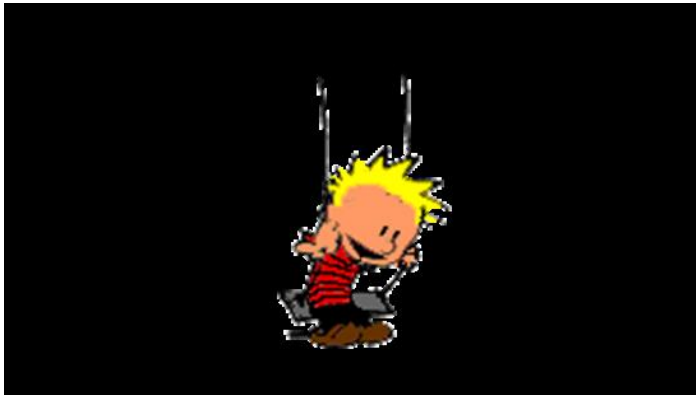
所以，分析得到直方图均匀化的灰度变换函数为：

$$S = S(s) = T(r) = \int_{0}^{T} p_r(r)dr$$

***S*(*s*)是原始图像的一个累计分布函数 CDF(Cumulative Distribution Function)；**

条件：单值单增，保证灰度从0到L； $0 \le T(r) \le \mathrm{L}-1$

L为灰度级个数；



分析离散情况。总像素为N，L个灰度级；积分变成求和；

$\because\ p_r(r_k) = n_k/N$ (1)

$s_k = T(r_k) = \sum p_r(r_j)$ ，$j = 0,\ldots,k$，将(1)式代入，得

$$\therefore \quad s_k = T(r_k) = \sum_{j=0}^{k} \frac{n_j}{N}, \quad k = 0,1,\ldots\ldots,L-1 \qquad (2)$$

即第k个新的灰度级等于原图像中前k个n_j/N 的累计和；

当累积和与某均匀灰度级最接近时，作为均衡化的灰度级。

逆变换 $r_k = T^{-1}(s_k)$ ； $0 \le s_k \le 1$ ；

三、直方图均匀化变换过程举例

设一幅图像64*64=4096个像素(即N=4096)，共有8个灰度级(0-7)；

分布情况如下：

灰度级 r_k	$r_0=0$	$r_1=1/7$	$r_2=2/7$	$r_3=3/7$	$r_4=4/7$	$r_5=5/7$	$r_6=6/7$	$r_7=1$
像素数 n_k	790	1023	850	656	329	245	122	81
概率$p_r(r_k)$	0.19	0.25	0.21	0.16	0.08	0.06	0.03	0.02

均匀化过程 $s_k = \mathrm{T}(r_k) = \sum p_r(r_j)$，$j = 0,\ldots,k$

原灰度级	变换函数 $T(r_k)$ 的值	像素数	量化级	新灰度级	新灰度级分布
$r_0=0$	$T(r_0)= s_0=0.19$	790	$r_0=0.0$		
$r_1=1/7=0.14$	$T(r_1)= s_1=0.19+0.25=0.44$	1023	$r_1=0.14$	s_0' (790)	790/4096=0.19
$r_2=2/7=0.29$	$T(r_2)= s_2=0.44+0.21=0.65$	850	0.29		
$r_3=3/7=0.43$	$T(r_3)= s_3=0.65+0.16=0.81$	656	0.43	s_1' (1023)	1023/4096=0.25
$r_4=4/7=0.57$	$T(r_4)= s_4=0.81+0.08=0.89$	329	0.57		
$r_5=5/7=0.71$	$T(r_5)= s_5=0.89+0.06=0.95$	245	0.71	s_2' (850)	850/4096=0.21
$r_6=6/7=0.86$	$T(r_6)= s_6=0.95+0.03=0.98$	122	0.86	s_3' (985)	985/4096=0.24
$r_7=1$	$T(r_7)= s_7=0.98+0.02=1$	81	1	s_4' (448)	448/4096=0.11

计算过程说明及结果分析：

原r_k是等间隔的，新灰度级取$T(r_k)$累计值与原灰度级中最接近的；

如$T(r_1)=0.44$，与$r_3=3/7=0.43$靠近，得到新的灰度级，s_1' (1023)；

如$T(r_2)=0.65$，与$r_5=5/7=0.71$靠近，得到新的灰度级，s_2' (850)；

如$T(r_3)=0.81$，$T(r_4)=0.89$，与$r_6=6/7=0.86$靠近，合起来得到一个新的灰度级，s_3' (656+329=985)；$p_s(s_3')=985/4096=0.24$；

如$T(r_5)=0.95$，$T(r_6)=0.98$，$T(r_7)=1.0$，与$r_7=7/7=1.0$靠近，合起来得到一个新的灰度级，s_4' (245+122+81=448)；

$p_s(s_4')=448/4096=0.11$，映射关系为原灰度级5、6、7 -> s_4'灰度级

结果分析：新的灰度分布概率较原先要均匀一些；

直方图均匀化处理算法的计算步骤小结:

1. 统计原始图像的直方图 $p_r(r_k)=\dfrac{n_k}{n}$

r_k 是归一化的输入图像的原灰度级;

2. 计算直方图累积分布曲线

$$s_k=T(r_k)=\sum_{j=0}^{k}p_r(r_j)=\sum_{j=0}^{k}\frac{n_j}{n}$$

3. 用累积分布函数作为变换函数进行图像灰度变换，建立输入图像与输出图像灰度级之间的对应关系，得到新灰度级图像。

四、实验举例

1. 变换后直方图趋向平坦，灰级减少，灰度合并。

2. 原始图像零灰度级像素个数较多，变换后零灰度级消失，含有像素数多的几个灰级间隔被拉大了，压缩的只是像素数少的几个灰度级，实际视觉能够接收的信息量却大大地增强了。

§ 3.3.2 直方图规定化

直方图规定化的含义：改变直方图使之成为某个指定的形状。

采用比较灵活的直方图规定化方法，正确地选择规定化的函数，可获得比直方图均衡化更好的结果。

方法一：用一个规定的概率密度函数(变换函数)来表示直方图；

1. 均匀分布 概率密度函数 $P_s=1/(S_{max}-S_{min})$，

变换函数 $S=S_{min}+[S_{max}-S_{min}]P_r(r)$

2. 指数分布 概率密度函数 $P_s=\alpha\exp[-\alpha(S-S_{min})]$，

变换函数 $S=S_{min}-(1/\alpha)\ln(1-P_r(r))$；

方法二：控制一组直线段来构成直方图，使其满足希望的形状。

实现步骤如下：

step 1 对原始图的直方图进行灰度均匀化

$\mathbf{S=T(}r\mathbf{)} = \int p_r(w)\mathrm{d}w$； 连续情况

或 $s_k = \mathrm{EH_r}(r_i) = \sum p_r(\mathrm{r_i})$，$i = 1,\ldots,k$； 离散情况

step 2 给出规定的直方图，并计算直方图均匀化

$\mathbf{V=G(}v\mathbf{)} = \int p_u(w)\mathrm{d}w$； 连续情况

或 $v_l = \mathrm{EH_u}(u_j) = \sum p_u(u_j)$，$j = 1,\ldots,l$； 离散情况

step 3 将原始直方图对应映射到规定的直方图

映射方法有单映射和组映射两种；

直方图规定化举例(灰度数据同前例):

step 1 原始图的均匀化概率分布情况

$P_r(r_i)$	0.19	0.25	0.21	0.16	0.08	0.06	0.03	0.02
	0.19	0.44	0.65	0.81	0.89	0.95	0.98	1.00

step 2 增强图的规定概率分布情况及均匀化

$p_v(u_i)$	0.00	0.00	0.00	0. 20	0.00	0.60	0.00	0.20
	0.00	0.00	0.00	0.20	0.20	0.80	0.80	1.00

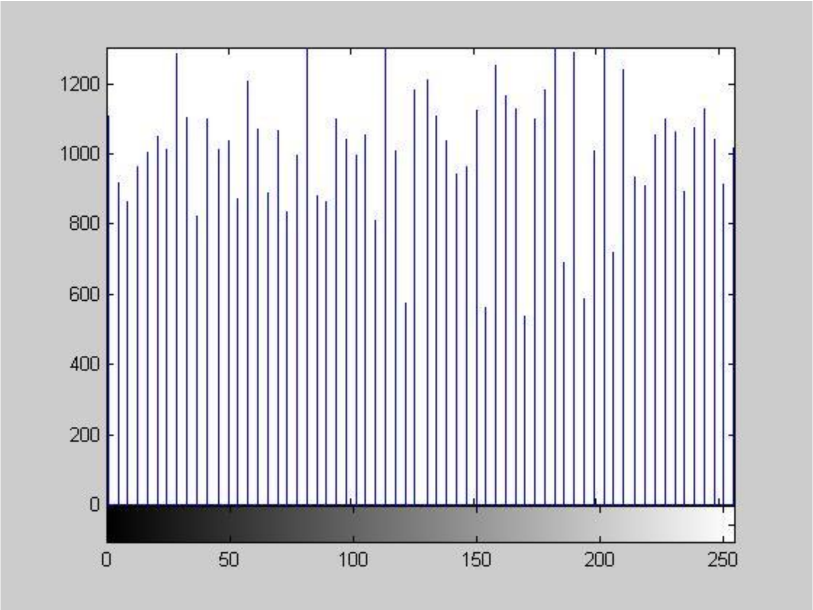
step 3 映射（成组映射）

$\sum r_i = 0.19$； 与0.2靠近，r_0 映射到v_3;

$\sum r_i = 0.19+\underline{0.25+0.21+0.16} = 0.81$ 与0.2+0.6靠近，$r_1r_2r_3$ 映射到v_5;

$\sum r_i = 0.81+\underline{0.08+0.06+0.03+0.02} = 1$； 与0.2 +0.6+0.2靠近，$r_4r_5r_6r_7$ 映射到v_7;

比较得出的直方图和规定的直方图，两者很接近。


1200
1000
800
600
400
200
0
0
50
100
150
200
250

§3.4 图像平滑

图像平滑的目的：

情况1：去掉图像中太小的细节，提取较大的目标；但可能会造成图像模糊；

情况2：消除噪声，达到图像增强的目的；

噪声分类：热噪声（电磁、闪电等引起的噪声）；

散粒噪声或颗粒噪声，斑点；干扰引起的噪声；

一般属于加性噪声，均值为0，互不相关，方差为σ^2；

信噪比SNR：含噪图像的均值与噪声方差σ^2之比；

图像中的脉冲噪声模型

- **椒盐噪声(Salt-Pepper Impulsive Noise)**

  受噪声干扰的图像像素以50%的相同概率等于图像灰度的最大(盐-白)或最小(胡椒-黑)的可能取值;

- **随机值脉冲噪声**

  受噪声干扰图像点取值均匀分布于图像灰度的最大与最小可能取值之间；一般采用高斯白噪声。

图像中的加噪举例

(a) 原始图像　　(b) 3%椒盐噪声　　(c) 3%随机值脉冲

J=imnoise(I,type,paramater);　　%加噪函数

类型type有'salt & pepper'和'gaussian'等;

图像平滑方法（去噪）方法分类：

1）空间域方法　　频率域方法

2）全局处理　　局部处理

3）线性平滑　　非线性平滑

4）自适应平滑

图像平滑函数（邻域处理）

$$g(x,y)=\phi\left[x,y,f\left(x',y'\right):\left(x',y'\right)\in N(x,y)\right]$$

结果图像 像素灰度值 邻域像素集合

如 $$g(x,y)=\frac{1}{N}\sum_{i=-(M+1)}^{M+1}\sum_{j=-(M+1)}^{M+1}f(x+i,y+j)h(i,j)$$

结果图像 邻域像素灰度值 加权系数

设邻域模板尺寸取 $(2M+1)\times(2M+1)$ ，**f(x,y)**为模板中心像素。

若 $M=1$ 即 3×3 模板。

§3.4.1 图像平均处理-邻域平均法

一、邻域平均法的计算及分析

平滑后图像中的每个像素为原图像预定邻域像素灰度的平均值；是一种局部空域方法。

$$g(x,y)=\frac{1}{N\times N}\sum_{(x,y)\in A}f(x,y)$$

式中，**$(x,y)\in A$，A**是中心点**(x,y)**邻域坐标集合；M=N*N为A内坐标点的总数；

例：M=4时，

$$\begin{array}{ccc} & + & \\ + & + & + \\ & + & \end{array}$$

中心**(x,y)**

A={$(x-1,y)$, $(x,y-1)$, $(x+1,y)$, $(x,y+1)$};

优点：算法简单，计算速度快；

缺点：图像会产生模糊，邻域半径越大，模糊程度越大(边缘和细节被模糊)；

邻域平均法在一定程度上抑制噪声，但是邻域平均法的平均作用会引起模糊现象，模糊程度与邻域半径成正比。

改进方法：采用阈值法，引进一个非负阈值；

当某些点和它们邻域的差值不超过阈值时，不进行平滑处理；

当超过该阈值时，必然是噪声，需要进行平滑处理。

超限邻域平均法 (算法思路与公式)

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。

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

T为某一给定的阈值。


(a) 3%椒盐噪声干扰的噪声图像

(b)3%随机值脉冲噪声干扰的噪声图像


用3×3大小窗口邻域平均法
对(a)图进行平滑

用3×3大小窗口邻域平均法
对(b)图进行平滑


用3×3窗口超限邻域平均法
对(a)图进行平滑

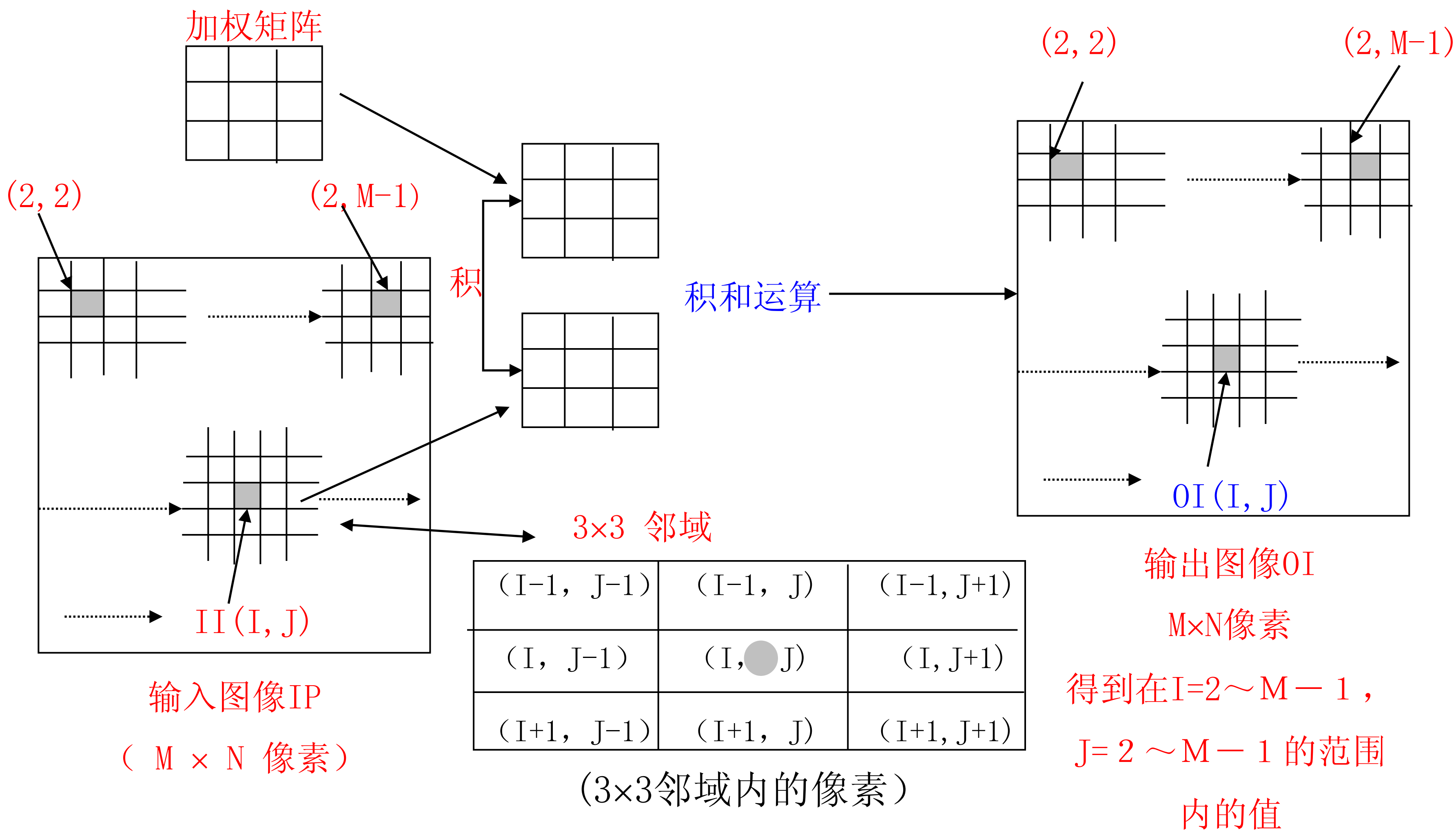
3×3窗口超限邻域平均法
对(b)图进行平滑

可以看出，“超限邻域平均法”比一般邻域平均法的效果要好;

在操作中对窗口的大小及门限的选择要慎重，T 太小，噪声消除不干净；T 太大，易使图像模糊。

在实际应用中，一般用3×3窗口，而且还可以对邻域中各个像素乘以不同的权重，然后再平均，由此可得到不同的加权矩阵。

$$\boldsymbol{H}_1 = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \boldsymbol{H}_2 = \frac{1}{10}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \boldsymbol{H}_3 = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$



采用3×3邻域的邻域处理与3×3加权矩阵的积和运算

局部平滑的降噪能力分析:

假设 $f(x,y) = f'(x,y) + n(x,y)$

无噪图像 噪声图像(均值为0、σ^2)

局部平滑图像,

$$\bar{g}(x,y) = \frac{1}{N}\sum_{i=-M}^{M}\sum_{j=-M}^{M}[f'(x,y)+n(x,y)]$$

$$= \frac{1}{N}\sum_{i=-M}^{M}\sum_{j=-M}^{M} f'(x,y) + \frac{1}{N}\sum_{i=-M}^{M}\sum_{j=-M}^{M} n(x,y)$$

因为噪声图像均值为0, $E\left\{\frac{1}{N}\sum_{i=-M}^{M}\sum_{j=-M}^{M} n(x,y)\right\} = 0$

所以，平滑后图像的均值和方差分别为

$$E\{g(x,y)\}=E\left\{\frac{1}{N}\sum_{i=-M}^{M}\sum_{j=-M}^{M}f'(x,y)\right\}=\bar{f}(x,y)$$

$$D\{g(x,y)\}=D\left\{\frac{1}{N}\sum_{i=-M}^{M}\sum_{j=-M}^{M}f(x,y)\right\}=\frac{\sigma^2}{N}$$

结论：平滑像素后的图像方差，降为原来的1/N (N为区域内像素的总数)，波动范围减小。

二、与邻域平均法相关的其他一些方法

图像平滑在去除噪声的同时，也将图像本身变模糊。如何区分开图像与噪声，加大对噪声平滑力度，维持图像本身不变或少变，是一个感兴趣的研究内容。

1. 超限像素平滑（Out range pixel smoothing）方法：

(见前超限邻域平滑法，取一阈值作为门限)

2. K最近邻法：与中心像素灰度接近的K个像素灰度求平均。

一般， 3x3窗口，K=6。

3. 在窗口中划分子窗口,将方差最小子窗口像素取均值。

4. 二值图像时的平滑方法（有利于区分目标与非目标）

二值图像是目标为“1”，背景为“0”两个灰度的图像。目标区域可能混入个别的背景像素点或小区，即在目标图像中出现一些为“0”的单点或空洞；背景区域也可能出现个别的目标像素点或小区。这些都相当于噪声干扰，会影响后续的特征提取和识别。

平滑去噪的过程是：

填充单点空洞：将8邻域都为“ 1 ”的中心像素赋“ 1 ”

收缩时：将像素八邻域全为“ 1 ”的像素位置赋值“ 1 ”

扩张时：将位置值为“ 1 ”的 8 邻域像素全赋值为“ 1 ”

5. 采用图像邻域操作进行图像平滑属于低层像素级处理，是一种简单重复、处理数据量大的费时操作。如果用软件完成，可能很难满足实时性的要求。

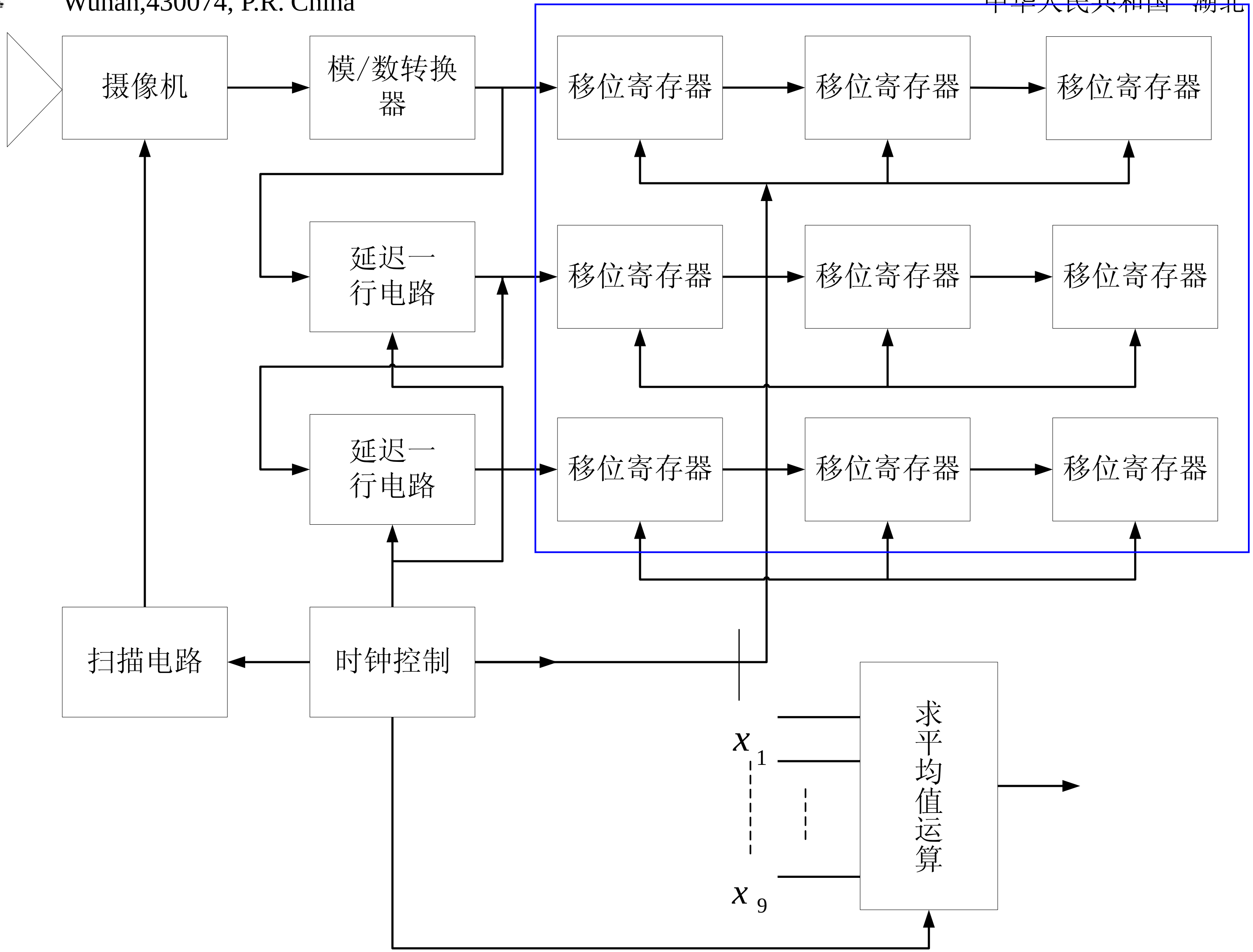
因此可设计硬件电路，按像素时钟频率实时完成平滑任务。

邻域平均法电路框图和邻域平均运算电路框图如下：

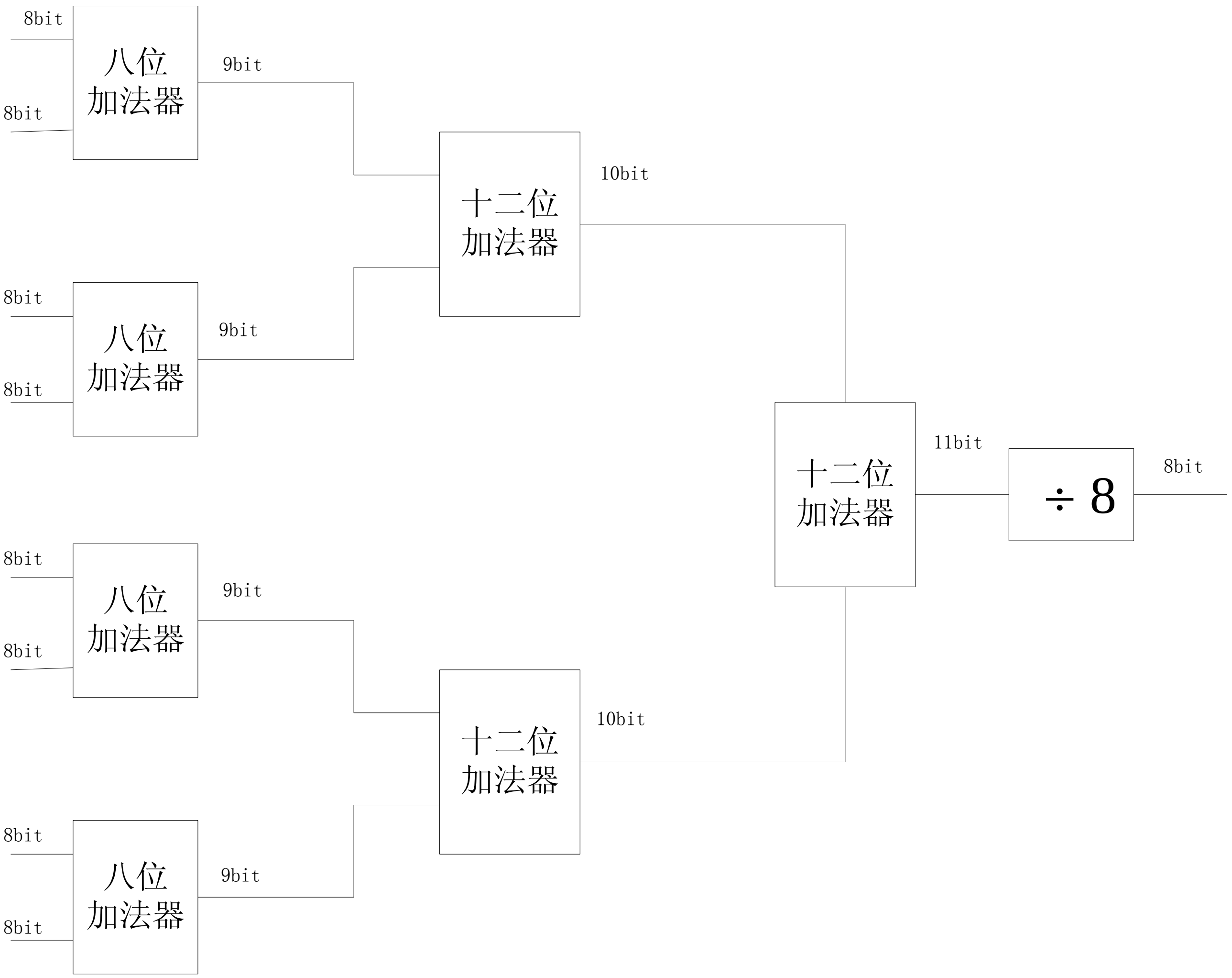
考虑以下8邻域平均，

$$\frac{1}{8}\begin{bmatrix}1 & 1 & 1\\ 1 & 0 & 1\\ 1 & 1 & 1\end{bmatrix}$$

要形成3*3窗口，必须有延迟电路，(因逐行扫描的原因)；


摄像机
模/数转换器
移位寄存器
移位寄存器
移位寄存器
延迟一行电路
移位寄存器
移位寄存器
移位寄存器
延迟一行电路
移位寄存器
移位寄存器
移位寄存器
扫描电路
时钟控制
x_1
x_9
求平均值运算

邻域平均法电路框图


8bit
8bit
八位
加法器
9bit
8bit
8bit
八位
加法器
9bit
十二位
加法器
10bit
8bit
8bit
八位
加法器
9bit
8bit
8bit
八位
加法器
9bit
十二位
加法器
10bit
十二位
加法器
11bit
÷ 8
8bit

邻域平均的运算电路

§ 3.4.2 多图像平均法

图像采集过程中，出现噪声是不可避免的，特别在采用信噪比较低的传感器时。在加性噪声情况下，如果处理静止场景图像，则可将**M**帧图像进行加权求平均的方法，降低噪声影响。其运算表达式为：

$$g(x,y)=\frac{1}{M}[f_1(x,y)+f_2(x,y)+\ldots\ldots+f_M(x,y)]$$

$\{f_i(x,y)\}$为一批静止图像，i 为帧号，噪声是随机加性。

$g(x,y)$ 是平滑处理的输出图像。

平滑后噪声方差会下降，而且参与平均的图像愈多，噪声抑制的效果愈好。

一般选用M=8幅图像取平均，因太多图像配准较困难。

利用同一景物的多幅图像取平均来消除噪声产生的高频成分，平滑图像。

效果分析：∵ $g_i(x,y) = f_i(x,y) + n_i(x,y)$；

M幅含有不同噪声的图像，求平均得

$$g(x,y) = (1/M) \sum g_i(x,y) \quad (i=1,\ldots,M)$$

$f(x,y) = E\{g(x,y)\}$，数学期望

通过分析可得：

$$\sigma_g^2 = \frac{\sigma_n^2}{M}$$

结论：M幅图像平均可把噪声方差减少M倍。