

180206081104P2009H

## 图像处理与分析

### 第六讲：图像增强

---

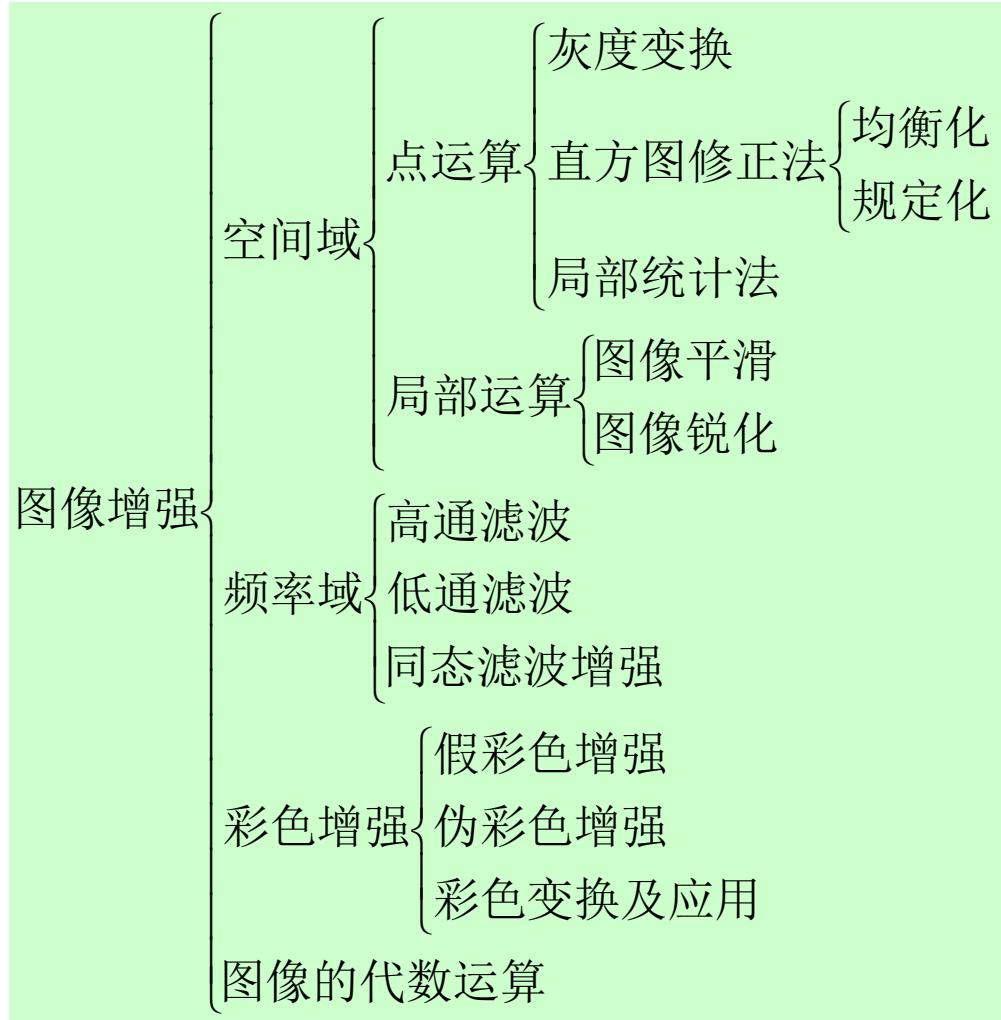
图像增强是采用一系列技术去改善**图像的视觉效果**，或将图像转换成一种更适合于人或机器进行分析和处理的形式。例如采用一系列技术有选择地突出某些感兴趣的信息，同时抑制一些不需要的信息，提高图像的使用价值。

图像增强方法从增强的作用域出发，可分为空间域增强和频率域增强两种。

**空间域增强**是直接对图像各像素进行处理；

**频率域增强**是对图像经傅立叶变换后的频谱成分进行处理，然后逆傅立叶变换获得所需的图像。

# 内容概要

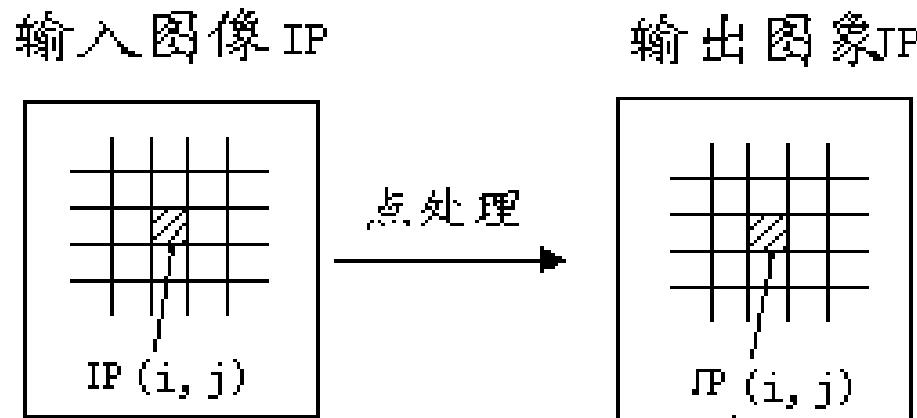


# 目的

- 熟悉并掌握本章基本概念、空间域图像增强的原理、方法及其特点；
- 重点掌握直方图修正方法、特点及其应用；空间域平滑、锐化和彩色增强技术。

# 图像的点处理

输出值 $JP(i,j)$ 仅与 $IP(i,j)$ 有关



点处理的计算表达式：

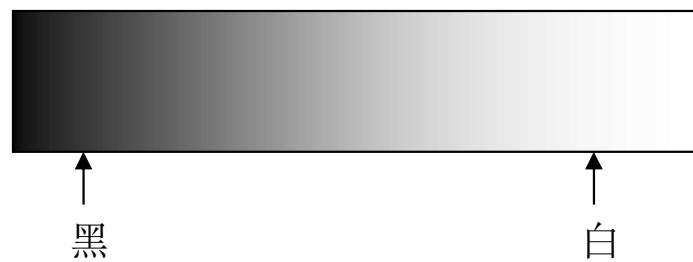
$$JP(i, j) = \phi_p(IP(i, j))$$

点运算的典型方法是以图像的灰度直方图作为分析处理的依据。

- 灰度变换
- 直方图修正法

# 图像的线性灰度变换

灰度变换可调整图像的灰度动态范围或图像对比度，是图像增强的重要手段之一。



## 1. 线性变换

令图像 $f(i, j)$ 的灰度范围为 $[a, b]$ ,

线性变换后图像 $g(i, j)$ 的范围为 $[a', b']$ ,  
如图,  $g(i, j)$ 与 $f(i, j)$ 之间的关系式为:

$$g(i, j) = a' + \frac{b' - a'}{b - a} (f(i, j) - a) \quad (5-1)$$

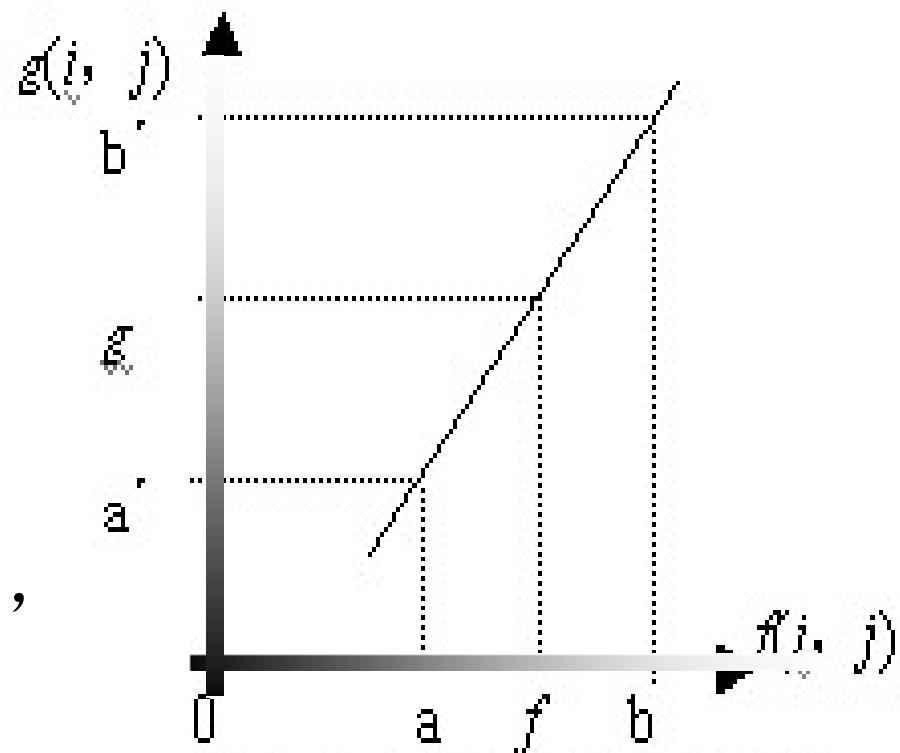
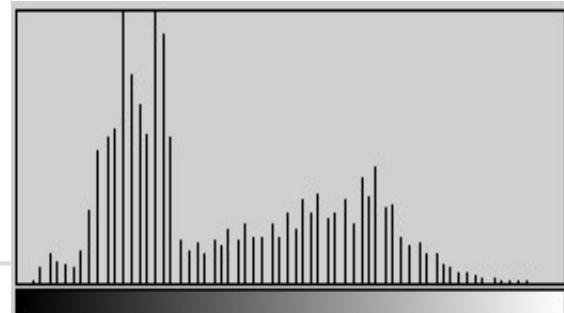
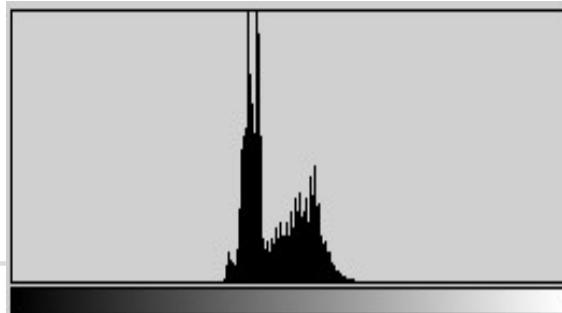


图 4.1.1 线性变换示意图

在曝光不足或过度的情况下，图像灰度可能会局限在一个很小的范围内。这时在显示器上看到的将是一个模糊不清、似乎没有灰度层次的图像。

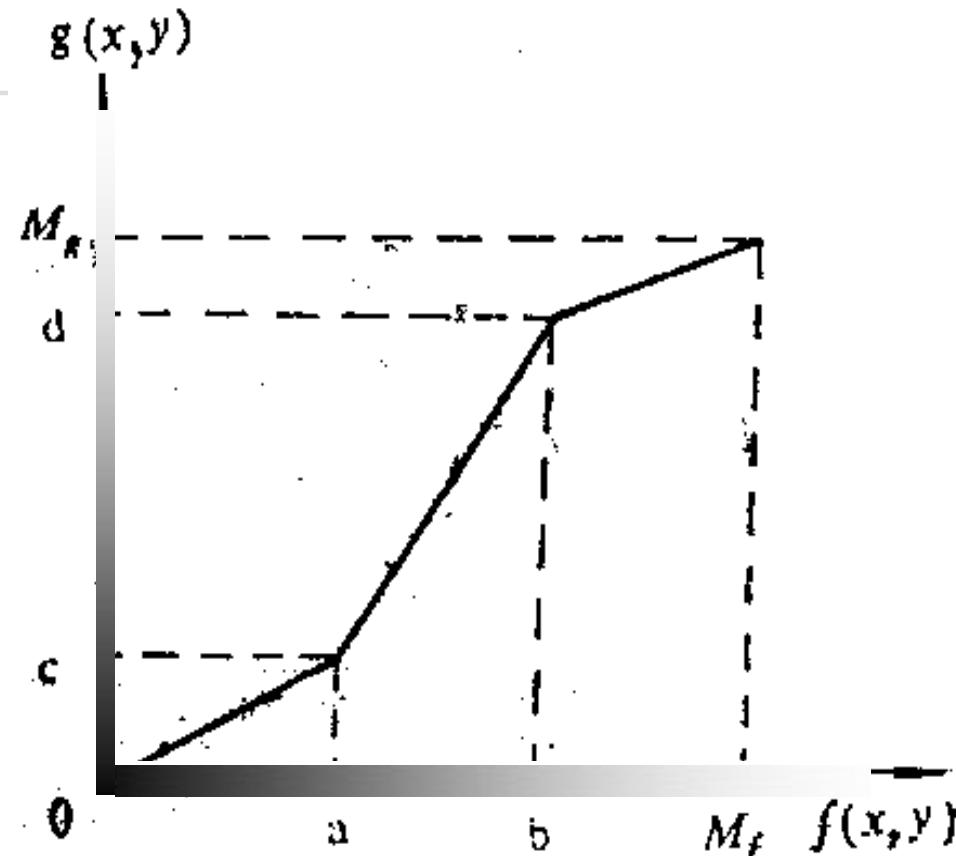
下图是对曝光不足的图像采用线性变换对图像每一个像素灰度作线性拉伸。可有效地改善图像视觉效果。



## 2. 分段线性变换

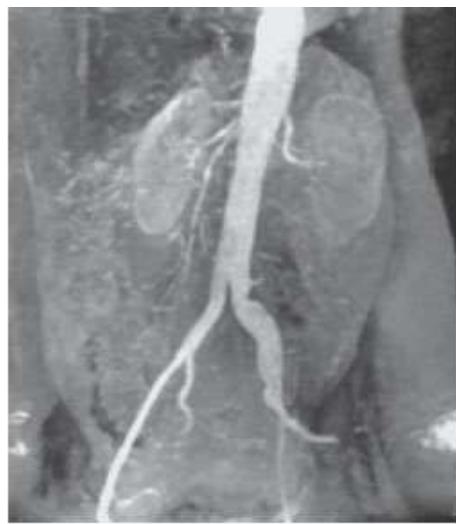
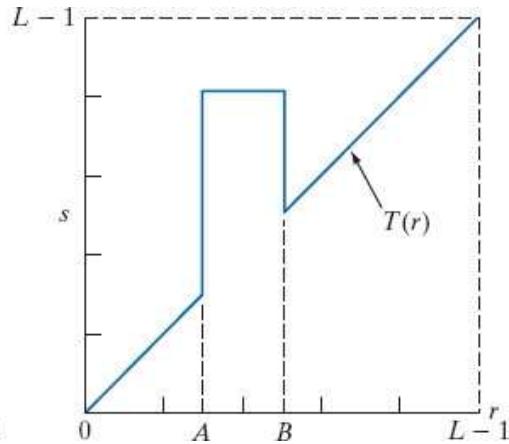
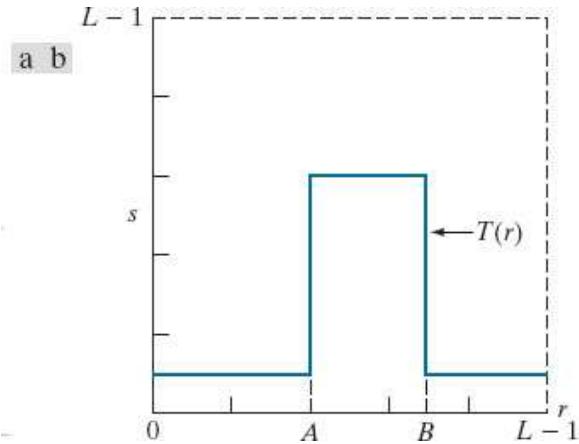
为了突出感兴趣目标所在的灰度区间，相对抑制那些不感兴趣的灰度区间，可采用分段线性变换。

设原图像 $f(x,y)$ 在 $[0, M_f]$ ，感兴趣目标的灰度范围在 $[a,b]$ ，欲使其灰度范围拉伸到 $[c,d]$ ，则对应的分段线性变换表达式为



$$g(x, y) = \begin{cases} (c/a)f(x, y) & 0 \leq f(x, y) < a \\ [(d - c)/(b - a)][f(x, y) - a] + c & a \leq f(x, y) < b \\ [(M_g - d)/(M_f - b)][f(x, y) - b] + d & b \leq f(x, y) \leq M_f \end{cases}$$

通过细心调整折线拐点的位置及控制分段直线的斜率，可对任一灰度区间进行拉伸或压缩。



---

**例1、在MATLAB环境中，采用图像线性变换进行  
图像增强。应用MATLAB的函数imadjust将图像  
 $0.3 \times 255 \sim 0.7 \times 255$ 灰度之间的值通过线性变换映  
射到0~255之间。**

**解：分别取：  $a = 0.3 \times 255$ ,  $b = 0.7 \times 255$   
 $c=0$ ,  $d=255$**

---

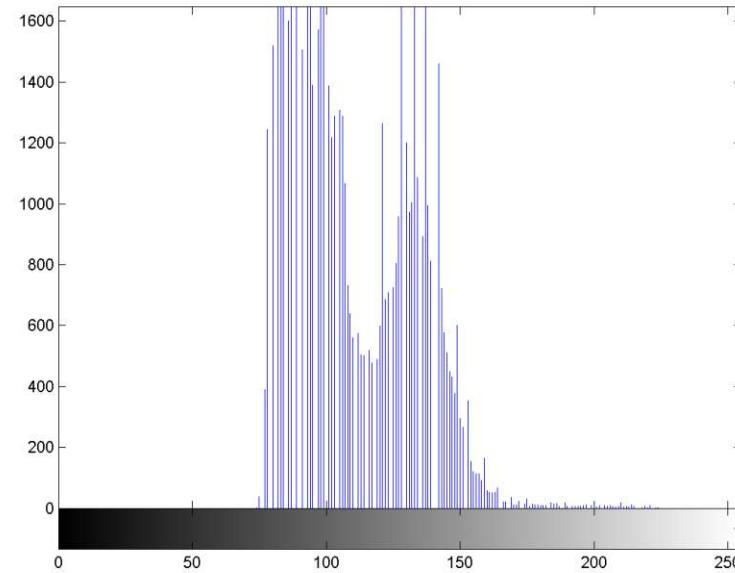
# 实现的程序

---

- A=imread('pout.tif'); %读入图像
- imshow(A); %显示图像
- figure
- imhist(A); %显示图像的直方图
- J1=imadjust(A,[0.3 0.7],[]);  
%函数将图像在0.3\*255~0.7\*255灰度之间的  
值通过线性变换映射到0~255之间
- figure,imshow(J1); %输出图像效果图
- figure
- imhist(J1) %输出图像的直方图



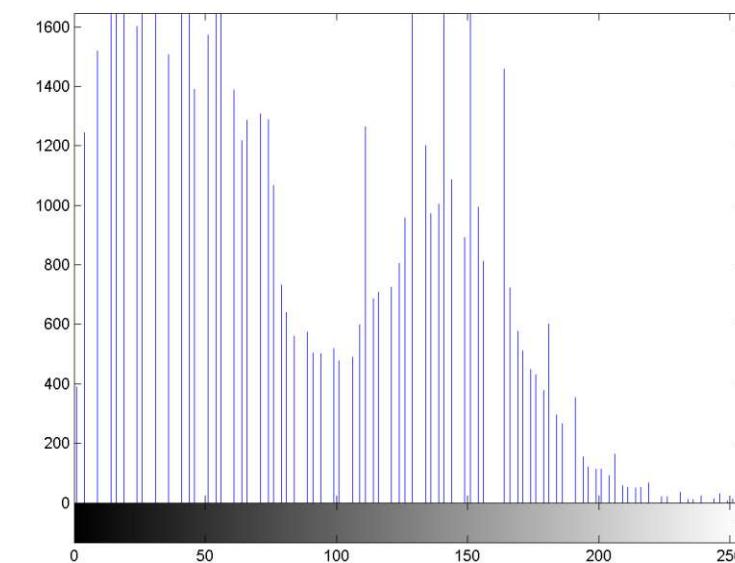
(a) 原图



(b) 原图的直方图



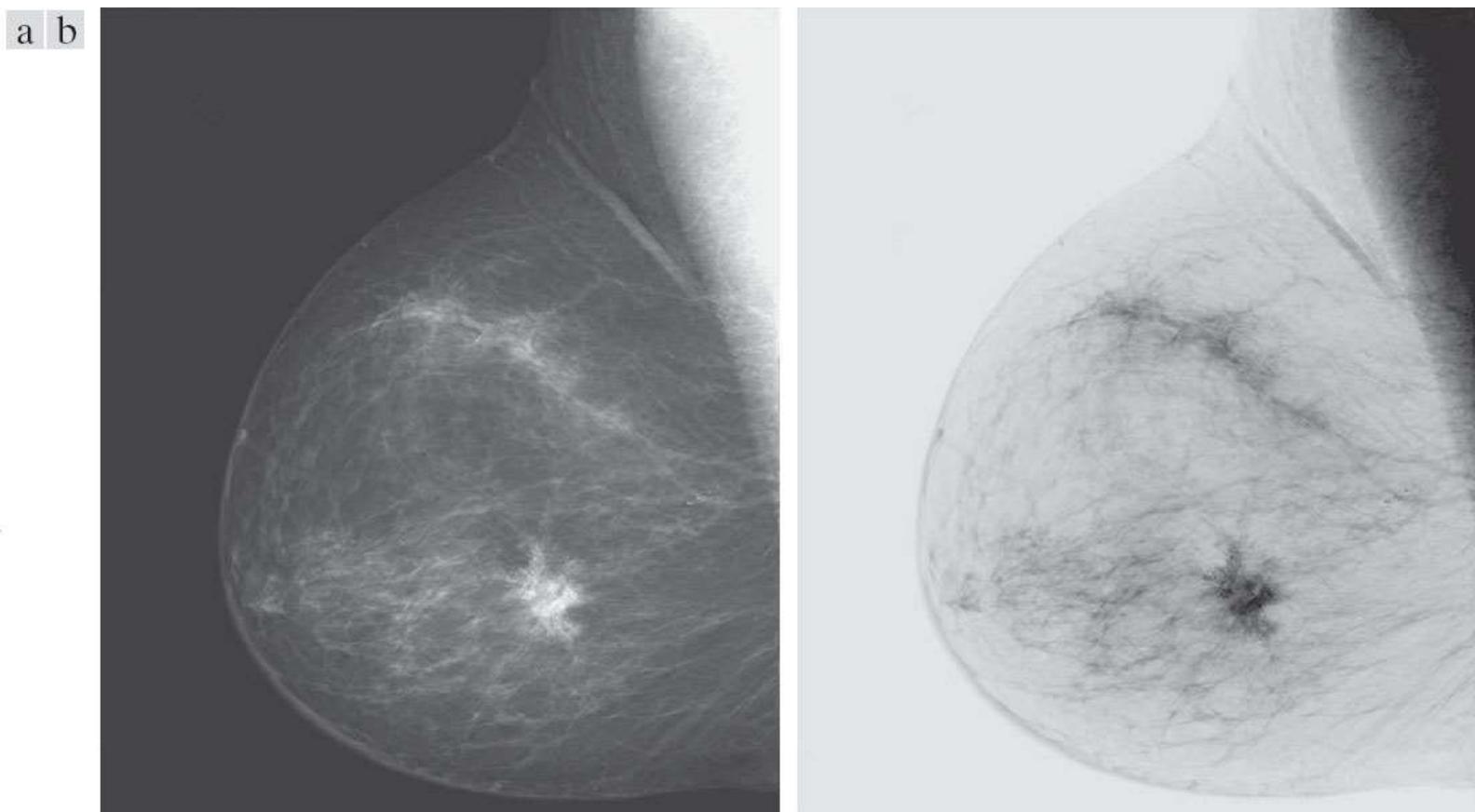
(c) 输出图像



(d) 输出图像的直方图

# 图像的反转变换

$$s = L - 1 - r \quad \text{where } s \in [0, L - 1]$$



# 图像的非线性灰度变换

当用某些非线性函数如对数函数、指数函数等，作为映射函数时，可实现图像灰度的非线性变换。

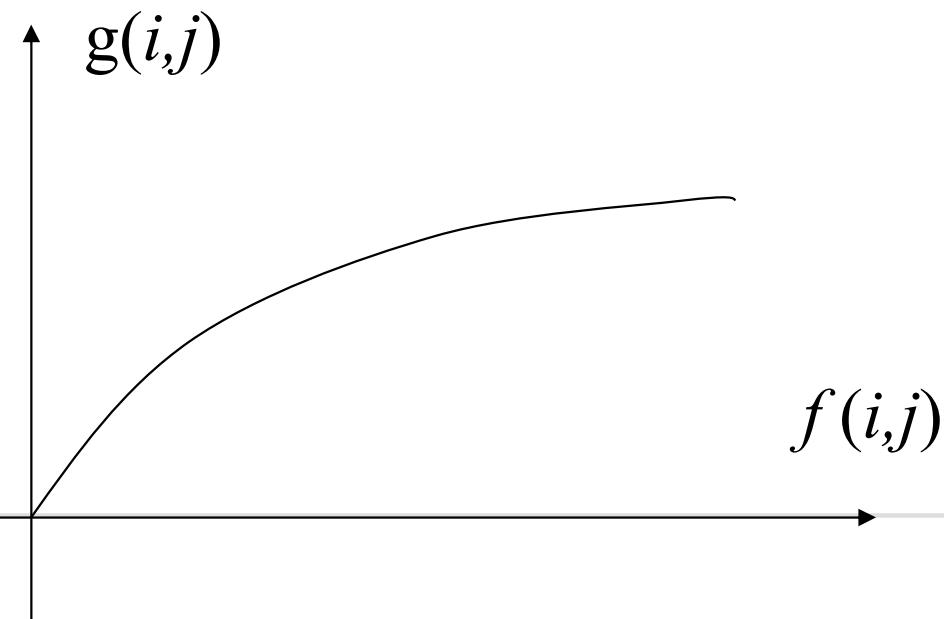
## ① 对数变换

对数变换的一般表达式为

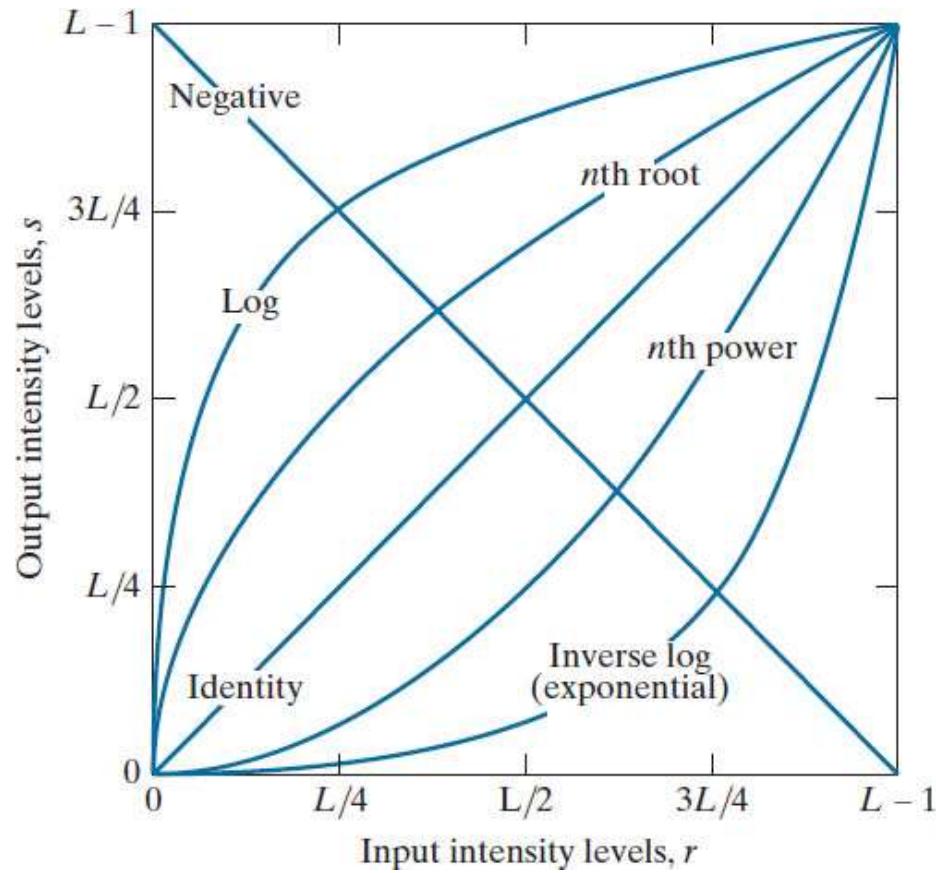
$$g(i,j) = a + \frac{\ln[f(i,j) + 1]}{b \cdot \ln c} \quad (5-2)$$

这里 $a, b, c$ 是为了调整曲线的位置和形状而引入的参数。

当希望对图像的低灰度区较大的拉伸而对高灰度区压缩时，可采用这种变换，它能使图像灰度分布与人的视觉特性相匹配。



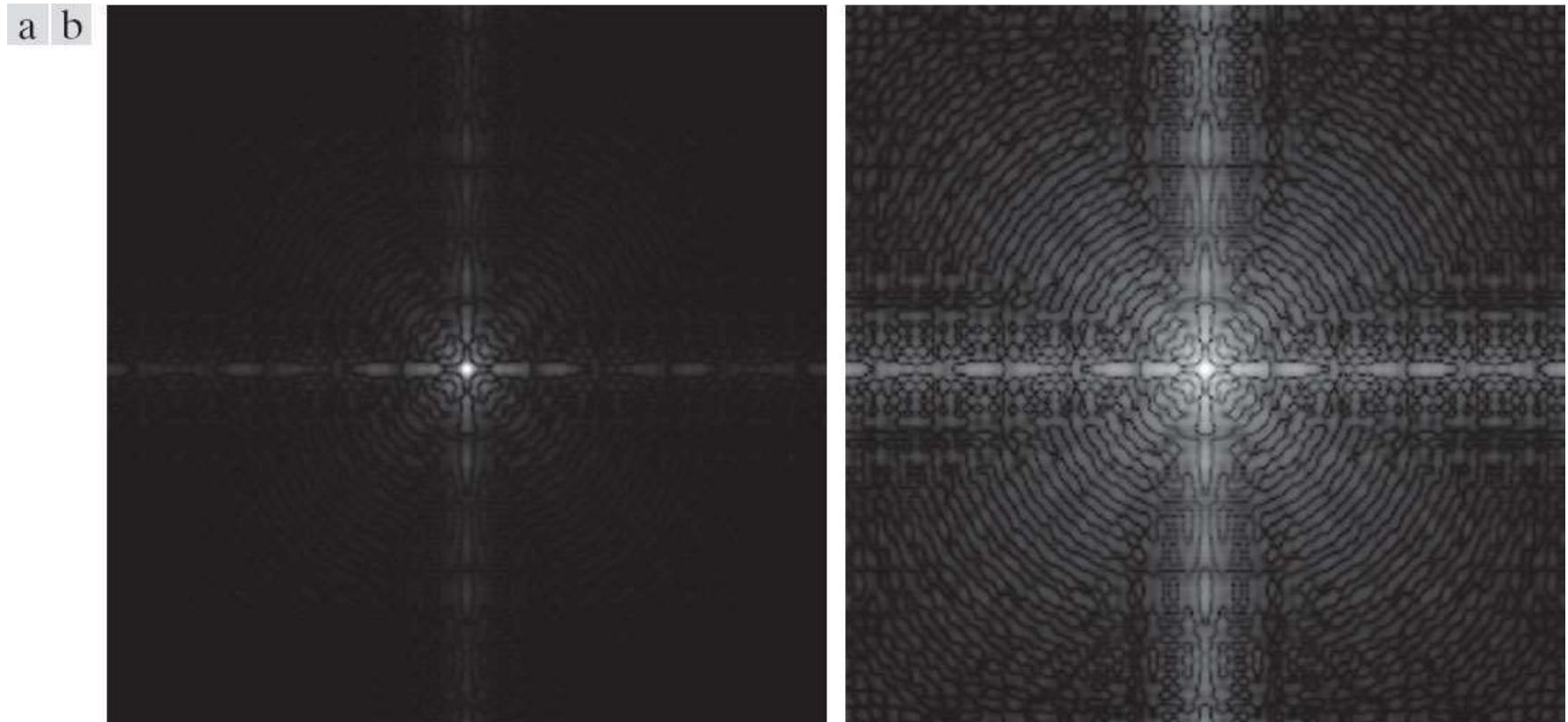
# 常用的图像的非线性变换



# 常用的图像的非线性变换

$$s = c \log(1 + r)$$

- (a) Fourier spectrum displayed as a grayscale image.  
(b) Result of applying the log transformation in Eq. (3-4)  
with **c = 1**. Both images are scaled to the range [0, 255].

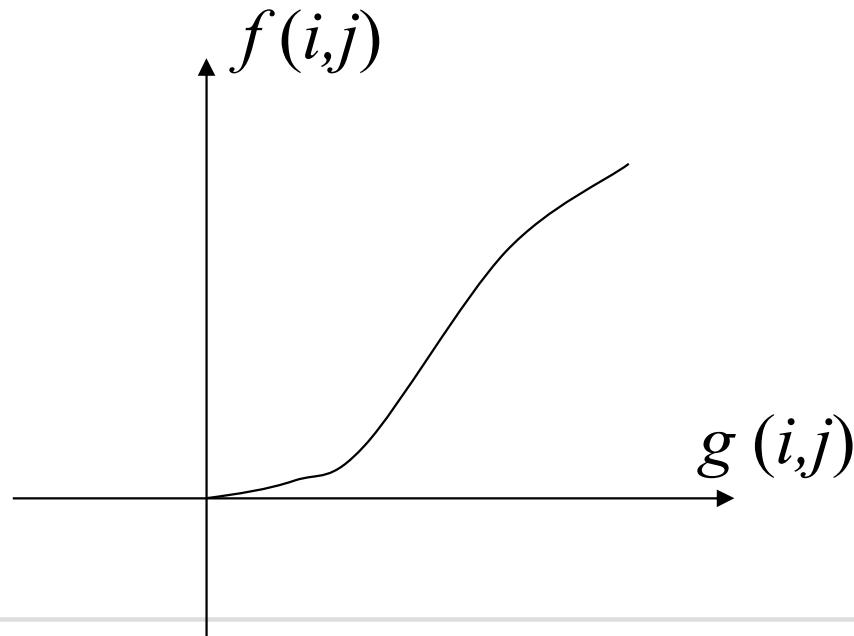


## ②指数变换

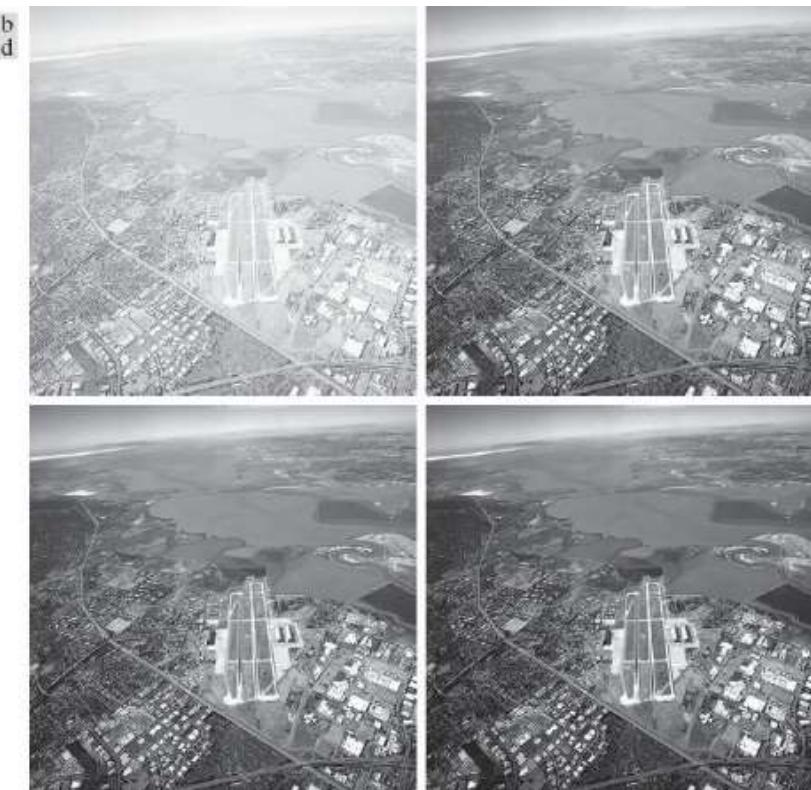
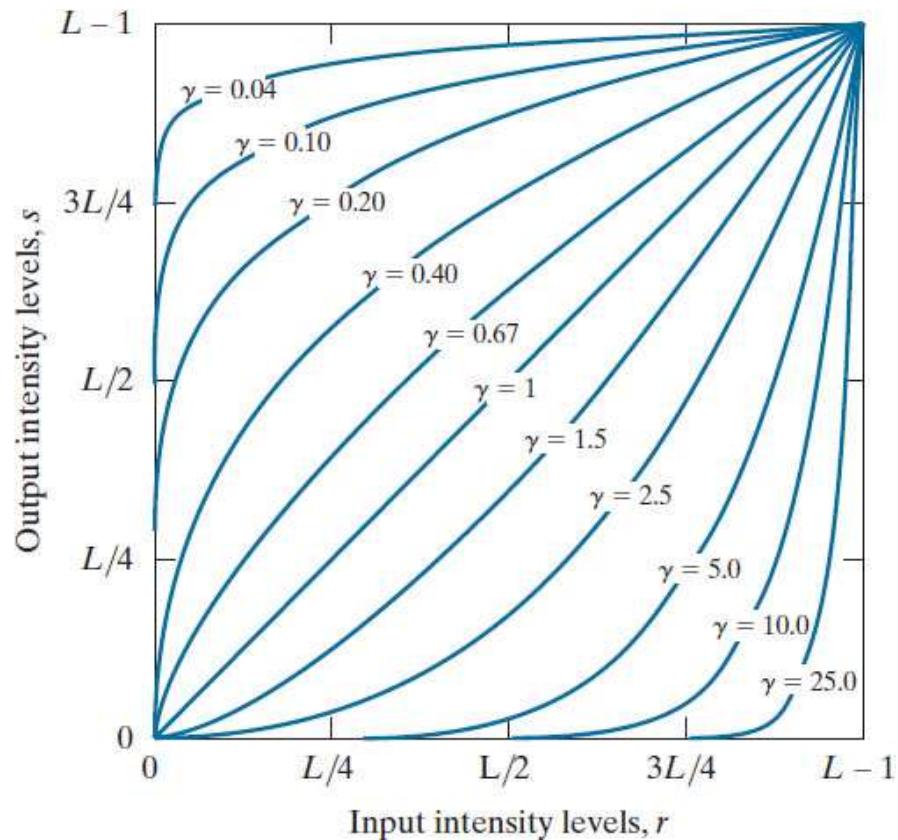
指数变换的一般表达式为

$$g(i,j) = b^{c[f(i,j)-a]} - 1 \quad (5-3)$$

这里参数a, b, c用来调整曲线的位置和形状。这种变换能对图像的高灰度区给予较大的拉伸。



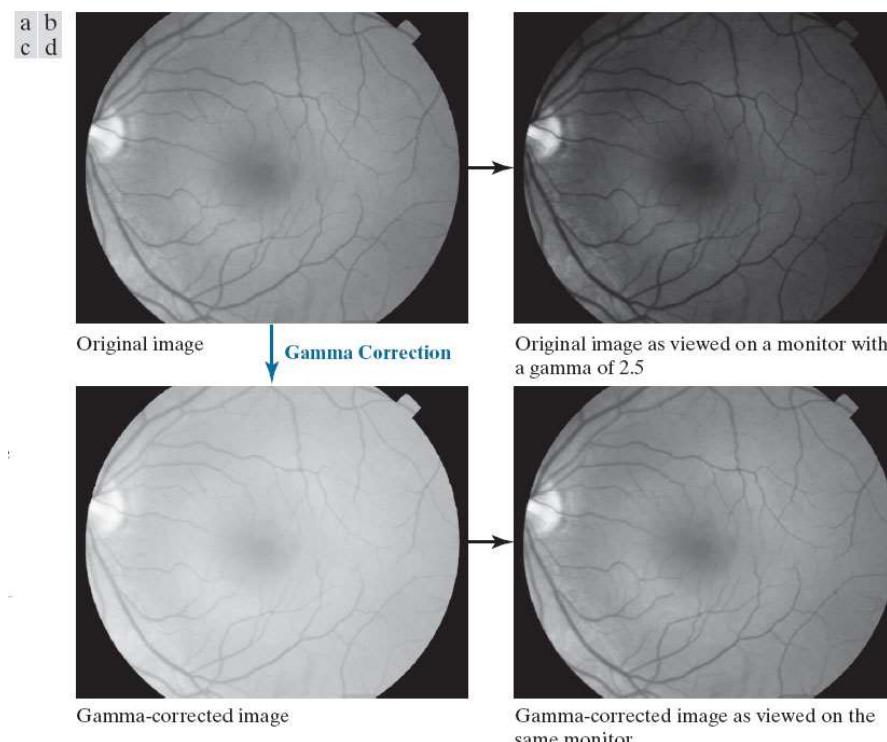
# 图像的伽马变换



$$s = cr^\gamma$$

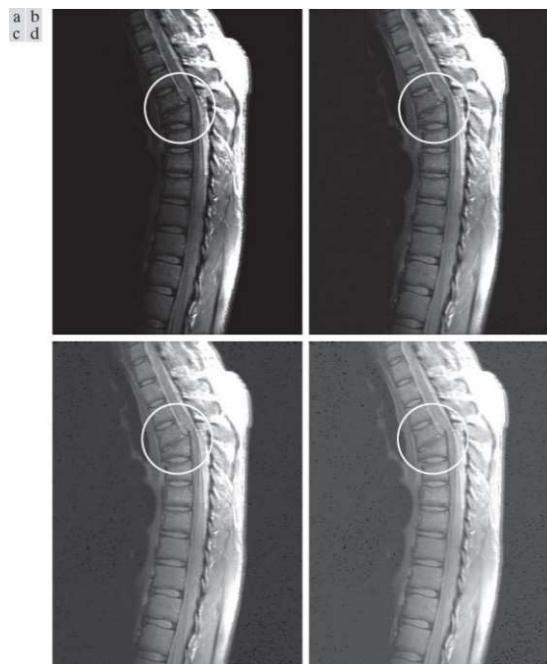
# 图像的伽马变换

(a) Image of a human retina. (b) Image as it appears on a monitor with a gamma setting of 2.5 (note the darkness). (c) Gamma-corrected image. (d) Corrected image, as it appears on the same monitor (compare with the original image). (Image (a) courtesy of the National Eye Institute, NIH)



# 图像的伽马变换

(a) Magnetic resonance image (MRI) of a fractured human spine (the region of the fracture is enclosed by the circle). (b)–(d) Results of applying the transformation in Eq. (3-5) with  $c = 1$  and and 0.3, respectively. (Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)



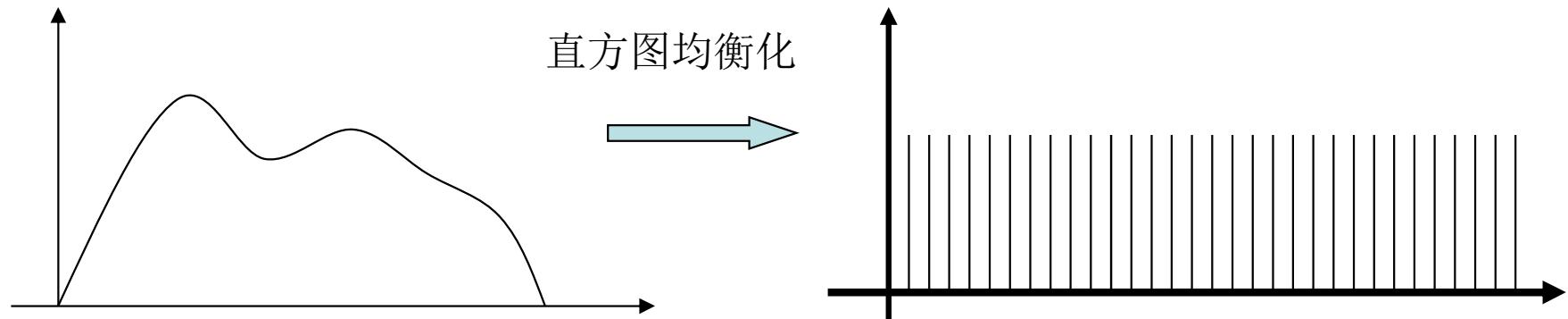
# 图像的直方图调整法

灰度直方图反映了数字图像中每一灰度级与其出现频率间的关系，它能描述该图像的概貌。通过修改直方图的方法增强图像是一种实用而有效的处理技术。

直方图调整法包括直方图均衡化及直方图规定化两类。

## 1. 直方图均衡化

直方图均衡化是将原图像通过某种变换，得到一幅灰度直方图为均匀分布的新图像的方法。



下面先讨论连续变化图像的均衡化问题，然后推广到离散的数字图像上。

设 $r$ 和 $s$ 分别表示归一化了的原图像灰度和经直方图修正后的图像灰度。即

$$0 \leq r, s \leq 1 \quad (5-4)$$

在 $[0, 1]$ 区间内的任一个 $r$ 值，都可产生一个 $s$ 值，且

$$s = T(r) \quad (5-5)$$

$T(r)$ 作为变换函数，满足下列条件：

①在 $0 \leq r \leq 1$ 内为单调递增函数，保证灰度级从黑到白的次序不变；

②在 $0 \leq r \leq 1$ 内，有 $0 \leq T(r) \leq 1$ ，确保映射后的像素灰度在允许的范围内。

反变换关系为

$$r = T^{-1}(s) \quad (5-6)$$

$T^{-1}(s)$ 对s同样满足上述两个条件。

由概率论理论可知，如果已知随机变量r的概率密度为 $p_r(r)$ ，而随机变量s是r的函数，则s的概率密度 $p_s(s)$ 可以由 $p_r(r)$ 求出。

假定随机变量s的分布函数用 $F_s(s)$ 表示，根据分布函数定义

$$F_s(s) = \int_{-\infty}^s p_s(s) ds = \int_{-\infty}^r p_r(r) dr \quad (7-7)$$

利用密度函数是分布函数的导数的关系，等式两边对s求导，有：

$$P_s(s) = \frac{d}{ds} \left[ \int_{-\infty}^r p_r(r) dr \right] = p_r \frac{dr}{ds} = p_r \frac{d}{ds} [T^{-1}(s)] \quad (5-8)$$

可见，输出图像的概率密度函数可以通过变换函数T(r)控制原图像灰度级的概率密度函数得到，因而改善原图像的灰度层次，这就是直方图修改技术的基础。

从人眼视觉特性来考虑，一幅图像的直方图如果是均匀分布的，即  $P_s(s)=k$  (归一化时  $k=1$ ) 时，该图像色调给人的感觉比较协调。因此将原图像直方图通过  $T(r)$  调整为均匀分布的直方图，这样修正后的图像能满足人眼视觉要求。

因为归一化假定

$$P_s(s) = 1 \quad (5-9)$$

由 (5-8) 则有

$$ds = p_r(r) dr$$

两边积分得

$$s = T(r) = \int_0^r p_r(r) dr \quad (5-10)$$

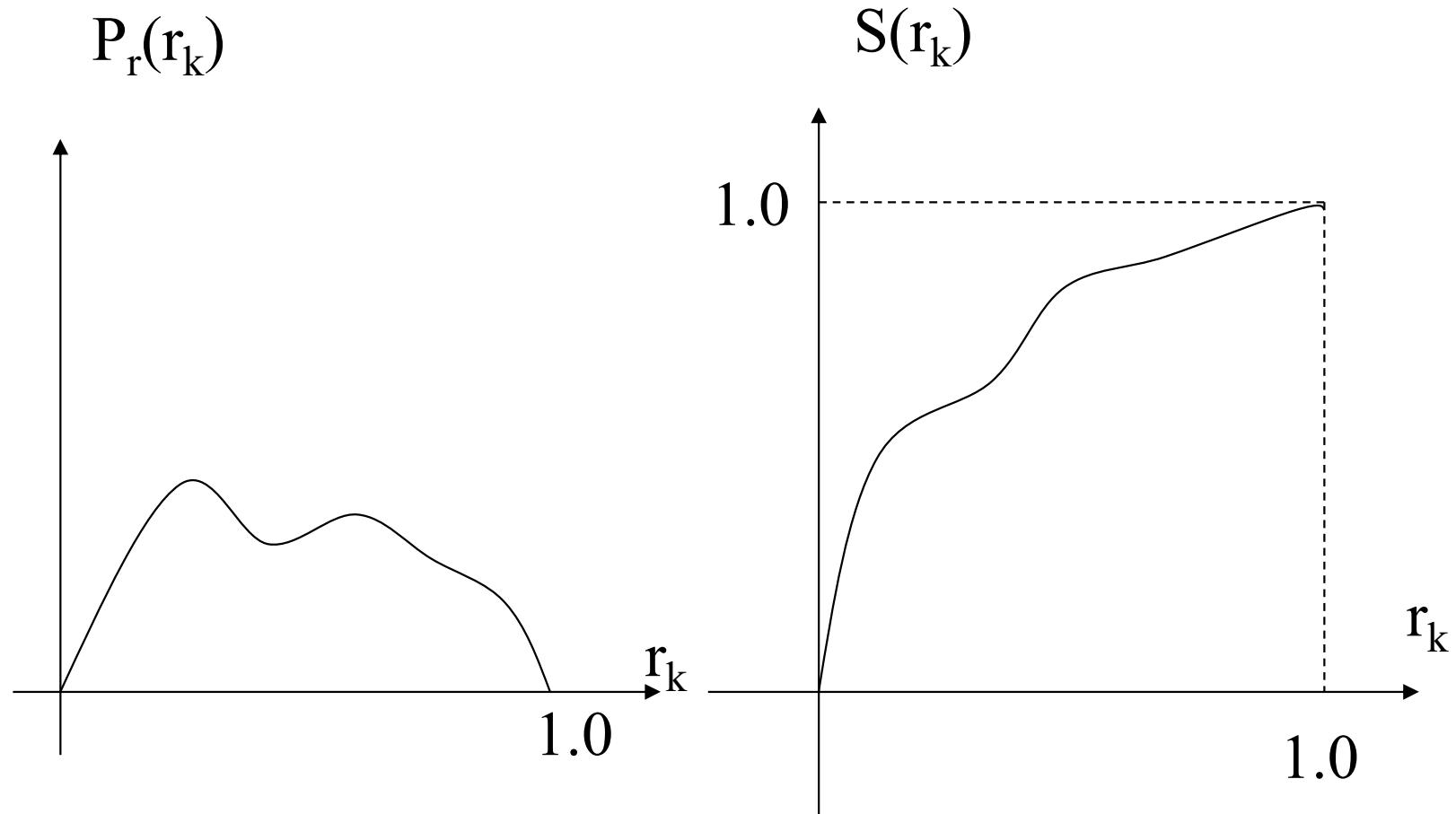
上式表明，当变换函数为r的累积直方图函数时，能达到直方图均衡化的目的。

对于离散的数字图像，用频率来代替概率，则变换函数  $T(r_k)$  的离散形式可表示为：

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

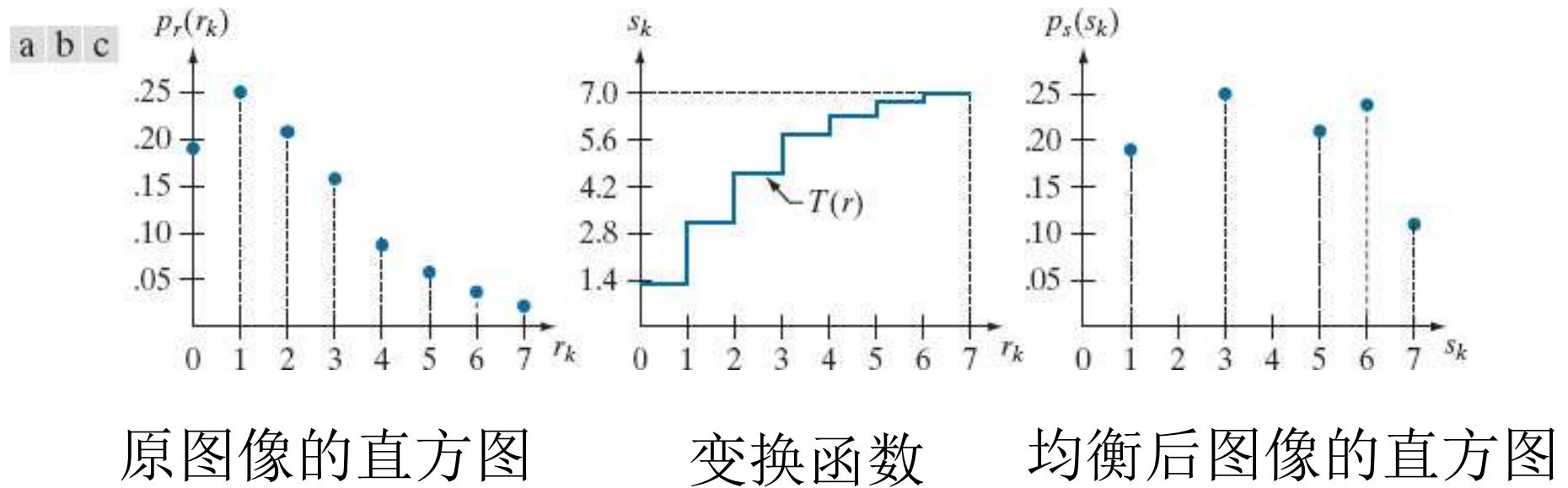
上式表明，均衡后各像素的灰度值  $s_k$  可直接由原图像的直方图算出。

一幅图像的 $s_k$ 与 $r_k$ 之间的关系称为该图像的累积灰度直方图。



下面举例说明直方图均衡过程。

# 直方图均衡化示例

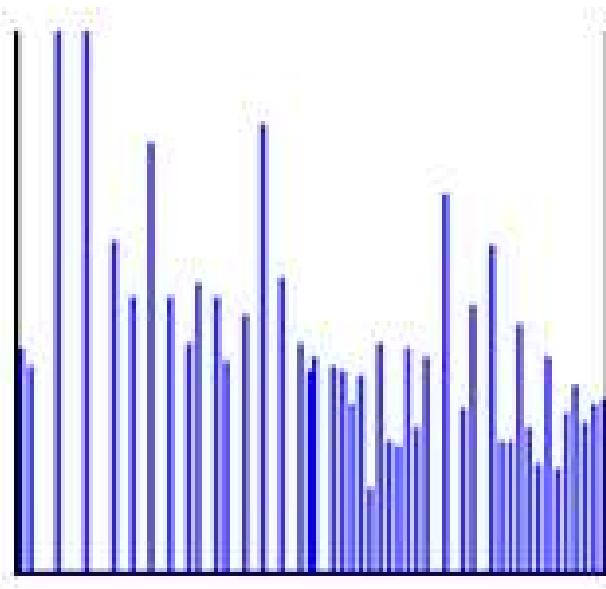
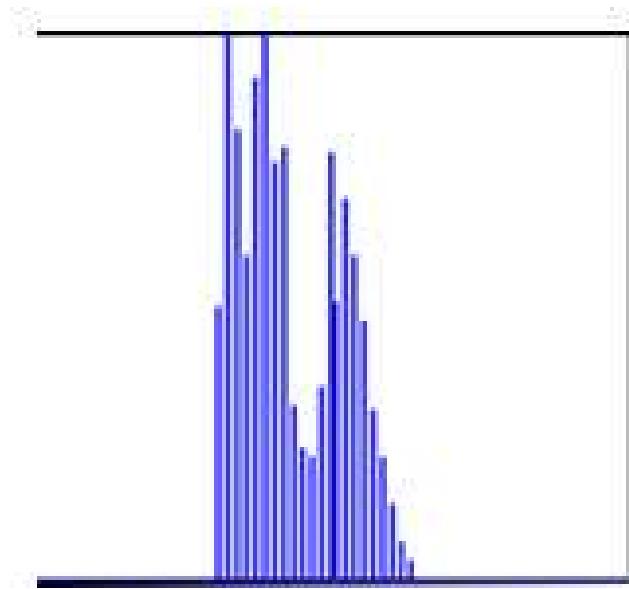


## 例题

**例2、在MATLAB环境中，采用直方图均衡的方法进行图像增强。**

解：程序如下

```
A=imread('pout.tif');
I=histeq(A); %调用函数完成直方图均衡化
subplot(1,2,1),imshow(A); %直方图均衡化前的图像效果
subplot(1,2,2),imshow(I); %直方图均衡化后的图像效果
figure,subplot(1,2,1),imhist(A); %均衡化前的直方图
subplot(1,2,2),imhist(I); %均衡化后的直方图
```



## 2. 直方图规定化

在某些情况下，并不一定需要具有均匀直方图的图像，有时需要具有特定的直方图的图像，以便能够增强图像中某些灰度级。**直方图规定化方法**就是针对上述思想提出来的。直方图规定化是使原图像灰度直方图变成规定形状的直方图而对图像作修正的增强方法。

可见，它是对直方图均衡化处理的一种有效的扩展。直方图均衡化处理是直方图规定化的一个特例。

对于直方图规定化，下面仍从灰度连续变化的概率密度函数出发进行推导，然后推广出灰度离散的图像直方图规定化算法。

假设 $p_r(r)$ 和 $p_z(z)$ 分别表示已归一化的原始图像灰度分布的概率密度函数和希望得到的图像的概率密度函数。

首先对原始图像进行直方图均衡化，即求变换函数：

$$s = T(r) = \int_0^r p_r(r) dr \quad (7-11)$$

假定已得到了所希望的图像，对它也进行均衡化处理，即

$$v = G(z) = \int_0^z p_z(r) dr \quad (7-12)$$

它的逆变换是

$$z = G^{-1}(v) \quad (7-13)$$

这表明可由均衡化的灰度得到希望图像的灰度。

若对原始图像和希望图像都作了均衡化处理，则二者均衡化的 $p_s(s)$ 和 $p_v(v)$ 相同，即都为均匀分布的密度函数。

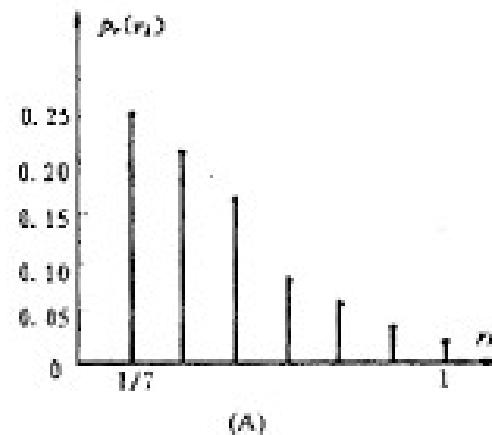
由 $s$ 代替 $v$ 得  $z = G^{-1}(s)$

这就是所求得的变换表达式。根据上述思想，可总结出直方图规定化增强处理的步骤如下：

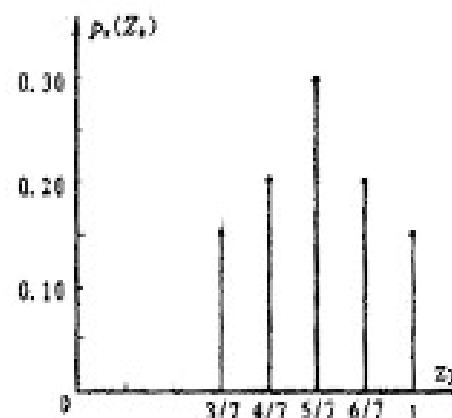
- ①对原始图像作直方图均衡化处理；
- ②按照希望得到的图像的灰度概率密度函数 $p_z(z)$ ，求得变换函数 $G(z)$ ；
- ③用步骤①得到的灰度级 $s$ 作逆变换 $z = G^{-1}(s)$ 。

经过以上处理得到的图像的灰度级将具有规定的概率密度函数 $p_z(z)$ 。

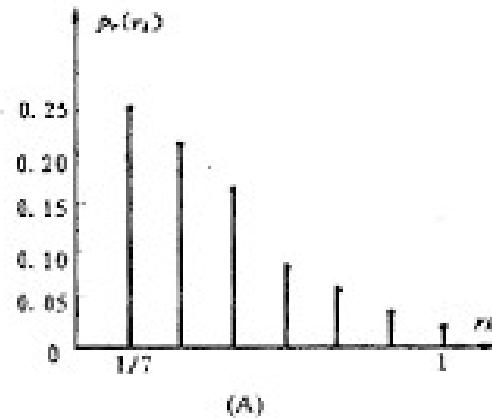
采用与直方图均衡相同的原始图像数据（ $64 \times 64$ 像素且具有8级灰度），其灰度级分布列于表中。给定的直方图的灰度分布列于表中。对应的直方图如下：



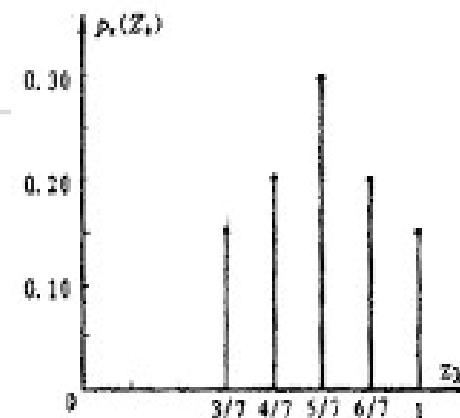
原图像的直方图



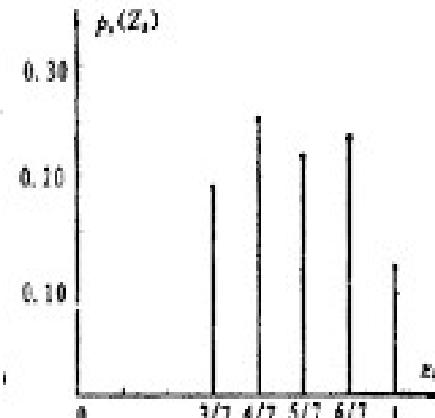
规定化直方图



原图像的直方图



规定的直方图



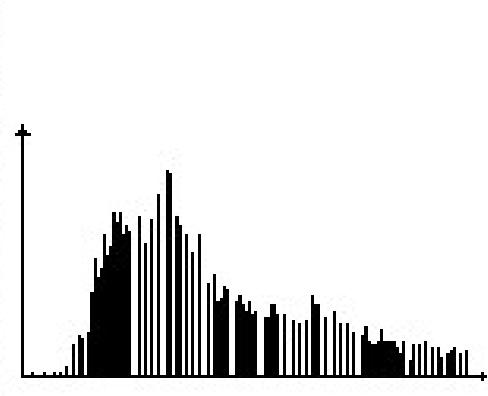
规定化后图像的直方图

利用直方图规定化方法进行图像增强的主要困难在于要构成有意义的直方图。图像经直方图规定化，其增强效果要有利于人的视觉判读或便于机器识别。

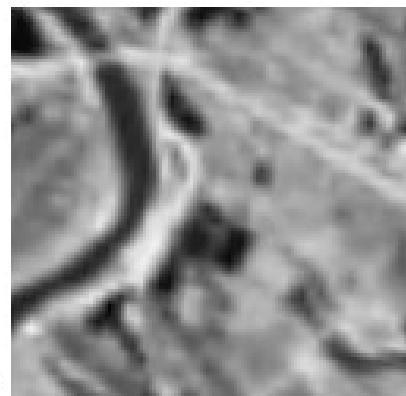
下面是一个直方图规定化应用实例。



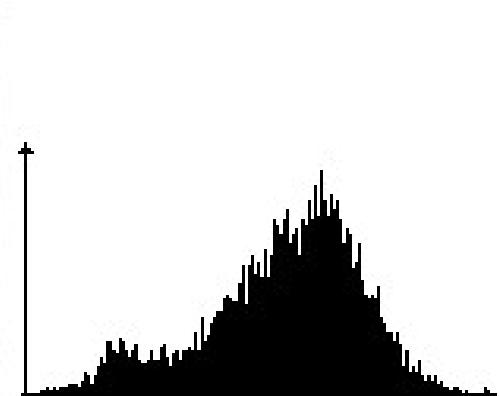
(A)



(a)



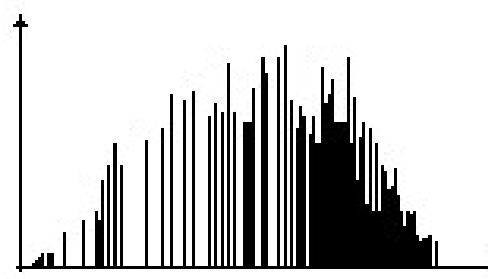
(B)



(b)

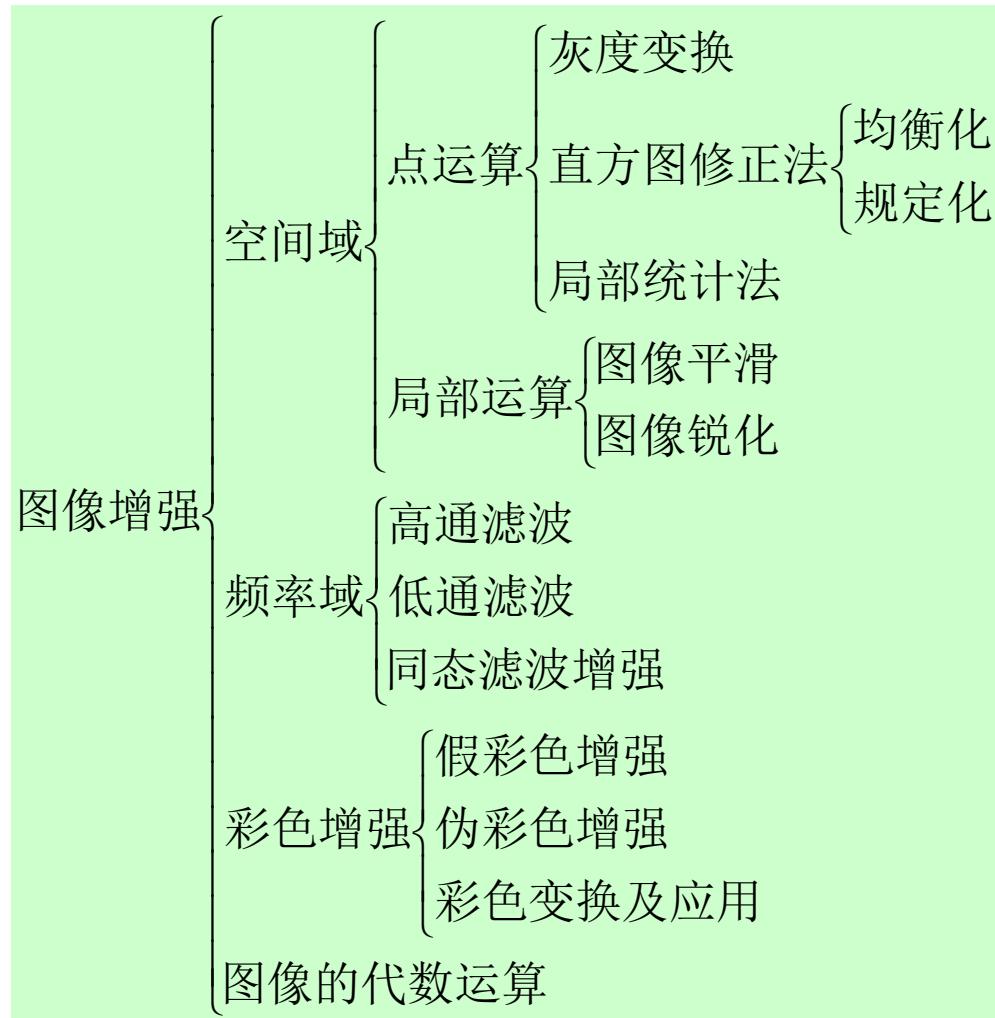


(C)

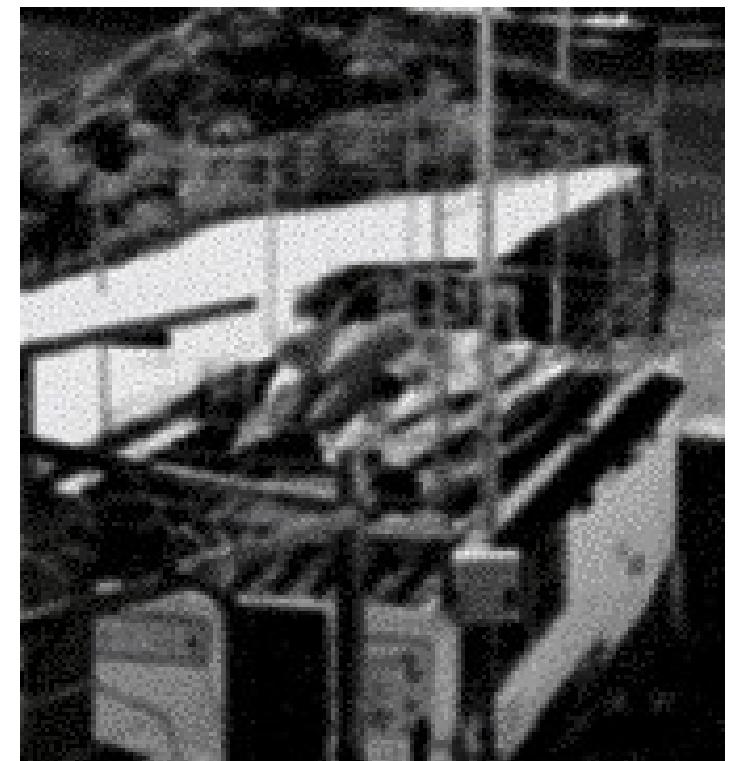
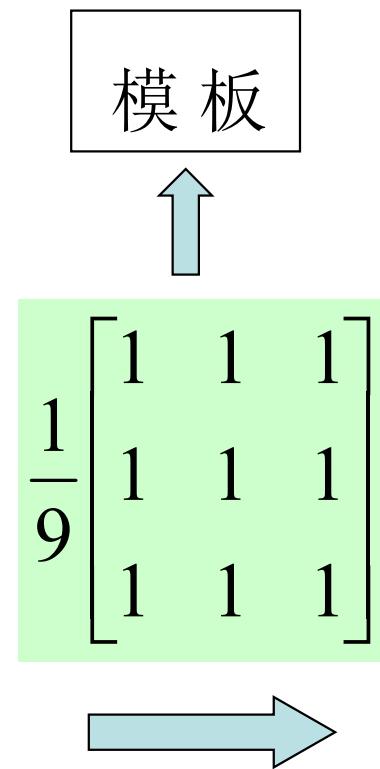
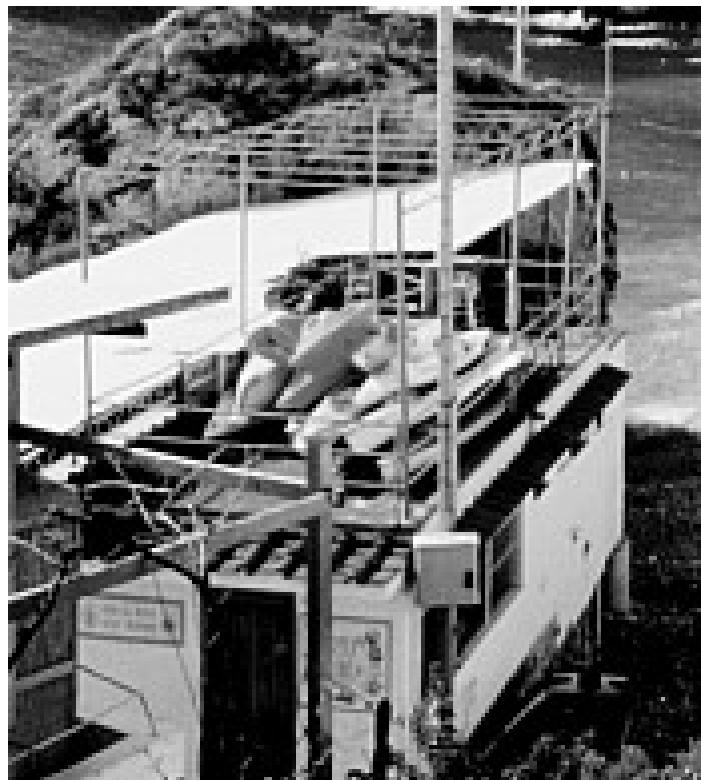


(c)

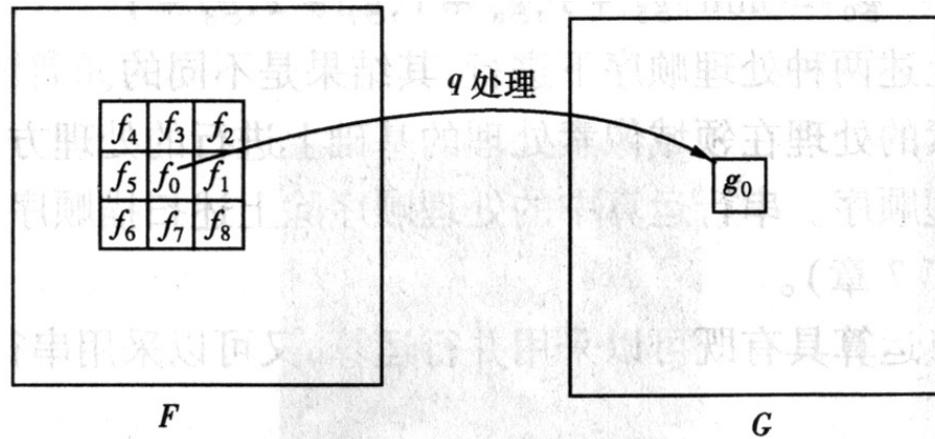
图(C)、(c)是将图像(A)按图(b)的直方图进行规定化得到的结果及其直方图。通过对比可以看出图(C)的对比度同图(B)接近一致，对应的直方图形状差异也不大。这样有利于影像融合处理，保证融合影像光谱特性变化小。



# 图像的空间域平滑

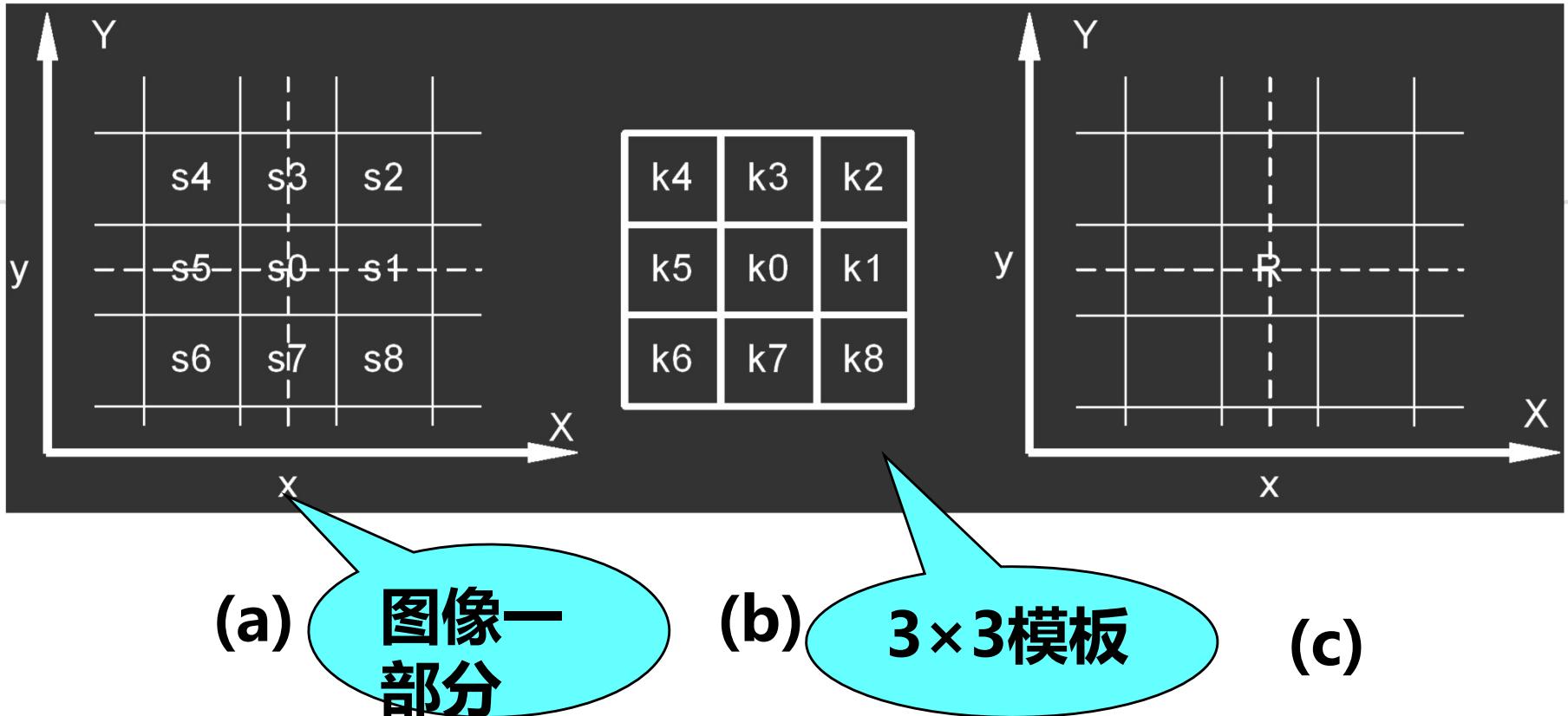


邻域运算：



表达式：

$$g(i, j) = \phi_N(N(f(i, j)))$$



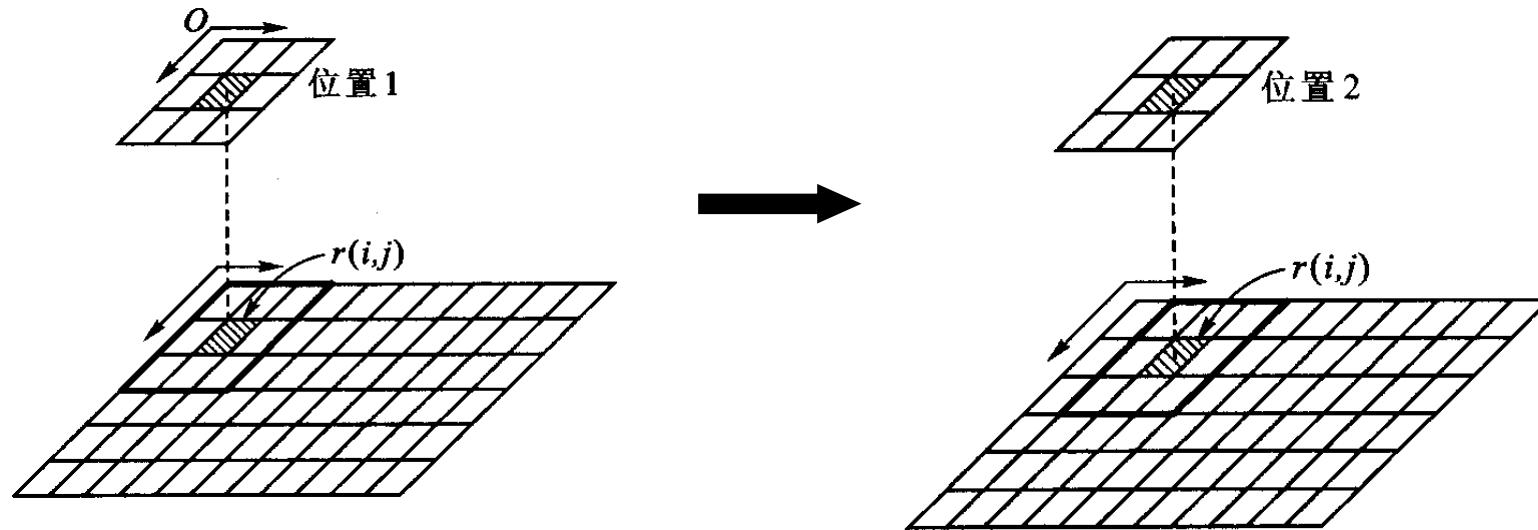
将 $k_0$ 的位置于图中灰度值为 $s_0$ 的象素重合（即将模板中心放在图中 $(x,y)$ 位置），模板的输出响应 $R$ 为：

$$R = k_0 s_0 + k_1 s_1 + \cdots + k_8 s_8$$

将 $R$ 赋给增强图，作为在 $(x,y)$ 位置的灰度值(图c)

# 模板运算

$$r(i, j) = \sum_{m=1}^M \sum_{n=1}^N \phi(m, n) t(m, n)$$



模板运算的步骤：

- (1) 将模板在图上漫游。将模板中心与图中某个像素位置重合；
- (2) 将模板上的系数与模板下对应像素相乘；
- (3) 将所有乘积相加
- (4) 将结果赋给输出图中对应模板中心位置的像素，

得到输出图上对应的像素值

注：如果模板始终不变，上述计算对应的是用对称的模板进行卷积运算。

---

## 空间域中常用的平滑方法：

- 均值平滑
- 中值滤波

## 均值平滑

---

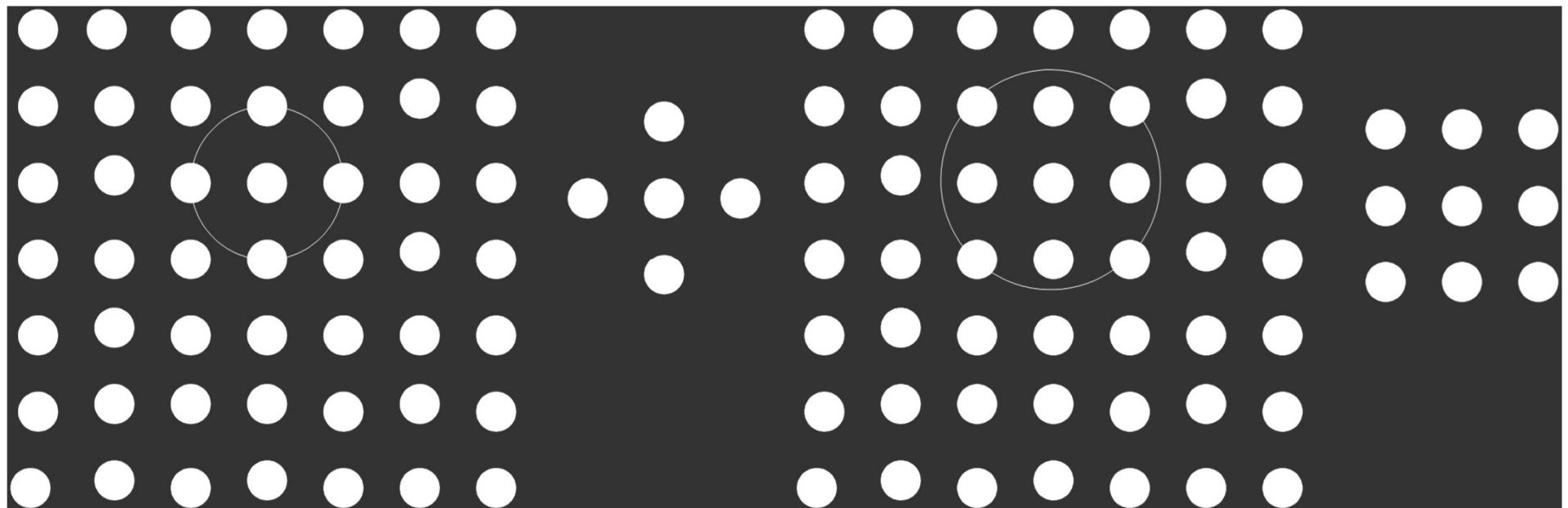
**方法：**将每个像元在以其为中心的邻域内取平均值来代替该像元值，以达到去掉尖锐“噪声”和平滑影像目的。

公式为：

$$g(i, j) = \frac{1}{K} \sum_{m, n \in S} f(m, n)$$

K表示集合内像素的总数

常用的邻域为4-邻域和8-邻域



3×3的模板形式：

0	1	0
1	1	1
0	1	0

1	1	1
1	1	1
1	1	1

为了避免中心像元值过高影响平均值，可采用模板

0	1	0
1	0	1
0	1	0

1	1	1
1	0	1
1	1	1



(a)



(b)



(c)



(d)

(a) 原图像 (b) 加椒盐噪声的图像 (c)  $3 \times 3$ 邻域平滑 (d)  $5 \times 5$ 邻域平滑

## 均值平滑的特点：

---

优点：简单，计算速度快；

缺点：造成图像模糊，削弱了图像的边缘和细节信息  
而且随着邻域范围的扩大，在去噪能力增强的同时图  
像的模糊程度越严重。

为克服简单局部平均法的弊病，已提出许多**保边  
缘、细节的局部平滑算法**。它们的出发点都集中在如  
何选择邻域的大小、形状和方向、参加平均的点数以  
及邻域各点的权重系数等。

## 5.2.2 超限像素平滑法

对邻域平均法稍加改进，可导出超限像素平滑法。它是将 $f(x, y)$ 和邻域平均 $g(x, y)$ 差的绝对值与选定的阈值进行比较，根据比较结果决定点 $(x, y)$ 的最后灰度 $g'(x, y)$ 。其表达式为

$$g'(x, y) = \begin{cases} g(x, y), & \text{当 } |f(x, y) - g(x, y)| > T \\ f(x, y), & \text{否则} \end{cases}$$

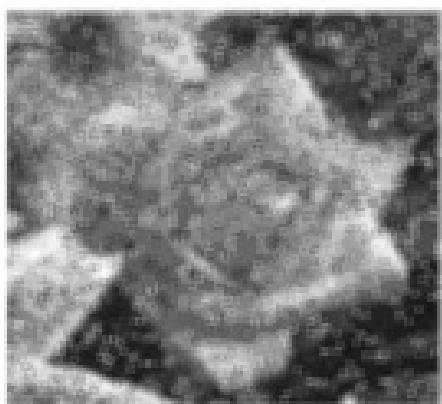
这算法对抑制椒盐噪声比较有效，对保护仅有微小灰度差的细节及纹理也有效。可见随着邻域增大，去噪能力增强，但模糊程度也大。同局部平滑法相比，超限像元平滑法去椒盐噪声效果更好。



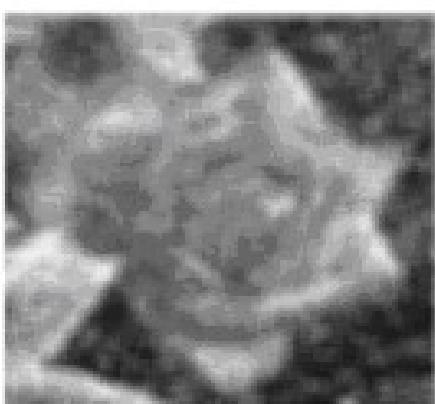
(a)



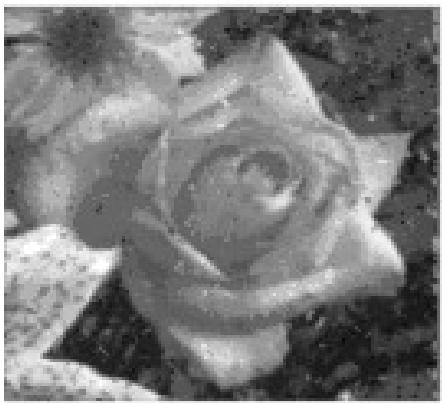
(b)



(c)



(d)



(e)



(f)

(a) 原图像

(b) 对 (a) 加椒盐噪声的图像

随机出现的亮点或者暗点

(c)  $3 \times 3$  邻域平滑(d)  $5 \times 5$  邻域平滑(e)  $3 \times 3$  超限像素平滑 ( $T=64$ )(f)  $5 \times 5$  超限像素平滑 ( $T=48$ )

### 5.2.3 灰度K近邻平均法

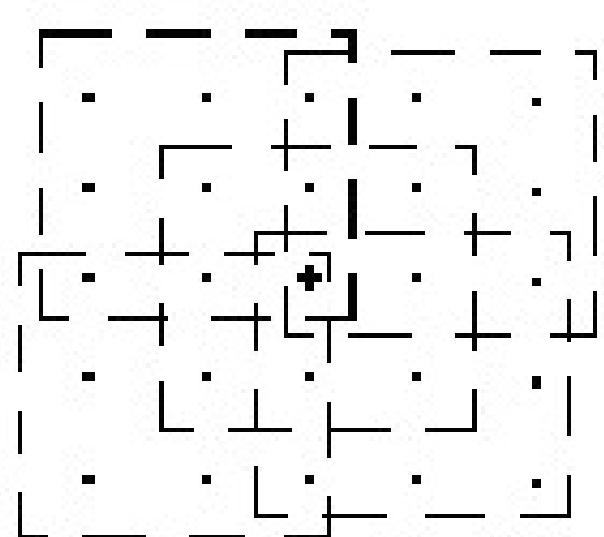
该算法的出发点是：在 $n \times n$ 的窗口内，属于同一集合体的像素，它们的灰度值将高度相关。因此，可用窗口内与中心像素的灰度最接近的K个邻像素的平均灰度来代替窗口中心像素的灰度值。这就是**灰度K近邻平均法**。

较小的K值使噪声方差下降较小，但保持细节效果较好；而较大的K值平滑噪声较好，但会使图像边缘模糊。

实验证明，对于 $3 \times 3$ 的窗口，取K=6为宜。

### 5.2.4 最大均匀性平滑

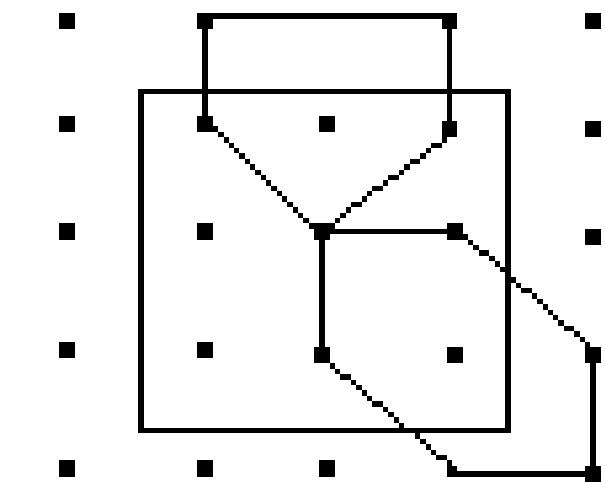
为避免消除噪声引起边缘模糊，该算法先找出环绕图像中每像素的最均匀区域，然后用这区域的灰度均值代替该像素原来的灰度值。



### 5.2.5 有选择保边缘平滑法

该方法对图像上任一像素 $(x, y)$ 的 $5 \times 5$ 邻域，采用9个掩模，其中包括一个 $3 \times 3$ 正方形、4个五边形和4个六边形。计算各个掩模的均值和方差，对方差进行排序，**最小方差所对应的掩模区的灰度均值就是像素 $(x, y)$ 的输出值。**

该方法以方差作为各个区域灰度均匀性的测度。若区域含有尖锐的边缘，它的灰度方差必定很大，而不含边缘或灰度均匀的区域，它的方差就小，那么最小方差所对应的区域就是灰度最均匀区域。因此**有选择保边缘平滑法既能够消除噪声，又不破坏区域边界的细节**。另外，五边形和六边形在 $(x, y)$ 处都有锐角，这样，即使像素 $(x, y)$ 位于一个复杂形状区域的锐角处，也能找到均匀的区域。从而在平滑时既不会使尖锐边缘模糊，也不会破坏边缘形状。



例如，某像素 $5 \times 5$ 邻域的灰度分布如图4.2.4，经计算9个掩模区的均值和方差为

均值	4	4	3	2	3	4	2	3	3	3	6	4	2	1
对应的 方差	54	7	17	17	28	31	23	26	0	4	7	3	2	4

8	4	1 <sup>a</sup>	4	3	6
4	2	1	5	3	
4	3	2	1	6	

最小方差为0，对应的灰度均值3，  
采用有选择保边缘平滑，该像素的输出值为3。

## 5.2.6 空间低通滤波法

邻域平均法可看作一个掩模作用于图像 $f(x,y)$ 的低通空间滤波，掩模就是一个滤波器，它的响应为 $H(r, s)$ ，于是滤波输出的数字图像 $g(x,y)$ 用离散卷积表示为

$$g(x, y) = \sum_{r=-k}^k \sum_{s=-l}^l f(x-r, y-s) H(r, s) \quad (5-15)$$

常用的掩模有

$$H_1 = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$H_2 = \frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$H_3 = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

$$H_4 = \frac{1}{8} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$H_5 = \frac{1}{2} \begin{bmatrix} 0 & \frac{1}{4} & 0 \\ \frac{1}{4} & 1 & \frac{1}{4} \\ 0 & \frac{1}{4} & 0 \end{bmatrix}$$

掩模不同，中心点或邻域的重要程度也不相同，因此，应根据问题的需要选取合适的掩模。但不管什么样的掩模，必须保证全部权系数之和为单位值，这样可保证输出图像灰度值在许可范围内，不会产生“溢出”现象。

## 5.2.7 中值滤波

中值滤波是对一个滑动窗口内的诸像素灰度值排序，用中值代替窗口中心像素的原来灰度值，因此它是一种非线性的图像平滑法。

例：采用 $1 \times 3$ 窗口进行中值滤波

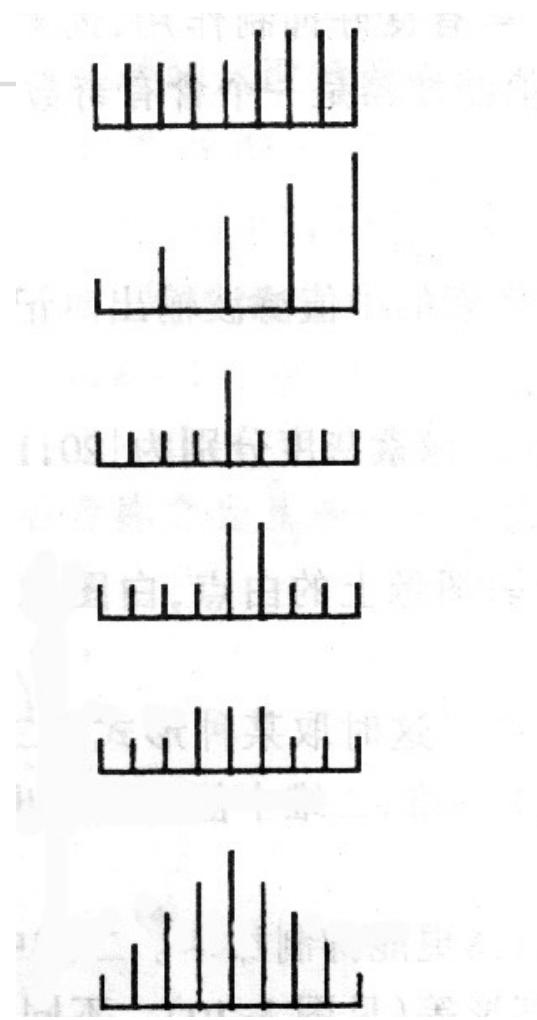
原图像为： 2 2 6 2 1 2 4 4 4 2 4

处理后为： 2 2 2 2 2 2 4 4 4 4 4

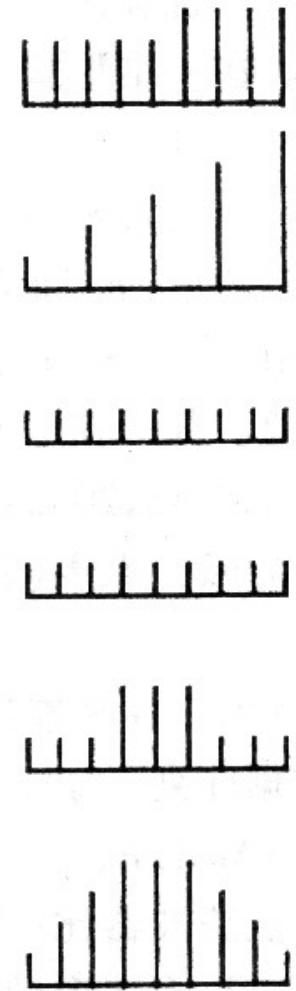
它对脉冲干扰及椒盐噪声的抑制效果好，在抑制随机噪声的同时能有效保护边缘少受模糊。但它对点、线等细节较多的图像却不太合适。

对中值滤波法来说，正确选择窗口尺寸的大小是很重要的环节。一般很难事先确定最佳的窗口尺寸，需通过从小窗口到大窗口的中值滤波试验，再从中选取最佳的。

原图像



中值滤波

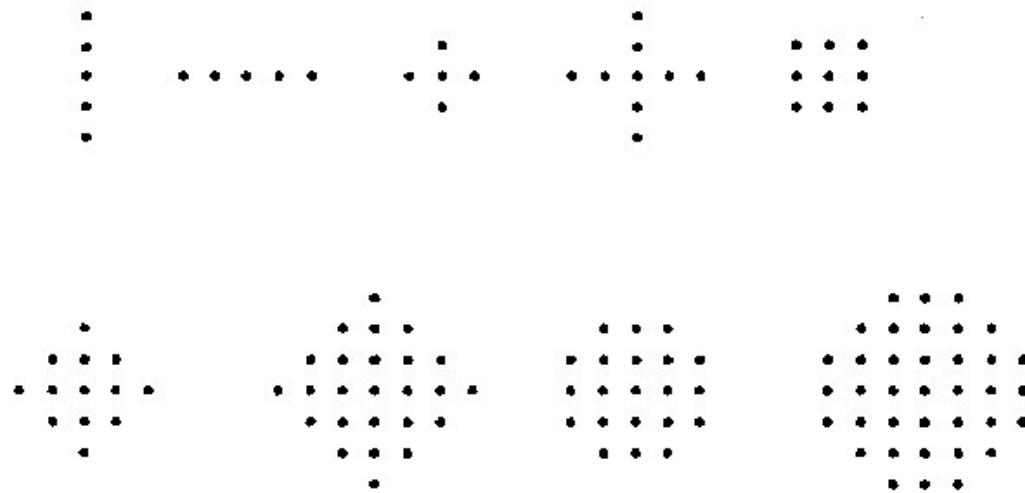


一维中值滤波的几个例子 ( $N=5$ )

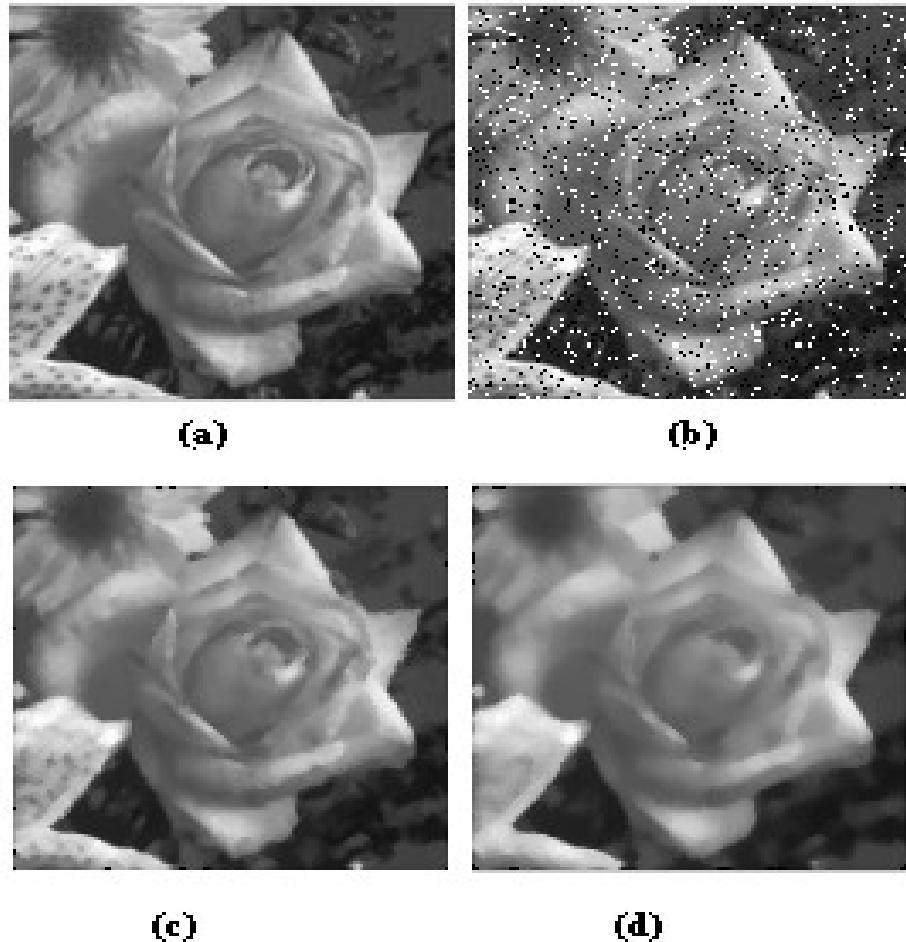
离散阶跃信号、斜坡信号没有受到影响。离散三角信号的顶部则变平了。对于离散的脉冲信号，当其连续出现的次数小于窗口尺寸的一半时，将被抑制掉，否则将不受影响。

一维中值滤波的概念很容易推广到二维。一般来说，二维中值滤波器比一维滤波器更能抑制噪声。

二维中值滤波器的窗口形状可以有多种，如线状、方形、十字形、圆形、菱形等（见图）。



不同形状的窗口产生不同的滤波效果，使用中必须根据图像的内容和不同的要求加以选择。从以往的经验看，方形或圆形窗口适宜于外轮廓线较长的物体图像，而十字形窗口对有尖顶角状的图像效果好。

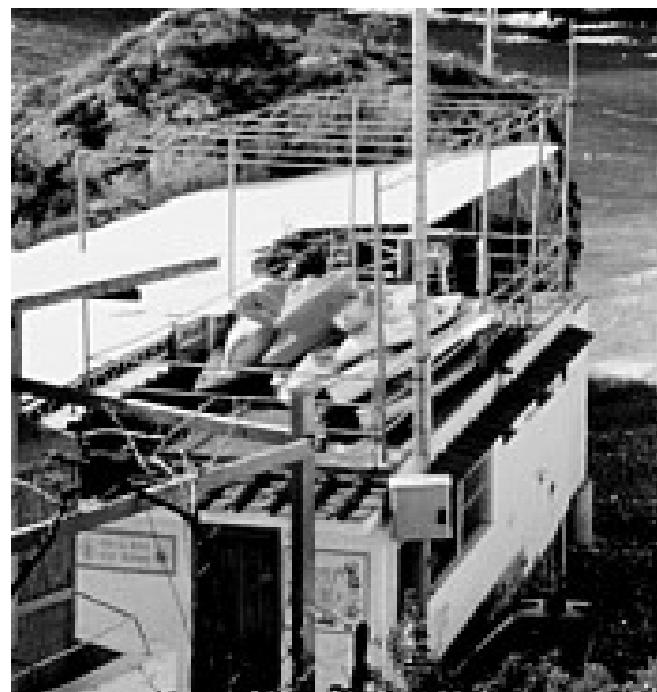


图(a)为原图像；图(b)为加椒盐噪声的图像；图(c)和图(d)分别为 $3 \times 3$ 、 $5 \times 5$ 模板进行中值滤波的结果。

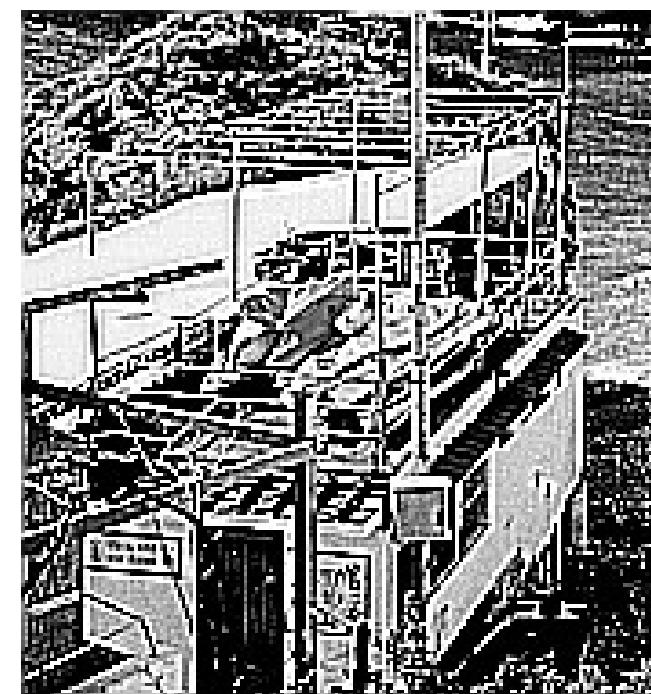
可见中值滤波法能有效削弱椒盐噪声，且比邻域、超限像素平均法更有效。

---

## 5.3 图像空间域锐化



$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$



在图像的识别中常需要突出边缘和轮廓信息。**图像锐化**就是增强图像的边缘或轮廓。

图像平滑通过积分过程使得图像边缘模糊，图像锐化则通过微分而使图像边缘突出、清晰。

### 5.3.1 梯度锐化法

图像锐化法最常用的是梯度法。对于图像 $f(x, y)$ ，在 $(x, y)$ 处的梯度定义为

$$\text{grad}(x, y) = \begin{bmatrix} f'_x \\ f'_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f(x, y)}{\partial x} \\ \frac{\partial f(x, y)}{\partial y} \end{bmatrix} \quad (7.3-1)$$

梯度是一个矢量，其大小和方向为

$$\begin{aligned} \text{grad}(x, y) &= \sqrt{f'_x^2 + f'_y^2} = \sqrt{\left(\frac{\partial f(x, y)}{\partial x}\right)^2 + \left(\frac{\partial f(x, y)}{\partial y}\right)^2} \\ \theta &= \tan^{-1}(f'_y / f'_x) = \tan^{-1}\left(\frac{\partial f(x, y)}{\partial y} / \frac{\partial f(x, y)}{\partial x}\right) \end{aligned} \quad (7.3-2)$$

对于离散图像处理而言，常用到梯度的大小，因此把梯度的大小习惯称为“梯度”。并且一阶偏导数采用一阶差分近似表示，即

$$f_x' = f(x+1, y) - f(x, y)$$

$$f_y' = f(x, y+1) - f(x, y)$$

为简化梯度的计算，经常使用

$$\text{grad}(x, y) = \text{Max}(|f_x'|, |f_y'|) \quad (5.3-4)$$

$$\text{或} \quad \text{grad}(x, y) = |f_x'| + |f_y'| \quad (5.3-5)$$

除梯度算子以外，还可采用 Roberts、Prewitt 和 Sobel 算子计算梯度，来增强边缘。

Roberts 对应的模板如图 5.3.2 所示。差分计算式如下

$$f_x' = |f(x+1, y+1) - f(x, y)|$$

$$f_y' = |f(x+1, y) - f(x, y+1)|$$

-1				-1
	1		1	

图 5.3.2 Roberts 梯度算子

为在锐化边缘的同时减少噪声的影响，Prewitt从加大边缘增强算子的模板大小出发，由 $2\times 2$ 扩大到 $3\times 3$ 来计算差分，如图(a)所示。

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

(a) Prewitt 算子

(b) Sobel算子

Sobel在Prewitt算子的基础上，对4-邻域采用带权的方法计算差分，对应的模板如图(b)。

根据梯度计算式就可以计算Roberts、Prewitt和Sobel梯度。一旦梯度算出后，就可根据不同的需要生成不同的梯度增强图像。

## 第一种输出形式

$$g(x,y) = \text{grad}(x,y) \quad (5.3-7)$$

此法的缺点是增强的图像仅显示灰度变化比较陡的边缘轮廓，而灰度变化比较平缓或均匀的区域则呈黑色。

## 第二种输出形式

$$g(x,y) = \begin{cases} \text{grad}(x,y), & \text{grad}(x,y) \geq T \\ f(x,y), & \text{其它} \end{cases}$$

式中T是一个非负的阈值。适当选取T，可使明显的边缘轮廓得到突出，又不会破坏原来灰度变化比较平缓的背景

## 第三种输出形式

$$g(x,y) = \begin{cases} L_G, & \text{grad}(x,y) \geq T \\ f(x,y), & \text{其他} \end{cases}$$

它将明显边缘用一固定的灰度级 $L_G$ 来表现。

#### 第四种输出形式

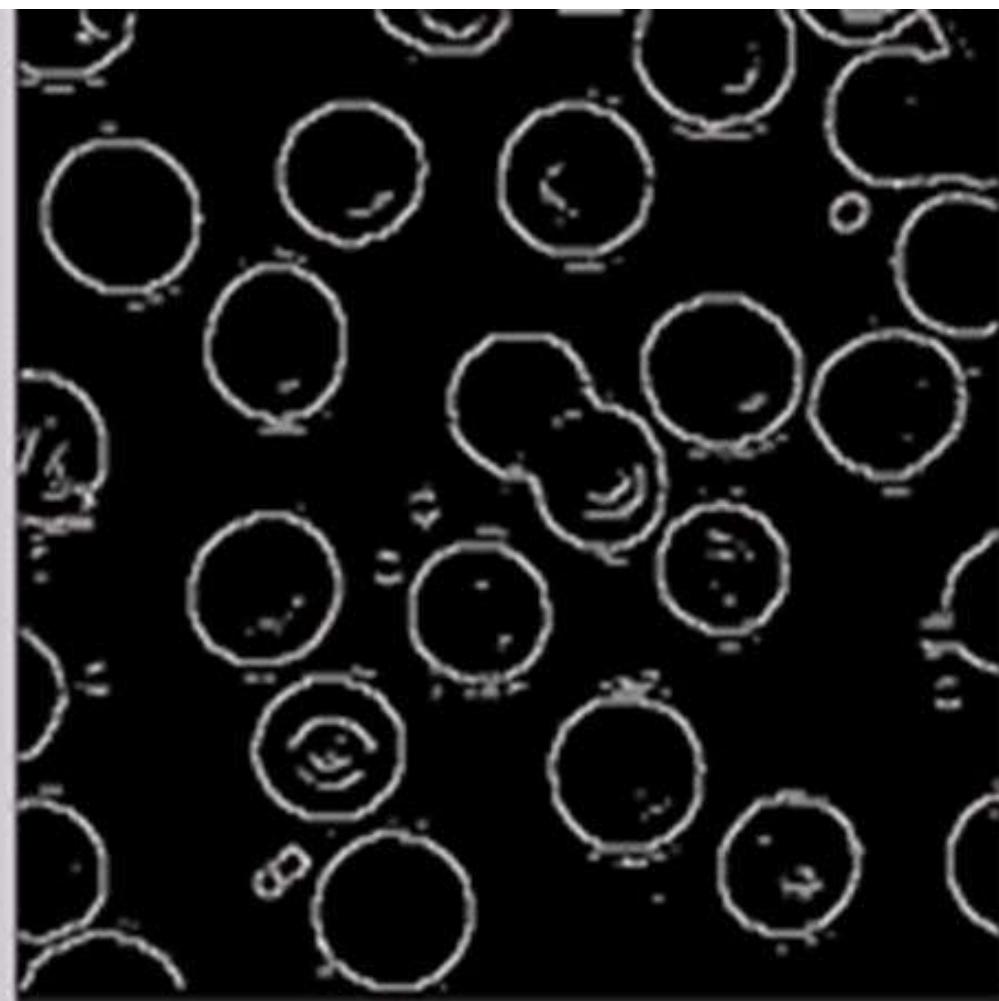
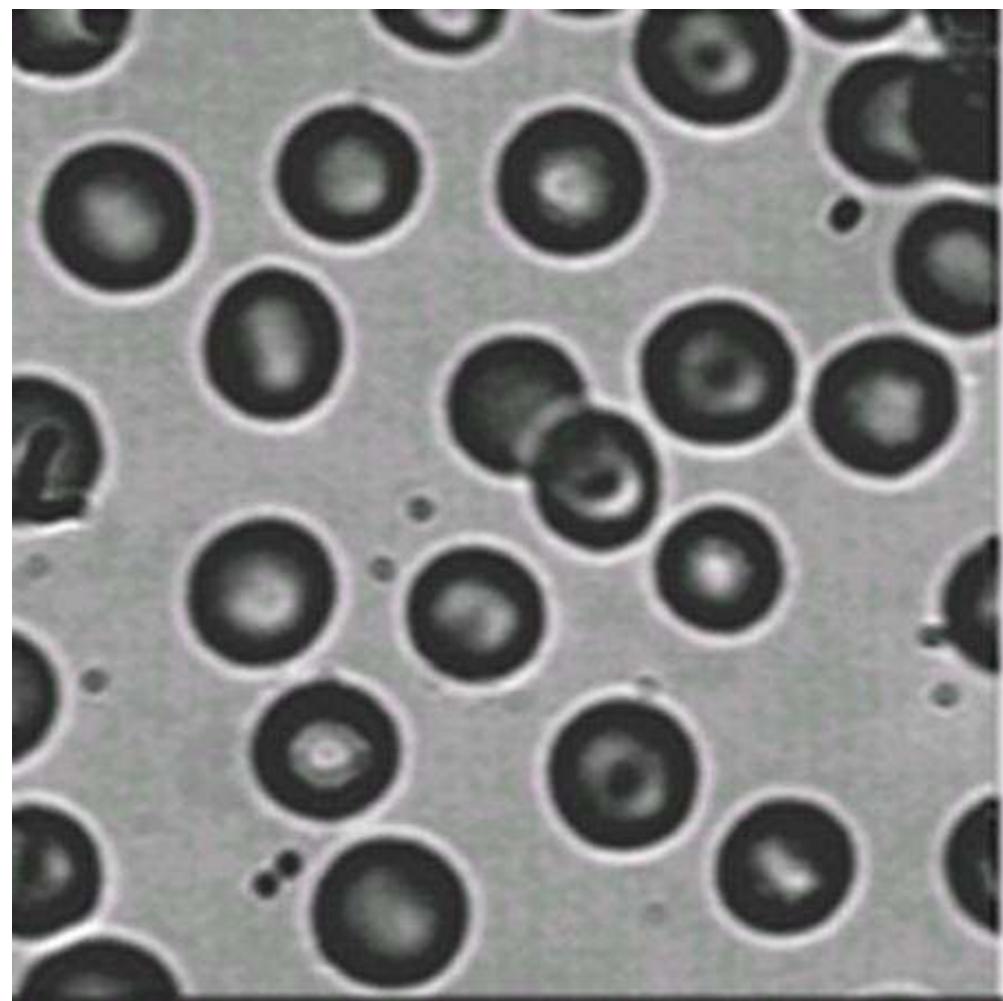
$$g(x, y) = \begin{cases} grad(x, y) & , grad(x, y) \geq T \\ L_B, & \text{其他} \end{cases}$$

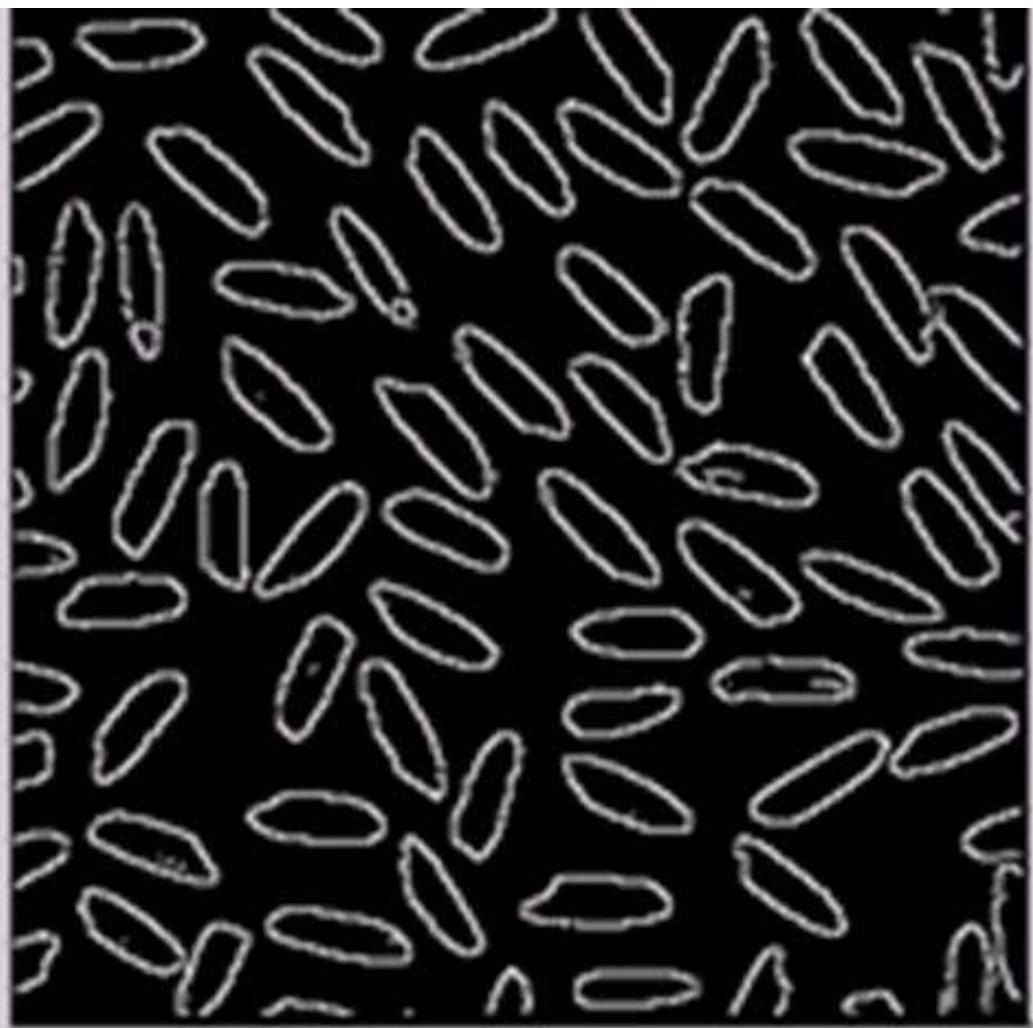
此方法将背景用一个固定的灰度级  $L_B$  来表现，便于研究边缘灰度的变化。

#### 第五种输出形式

$$g(x, y) = \begin{cases} L_G & , grad(x, y) \geq T \\ L_B, & \text{其他} \end{cases}$$

这种方法将明显边缘和背景分别用灰度级  $L_G$  和  $L_B$  表示，生成二值图像，便于研究边缘所在位置。







(a)



(b)



(c)



(d)



(e)



(f)

## 5.3.2 Laplacian增强算子

Laplacian 算子是线性二阶微分算子。即

$$\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2}$$

对离散的数字图像而言，二阶偏导数可用二阶差分近似，可推导出Laplacian算子表达式为

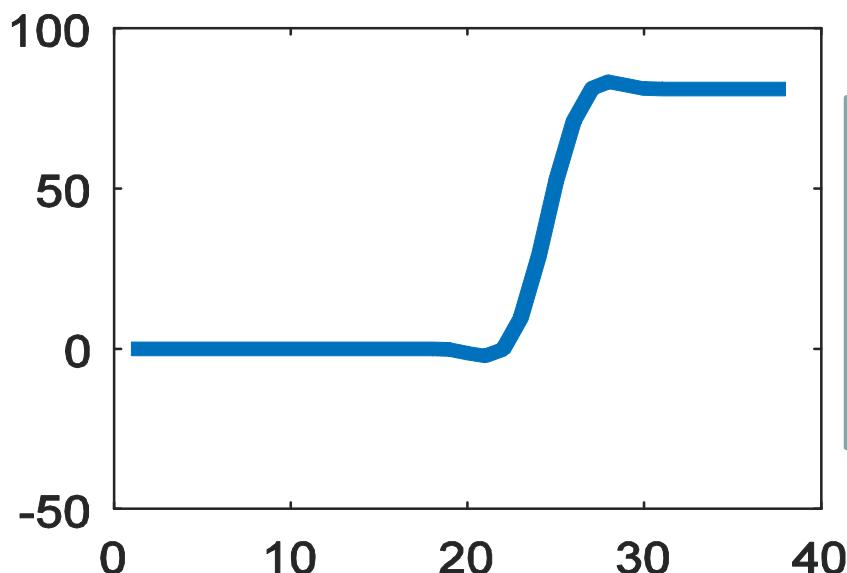
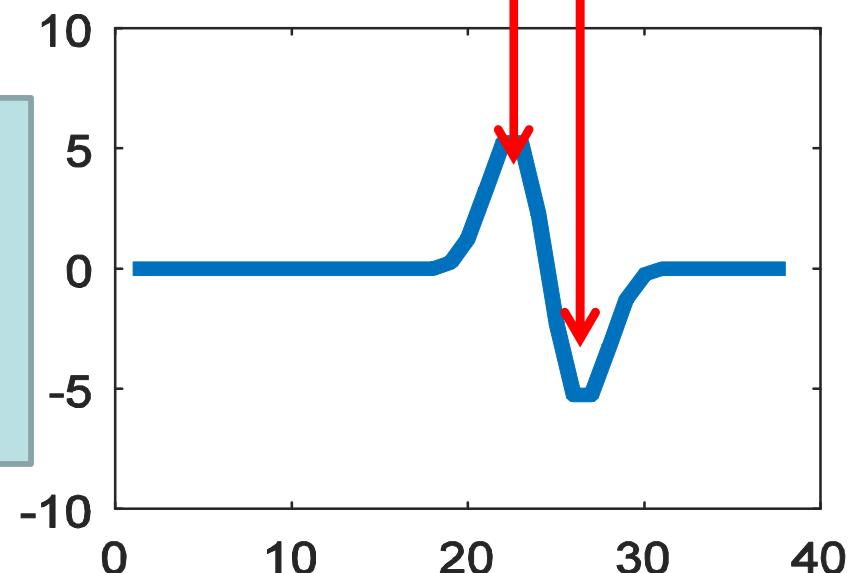
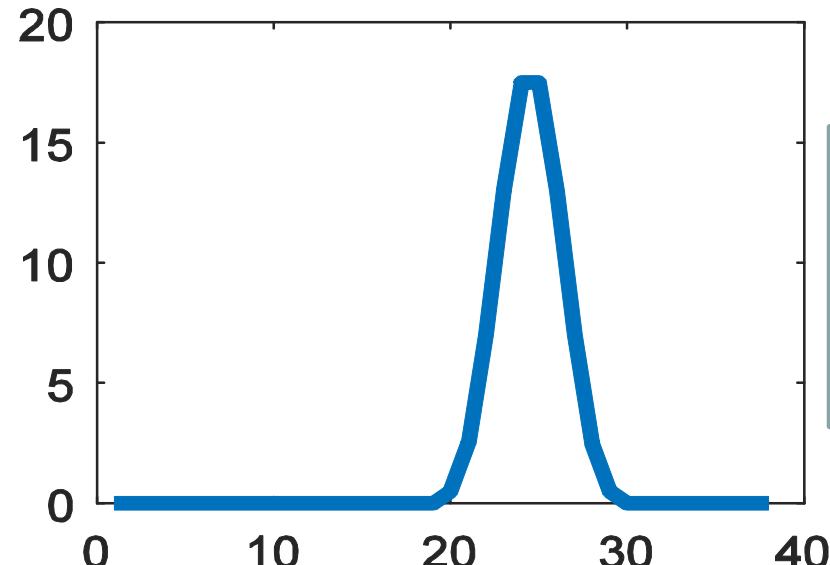
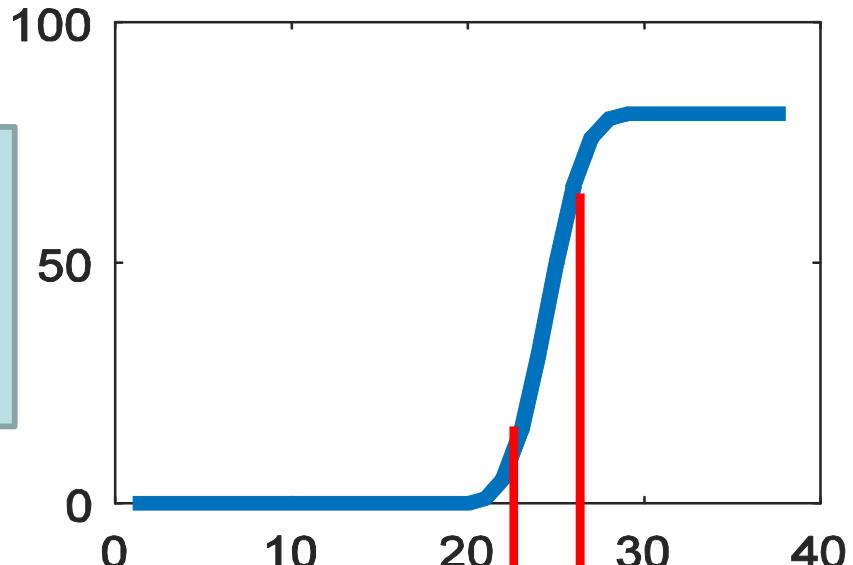
$$\begin{aligned}\nabla^2 f(x, y) &= f(x+1, y) + f(x-1, y) + \\ &\quad f(x, y+1) + f(x, y-1) - 4f(x, y)\end{aligned}$$

Laplacian增强算子为：

$$\begin{aligned}g(x, y) &= f(x, y) - \nabla^2 f(x, y) \\ &= 5f(x, y) - [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)]\end{aligned}$$

0	1	0
1	-4	1
0	1	0
Laplace算子		

0	-1	0
-1	5	-1
0	-1	0
增强算子		



Laplacian 增强示意

其特点是：

- 1、在灰度均匀的区域或斜坡中间  $\nabla^2 f(x, y)$  为0，增强图像上像元灰度不变；
- 2、在斜坡底或低灰度侧形成“下冲”；而在斜坡顶或高灰度侧形成“上冲”。

### 7.3.3 高通滤波法

高通滤波法就是用高通滤波算子和图像卷积来增强边缘。常用的算子有：

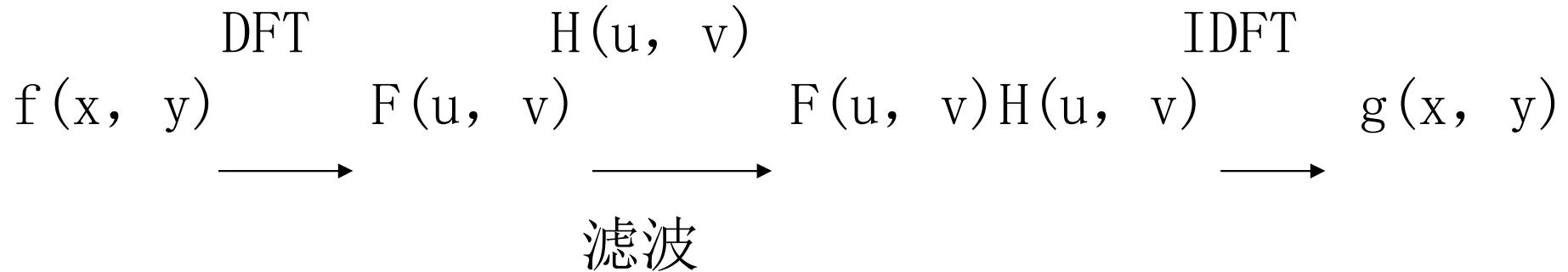
$$H1 = \begin{matrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{matrix} \quad H2 = \begin{matrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{matrix}$$

## 5.4 图像的频率域增强

图像增强的主要目的包括：①消除噪声，改善图像的视觉效果；②突出边缘，有利于识别和处理。

假定原图像为 $f(x, y)$ ，经傅立叶变换为 $F(u, v)$ 。频率域增强就是选择合适的滤波器 $H(u, v)$ 对 $F(u, v)$ 的频谱成分进行处理，然后经逆傅立叶变换得到增强的图像 $g(x, y)$ 。

频率域增强的一般过程如下：



# 彩色图像增强

人眼的视觉特性：

- 分辨的灰度级介于十几到二十几级之间；
- 彩色分辨能力可达到灰度分辨能力的百倍以上。

彩色增强技术利用人眼的视觉特性，即对色彩的分辨能力远远大于对灰度级的分辨能力，将灰度图像变为彩色图像以及进行各种彩色变换，改善图像的可视性。

可分为：

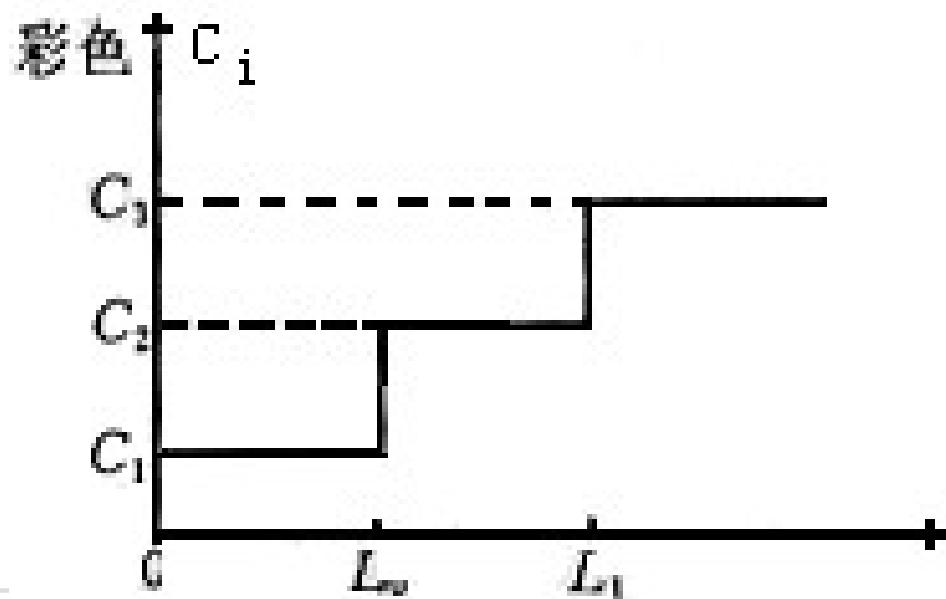
- 伪彩色增强
  - 彩色图像增强
- 方法分类

密度分割  
灰度级-彩色变换  
频域滤波

# 1. 密度分割法

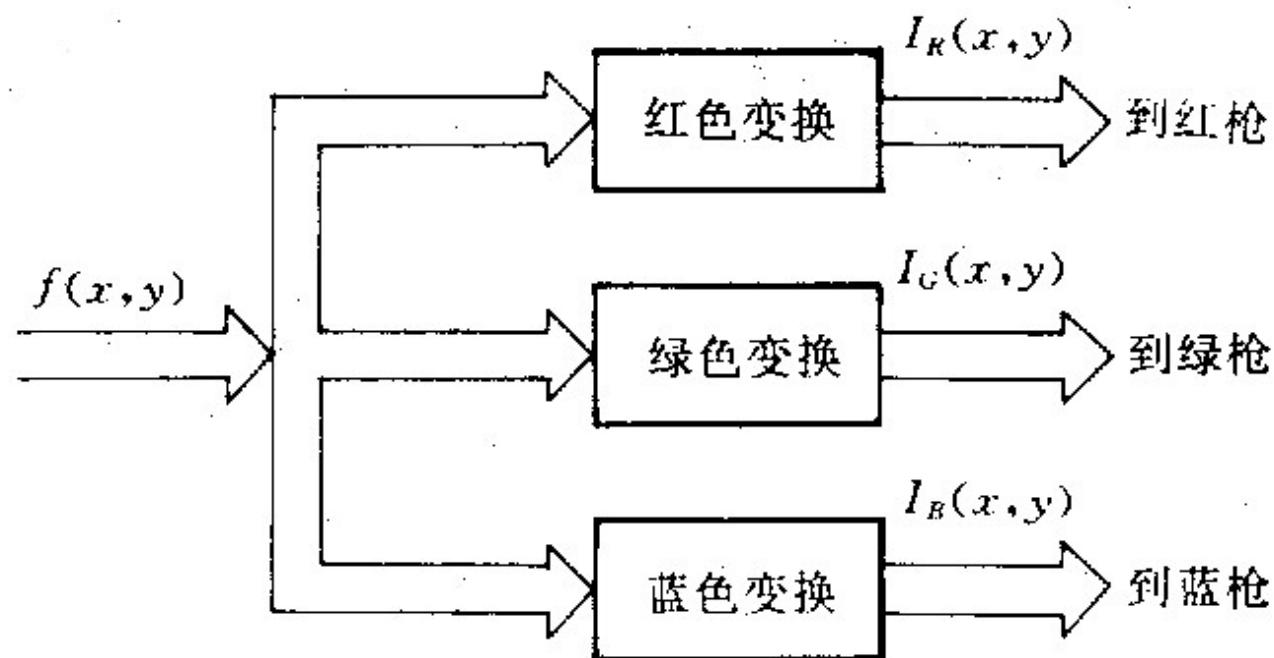
密度分割法是把黑白图像的灰度级从0（黑）到 $M_0$ （白）分成N个区间 $I_i$  ( $i=1, 2, \dots, N$ )，给每个区间 $I_i$ 指定一种彩色 $C_i$ ，这样，便可以把一幅灰度图像变成一幅伪彩色图像。

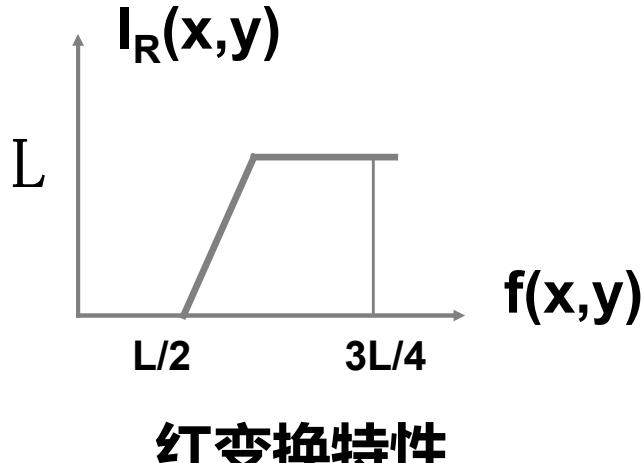
该方法比较简单、直观。缺点是变换出的彩色数目有限。



- 
- 密度分割中的彩色是人为赋予的，与场景的真实色彩毫无关系。
  - 灰度图像经过密度分割后，图像的可分辨力得到明显提高，如果分层方案与场景目标的光谱特性差异对应较好，可以较准确的区分出目标类别。

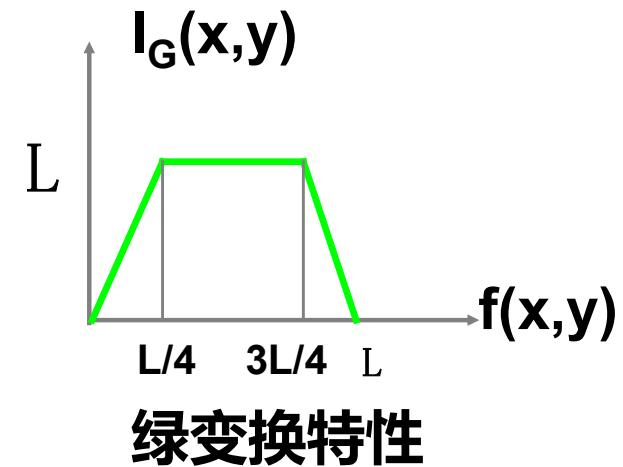
## 2. 空间域灰度级一彩色变换合成法



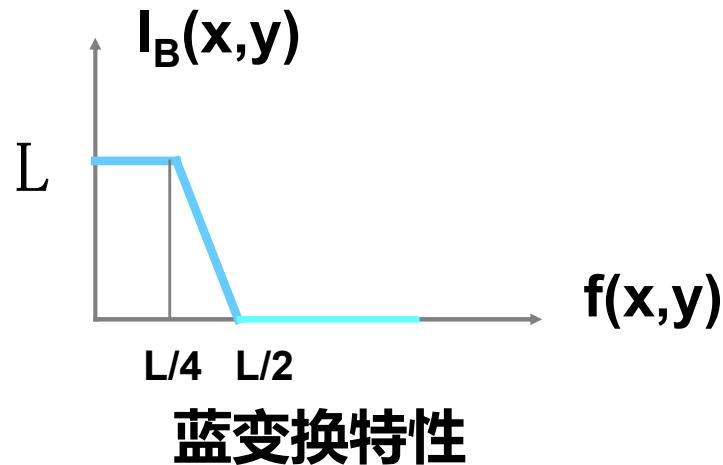


**红变换特性**

将低于 $L/2$ 地所有灰度  
映射成最暗的红色，  
在 $L/2$ 到 $3L/4$ 之间的灰  
度映射为线性增加饱  
和度的红色，在 $3L/4$   
到 $L$ 之间的灰度映射为  
最亮的红色

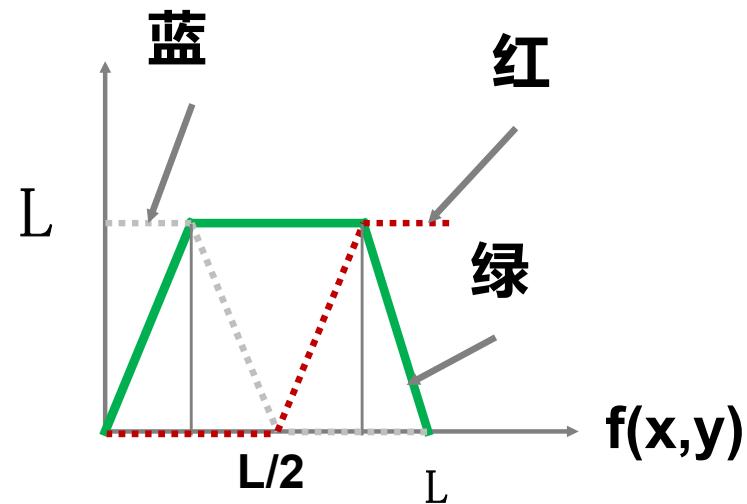


**绿变换特性**



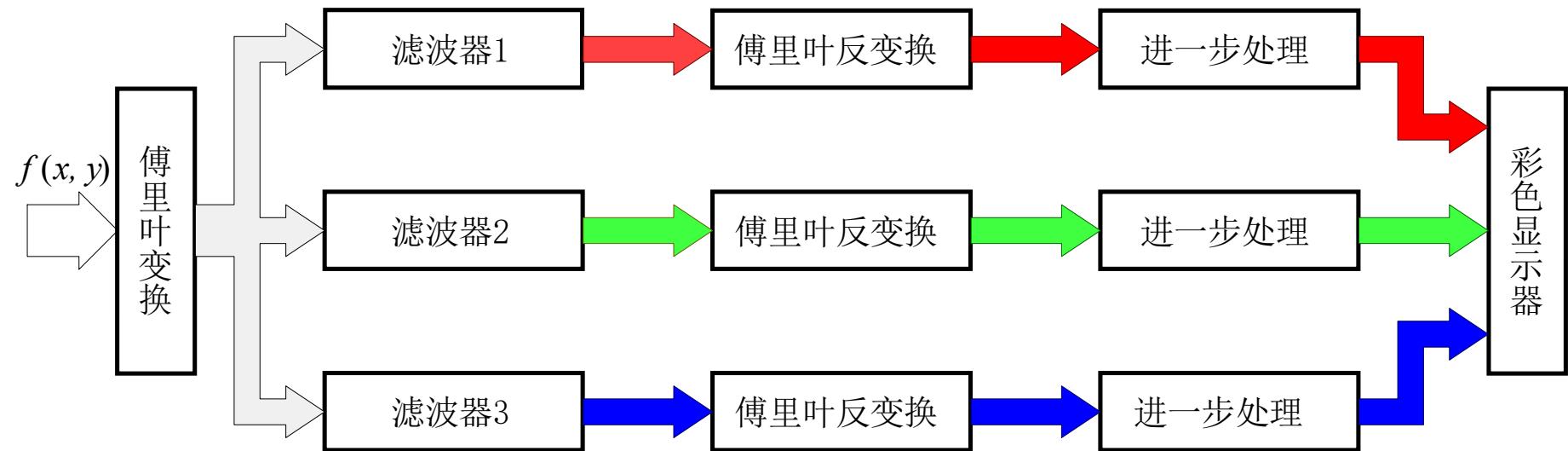
**典型的变换函数图**

# 三种变换合成特性



### 3. 频率域伪彩色增强

**基本思想：**对原来灰度图像中的不同频率分量赋予不同的颜色



# 伪彩色技术早期在遥感图像处理中得到广泛的应用



## 二、彩色图像增强

---

包括基于三基色的增强、基于色彩三要素的增强。

### 1. 假彩色增强

假彩色增强对一幅自然彩色图像或同一景物的多光谱图像，通过映射函数变换成新的三基色分量，彩色合成使感兴趣目标呈现出与原图像中不同的、奇异的彩色。

假彩色增强目的：一是使感兴趣的目标呈现出突出的彩色；一是使景物呈现出与人眼色觉相匹配的颜色，以提高对目标的分辨力。

多光谱图像的假彩色增强可表示为：

$$\left. \begin{aligned} R_F &= f_R \{g_1, g_2, \dots, g_i, \dots\} \\ G_F &= f_G \{g_1, g_2, \dots, g_i, \dots\} \\ B_F &= f_B \{g_1, g_2, \dots, g_i, \dots\} \end{aligned} \right\}$$

将可见光与非可见光波段结合起来，通过假彩色处理，就能获得更丰富的信息，便于对地物识别。

对于自然景色图像，通用的线性假彩色映射可表示为

$$\begin{bmatrix} R_F \\ G_F \\ B_F \end{bmatrix} = \begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{bmatrix} \cdot \begin{bmatrix} R_f \\ G_f \\ B_f \end{bmatrix}$$

例如采用以下的映射关系

$$\begin{bmatrix} R_F \\ G_F \\ B_F \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} R_f \\ G_f \\ B_f \end{bmatrix}$$

则原图像中绿色物体会呈红色，蓝色物体会呈绿色，红色物体则呈兰色。

伪彩色增强与假彩色增强有何区别？

## 2. 真彩色增强

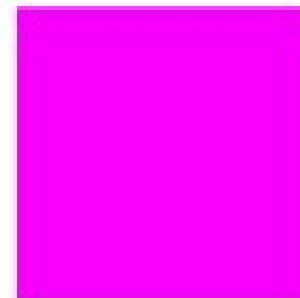
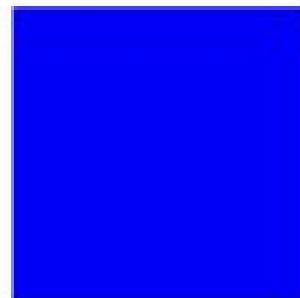
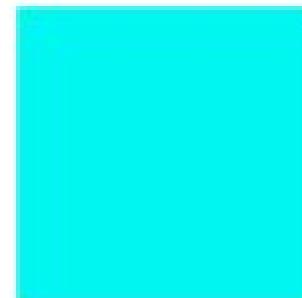
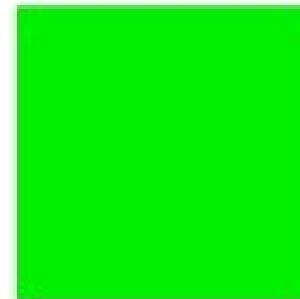
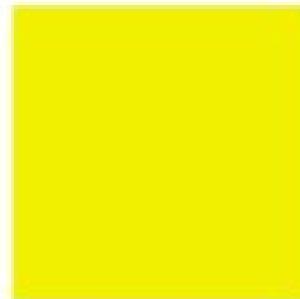
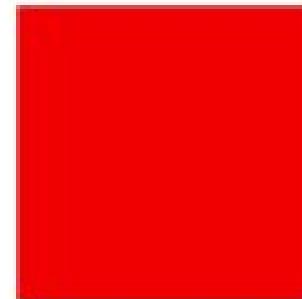
RGB模型 HIS模型

HIS模型

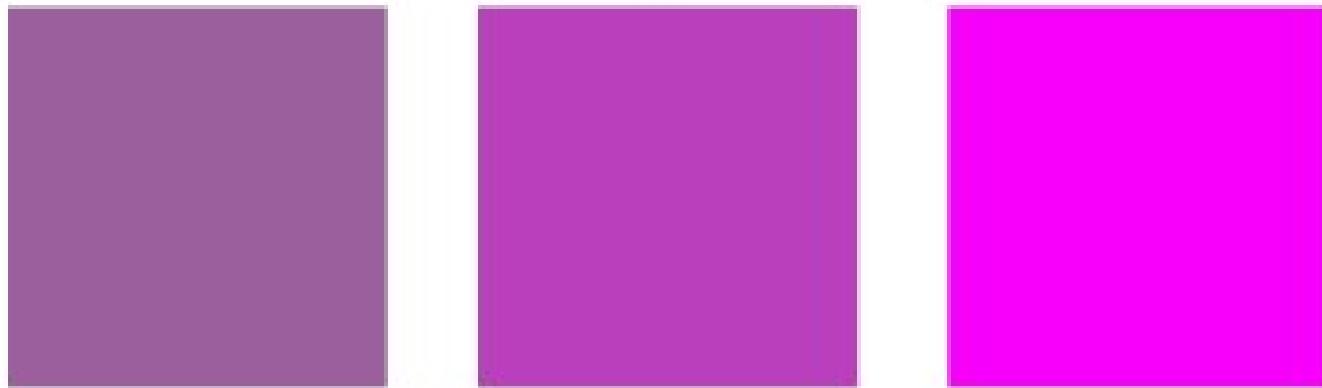
H-色度

I-亮度

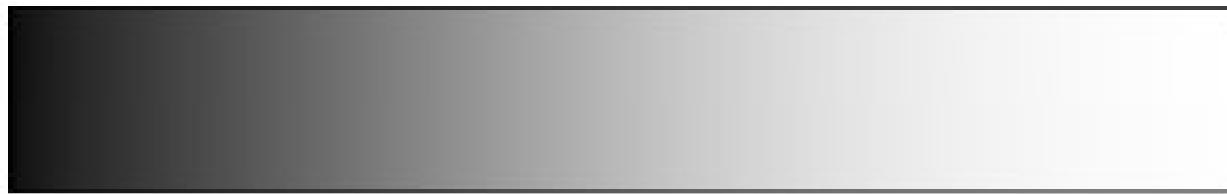
S-饱和度



## S-饱和度



## I-亮度



I: 小 → 大

## 方法：

- (1) 将R、G、B分量转换为I、H、S分量；
- (2) 利用对灰度图像增强的方法对I、H、S其中的某个分量进行增强；
- (3) 再将结果转换为R、G、B分量。

并不改变原图的彩色内容，但增强后的图看起来可能会有些不同。这是因为色调、饱和度或亮度分量得到了增强。

## 5.6 图像代数运算

---

### 一、概念

是指两幅输入图像之间进行点对点的加、减、乘、除算术运算，得到输出图像的过程。

运算是以图像为单位进行的操作，逐像素进行。

一般用于灰度图像。

(1) 加法：记为 $p+q$

(2) 减法：记为 $p-q$

(3) 乘法：记为 $p*q$

(4) 除法：记为 $p\div q$

- 两个像素的灰度值通过运算得到一个新的灰度值，作为对应新图像同位置处像素的灰度值。
- 新灰度值可能超出原图像的动态范围，需要进行灰度映射，来对运算结果的灰度值限制或调整到原图允许的动态范围内。

## 二、代数运算的应用

### 1. 加法运算的应用

图像平均（图像加法） ---- 图像平滑

目的：消除随机加性噪声

主要用于对同一场景的多幅图像求平均，以便有效地降低加性随机噪声。

多幅图像：相同条件（背景、光线等）下的不同时刻的图像

加性随机噪声：加性噪声和图像信号强度不相关

$$g(x,y) = f(x,y) + n(x,y)$$



噪声图像1



噪声图像2



噪声图像3



噪声图像4



噪声图像5



噪声图像6



噪声图像7



噪声图像8



原始图像



降噪后图像

## 2) 生成图像叠加效果

对于两个图像 $f(x, y)$ 和 $h(x, y)$ 的均值有：

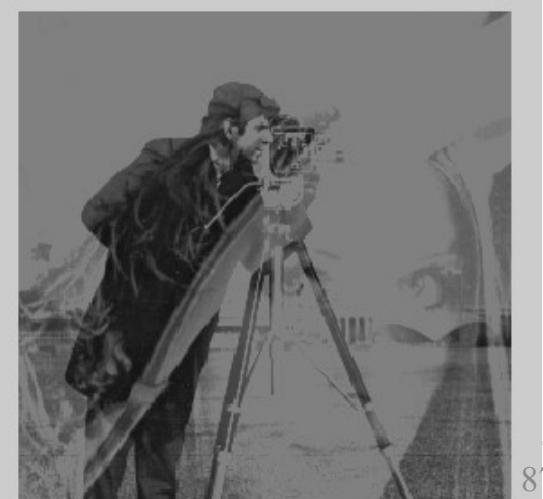
$$g(x, y) = 1/2f(x, y) + 1/2h(x, y)$$

推广这个公式为：

$$g(x, y) = \alpha f(x, y) + \beta h(x, y)$$

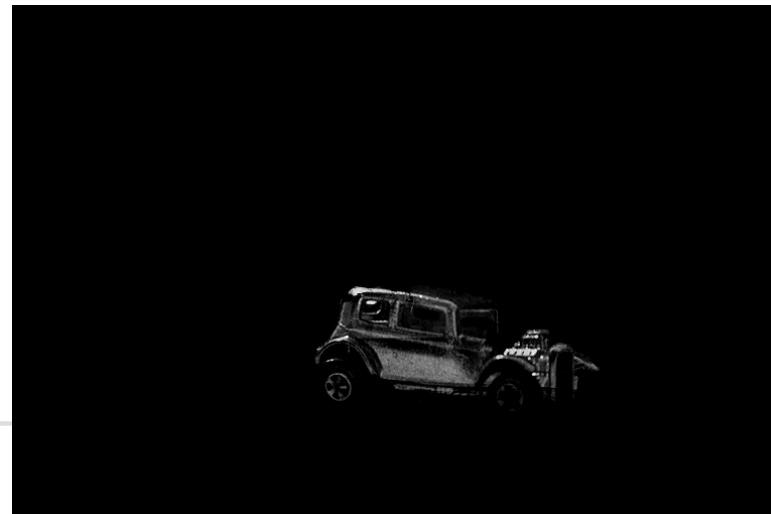
其中 $\alpha+\beta=1$

可以得到各种图像合成的效果

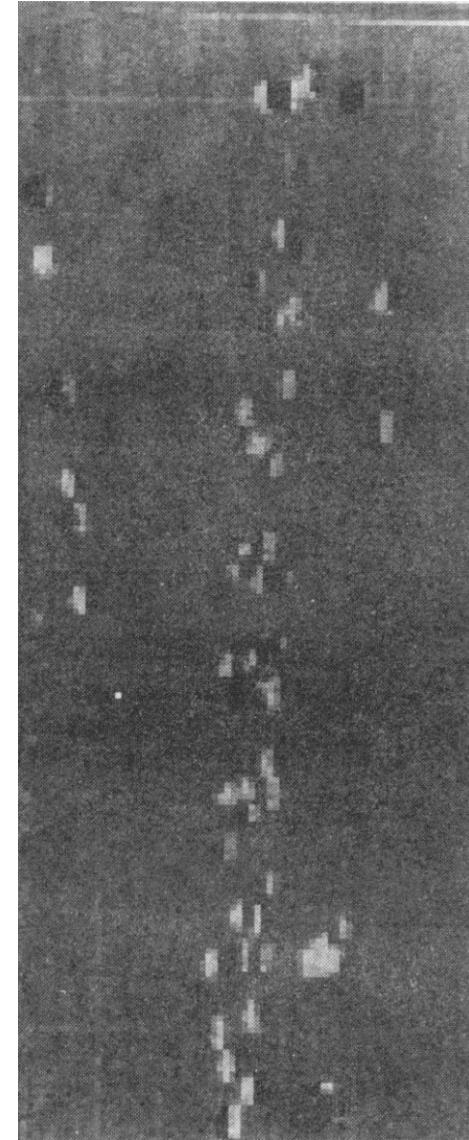
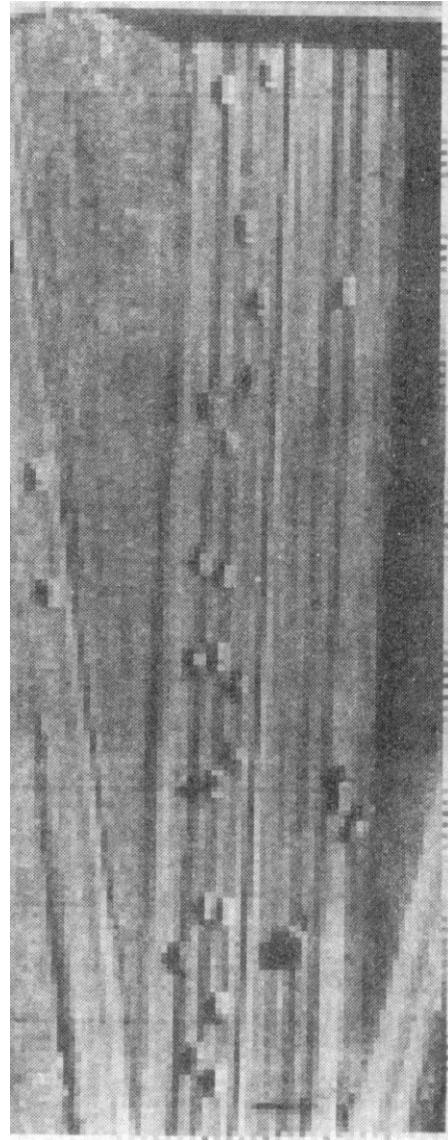
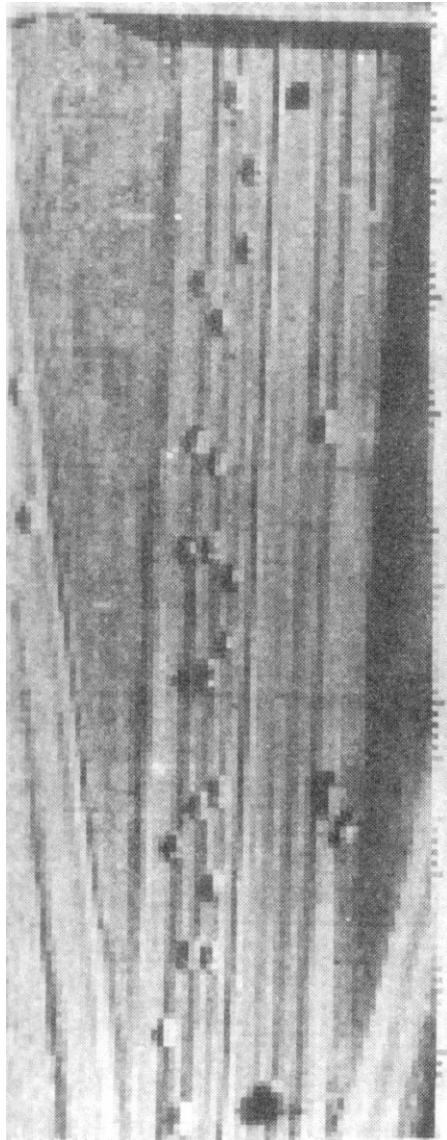


## 2. 减法运算的应用

1) 图像相减可去除图像中不需要的加性图案，如背景消除

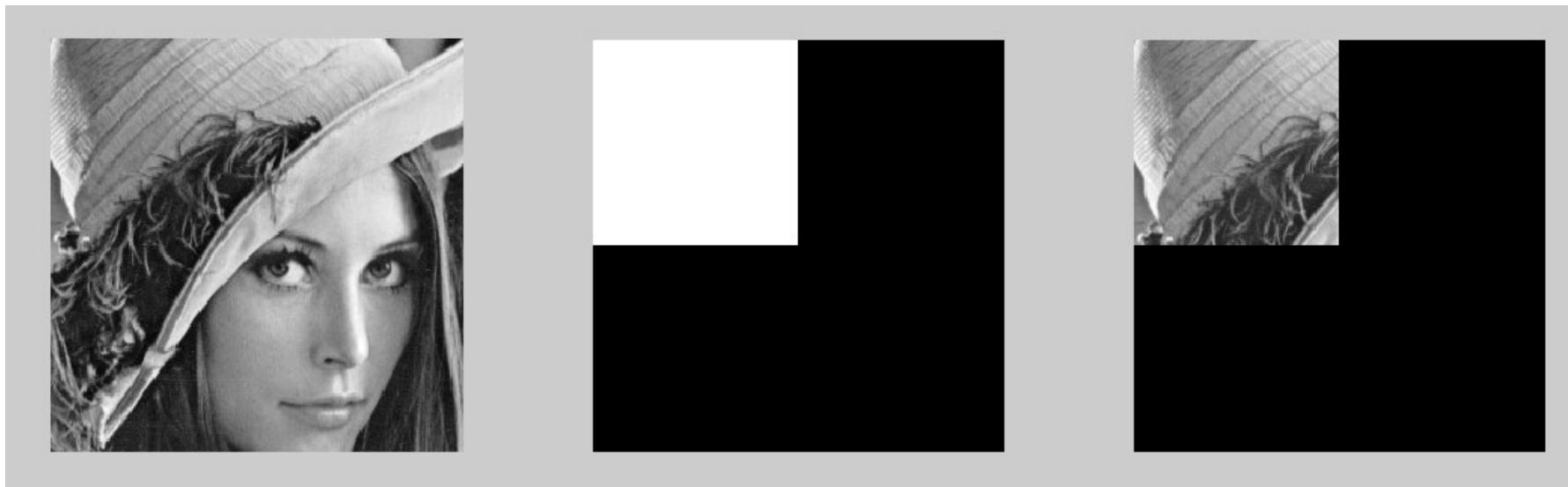


2) 图像相减也可用于动态监测、运动目标检测、  
目标识别



### 3.乘法运算的应用

用来遮掉图像的某些部分。

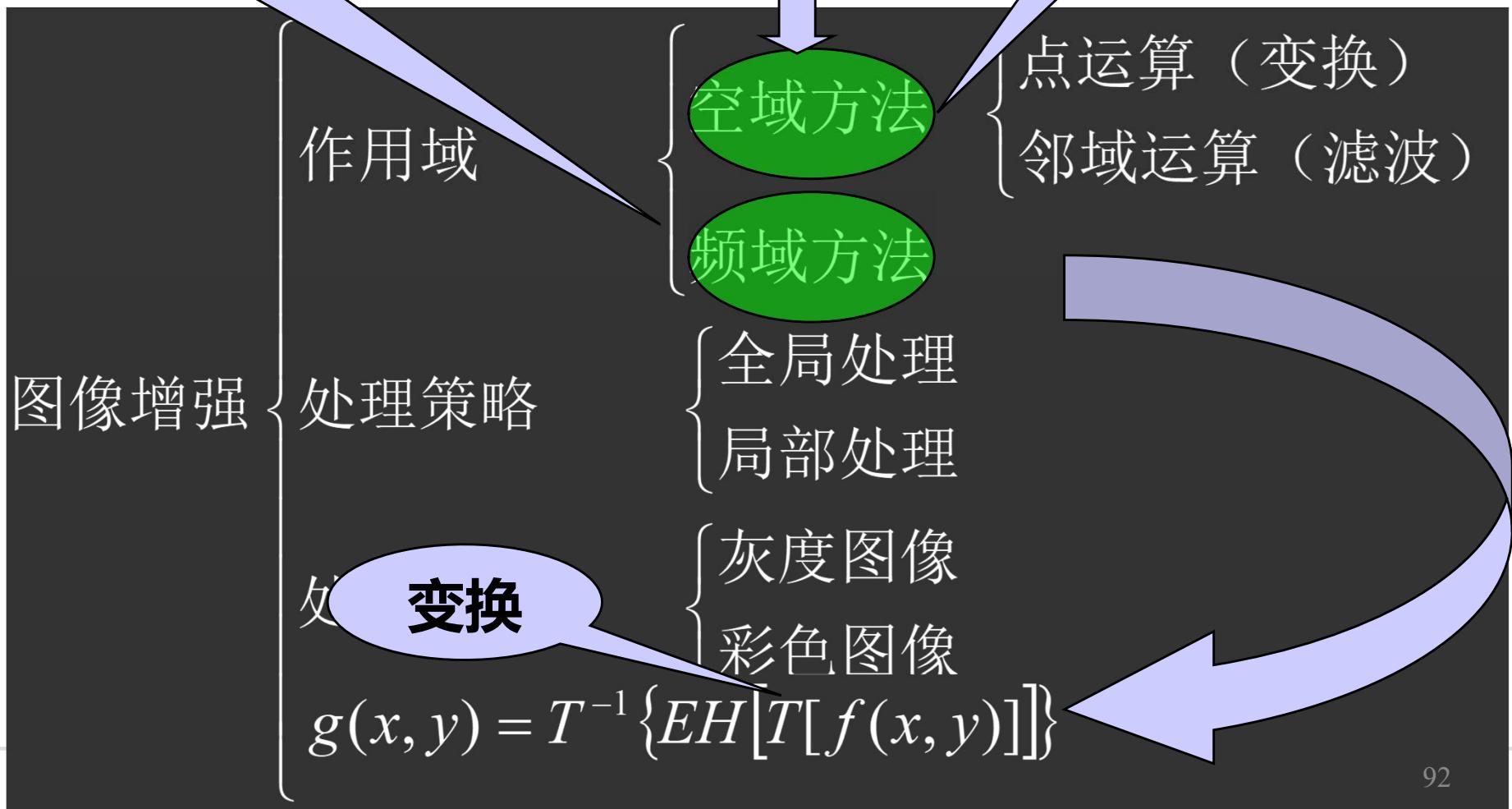


也称比值运算，在遥感图像处理中经常应用。比值运算可以降低传感器灵敏度随空间变化造成的影响，增强图像中特定的区域；降低地形导致的阴影影响，突出季节的差异。

地形部位	波段		
	TM1	TM2	TM1/TM2
阳坡	28	43	0.65
阴坡	22	34	0.65

## 直接对象素 灰度值运算

### 对图像进 行变换



# 基于深度学习的增强方法

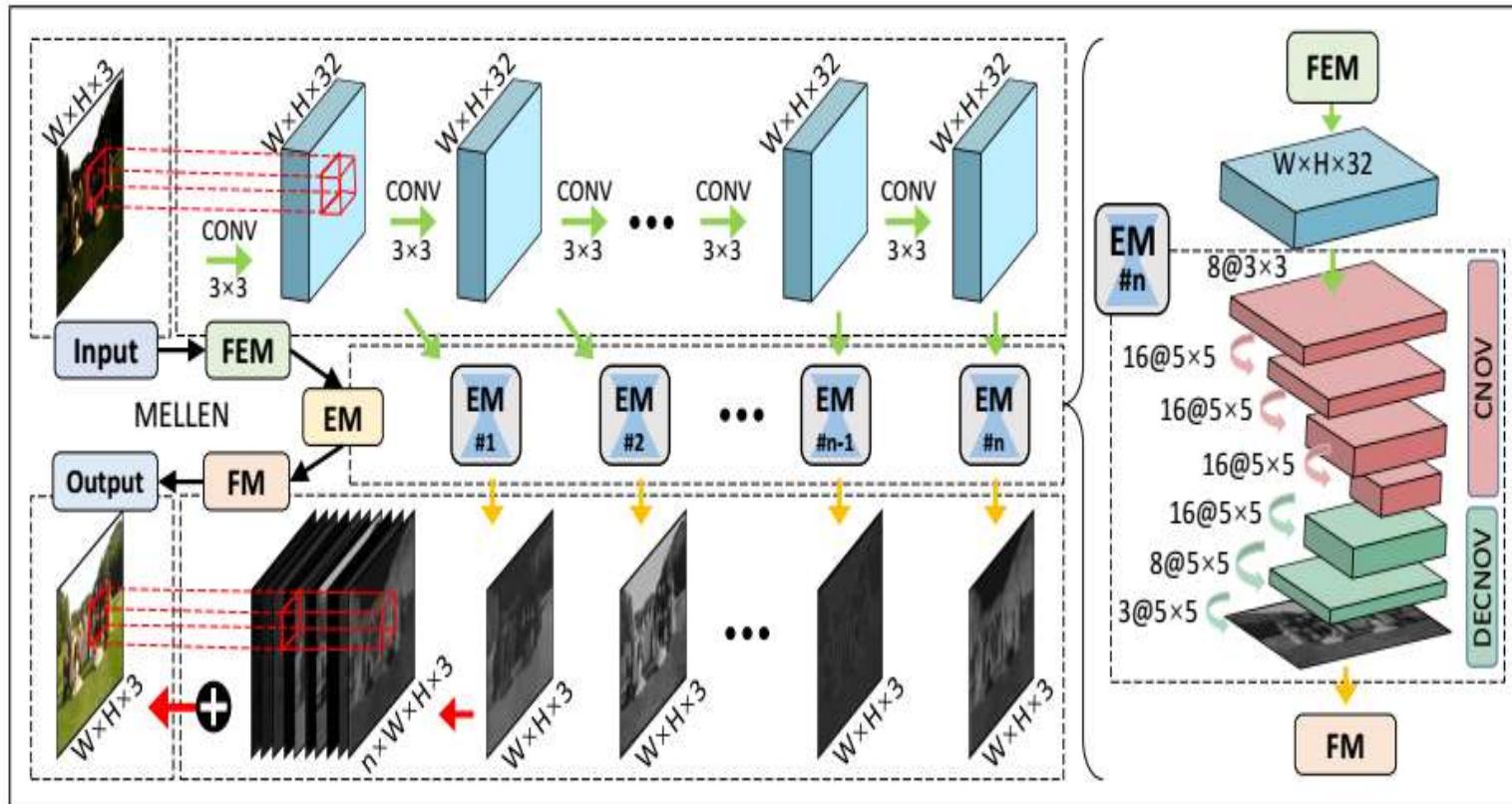
---

- 利用网络直接学习数据映射
  - CNN, GAN
- 结合经典图像模型或先验设计网络
  - Retinex, 暗通道

# 基于深度学习的图像增强

## 数据映射

- 利用(3D)卷积网络，实现图像/视频的端对端增强学习
- 可以挖掘不同层次的特征，并且适应视频多帧处理需求



MBLLEN: Low-light Image/Video Enhancement Using CNNs

# MBLLEN: Low-light Image/Video Enhancement Using CNNs

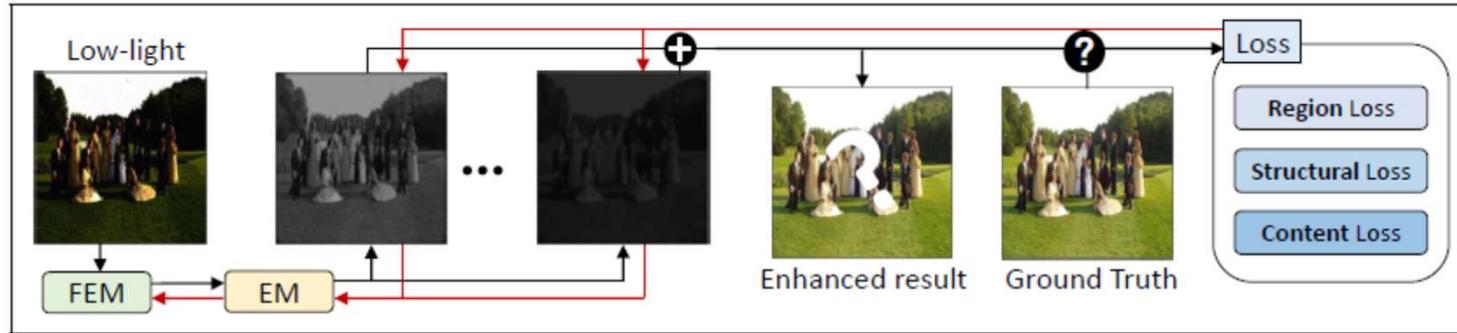


Figure 3: Data flow for training. The proposed loss function consists of three parts.

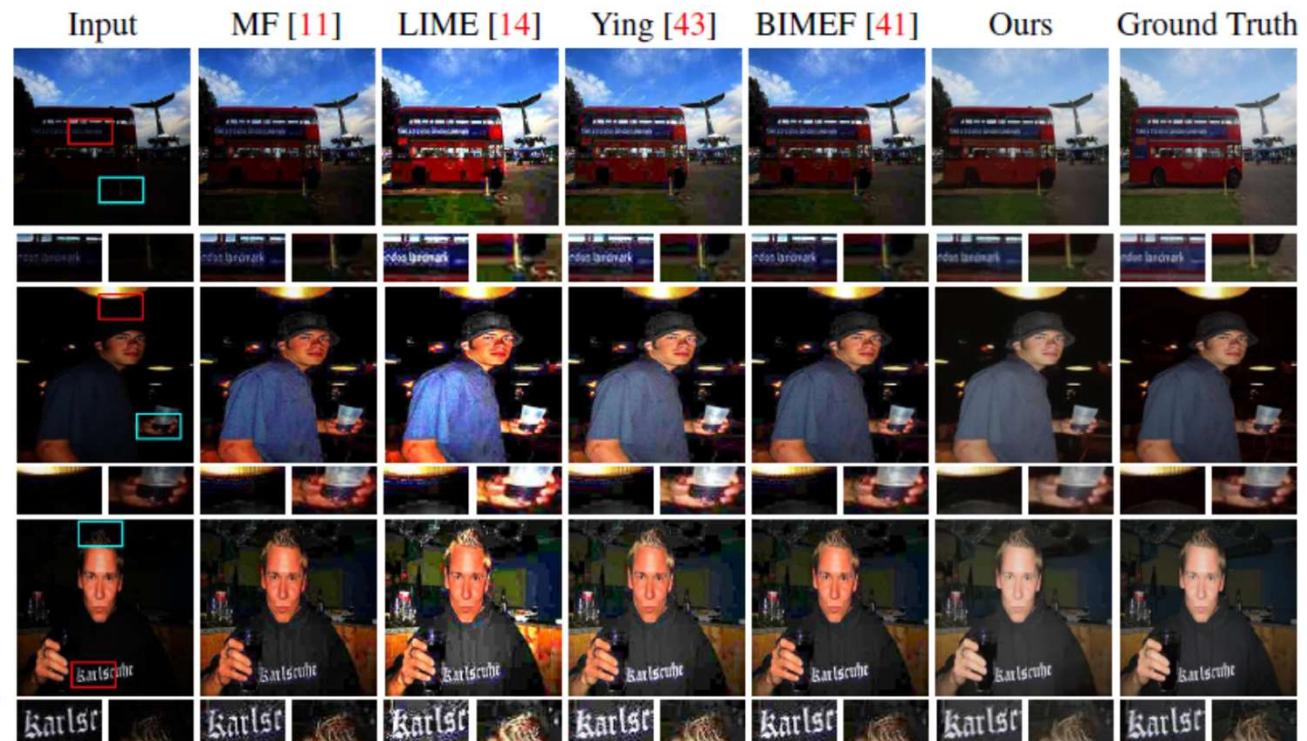
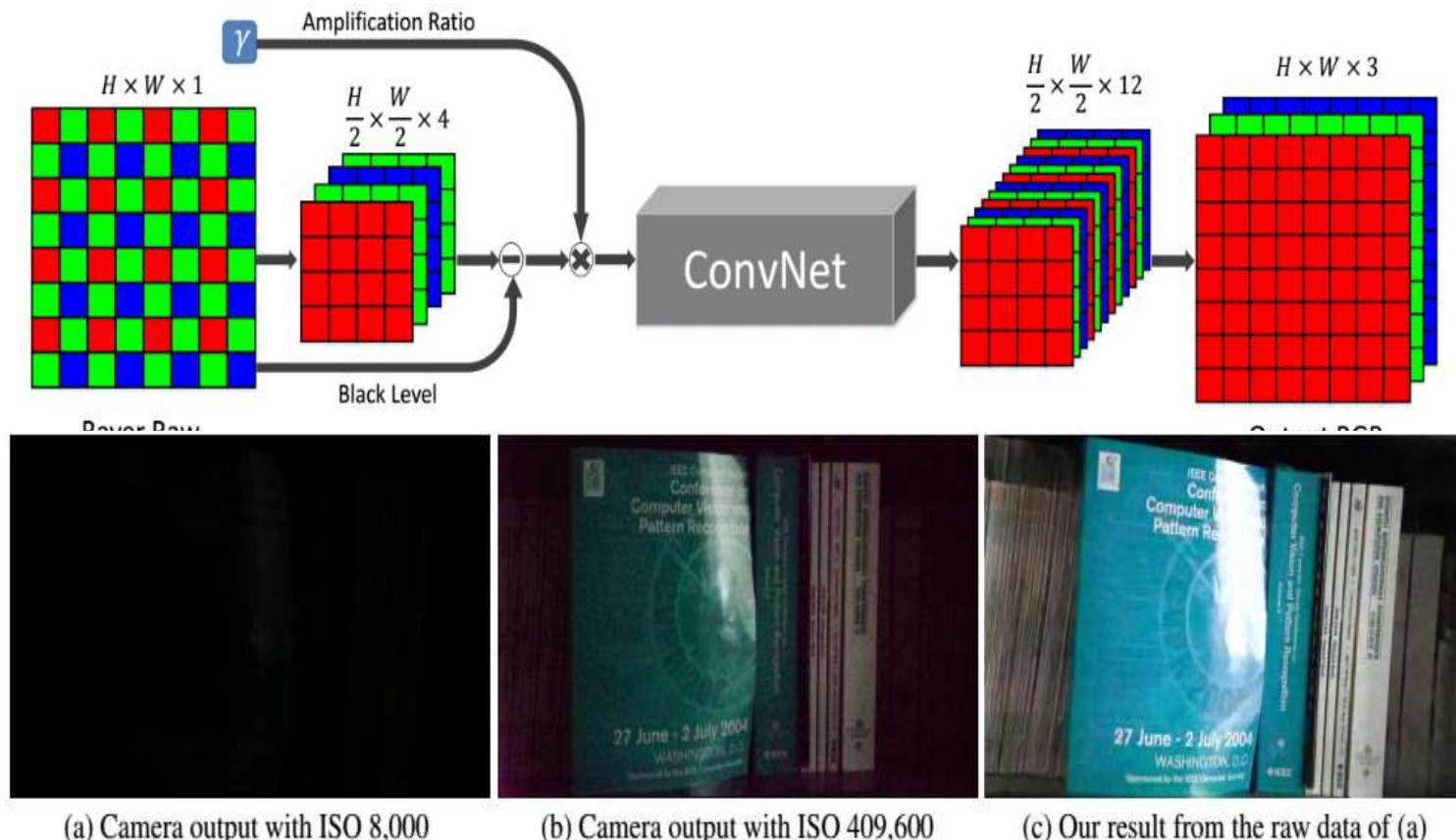


Figure 4: Comparison of low-light(no additional noise) images.

# 基于深度学习的图像增强

## 数据映射

- 除了图对图的映射外，可使用原始传感器数据，端到端学习映射



Learn to see in the dark

# Learn to see in the dark

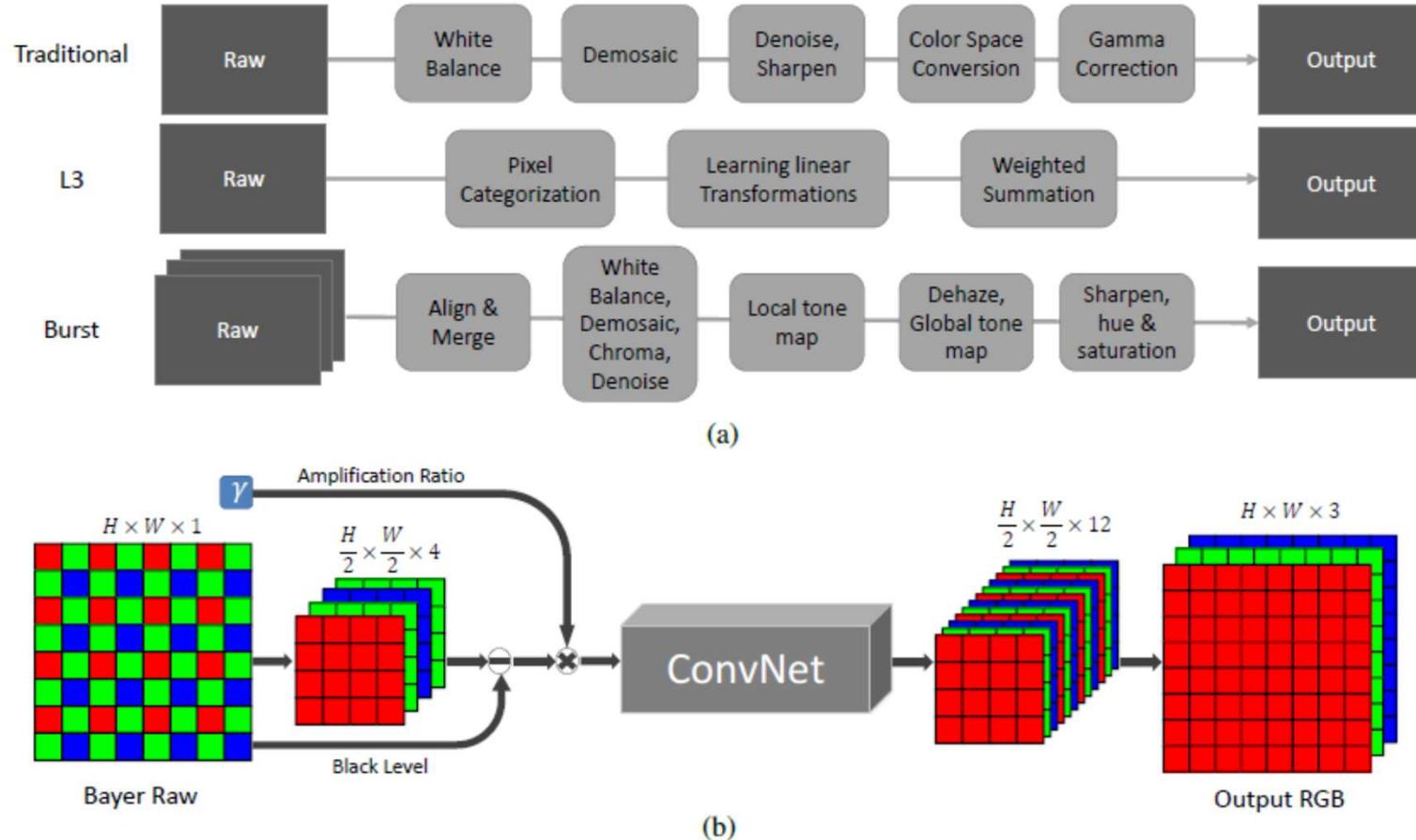
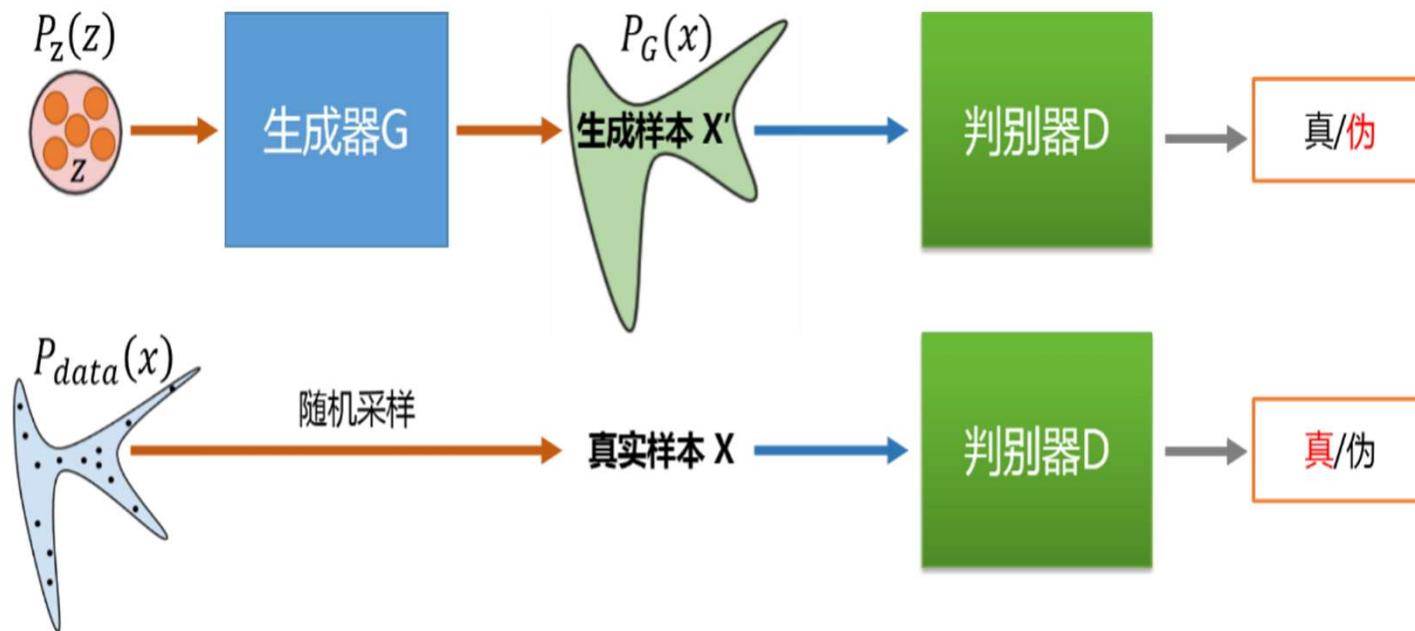


Figure 3. The structure of different image processing pipelines. (a) From top to bottom: a traditional image processing pipeline, the L3 pipeline [18], and a burst imaging pipeline [14]. (b) Our pipeline.

# 基于深度学习的图像增强

数据映射--基于生成对抗网络的图像增强

- GAN(生成对抗网络)
  - 在二人零和博弈中，生成能力和判别能力同时提高



$$\min_G \max_D V(D, G) = E_{x \sim P_{data}} [\log D(x)] + E_{z \sim P_z} [\log(1 - D(G(z)))]$$

# 基于深度学习的图像增强

## 数据映射--基于生成对抗网络的图像增强

- EnhanceGAN

- 目的是提取特征来学习图像gamma变换的超参数
- 利用判别器对高质量与低质量的判别，来指导超参数的生成
- 并且可以扩展到不同的增强方式（滤波、裁剪等）的组合上

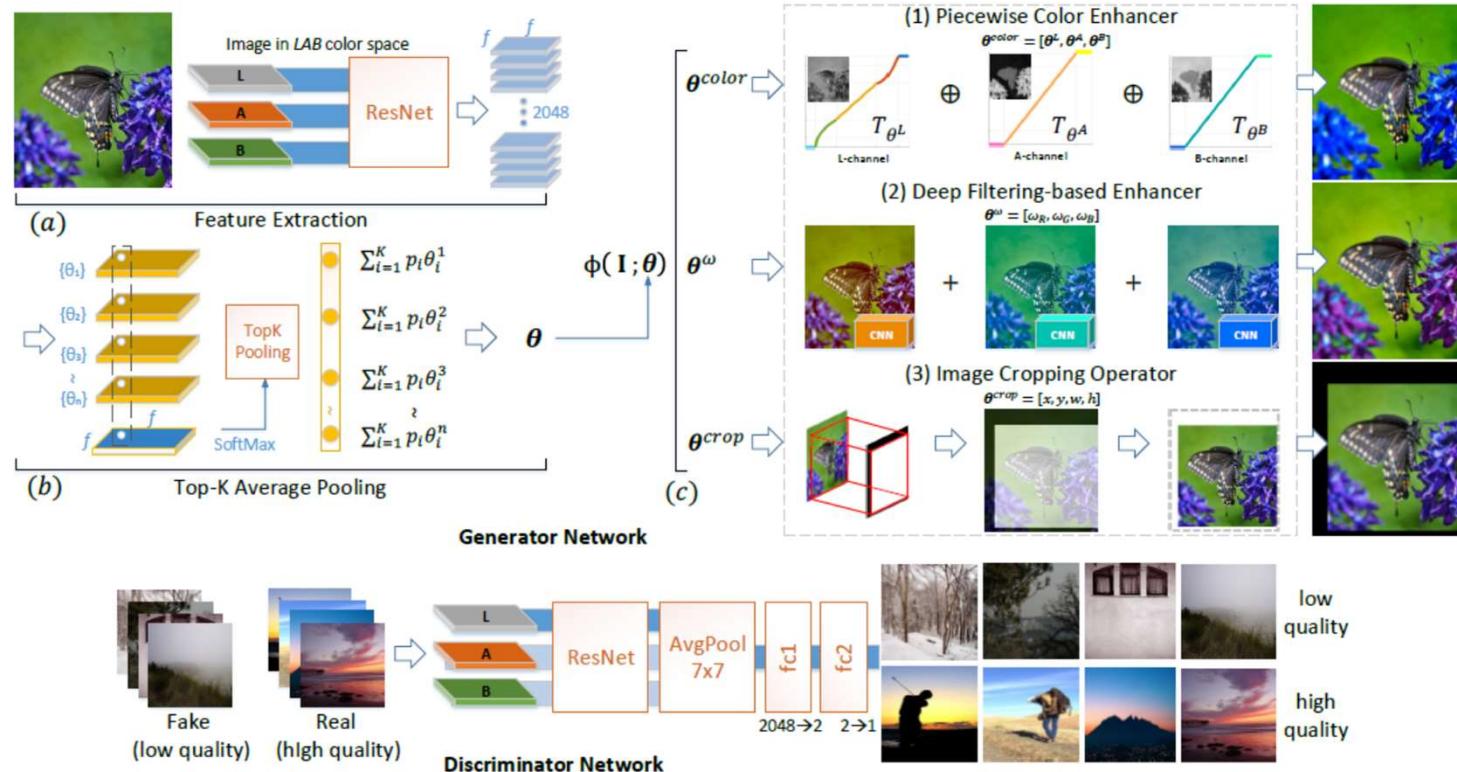


Figure 2: The architecture of our proposed EnhanceGAN framework. ResNet module is the feature extractor; in this work, we use the ResNet-101 [16] and removed the last average pooling layer and the final fc layer. (Best viewed in color.)

# EnhanceGAN

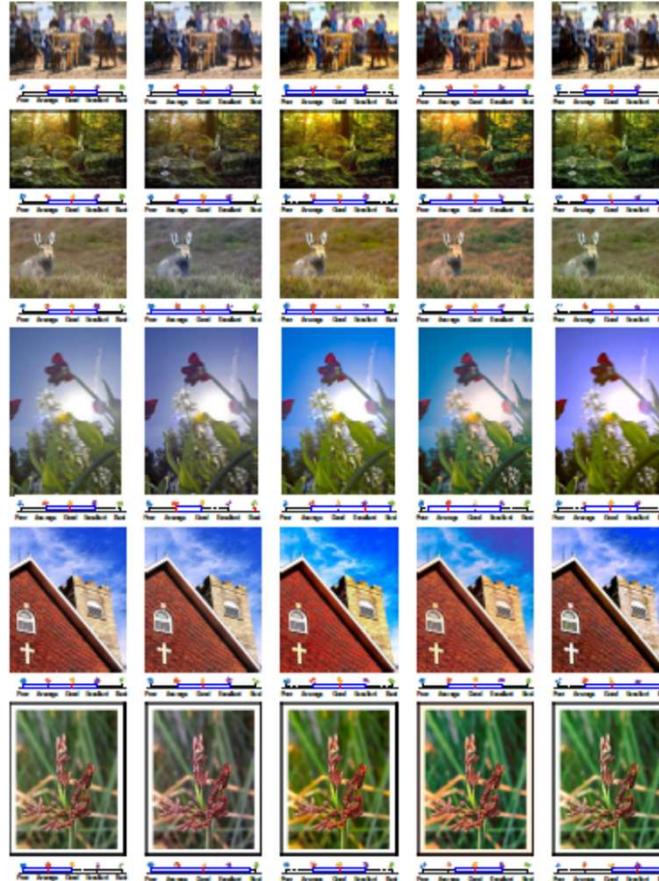
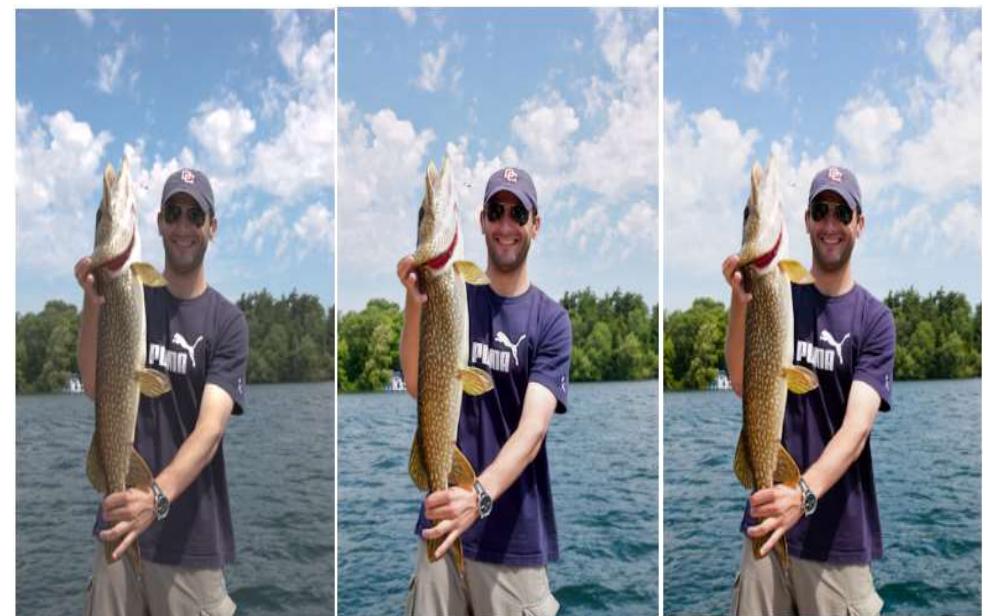
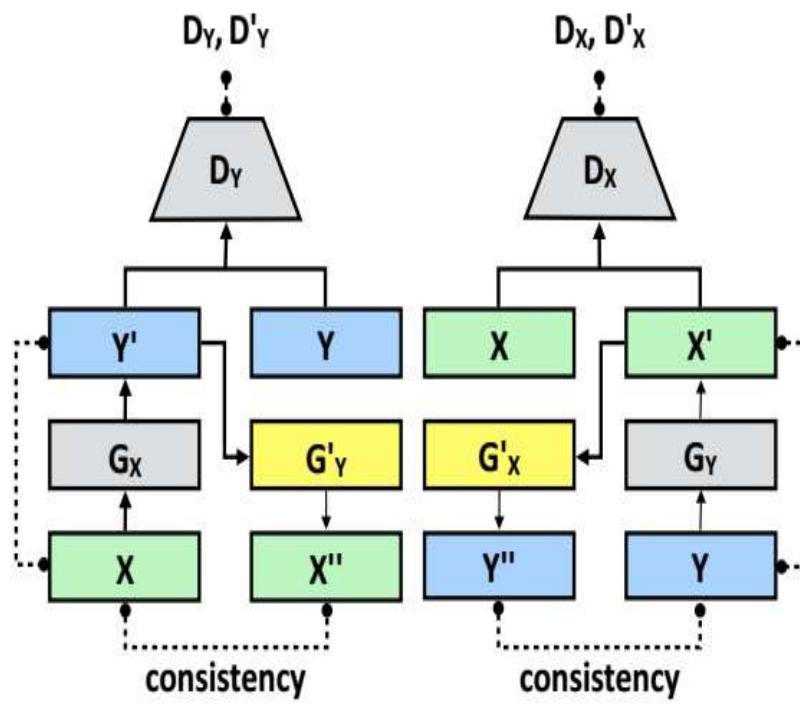


Figure 8: Visual results from our user study. From left to right: (a) Original image; (b) Photoshop-Auto (c) EnhanceGAN - piecewise color enhancer; (d) EnhanceGAN - deep filtering-based enhancer; (e) Human editing. The box plot below shows the ranking for each image, and the amount of dots denotes the number of users who gives a particular rank as in {Poor, Average, Good, Excellent, Best}. We show the complete results in the supplementary material.

# 基于深度学习的图像增强

数据映射--基于生成对抗网络的图像增强

- Deep Photo Enhancer
  - 利用GAN的生成数据分布的能力，实现由源域向增强域的图像迁移
  - 利用双路GAN实现非配对训练。



source  
translated  
target

# Deep Photo Enhancer

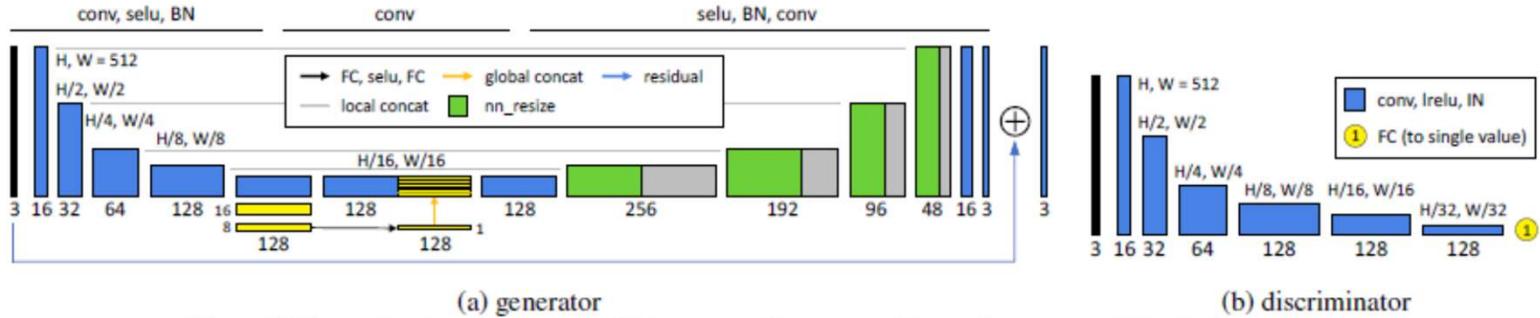


Figure 2. The network architectures of the proposed generator (a) and the proposed discriminator (b).

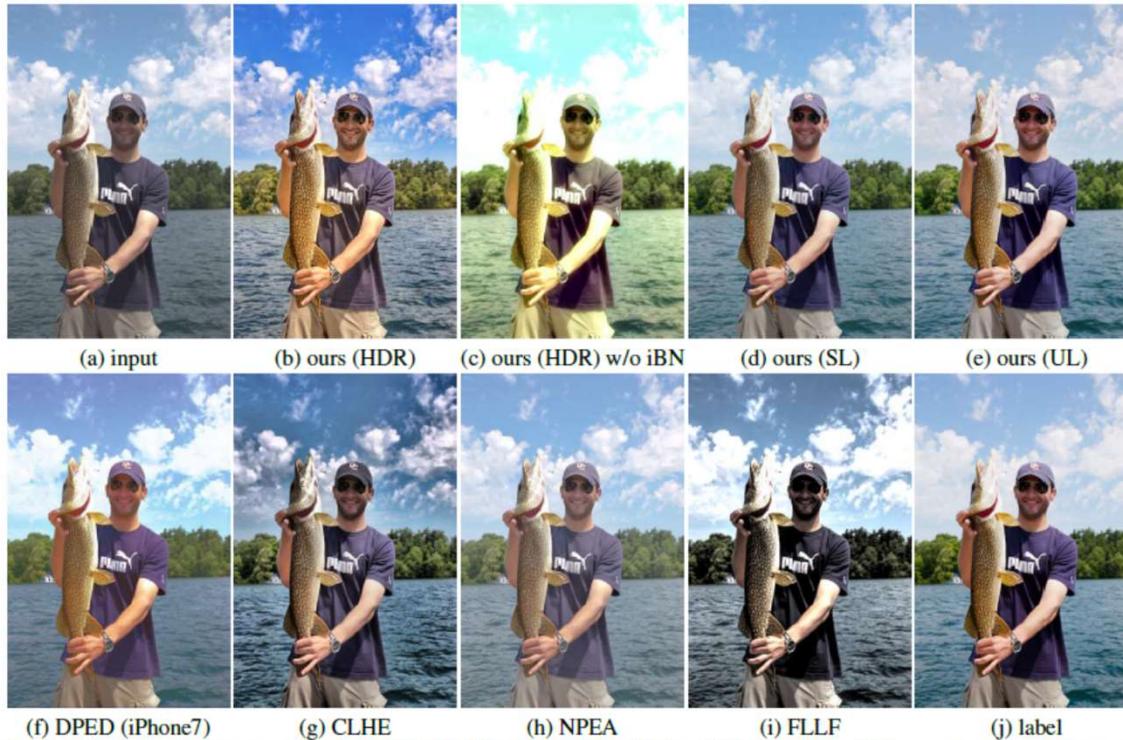


Figure 4. Comparisons of our models with DPED [10], CLHE [21], NPEA [22] and FLLF [2]. SL and UL respectively mean supervised and unpaired learning on the MIT-Adobe 5K dataset. The label is the retouched image by a photographer.

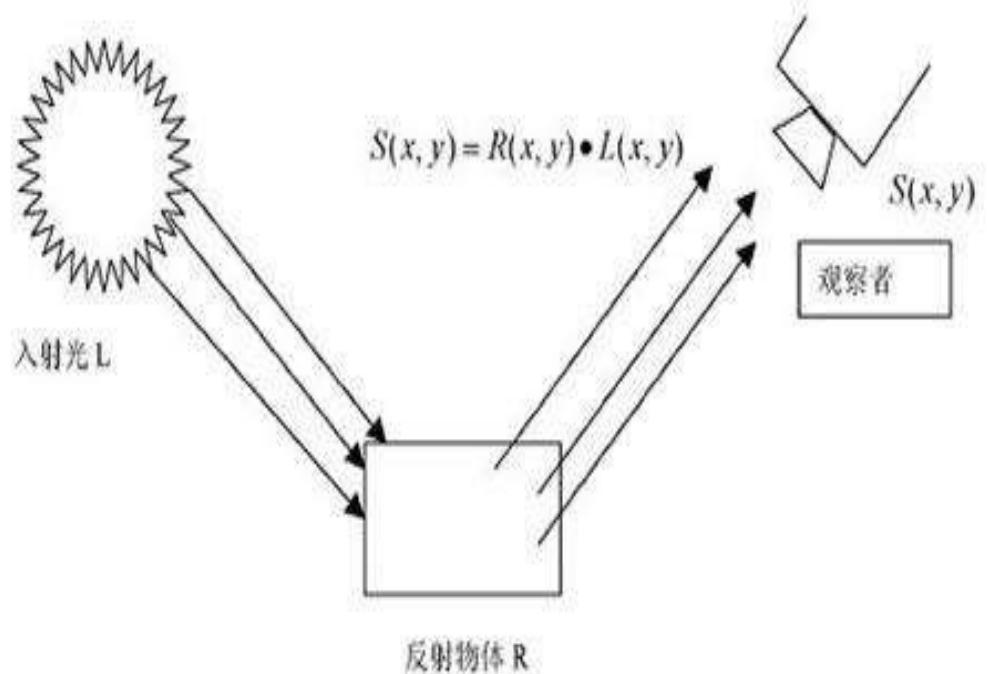
# 基于深度学习的图像增强

## 深度学习方法 视觉模型

$$r(x, y) = \log R(x, y) = \log \frac{S(x, y)}{L(x, y)}$$
$$= \log S(x, y) - \log L(x, y)$$

S -- 成像    L -- 光照分量

R -- 反射分量(理想图片)

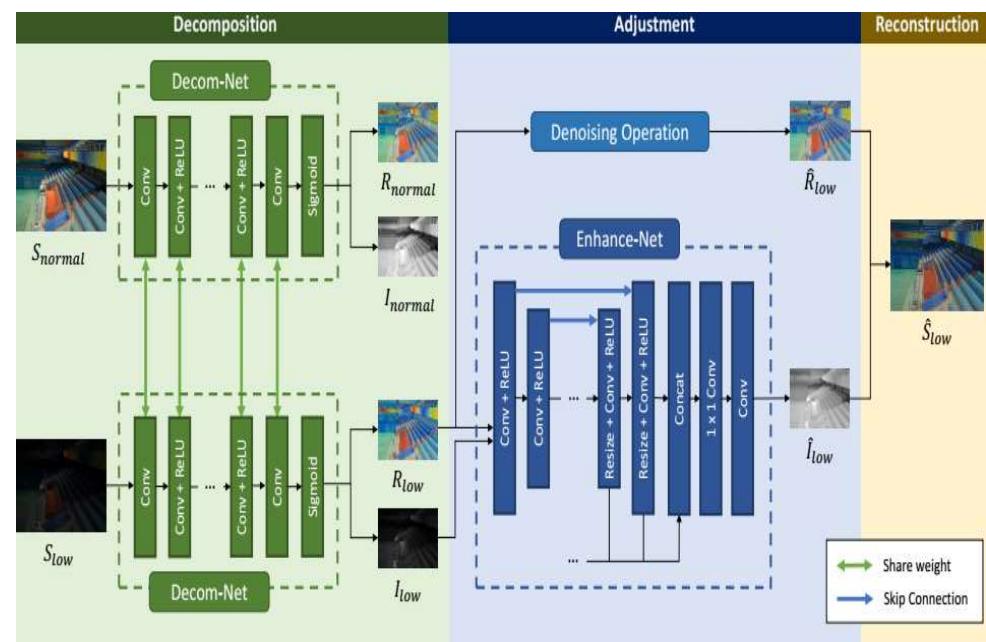
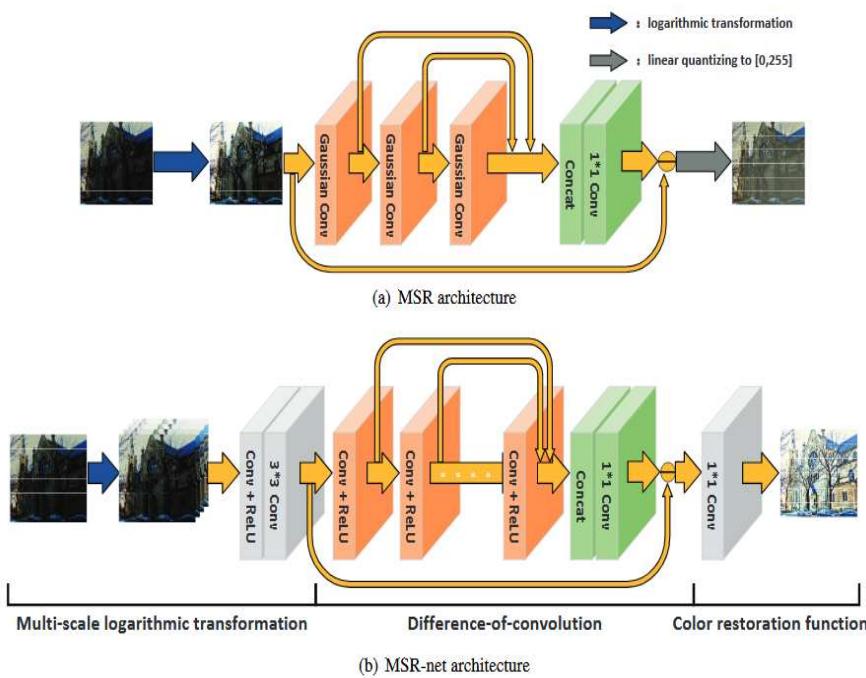


经典方法是利用低通滤波来估计L，进而求得R

# 基于深度学习的图像增强

## 结合模型假设

- MSRnet--基于Retinex的经典方法，把滤波器部分由深度网络来拟合
- Deep Retinex一分解R和L分量，学习source和label各自的L分量之间的映射。



MSRnet

Deep Retinex Decomposition

# MSRNet

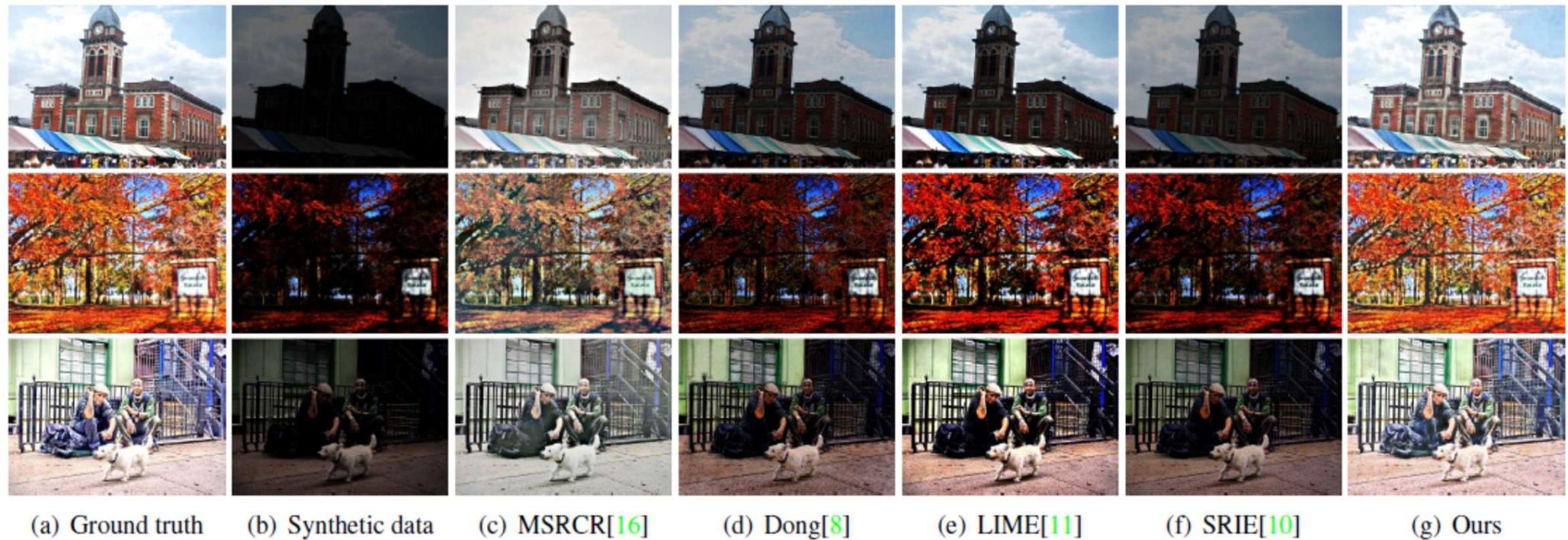


Figure 4. Results using different methods on synthesized test images

# Deep Retinex

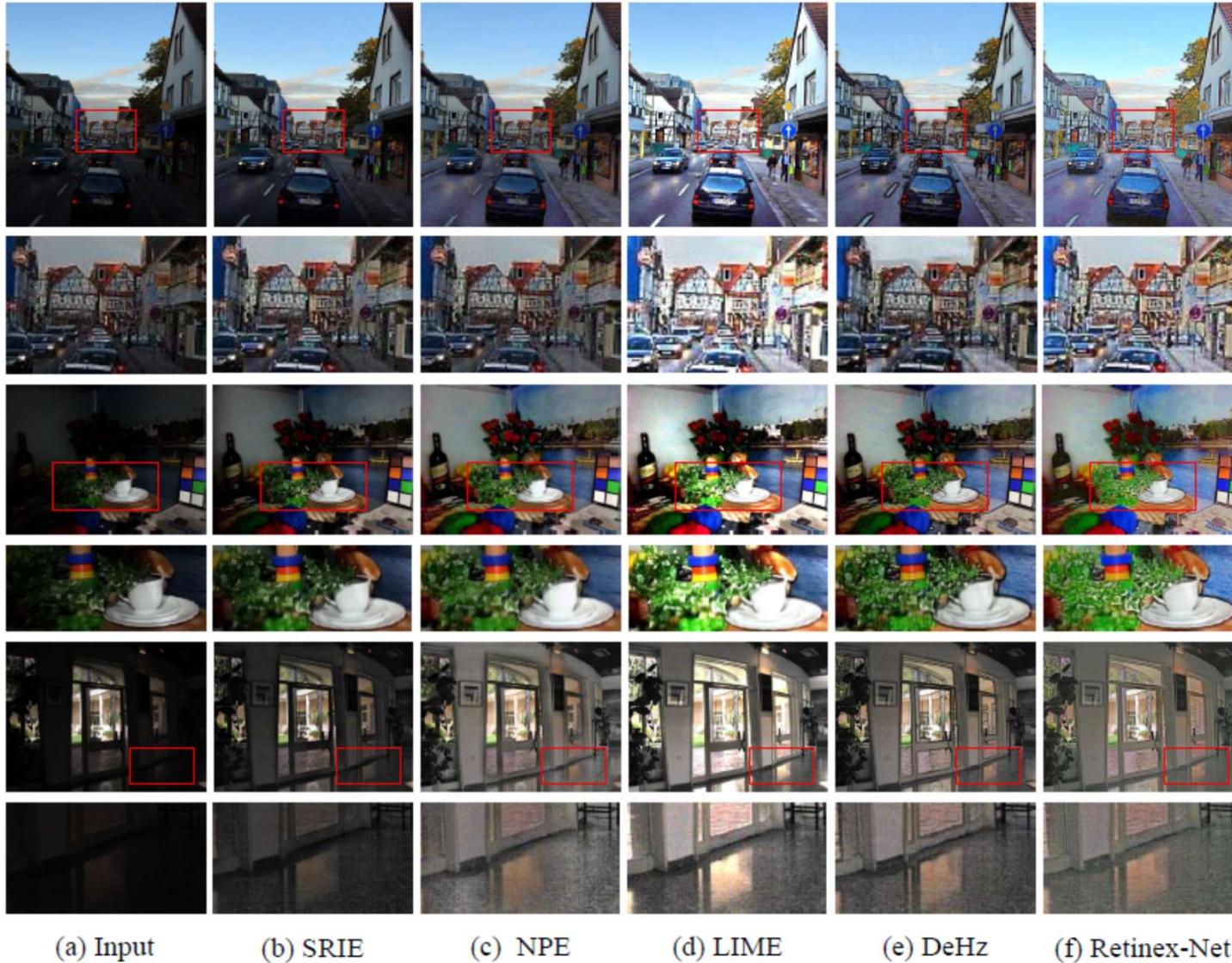


Figure 6: The results using different methods on natural images: (top-to-bottom) *Street* from LIME dataset, *Still lives* from LIME dataset, and *Room* from MEF dataset.

# 基于深度学习的图像增强

- 暗通道先验模型

$$I_d(x) = \min_{y \in \Omega(x)} (\min_{c \in \{r,g,b\}} I^c(y))$$

$$I_d(x) \rightarrow 0$$

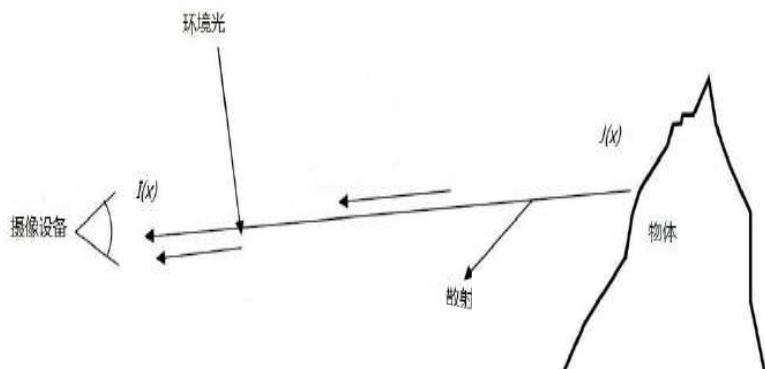
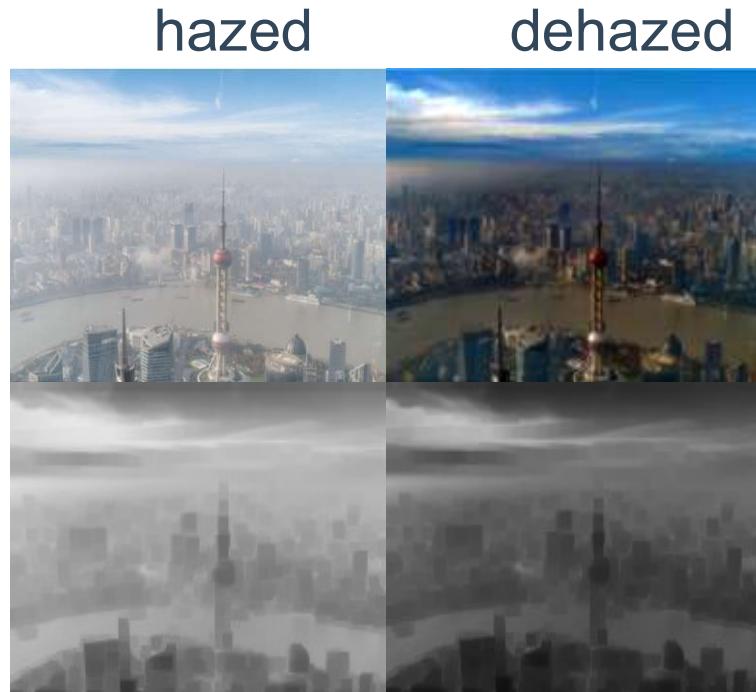
在绝大多数无雾的非天空的局部区域内，某些像素至少一个颜色通道具有很低的值

- 大气散射模型

$$I(x) = J(x) \times t(x) + A \times (1 - t(x))$$

观测图 清晰图      大气光      透射率

$$J(x) = (I(x) - A) / t(x) + A$$

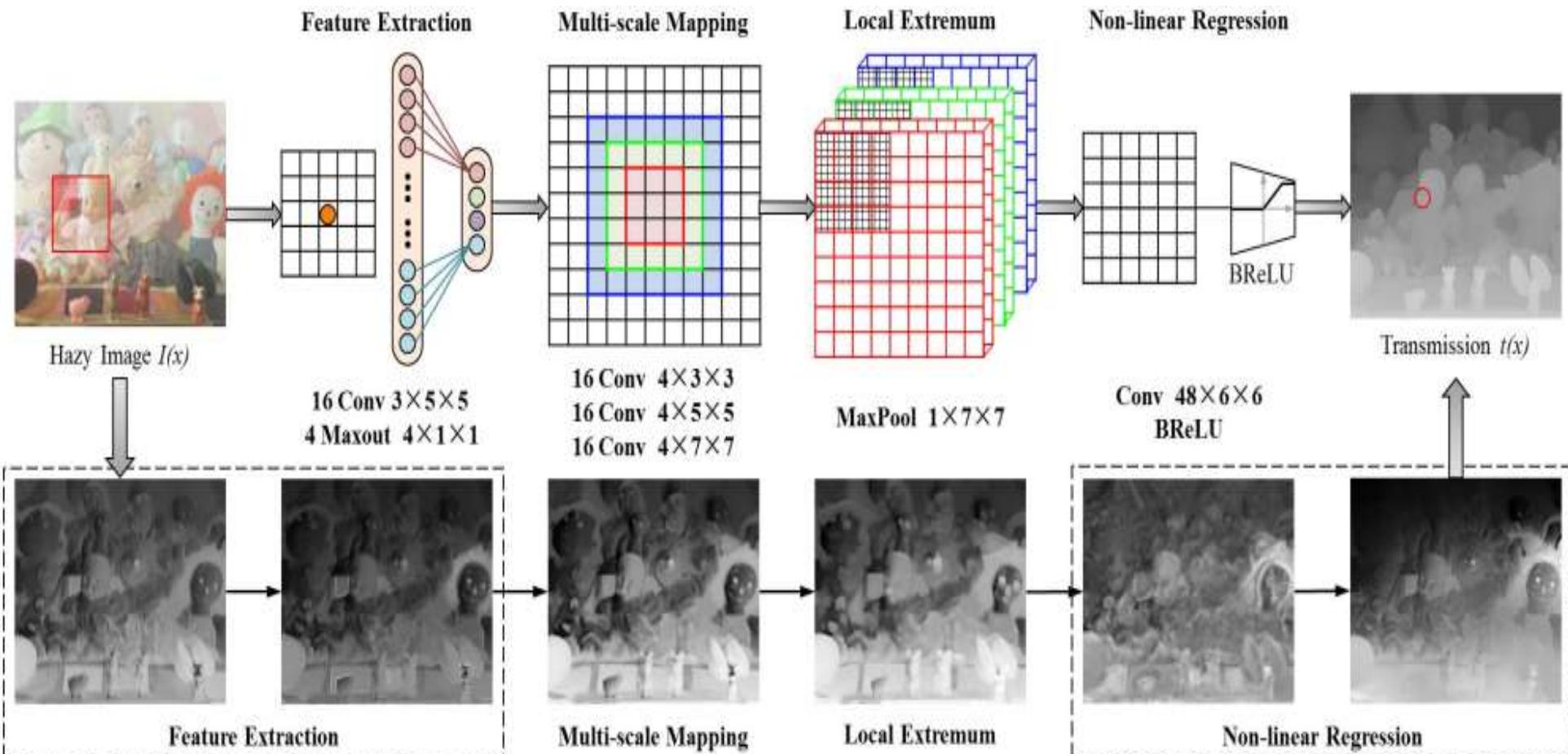


要求去雾图  $J(x)$ ，需要得到  $t(x)$  和  $A$

# 基于深度学习的图像增强

结合大气散射模型和经典去雾先验，设计去雾网络

- 利用大气散射模型，从清晰图像仿真出有雾图像，并得到透射率数据，制作训练集。
- 设置大气光值为1，端到端的学习有雾图像的透射率矩阵



# 图像复原

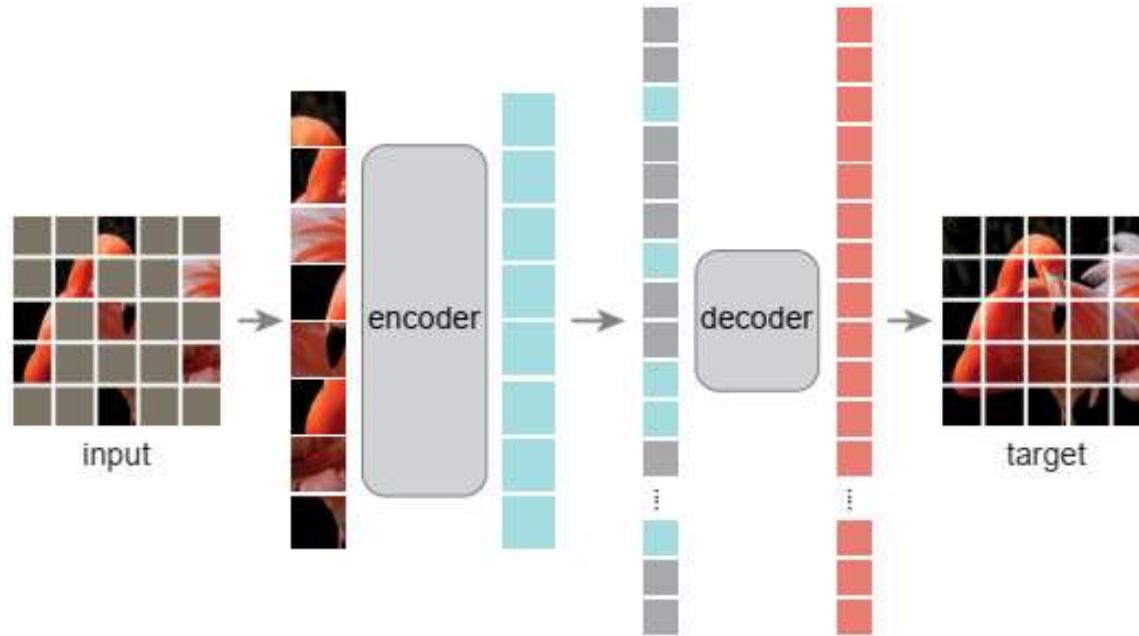
---

# **Masked Autoencoders Are Scalable Vision Learners**

Kaiming He<sup>\*,†</sup> Xinlei Chen<sup>\*</sup> Saining Xie Yanghao Li Piotr Dollár Ross Girshick

<sup>\*</sup>equal technical contribution      <sup>†</sup>project lead

Facebook AI Research (FAIR)



**Figure 1. Our MAE architecture.** During pre-training, a large random subset of image patches (*e.g.*, 75%) is masked out. The encoder is applied to the small subset of *visible patches*. Mask tokens are introduced *after* the encoder, and the full set of encoded patches and mask tokens is processed by a small decoder that reconstructs the original image in pixels. After pre-training, the decoder is discarded and the encoder is applied to uncorrupted images (full sets of patches) for recognition tasks.

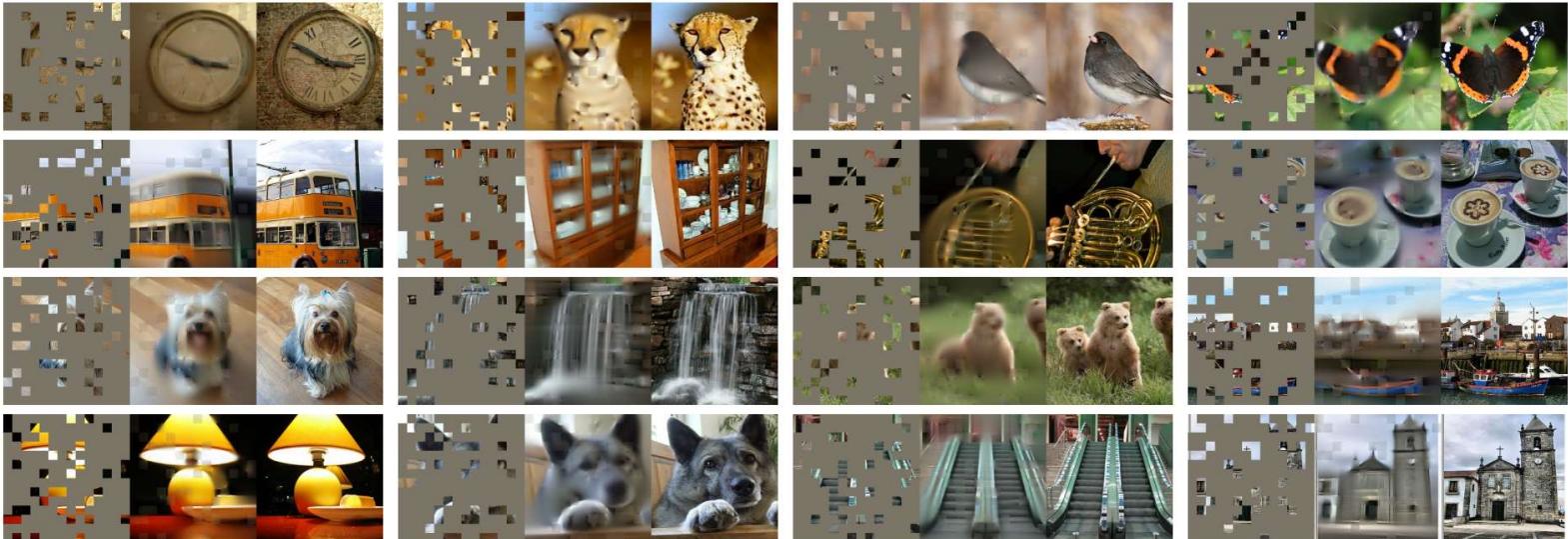


Figure 2. Example results on ImageNet *validation* images. For each triplet, we show the masked image (left), our MAE reconstruction<sup>†</sup> (middle), and the ground-truth (right). The masking ratio is 80%, leaving only 39 out of 196 patches. More examples are in the appendix.

<sup>†</sup>As no loss is computed on visible patches, the model output on visible patches is qualitatively worse. One can simply overlay the output with the visible patches to improve visual quality. We intentionally opt not to do this, so we can more comprehensively demonstrate the method's behavior.

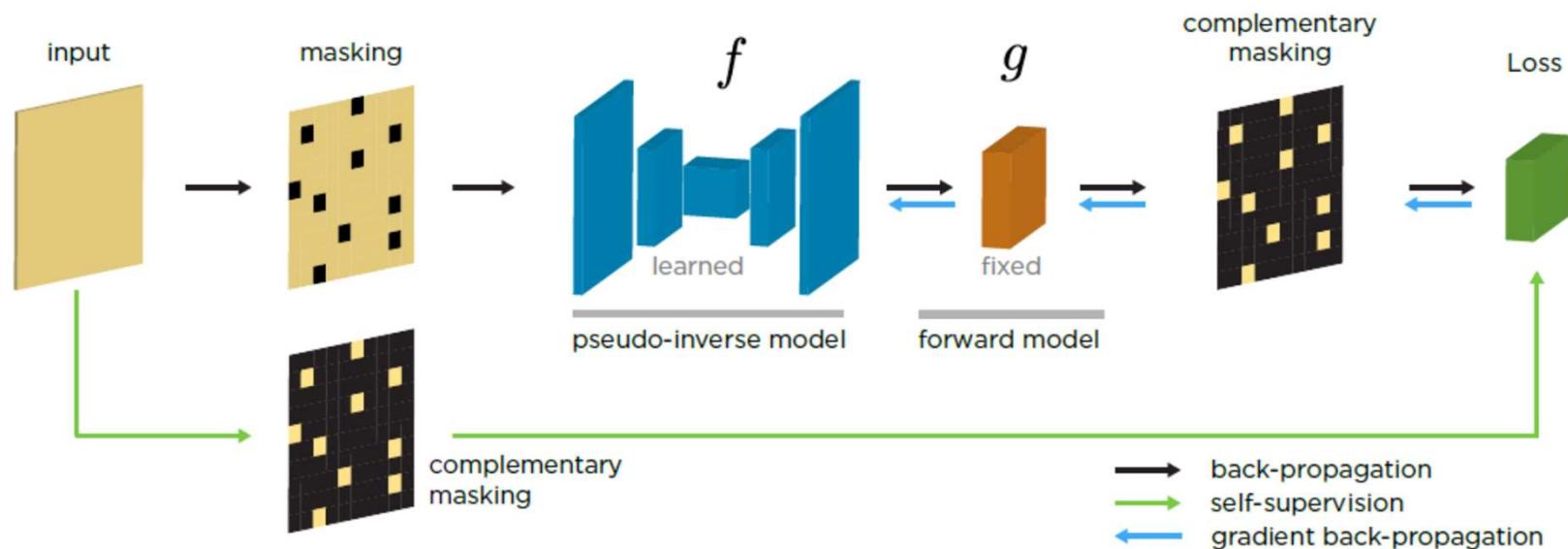


Figure 3. Example results on COCO validation images, using an MAE trained on ImageNet (the same model weights as in Figure 2). Observe the reconstructions on the two right-most examples, which, although different from the ground truth, are semantically plausible.

# 图像去模糊：无监督深度学习方法

## Image Deconvolution via Noise-Tolerant Self-Supervised Inversion

Hirofumi Kobayashi <sup>1</sup> Ahmet Can Solak <sup>1</sup> Joshua Batson <sup>1</sup> Loic A. Royer <sup>\*1</sup>



---

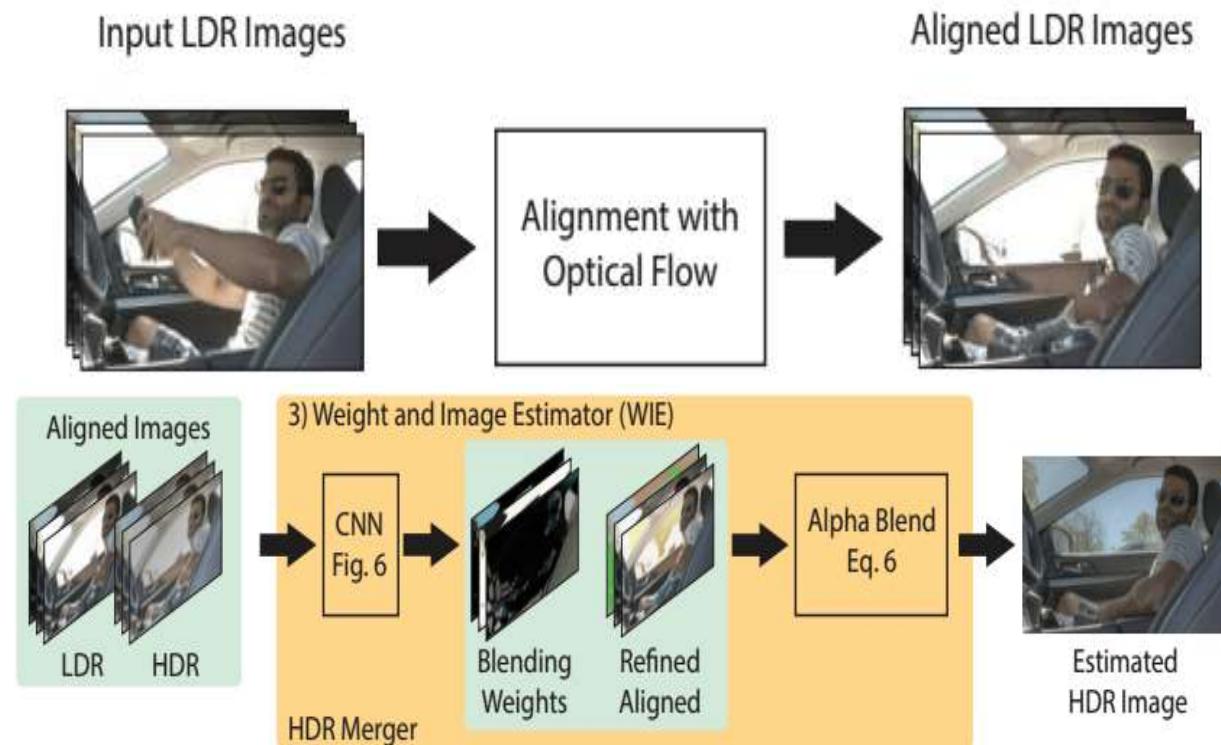
# Questions?

---

# 基于深度学习的图像增强

## 数据映射

- 基于图像合成的动态范围增强，深度网络作为融合器
  - 首先对LDR图片进行对齐
  - 使用CNN进行对齐修正和合成



Deep High Dynamic Range Imaging of Dynamic Scenes

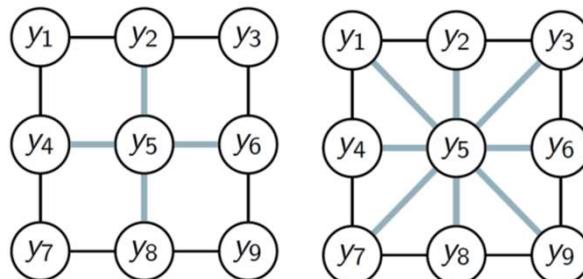
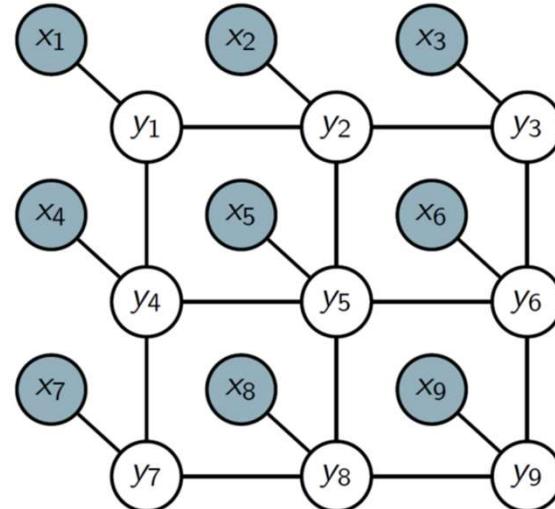
## 基本概念 (III)

- 后验条件概率模型

$$P(y|x) = \frac{1}{Z(x)} \exp\{-E(y; x)\}$$

where  $Z(x) = \sum_{y \in L^n} \exp\{-E(y; x)\}$   
Partition function

$$\begin{aligned} E(y; x) &= \sum_c \psi_c(y_c; x) \\ &= \sum_{i \in V} \psi_i^U(y_i; x) + \sum_{ij \in E} \psi_{ij}^P(y_i, y_j; x) + \sum_{c \in C} \psi_c^H(y_c; x) \end{aligned}$$



4-connected,  $\mathcal{N}_4$

8-connected,  $\mathcal{N}_8$

# 基本概念 (IV)

## Binary MRF Example

Consider the following energy function for two binary random variables,  $y_1$  and  $y_2$ .

$$\begin{array}{c} \text{0} \\ \text{1} \end{array} \begin{array}{|c|} \hline 5 \\ \hline 2 \\ \hline \end{array} \quad \begin{array}{c} \text{0} \\ \text{1} \end{array} \begin{array}{|c|} \hline 1 \\ \hline 3 \\ \hline \end{array} \quad \begin{array}{c} \text{0} \\ \text{1} \end{array} \begin{array}{|c|c|} \hline & 1 \\ \hline 0 & 3 \\ \hline 1 & 4 \\ \hline 0 & \\ \hline \end{array}$$

$$\begin{aligned} E(y_1, y_2) &= \psi_1(y_1) + \psi_2(y_2) + \psi_{12}(y_1, y_2) \\ &= \underbrace{5\bar{y}_1 + 2y_1}_{\psi_1} \\ &\quad + \underbrace{\bar{y}_2 + 3y_2}_{\psi_2} \\ &\quad + \underbrace{3\bar{y}_1 y_2 + 4y_1 \bar{y}_2}_{\psi_{12}} \end{aligned}$$

where  $\bar{y}_1 = 1 - y_1$  and  $\bar{y}_2 = 1 - y_2$ .

## Graphical Model



## Probability Table

$y_1$	$y_2$	$E$	$P$
0	0	6	0.244
0	1	11	0.002
1	0	7	0.090
1	1	5	0.664

## 基本概念 (V)

---

Consider a 1 mega-pixel image, e.g.,  $1000 \times 1000$  pixels. We want to annotate each pixel with a label from  $\mathcal{L}$ . Let  $L = |\mathcal{L}|$ .

- There are  $L^{10^6}$  possible ways to label such an image.
- A naive encoding—i.e., one big table—would require  $L^{10^6} - 1$  parameters.
- A pairwise MRF over  $\mathcal{N}_4$  requires  $10^6 L$  parameters for the unary terms and  $2 \times 1000 \times (1000 - 1)L^2$  parameters for the pairwise terms, i.e.,  $O(10^6 L^2)$ . Even less are required if we share parameters.

## 基本概念 (VI)

$$P(y|x) = \frac{1}{Z(x)} \exp\{-E(y; x)\}$$

$$\text{where } Z(x) = \sum_{y \in L^n} \exp\{-E(y; x)\}$$

$$y^* = \arg \max_y P(y|x) = \arg \min_y E(y; x)$$

# MRF-Gibbs 等价关系 (I)

- Gibbs分布

$$P(X = x) = \frac{1}{Z} \exp(-E(x))$$

$$Z = \sum_x \exp(-E(x))$$

- Hammersley-Clifford定理

任何条件随机分布存在一个Gibbs分布如果满足如下三个条件

Positivity:  $P(X = x) > 0$

Locality:  $P(X_s = x_s | X_t = x_t, t \neq s, t \in S) = P(X_s = x_s | X_t = x_t, t \in G_s)$

Homogeneity:  $P(X_s = x_s | X_t = x_t, t \in G_s)$  is the same for all sites  $s$

## MRF-Gibbs 等价关系 (II)

$$\left( \begin{array}{l} X \text{ is a Markov random field} \\ \& \\ \forall x, P\{X = x\} > 0 \end{array} \right) \iff \left( \begin{array}{l} P\{X = x\} \text{ has the form} \\ \text{of a Gibbs distribution} \end{array} \right)$$

直观上讲，马尔可夫随机场模型假设一个像素的灰度仅取决于其邻域像素的灰度。在这种条件下，图像可以由一个全局的联合概率密度函数描述。

# 示例1：图像的语义分割

- **Label space:** pre-defined label set, e.g.,



$$\mathcal{L} = \{\text{sky}, \text{tree}, \text{grass}, \dots\}$$

- **Unary term:** Boosted decision-tree classifiers over “textron-layout” features [Shotton et al., 2006]

$$\psi_i^U(y_i = \ell; \mathbf{x}) = \theta_\ell \log P(\phi_i(\mathbf{x}) \mid \ell)$$

- **Pairwise term:** contrast-dependent smoothness prior

$$\psi_{ij}^P(y_i, y_j; \mathbf{x}) = \begin{cases} \lambda_0 + \lambda_1 \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\beta}\right), & \text{if } y_i \neq y_j \\ 0, & \text{otherwise} \end{cases}$$

## 示例2：图像去噪声



- **Label space:** pixel intensity or colour

$$\mathcal{L} = \{0, 1, \dots, 255\}$$

- **Unary term:** square distance

$$\psi_i^U(y_i; \mathbf{x}) = \|y_i - x_i\|^2$$

- **Pairwise term:** truncated  $L_2$  distance

$$\psi_{ij}^P(y_i, y_j) = \max \left\{ \|y_i - y_j\|^2, d_{\max}^2 \right\}$$