

第五章 图像预处理

一般情况下，成像系统获取的图像(即原始图像)由于受到种种条件限制和随机干扰，往往不能在视觉系统中直接使用，必须在视觉信息处理的早期阶段对原始图像进行灰度校正、噪声过滤等图像预处理。对机器视觉系统来说，所用的图像预处理方法并不考虑图像降质原因，只将图像中感兴趣的特征有选择地突出，衰减其不需要的特征，故预处理后的输出图像并不需要去逼近原图像。这类图像预处理方法统称为图像增强。图像增强技术主要有两种方法：空间域法和频率域法。空间域方法主要是在空间域内对图像像素直接运算处理。频率域方法就是在图像的某种变换域，对图像的变换值进行运算，如先对图像进行付立叶变换，再对图像的频谱进行某种计算(如滤波等)，最后将计算后的图像逆变换到空间域。本章首先讨论直方图修正方法，然后介绍各种滤波技术，其中，高斯平滑滤波器将作比较深入的讨论。

5.1 直方图修正

许多图像的灰度值是非均匀分布的，其中灰度值集中在一个小区间内的图像是很常见的(图 5.2(a)所示的对比度很弱的图像)。直方图均衡化是一种通过重新均匀地分布各灰度值来增强图像对比度的方法。经过直方图均衡化的图像对二值化阈值选取十分有利。一般来说，直方图修正能提高图像的主观质量，因此在处理艺术图像时非常有用。

直方图修正的一个简单例子是图像尺度变换：把在灰度区间 $[a, b]$ 内的像素点映射到 $[z_1, z_k]$ 区间。一般情况下，由于曝光不充分，原始图像灰度区间 $[a, b]$ 常常为空间 $[z_1, z_k]$ 的子空间，此时，将原区间内的像素点 z 映射成新区间内像素点 z' 的函数表示为

$$z' = \frac{z_k - z_1}{b - a}(z - a) + z_1 \quad (5.1)$$

上述函数的曲线形状见图 5.1(a)。上述映射关系实际上将 $[a, b]$ 区间扩展到区间 $[z_1, z_k]$ 上，使曝光不充分的图像黑的更黑，白的更白。

如果图像的大多数像素灰度值分布在区间 $[a, b]$ ，则可以使用图 5.1(b)所示的映射函数：

$$z' = \begin{cases} \frac{z_k - z_1}{b - a}(z - a) + z_1 & a \leq z \leq b \\ z_1 & z < a \\ z_k & z > b \end{cases} \quad (5.2)$$

若要突出图像中具有某些灰度值物体的细节，而又不牺牲其它灰度上的细节，可以采用分段灰度变换，使需要的细节灰度值区间得到拉伸，不需要的细节得到压缩，以增强对比度，如图 5.1(c)所示。当然也可以采用连续平滑函数进行灰度变换，见图 5.1(d)。

这一方法存在的问题是，当直方图被延伸后，所得到的新直方图并不均匀，也就是说，各灰度值所对应的像素数并不相等。因此，更好的方法应该是既能扩展直方图，又能使直方图真正地呈现均匀性。

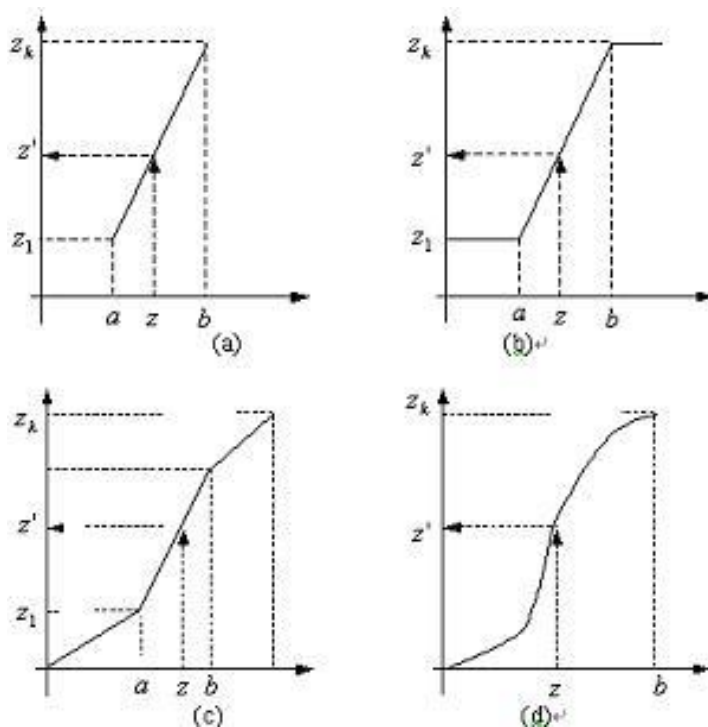


图 5. 1 灰度变换

如果预先设定灰度值分布，那么就可以用下面的方法：假定 p_i 是原直方图中在灰度级 z_i 上的像素点的数目， q_i 是要得到的直方图在灰度级 z_i 上的像素点的数目。从原直方图的左边起，找到灰度值 k_1 ，使得：

$$\sum_{i=1}^{k_1-1} p_i \leq q_1 < \sum_{i=1}^{k_1} p_i \quad (5. 3)$$

灰度级 $z_1, z_2, \dots, z_{k_1-1}$ 上的像素点将映射到新图像的灰度级为 z_1 的像素点上。现在求灰度值 k_2 使得：

$$\sum_{i=1}^{k_2-1} p_i \leq q_1 + q_2 < \sum_{i=1}^{k_2} p_i \quad (5. 4)$$

下一区间像素值 $z_{k_1}, \dots, z_{k_2-1}$ 被映射到灰度级 z_2 上。重复这一过程直到原始图像的所有灰度值都得到处理。这一方法的处理结果示于图 5. 2 中。在那里，原始图像对比度很弱，原因是灰度值分布在一小区间内。直方图均衡化通过映射灰度值来逼近均匀分布，从而改善了对比度。但是这一方法在均衡化后的直方图中仍然留下了间隙，除非输入图像中具有同一灰度级的像素点在输出图中被延伸至几个灰度级。

如果直方图被延伸，则在原始图像中具有相同灰度值的像素点在新的图像中可能会被延伸成不同的灰度值。最简便的方法就是为相同灰度值的每一个像素点分配一个随机的输出值。为了把像素点均匀地分布在 n 个输出值 $q_k, q_{k+1}, \dots, q_{k+n-1}$ 的范围内，假定使用一个随机数发生器，其产生的随机数均匀地分布在 $[0,1)$ 内。输出的像素点标号可以由随机数 r 通过计算公式 $k + [n \times r]$ 得到。换句话说，对每一次决策，抽出一个随机数，乘以区间内的输出值数目 n 后四舍五入取整，最后将这一偏移量加到最低标号 k 上。

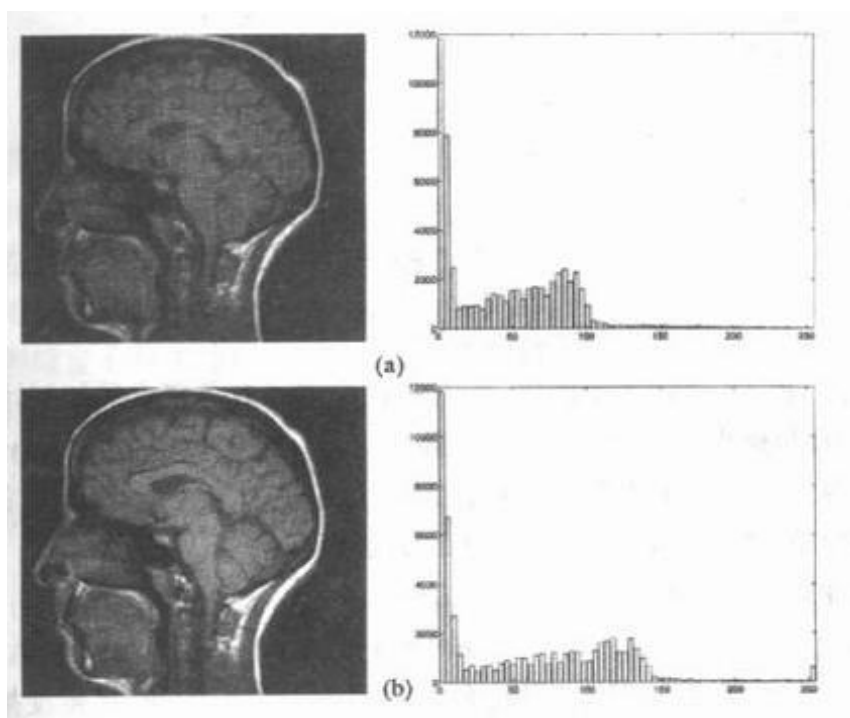
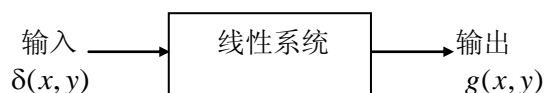


图 5. 2 上图为原始图像及其直方图。下图为直方图均衡化后的图像及其直方图。

5. 2 图像线性运算

5. 2. 1 线性系统

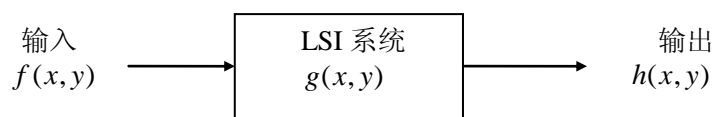
许多图像处理系统都可以用一个线性系统作为模型：



对于线性系统，当系统输入是一个中心在原点的脉冲 $\delta(x,y)$ 时，输出 $g(x,y)$ 就是系统的脉冲响应。此外，如果系统响应与输入脉冲的中心位置无关，则该系统称为空间不变系统。



线性空间不变系统(Linear Space Invariant, LSI)完全能用其脉冲响应来描述：



其中， $f(x,y)$ 和 $h(x,y)$ 是输入和输出图像。上面的系统必须满足关系式：

$$a \cdot f_1(x, y) + b \cdot f_2(x, y) \Rightarrow a \cdot h_1(x, y) + b \cdot h_2(x, y)$$

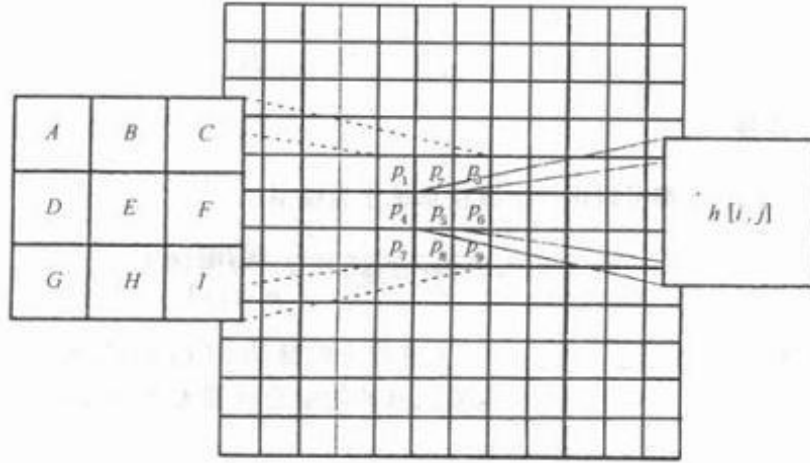
其中， $f_1(x,y)$ 和 $f_2(x,y)$ 是输入图像， $h_1(x,y)$ 和 $h_2(x,y)$ 是对应于 $f_1(x,y)$ 和 $f_2(x,y)$ 的输出图像， a 和 b 是常数比例因子。对这样的系统，其输出 $h(x,y)$ 可以用输入 $f(x,y)$ 与

其脉冲响应 $g(x, y)$ 的卷积来定义:

$$\begin{aligned} h(x, y) &= f(x, y) * g(x, y) \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x', y') g(x - x', y - y') dx' dy'. \end{aligned} \quad (5.5)$$

若为离散函数, 上式变为

$$\begin{aligned} h[i, j] &= f[i, j] * g[i, j] \\ &= \sum_{k=0}^{n-1} \sum_{l=0}^{m-1} f[k, l] g[i - k, j - l] \end{aligned} \quad (5.6)$$



$$h[i, j] = Ap_1 + Bp_2 + Cp_3 + Dp_4 + Ep_5 + Fp_6 + Gp_7 + Hp_8 + Ip_9$$

图 5.3 3×3 阶的卷积模板示意图, 卷积模板原点对应于位置 E , 而权 A, B, \dots, I 是 $g[-k, -l]$ 的值, 其中 $k, l = -1, 0, +1$.

如果 $f(x, y)$ 和 $h(x, y)$ 表示图像, 则卷积就变成了对像素点的加权计算, 脉冲响应 $g[i, j]$ 就是一个卷积模板. 对图像中每一像素点 $[i, j]$, 输出响应值 $h(x, y)$ 是通过平移卷积模板到像素点 $[i, j]$ 处, 计算模板与像素点 $[i, j]$ 邻域加权得到的, 其中各加权值对应卷积模板的各对应值. 图 5.3 是模板为 3×3 的示意图.

卷积是线性运算, 因为

$$g[i, j] * \{a_1 f_1[i, j] + a_2 f_2[i, j]\} = a_1 \{g[i, j] * f_1[i, j]\} + a_2 \{g[i, j] * f_2[i, j]\}$$

对任何常量 a_1 和 a_2 都成立. 换句话说, 和的卷积等于卷积的和, 尺度变换后的图像卷积等于卷积后作相应的尺度变换. 卷积是空间不变算子, 因为在整幅图像中都使用相同的权重系数. 但空间可变系统则在图像的不同部分要求不同的滤波权重因子, 因此这种运算无法用卷积来表示.

5.2.2 付立叶变换

$n \times m$ 图像可用下列频率分量表示:

$$f[k, l] = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} F(\xi, \eta) e^{jk\xi} e^{jl\eta} d\xi d\eta \quad (5.7)$$

其中, $F(\xi, \eta)$ 是图像的付立叶变换. 付立叶变换对每一个频率分量的幅值和相位进行编码, 定义为

$$\begin{aligned} F(\xi, \eta) &= F\{f[k, l]\} \\ &= \sum_{k=0}^{n-1} \sum_{l=0}^{m-1} f[k, l] e^{-jk\xi} e^{-jl\eta} \end{aligned} \quad (5.8)$$

其中 F 代表付立叶变换运算符号. 在 (ξ, η) 平面原点附近的值称为付立叶变换的低频分量, 而远离原点的值称为高频分量. 注意, $F(\xi, \eta)$ 是一个连续函数.

图像域的卷积对应于频率域的乘积, 因此, 对于图像域中非常费时的大滤波器卷积, 若使用快速付立叶变换 (fast fourier transform, FFT), 可以大大地提高计算效率. FFT 是许多图像处理应用领域里十分重要的方法. 但是在机器视觉中, 由于大多数算法是非线性的或空间可变的, 因此不能使用付立叶变换方法. 对于视觉模型为线性的、空间不变的系统, 由于滤波尺度很小, 使用快速付立叶变换几乎得不到什么益处. 因此, 在视觉预处理阶段, 通常使用线性滤波器 (如平滑滤波器等) 来完成图像时域卷积.

5. 3 线性滤波器

图像常常被强度随机信号 (也称为噪声) 所污染. 一些常见的噪声有椒盐 (Salt & Pepper) 噪声、脉冲噪声、高斯噪声等. 椒盐噪声含有随机出现的黑白强度值. 而脉冲噪声则只含有随机的白强度值 (正脉冲噪声) 或黑强度值 (负脉冲噪声). 与前两者不同, 高斯噪声含有强度服从高斯或正态分布的噪声. 高斯噪声是许多传感器噪声的很好模型, 例如摄像机的电子干扰噪声.

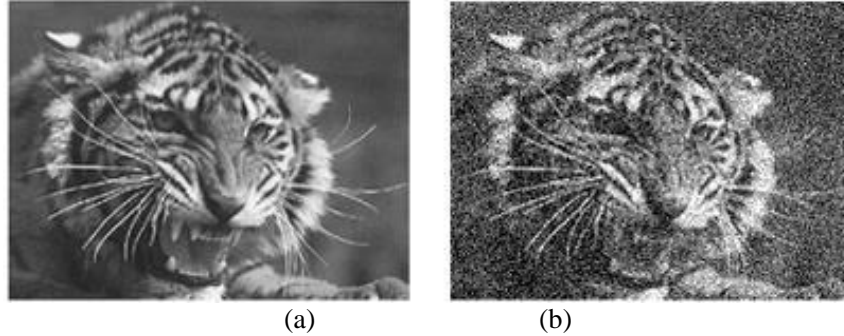


图 5. 4 被高斯噪声所污染的图像. (a) 原始图像, (b) 高斯噪声.

线性平滑滤波器去除高斯噪声的效果很好, 且在大多数情况下, 对其它类型的噪声也有很好的效果. 线性滤波器使用连续窗函数内像素加权和来实现滤波. 特别典型的是, 同一模式的权重因子可以作用在每一个窗口内, 也就意味着线性滤波器是空间不变的, 这样就可以使用卷积模板来实现滤波. 如果图像的不同部分使用不同的滤波权重因子, 且仍然可以用滤波器完成加权运算, 那么线性滤波器就是空间可变的. 任何不是像素加权运算的滤波器都属于非线性滤波器. 非线性滤波器也可以是空间不变的, 也就是说, 在图像的任何位置上可以进行相同的运算而不考虑图像位置或空间的变化. 5.4 节中所提出的中值滤波器就是空间不变的非线性滤波器. 下面主要介绍两种线性滤波器, 均值滤波器和高斯滤波器.

5. 3. 1 均值滤波器

最简单的线性滤波器是局部均值运算, 即每一个像素值用其局部邻域内所有值的均值置换:

$$h[i, j] = \frac{1}{M} \sum_{(k, l) \in N} f[k, l] \quad (5. 9)$$

其中, M 是邻域 N 内的像素点总数. 例如, 在像素点 $[i, j]$ 处取 3×3 邻域, 得到

$$h[i, j] = \frac{1}{9} \sum_{k=i-1}^{i+1} \sum_{l=j-1}^{j+1} f[k, l]. \quad (5. 10)$$

该方程与方程 5. 6 对比, 对于卷积模板中的每一点 $[i, j]$, 有 $g[i, j] = 1/9$, 那么方程 5. 6

就退化成方程 5.10 所示的局部均值运算。这一结果表明，均值滤波器可以通过卷积模板的等权值卷积运算来实现(见图 5.5)。实际上，许多图像处理运算都可以通过卷积来实现。

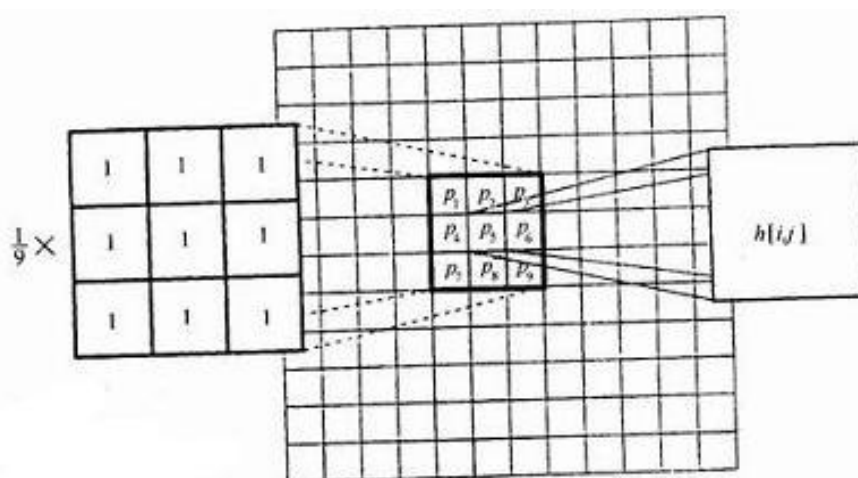


图 5.5 采用 3×3 邻域的均值滤波器示意图

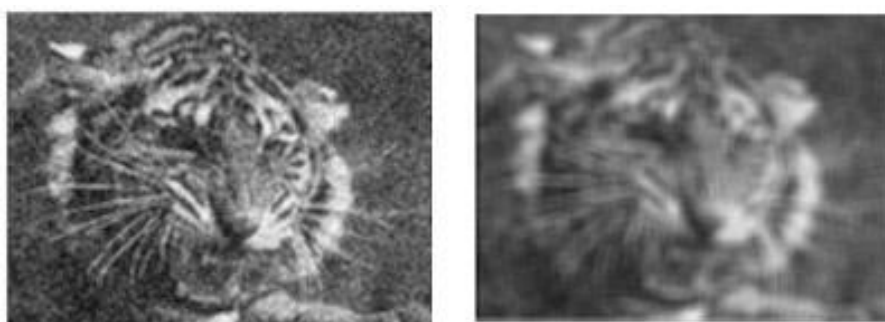


图 5.6 用 3×3 ， 7×7 窗函数对图 5.4 噪声图像进行均值滤波后的结果。

邻域 N 的大小控制着滤波程度，对应大卷积模板的大尺度邻域会加大滤波程度。作为去除大噪声的代价，大尺度滤波器也会导致图像细节的损失。不同尺度下均值滤波的结果见图 5.6。

在设计线性平滑滤波器时，选择滤波权值应使得滤波器只有一个峰值，称之为主瓣，并且在水平和垂直方向上是对称的。一个典型的 3×3 平滑滤波器的权值模板如下：

$\frac{1}{16}$	$\frac{1}{8}$	$\frac{1}{16}$
$\frac{1}{8}$	$\frac{1}{4}$	$\frac{1}{8}$
$\frac{1}{16}$	$\frac{1}{8}$	$\frac{1}{16}$

线性平滑滤波器去除了高频成分和图像中的锐化细节，例如：会把阶跃变化平滑成渐近变化，从而牺牲了精确定位的能力。空间可变滤波器能调节权值，使得在相对比较均匀的图像区域上加大平滑量，而在尖锐变化的图像区域上减小平滑量。

5.3.2 高斯平滑滤波

高斯滤波器是一类根据高斯函数的形状来选择权值的线性平滑滤波器。高斯平滑滤波器对去除服从正态分布的噪声是有效的。一维零均值高斯函数为：

$$g(x) = e^{-\frac{x^2}{2\sigma^2}} \quad (5.11)$$

其中，高斯分布参数 σ 决定了高斯滤波器的宽度。对图像处理来说，常用二维零均值离散高斯函数作平滑滤波器。这种函数的图形如图 5.7 所示，函数表达式为：

$$g[i, j] = e^{-\frac{(i^2 + j^2)}{2\sigma^2}} \quad (5.12)$$

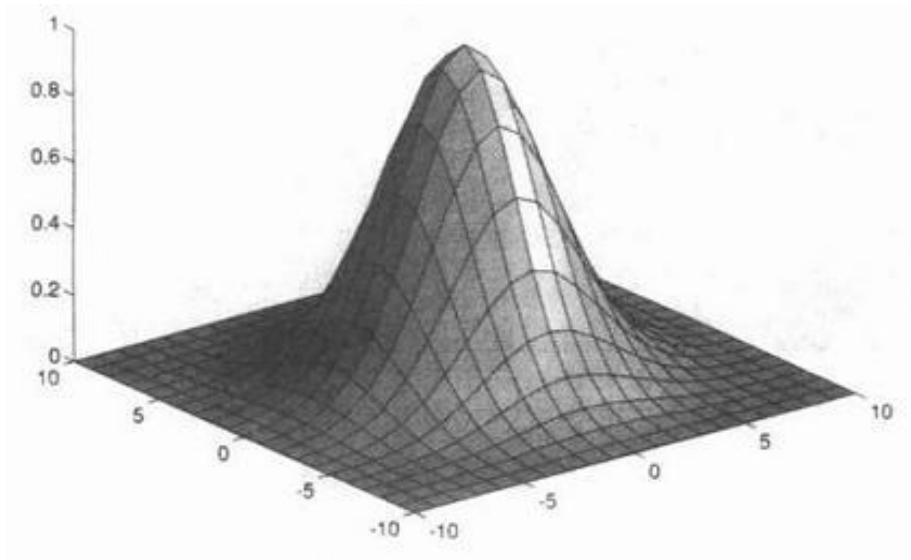


图 5.7 二维零均值高斯函数示意图

高斯函数具有五个重要的性质，这些性质使得它在早期图像处理中特别有用。这些性质表明，高斯平滑滤波器无论在空间域还是在频率域都是十分有效的低通滤波器，且在实际图像处理中得到了工程人员的有效使用。高斯函数具有五个十分重要的性质，它们是：

- 二维高斯函数具有旋转对称性，即滤波器在各个方向上的平滑程度是相同的。一般来说，一幅图像的边缘方向是事先不知道的，因此，在滤波前是无法确定一个方向上比另一方向上需要更多的平滑。旋转对称性意味着高斯平滑滤波器在后续边缘检测中不会偏向任一方向。
- 高斯函数是单值函数。这表明，高斯滤波器用像素邻域的加权均值来代替该点的像素值，而每一邻域像素点权值是随该点与中心点的距离单调增减的。这一性质是很重要的，因为边缘是一种图像局部特征，如果平滑运算对离算子中心很远的像素点仍然有很大作用，则平滑运算会使图像失真。
- 高斯函数的付立叶变换频谱是单瓣的。正如下面所示，这一性质是高斯函数付立叶变换等于高斯函数本身这一事实的直接推论。图像常被不希望的高频信号所污染(噪声和细纹理)。而所希望的图像特征(如边缘)，既含有低频分量，又含有高频分量。高斯函数付立叶变换的单瓣意味着平滑图像不会被不必要的高频信号所污染，同时保留了大部分所需信号。
- 高斯滤波器宽度(决定着平滑程度)是由参数 σ 表征的，而且 σ 和平滑程度的关系是非常简单的。 σ 越大，高斯滤波器的频带就越宽，平滑程度就越好。通过调节平滑程度参数 σ ，可在图像特征过分模糊(过平滑)与平滑图像中由于噪声和细纹理所引起的过多的不希望突变量(欠平滑)之间取得折衷。
- 由于高斯函数的可分离性，大高斯滤波器可以得以有效地实现。二维高斯函数卷积可以分两步来进行，首先将图像与一维高斯函数进行卷积，然后将卷积结果与方向垂直的相同一维高斯函数卷积。因此，二维高斯滤波的计算量随滤波模板宽度成线性增长而不是成平方增长。

下面详细解释这些性质。

(1) 旋转对称性

把高斯函数从直角坐标变换到极坐标，则可以清楚地看到高斯函数的旋转对称特性。二

维高斯函数为：

$$g[i, j] = e^{-\frac{i^2 + j^2}{2\sigma^2}} \quad (5.13)$$

由于极坐标中的矢径由公式 $r^2 = i^2 + j^2$ 确定，因此很容易得出高斯函数的极坐标表达式：

$$g(r, \theta) = e^{-\frac{r^2}{2\sigma^2}}, \quad (5.14)$$

它不依赖于极角 θ 自然也就旋转对称了。如果要求在某一特定的方向上加大平滑量，则应用旋转非对称高斯函数也是可能的。旋转非对称高斯函数的表达式见[Wozencraft 1965]给出，它们被用于通讯频道的概率统计分析中。

(2) 付立叶变换性质

高斯函数有一个十分有趣的性质，即它的付立叶变换也是一个高斯函数。由于高斯函数的付立叶变换是一个实函数，所以其付立叶变换前后的幅值不一样。高斯函数的付立叶变换通过下式计算：

$$\begin{aligned} F\{g(x)\} &= \int_{-\infty}^{\infty} g(x) e^{-j\omega x} dx \\ &= \int_{-\infty}^{\infty} e^{-\frac{x^2}{2\sigma^2}} e^{-j\omega x} dx \\ &= \int_{-\infty}^{\infty} e^{-\frac{x^2}{2\sigma^2}} (\cos \omega x - j \sin \omega x) dx \\ &= \int_{-\infty}^{\infty} e^{-\frac{x^2}{2\sigma^2}} \cos \omega x dx - j \int_{-\infty}^{\infty} e^{-\frac{x^2}{2\sigma^2}} \sin \omega x dx \end{aligned} \quad (5.15)$$

高斯函数是偶函数，而正弦函数是奇函数，因此第二个积分式的值必然等于零，从而整个付立叶变换可简化为：

$$\begin{aligned} F\{g(x)\} &= \int_{-\infty}^{\infty} e^{-\frac{x^2}{2\sigma^2}} \cos \omega x dx \\ &= \sqrt{2\pi\sigma} e^{-\frac{\omega^2}{2\nu^2}}, \quad \nu^2 = \frac{1}{\sigma^2}. \end{aligned} \quad (5.16)$$

空间域频率参数为 ω ，高斯函数在频率域内的散布由 ν 控制， ν 是空间域散布参数 σ 的倒数。这表明，高斯函数在空间域越窄，则在频率域里的频谱越宽，反之亦然。这一性质和高斯滤波器的抑制噪声能力有关。窄带空间域高斯函数的平滑能力较低，因为在频率域内其频带较宽，能通过更多的高频噪声和细纹理信号。随着高斯函数在空间域的宽度增加，高斯函数的平滑能力也增强了。也就是说，在频率域内，高斯函数越窄，通过高频噪声和细纹理信号就越少。图 5.8 所示的是不同散布参数 σ 对图像噪声的抑制程度和平滑程度。高斯函数在空间域的宽度与在频率域的频谱宽度之间的简单关系有利于高斯滤波器在实际设计中的应用。高斯函数付立叶变换的对偶性也解释了为什么空间域单瓣特性在频率域内也成立。

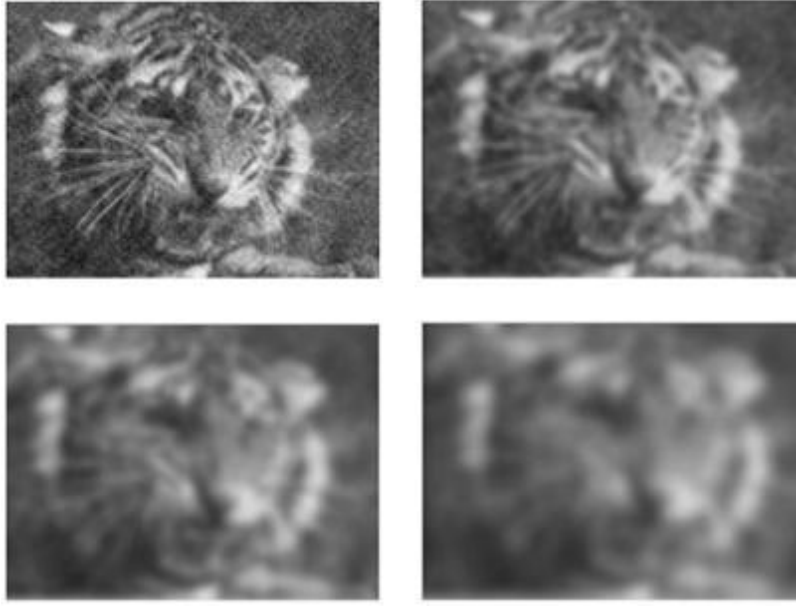


图 5.8 采用不同分布参数 σ 的高斯函数对图 5.4 (b) 噪声污染图像的滤波示意图。(a) $\sigma=1$ 的 7×7 模板, (b) $\sigma=2$ 的 13×13 模板, (c) $\sigma=3$ 的 19×19 模板, (d) $\sigma=4$ 的 25×25 模板。

(3) 高斯函数的可分离性

高斯函数的可分离性很容易表示为:

$$\begin{aligned}
 g[i, j] * f[i, j] &= \sum_{k=0}^{m-1} \sum_{l=0}^{n-1} g[k, l] f[i-k, j-l] \\
 &= \sum_{k=0}^{m-1} \sum_{l=0}^{n-1} e^{-\frac{(k^2+l^2)}{2\sigma^2}} f[i-k, j-l] \\
 &= \sum_{k=0}^{m-1} e^{-\frac{k^2}{2\sigma^2}} \left\{ \sum_{l=0}^{n-1} e^{-\frac{l^2}{2\sigma^2}} f[i-k, j-l] \right\}
 \end{aligned} \tag{5.17}$$

花括弧里的和式是输入图像 $f[i, j]$ 与一维水平高斯函数的卷积. 这一和式的结果是一个二维图像, 该图像在水平方向上被模糊化. 将该图像作为输入与相同的一维垂直高斯函数进行卷积, 使得图像在垂直方向上也被模糊化. 由于卷积是服从结合律和交换律的, 因此卷积次序可以颠倒, 即可以先进行垂直卷积, 将其结果作为输入再进行水平卷积. 图 5.9 是高斯函数卷积可分离性示意图。

这一方法可采用两个水平卷积模板来组合完成. 首先将输入 $f[i, j]$ 与一个水平高斯函数进行卷积, 并将其结果进行转置存入一个临时数组, 然后将临时数组作为输入, 与相同的高斯函数进行卷积, 以实现由水平卷积代替垂直卷积的目的. 第二次卷积后的输出信息再一次进行转置, 就得到了最终的平滑输出图像. 分离卷积的结果如图 5.10 所示。

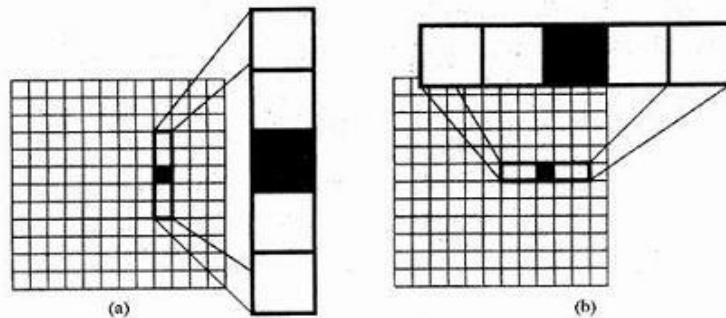


图 5. 9 高斯函数卷积可分离性示意图. 左: 纵向模板卷积. 右: 水平模板卷积.

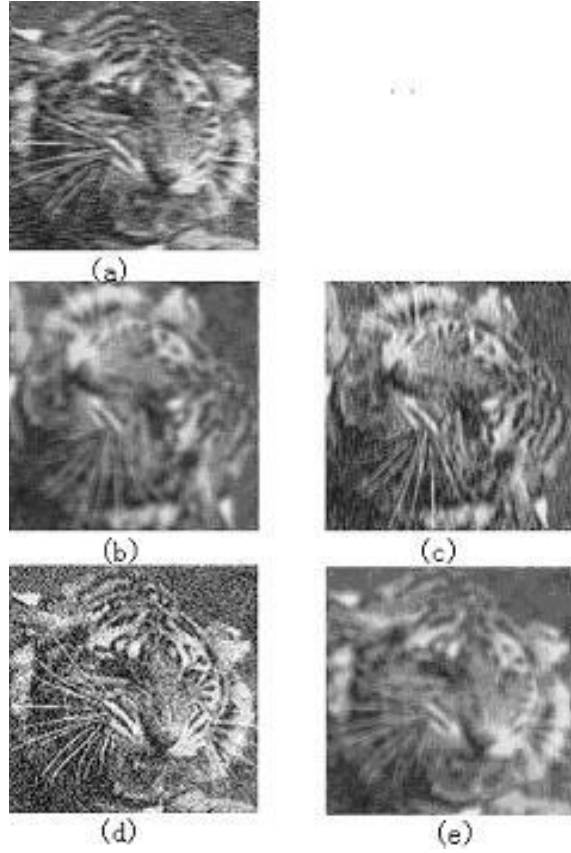


图 5. 10 采用单一水平卷积模板的分离高斯卷积结果. (a) 原始图像. (b) 水平高斯模板卷积结果. (c) 图(b)的转置. (d) 图(c)的水平模板卷积. (e) 图(d)的转置, 这是最终平滑图像.

(4) 级联高斯函数

与高斯函数有关的一个性质是高斯函数与其自身的卷积会产生一个与 σ 成比例的高斯函数, 该性质在一维情况下很容易地表达出来:

$$\begin{aligned}
 g(x) * g(x) &= \int_{-\infty}^{\infty} e^{-\frac{\xi^2}{2\sigma^2}} e^{-\frac{(x-\xi)^2}{2\sigma^2}} d\xi \\
 &= \int_{-\infty}^{\infty} e^{-\frac{(\frac{x}{2}+\xi)^2}{2\sigma^2}} e^{-\frac{(\frac{x}{2}-\xi)^2}{2\sigma^2}} d\xi, \quad \xi \rightarrow \xi + \frac{x}{2}. \\
 &= \int_{-\infty}^{\infty} e^{-\frac{(2\xi^2 + x^2/2)}{2\sigma^2}} d\xi. \\
 &= \sqrt{2\sigma} e^{-\frac{x^2}{2(\sqrt{2}\sigma)^2}}.
 \end{aligned} \tag{5.18}$$

具有散布参数为 σ 的两个高斯函数的卷积是一个具有散布参数为 $\sqrt{2}\sigma$ 的高斯函数, 这一结果也适用于二维的情形. 这说明, 如果一幅图像用散布参数为 σ 的高斯函数滤波, 以及相同的图像用散布参数为 $\sqrt{2}\sigma$ 的高斯函数滤波, 那么, 可以不用较大散布参数的高斯函数进行滤波, 而是用相同散布参数的滤波器对上一次滤波结果再进行滤波就可得到预期的图像. 这

就意味着在计算图像的多种尺度平滑时，计算量能得到显著减少。在具有不同散布参数 σ 的高斯函数级联运算中，也可以得到类似的结果。

(5) 高斯滤波器设计

高斯函数的最佳逼近由二项式展开的系数决定：

$$(1+x)^n = \binom{n}{0} + \binom{n}{1}x + \binom{n}{2}x^2 + \cdots + \binom{n}{n}x^n. \quad (5.19)$$

换句话说，用杨辉三角形（也称 Pascal 三角形）的第 n 行作为高斯滤波器的一个具有 n 个点的一维逼近，例如，五点逼近为：

$$\begin{bmatrix} 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

它们对应于 Pascal 三角形的第 5 行。这一模板被用来在水平方向上平滑图像。在高斯函数可分离性性质中曾指出，二维高斯滤波器能用两个一维高斯滤波器逐次卷积来实现，一个沿水平方向，一个沿垂直方向。实际中，这种运算可以通过使用单个一维高斯模板，对两次卷积之间的图像和最后卷积的结果图像进行转置来完成。

这一技术在模板尺寸 n 约为 10 时的滤波效果极好。对较大的滤波器，二项式展开系数对大多数计算机来说都太多。但是，任意大的高斯滤波器都能通过重复使用小高斯滤波器来实现。高斯滤波器的二项式逼近的 σ 可用高斯函数拟合二项式系数的最小方差来计算。

设计高斯滤波器的另一途径是直接从离散高斯分布中计算模板权值[Kohl 1992]：

$$g[i, j] = ce^{-\frac{(i^2+j^2)}{2\sigma^2}}. \quad (5.20)$$

其中 c 是归一化常量。把上式重新表示为：

$$\frac{g[i, j]}{c} = e^{-\frac{(i^2+j^2)}{2\sigma^2}}. \quad (5.21)$$

选择合适的 σ^2 值，我们就可以在 $n \times n$ 窗口上评价该值，以便获取核或模板，其中，在[0,0]处的值等于 1。例如，选 $\sigma^2 = 2$ ， $n = 7$ ，上面的式子产生下列数组：

$[i, j]$	-3	-2	-1	0	1	2	3
-3	.011	.039	.082	.105	.082	.039	.011
-2	.039	.135	.287	.368	.287	.135	.039
-1	.082	.287	.606	.779	.606	.287	.082
0	.105	.368	.779	1.000	.779	.368	.105
1	.082	.287	.606	.779	.606	.287	.082
2	.039	.135	.287	.368	.287	.135	.039
3	.011	.039	.082	.105	.082	.039	.011

为了计算方便，一般希望滤波器权值是整数值。在阵列的一个角点处取一个值，并选择 k 使得该角点处的值为 1。运用上面的例子，我们得到：

$$\frac{g[3,3]}{k} = e^{-\frac{(3^2+3^2)}{2(2^2)}} = 0.011 \implies k = \frac{g[3,3]}{0.011} = \frac{10}{0.011} = 91.$$

现在，用 k 乘余下的权值，有

$[i,j]$	-3	-2	-1	0	1	2	3
-3	1	4	7	10	7	4	1
-2	4	12	26	33	26	12	4
-1	7	26	55	71	55	26	7
0	10	33	71	91	71	33	10
1	7	26	55	71	55	26	7
2	4	12	26	33	26	12	4
3	1	4	7	10	7	4	1

这是高斯滤波器卷积模板的最后结果。但是，这一模板的权值之和并不等于 1。所以，在进行卷积的时候，像素点的输出值必须用模板的权值来规范化，以确保图像的均匀灰度区域不受影响。从上面的例中，有

$$\sum_{i=-3}^3 \sum_{j=-3}^3 g[i,j] = 1151.$$

所以，

$$h[i,j] = \frac{1}{1151} (f[i,j] * g[i,j])$$

其中， $g[i,j]$ 的权值为所有的整数。其它普通滤波器的模板见图 5. 11。

(6) 离散高斯滤波器

高斯滤波器的采样值或者高斯滤波器的二项式展开系数可以形成离散高斯滤波器。当用离散高斯滤波器进行卷积时，其结果是一个更大的高斯离散滤波器。若一幅图像用 $n \times n$ 离散高斯滤波器进行平滑，接着再用 $m \times m$ 离散高斯滤波器平滑的话，那么平滑结果就和用 $(n+m-1) \times (n+m-1)$ 离散高斯滤波器平滑的结果一样。换言之，在杨辉三角形中用第 n 行和第 m 行卷积形成了第 $n+m-1$ 行。

7 × 7 高斯模板

1	1	2	2	2	1	1
1	2	2	4	2	2	1
2	2	4	8	4	2	2
2	4	8	16	8	4	2
2	2	4	8	4	2	2
1	2	2	4	2	2	1
1	1	2	2	2	1	1

15 × 15 高斯模板

2	2	3	4	5	5	6	6	6	5	5	4	3	2	2
2	3	4	5	7	7	8	8	8	7	7	5	4	3	2
3	4	6	7	9	10	10	11	10	10	9	7	6	4	3
4	5	7	9	10	12	13	13	13	12	10	9	7	5	4
5	7	9	11	13	14	15	16	15	14	13	11	9	7	5
5	7	10	12	14	16	17	18	17	16	14	12	10	7	5
6	8	10	13	15	17	19	19	19	17	15	13	10	8	6
6	8	11	13	16	18	19	20	19	18	16	13	11	8	6
6	8	10	13	15	17	19	19	19	17	15	13	10	8	6
5	7	10	12	14	16	17	18	17	16	14	12	10	7	5
5	7	9	11	13	14	15	16	15	14	13	11	9	7	5
4	5	7	9	10	12	13	13	13	12	10	9	7	5	4
3	4	6	7	9	10	10	11	10	10	9	7	6	4	3
2	3	4	5	7	7	8	8	8	7	7	5	4	3	2
2	2	3	4	5	5	6	6	6	5	5	4	3	2	2

图 5. 11 其它常用的高斯模板

5. 4 非线性滤波

均值滤波和高斯滤波运算的主要问题是有可能模糊图像中的尖锐不连续部分. 下面介绍两种非线性滤波算法: 中值滤波和边缘保持滤波算法。

5. 5. 1 中值滤波

中值滤波的基本思想是用像素点邻域灰度值的中值来代替该像素点的灰度值, 该方法在去除脉冲噪声、椒盐噪声的同时又能保留图像边缘细节, 这是因为它不依赖于邻域内那些与典型值差别很大的值. 中值滤波器在处理连续图像窗函数时与线性滤波器的工作方式类似, 但滤波过程却不再是加权运算. 例如, 取 3×3 函数窗, 计算以点 $[i, j]$ 为中心的函数窗像素中值步骤如下:

- (1) 按强度值大小排列像素点.
- (2) 选择排序像素集的中间值作为点 $[i, j]$ 的新值.

这一过程如图 5. 12 所示. 一般采用奇数点的邻域来计算中值, 但如果像素点数为偶数时, 中值就取排序像素中间两点的平均值. 采用大小不同邻域的中值滤波器的结果见图 5. 13.

中值滤波在一定条件下, 可以克服线性滤波器 (如均值滤波等) 所带来的图像细节模糊, 而且对滤除脉冲干扰即图像扫描噪声最为有效. 在实际运算过程中并不需要图像的统计特性, 也给计算带来不少方便. 但是对一些细节多, 特别是线、尖顶等细节多的图像不宜采用中值滤波。

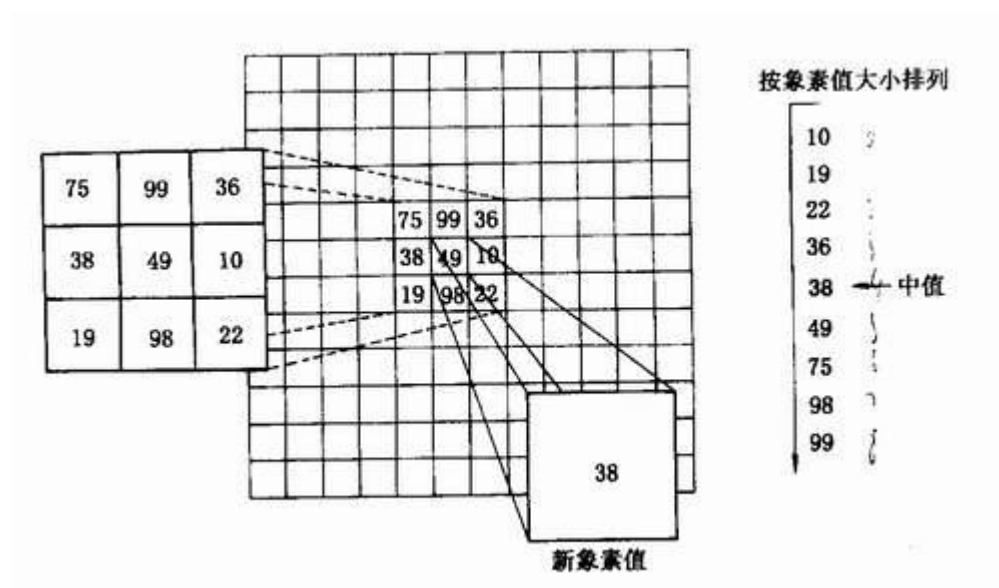


图 5. 12 采用 3×3 邻域的中值滤波器示意图.

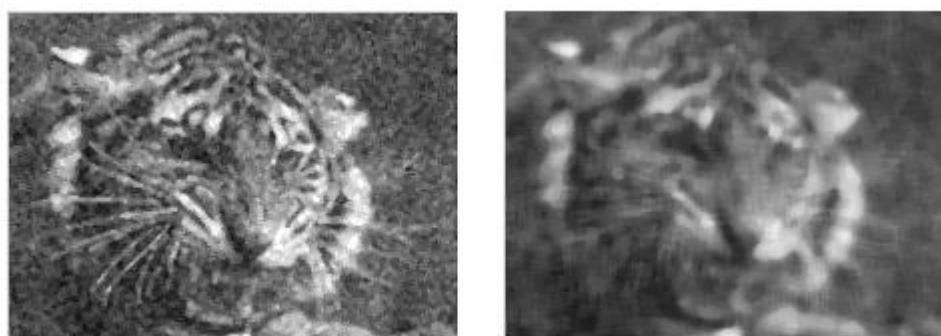


图 5. 13 用 3×3 , 7×7 窗函数对图 5. 4(b)噪声图像进行中值滤波后的结果.

5. 5. 2 边缘保持滤波器

均值滤波的平滑功能会使图像边缘模糊,而中值滤波在去除脉冲噪声的同时也将图像中的线条细节滤除掉,如图 5.1 4 所示.边缘保持滤波器是在上述两种滤波器的基础上发展的一种滤波器,该滤波器在滤除噪声脉冲的同时,又不致于使图像边缘十分模糊.

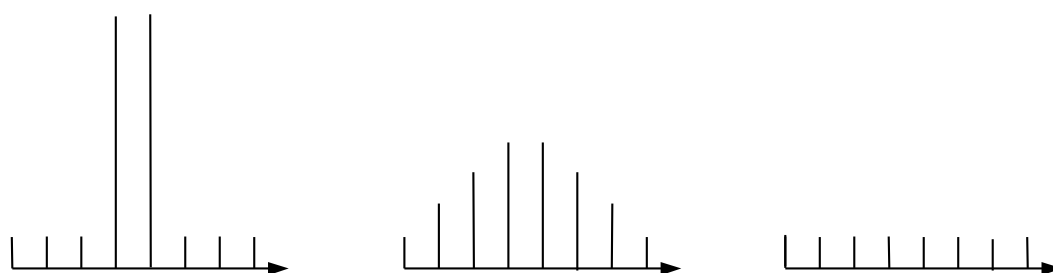


图 5.1 4 尖顶边缘滤波示意图. (a) 原始信号 (b) 均值滤波, (c) 中值滤波.
其中设定模板为 5×5 .

边缘保持算法的基本过程如下：对灰度图像的每一个像素点 $[i, j]$ 取适当大小的一个邻域（如 3×3 邻域），分别计算 $[i, j]$ 的左上角子邻域、左下角子邻域、右上角子邻域和右下角子邻域的灰度分布均匀度 V ，然后取最小均匀度对应区域的均值作为该像素点的新的灰度值。

计算灰度均匀度的公式为：

$$V = \sum f^2(i, j) - (\sum f(i, j))^2 / N \quad (5.10)$$

也可以用如下式计算：

$$V = \sum (f_{ij} - \bar{f})^2 \quad (5.11)$$

举例：计算下图所示的一个邻域均匀度：

0	1	1
0	0	1
0	0	1

0 1	1 1	0 0	0 1
0 0	0 1	0 0	0 1
$V = 3/4$	$V = 3/4$	$V = 0$	$V = 1$

图 5.15 计算邻域均匀度例子

由上面计算可见，分布越均匀， V 值越小。左下角邻域全为零，因此， $V = 0$ 为最小值。像素 $[i, j]$ 处的灰度值应为左下角邻域灰度值的均值，即 $f[i, j] = 0$ 。图 5.16 是用边缘保持滤波的实验结果。

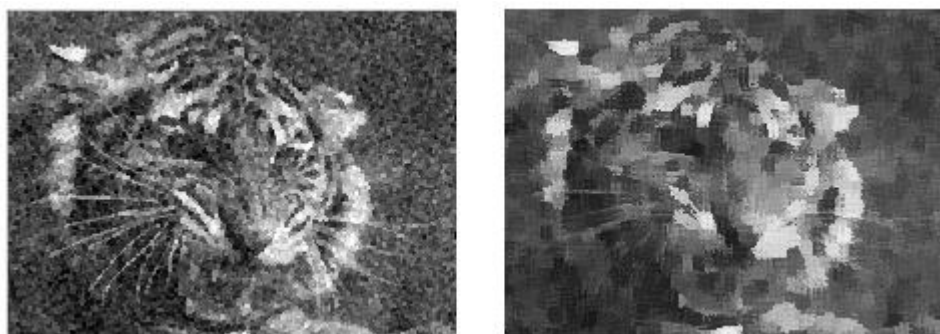


图 5.16 用 3×3 和 7×7 窗函数对图 5. 4(b)噪声图像进行边缘保持滤波的结果

思考题

- 5.1 均值滤波器是线性滤波器，而中值滤波器却不是，为什么？
- 5.2 比较中值滤波器和均值滤波器的特征并鉴别其使用条件。
- 5.3 高斯滤波常常是求均值的首选方法，为什么？
- 5.4 什么是高斯滤波可分离性质？为什么要求滤波是可分离的？
- 5.5 在许多应用中，一幅图像用几种不同尺度的高斯滤波器来进行平滑，为什么要用几种不同参数的滤波器来平滑呢？

5.6 高斯滤波器的级联性质是什么？它在机器视觉中有什么用处？

5.7 一幅 8×8 图像 $f[i, j]$ 的灰度值由下列方程给出：

$$f[i, j] = |i - j|; \quad i, j = 0, 1, 2, 3, 4, 5, 6, 7$$

用 3×3 中值滤波器作用于该图像上，求输出图像。注意边界像素保持不变。

计算机练习题

- 5.1 用直方图修正方法实现对比度伸展，其中用直方图区间 $[a, b]$ 和 $[z_1, z_k]$ 作为变量。取一幅对比度图很弱的图像做实验来看一看直方图修正法对对比度增强的影响。实现几种不同对比度增强方法，并对它们的性能进行评价。
- 5.2 实现高斯平滑滤波器。选择几个不同的 σ 值(至少为 5 个)对一幅图像进行滤波，观测平滑量。你将如何为一幅图像选择合适的 σ 值？
- 5.3 实现 3×3 和 7×7 中值滤波和边缘保持滤波。试比较它们的结果。