

- 数字图像、DIP 数字图像处理、图像获取、图像增强、图像复原、图像分割、彩色图像处理、形态学处理
- Pixel 像素、亮度、光强、电磁波谱、仿射变换、图像配准
- 二值图像、灰度图像、彩色图像、索引色图像
- Sampling 采样、Quantization 量化、空间分辨率、灰度分辨率、灰度均值、灰度方差、灰度变换、伽马校正、对比度拉伸、阈值处理
- Histogram 直方图、直方图归一化、直方图均衡化、PDF 概率密度函数、CDF 累积分布函数、直方图匹配
- 4 邻域、8 邻域、连通域
- 空间相关、卷积、空间滤波、平滑滤波器、锐化滤波器
- 椒盐噪声、高斯噪声、周期噪声
- 傅里叶级数、DFT、FFT、DCT、二维卷积定理、DC 频域直流分量、频域滤波
- 三基色、色度图、XYZ、Lab、RGB、CMYK、HSI、YUV、彩色补偿、彩色平衡（白平衡）
- 结构元素、膨胀、腐蚀、开运算、闭运算
- 图像梯度

数字图像是指物理图像的连续信号值被离散化后,由被称作像素的小块区域组成的二维矩阵。数字图像是图像的数字表示,像素是其最小的单位。数字图像的描述有:

1) 无彩色图像 2) 彩色图像

数字图像处理 digital image processing

- 1) 将一幅图像变为另一幅经过加工的图像,是图像到图像的过程。
- 2) 将一幅图像转化为一种非图像的表示,如一个决策等。

广义的**图像获取**指的是,将物体成像的过程。多媒体技术中的**图像获取**指,为了能在计算机上对图像进行加工处理,而将目标通过扫描仪等设备,转换成数字图像信息的过程。

图像的增强 加强图像的有用信息,消弱干扰和噪声

图像的恢复 把退化、模糊了的图像复原。模糊的原因有许多种,最常见的有运动模糊,散焦模糊等等。

图像分割与特征提取 图像分割是指通过某种方法,使得画面场景被分为“目标物”及“非目标物”两类,即将图像的像素变换为黑、白两种。因为结果图像为二值图像,所以通常又称图像分割为图像的二值化处理

彩色图像处理 针对图像处理中的颜色信息进行专门处理的技术

形态学处理: 从图像中提取对于表达和描绘区域形状有意义的图像分量,使后续的认识工作能够抓住目标对象最为本质(最具区分能力——most discrimination)的形状特征,像是边界、连通区域等。

像素: 是指在由一个数字序列表示的图像中的一个最小单位,称为像素。(不一定是方格)

亮度: 也称为灰度,它是颜色的明暗变化。亮度是指发光体(光强与人眼所“见到”的光

源面积之比，定义为该光源单位的亮度，即单位投影面积上的发光强度。亮度的单位是坎德拉/平方米（cd/m²）亮度是人对光的强度的感受。它是一个主观的量。与光照度不同的，由物理定义的客观的相应的量是光强。这两个量在一般的日常用语中往往被混淆。亮度也称明度，表示色彩的明暗程度。人眼所感受到的亮度是色彩反射或透射的光亮所决定的。

光强：观察者从一个光源感知的能量 流明 。发光强度简称光强，国际单位是 candela（坎德拉）简写 cd，其他单位有烛光，支光。1cd 即 1000mcd 是指单色光源（频率 540×10^{12} Hz）的光，在给定方向上（该方向上的辐射强度为（1/683）瓦特/球面度）的单位立体角发出的光通量。可以用基尔霍夫积分定理计算。

电磁波谱：电磁波按照它们的波长或频率、波数、能量的大小顺序进行排列

仿射变换：图像仿射变换提出的意义是采用通用的数学影射变换公式，来表示前面给出的几何变换。又称仿射映射，是指在几何中，一个向量空间进行一次线性变换并接上一个平移，变换为另一个向量空间。

图像配准(Image registration)就是将不同时间、不同传感器（成像设备）或不同条件下（天候、照度、摄像位置和角度等）获取的两幅或多幅图像进行匹配、叠加的过程，

二值图像（Binary Image）是指将图像上的每一个像素像素只能是黑或者白，没有中间的过渡

灰度图像（gray image）是每个像素的信息由一个量化的灰度级来描述的图像，没有彩色信息。

彩色图像是指每个像素由 R、G、B 分量构成的图像，其中 R、G、B 是由不同的灰度级来描述的。

索引色图像是一种把像素值直接作为 RGB 调色板下标的图像。索引图像可把像素值“直接映射”为调色板数值

采样（sampling）也称取样，指把时间域或空间域连续量转化成离散量的过程。

量化在数字信号处理领域，是指将信号的连续取值（或者大量可能的离散取值）近似为有限多个（或较少的）离散值的过程。

空间分辨率，是指遥感图像上能够详细区分的最小单元的尺寸或大小，是用来表征影像分辨地面目标细节的指标。

灰度分辨率，是用来体现显示器区分灰度的能力的色阶。

图像某一区域内**灰度的方差**确实能从一个角度反应区域内图像的对比度

方差越大，图像区域内像素灰度变化越剧烈

灰度的平均值表示图像的**总体亮度**

灰度变换是指根据某种目标条件按一定变换关系逐点改变源图像中每一个像素灰度值的方法。目的是为了改善画质，使图像的显示效果更加清晰。

伽马校正 又叫**伽马非线性化**或**伽马编码**，是用来针对影片或是影像系统里对于光线的辉度或是三色刺激值所进行非线性的运算或反运算。

灰度拉伸又叫**对比度拉伸**，它是最基本的一种灰度变换，使用的是最简单的分段线性变换函数，它的主要思想是提高图像处理时灰度级的动态范围。

【阈值处理】

图像**阈值处理**是实现图像分割的一种方法，

灰度直方图是灰度级的函数，是对图像中灰度级分布的统计。有两种表示形式

直方图归一化：求出图像 f 的总体像素个数， $N_f = m \times n$ （ m, n 分别为图像的长和宽），计算每个灰度级的像素个数在整个图像中所占的百分比。

$$h_s(i) = h(i) / N_f \quad (i=0, 1, \dots, 255)$$

直方图均衡方法的基本思想是对在图像中像素个数多的灰度级进行展宽，而对像素个数少的灰度级进行缩减。从而达到清晰图像的目的。

直方图匹配：直方图匹配又称为直方图规定化，是指将一幅图像的直方图变成规定形状的直方图而进行的图像增强方法。即将某幅影像或某一区域的直方图匹配到另一幅影像上。使两幅影像的色调保持一致。可以在单波段影像直方图之间进行匹配，也可以对多波段影像进行同时匹配。两幅图像比对前，通常要使其直方图形式一致。

PDF 概率密度函数：在数学中，连续型随机变量的概率密度函数（在不至于混淆时可以简称为密度函数）是一个描述这个随机变量的输出值，在某个确定的取值点附近的可能性的函数。

CDF 累积分布函数：累积分布函数(Cumulative Distribution Function)，又叫分布函数，是概率密度函数的积分，能完整描述一个实随机变量 X 的概率分布。一般以大写 CDF 标记，与概率密度函数 probability density function（小写 pdf）相对。

四连接：当前像素为黑，其四个近邻像素中至少有一个为黑；

八连接：当前像素为黑，其八个近邻像素中至少有一个为黑。

连通域：将相互连在一起的黑色像素的集合称为一个连通域。

4 邻域：当前像素的上下左右四个元素。

8 邻域：当前元素近邻的八个元素。

【空间相关】

卷积：卷积是通过两个函数 f 和 g 生成第三个函数的一种数学算子，表征函数 f 与 g 经过翻转和平移的重叠部分函数值乘积对重叠长度的积分。

空间滤波：空间滤波是一种采用滤波处理的影像增强方法。其理论基础是空间卷积和空间相关。目的是改善影像质量，包括去除高频噪声与干扰，及影像边缘增强、线性增强以及去模糊等。分为低通滤波（平滑化）、高通滤波（锐化）和带通滤波。处理方法有计算机处理（数字滤波）和光学信息处理两种。

平滑滤波器：平滑滤波器是一种低通滤波器，是在空间域实现的一种滤波器。通过缩小高频，扩大低频可以去除某些噪声。

锐化滤波器：是将图像的低频部分减弱或去除，保留图像的高频部分，即图像的边缘信息。

椒盐噪声的特征：出现位置是随机的，但噪声的幅值是基本相同的。

高斯噪声的特征：出现在位置是一定的（每一点上），但噪声的幅值是随机的。

周期噪声：来源：电或电磁干扰

特性：空间相关 周期性 - 在频率域很容易观察到

傅里叶级数：法国数学家傅里叶发现，任何周期函数都可以用正弦函数和余弦函数构成的无穷级数来表示（选择正弦函数与余弦函数作为基函数是因为它们是正交的），后世称傅里

叶级数为一种特殊的三角级数

DFT: 离散傅里叶变换 (DFT), 是傅里叶变换在时域和频域上都呈现离散的形式, 将时域信号的采样变换为在离散时间傅里叶变换 (DTFT) 频域的采样。在形式上, 变换两端 (时域和频域上) 的序列是有限长的, 而实际上这两组序列都应当被认为是离散周期信号的主值序列。即使对有限长的离散信号作 DFT, 也应当将其看作经过周期延拓成为周期信号再作变换。

FFT: 快速傅里叶变换 (fast Fourier transform), 即利用计算机计算离散傅里叶变换 (DFT) 的高效、快速计算方法的统称, 简称 FFT。FFT 的基本思想是把原始的 N 点序列, 依次分解成一系列的短序列。充分利用 DFT 计算式中指数因子 所具有的对称性质和周期性质, 进而求出这些短序列相应的 DFT 并进行适当组合, 达到删除重复计算, 减少乘法运算和简化结构的目的。

二维卷积定理: 二维卷积定理 ^[1] (two-dimensional convolution theorem) 数字信号处理的基本定理之一对于二维线性非移变系统, 输出序列 $y(m,n)$ 等于输入序列 $x(m,n)$ 与单位脉冲响应序列 $h(m,n)$ 的卷积和, 即

$$\begin{aligned} &= \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} x(n_1, n_2) h(m - n_1, n - n_2) \\ &= \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} h(n_1, n_2) x(m - n_1, n - n_2). \end{aligned}$$

DC 频域直流分量: 表示为图像的内部信息, 多为连续的颜色, 无频率变化。

频域滤波: 频率域滤波是对图像进行傅里叶变换, 将图像由图像空间转换到频域空间, 然后在频率域中对图像的频谱作分析处理, 以改变图像的频率特征。

以 700nm(红)、546.1nm(绿)、435.8nm(蓝)三个色光为**三基色**。

色度图: 色度图是黑体轨迹的函数表达式 $v=f(u)$ 在色度学中以色度坐标表示的平面图。

XYZ 三刺激值 亮度、色调、饱和度 $x+y+z=1$

Lab: L: 明亮度 a: 从绿色到红色 b: 从蓝色到黄色

RGB 色系: 700nm(红)、546.1nm(绿)、435.8nm(蓝)三个色光为**三基色**

HSI 色系:

H: 表示色度, 由角度表示。反映了该颜色最接近什么样的光谱波长。0° 为红色, 120° 为绿色, 240° 为蓝色。

I 表示光照强度或称为亮度。 **S:** 表示饱和度, 饱和度参数是色环的原点到彩色点的半径长度。在环的外围圆周是纯的或称饱和的颜色, 其饱和度值为 1。在中心是中性 (灰) 色, 即饱和度为 0。

CMYK 色系: 用于印刷行业, 是一种减色系统, 将从白光中滤出三种原色之后获得的颜色作为其表色系的三原色 CMY。K 为黑色, 为了印刷时对黑色可用黑色墨来印刷。C: 青色, 从白色中滤去红色。M: 品红, 从白色中滤去绿色。Y: 黄色, 从白色中滤去蓝色。

YUV: Y: 亮度; U, V: 色差信号

YCbCr: 这是常用于彩色图像压缩时的一种表色系。Y: 代表亮度; Cb、Cr: 代表色差

彩色补偿: 通过数学运算, 将扩散进来的颜色分量补偿掉。由此, 使不同的目标在不同的颜色分量中信号最强。

彩色平衡 (白平衡): 当一幅彩色图像数字化后, 在显示时颜色经常看起来有些不正常。这

是因为色通道的不同敏感度、增光因子、偏移量等原因导致。称之为三基色不平衡。将其校正的过程就是彩色平衡。

【结构元素】

腐蚀 是一种消除连通域的边界点，使边界向内收缩的处理。

膨胀是将与目标区域的背景点合并到该目标物中，使目标物边界向外部扩张的处理。

开运算是对原图先进行腐蚀处理，后再进行膨胀的处理。

闭运算是对原图先进行膨胀处理，后再进行腐蚀的处理。

图像梯度：图像梯度可以把图像看成二维离散函数，图像梯度其实就是这个二维离散函数的求导：图像梯度： $G(x, y) = dx(i, j) + dy(i, j)$ ；

$dx(i, j) = I(i+1, j) - I(i, j)$ ； $dy(i, j) = I(i, j+1) - I(i, j)$ ；

其中， I 是图像像素的值(如：RGB 值)， (i, j) 为像素的坐标。图像梯度一般也可以用中值

差分： $dx(i, j) = [I(i+1, j) - I(i-1, j)]/2$ ； $dy(i, j) = [I(i, j+1) - I(i, j-1)]/2$ ；

图像边缘一般都是通过对图像进行梯度运算来实现的。

复习概况

【第一章：基本概念】

数字图像：像素和颜色离散且有限。

数字图像的坐标表示 xy ；0 开始

可见光范围 380 – 780 nm

图像的坐标：矩阵坐标系

0 黑；1 白

索引色图像的调色板

均匀量化与非均匀量化：量化级分布不同

采样、量化

伪轮廓现象

【第三章：灰度变换】

对比度拉伸变换

阈值分割

负片、 \log 变换、 γ 变换 ($\gamma > 1$ ，向下弯， $\gamma < 1$ ，向上弯)、

γ 变换的矩阵计算

$$I = C \cdot L^\gamma$$

当 γ 不准确时的校准结果

对比度展宽

动态范围调整计算：线性动态调整

$$\therefore L = (I/C)^{1/\gamma} = \bar{C} \cdot I^{1/\gamma}$$

直方图均衡化：某一灰度级像素越多，与下一灰度级相隔越远。

直方图匹配：是指使一幅图像的直方图变成规定形状的直方图而进行的图像增强方法。

【第四章：图像的几何变换】

计算：图像缩小和放大矩阵计算

放大倍数过大：马赛克现象

图像的错切

位置变换：平移、镜像、旋转

旋转后处理：插值（行插值/列插值：填补空洞）、边界的锯齿（仿射变换）

【第五章：噪声抑制】

计算：滤波结果

K 近邻平滑滤波器 需要确定 K 值

■ 边界保持滤波器的核心是确定边界点与非边界点。

对称近邻平滑滤波器

最小方差平滑滤波器

Sigma 平滑滤波器

【第六章】

一阶微分锐化的方向

$$H = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad \text{水平锐化 已旋转} \qquad H = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad \text{垂直锐化}$$

单方向锐化后处理：

整体加一个正整数=>类似浮雕效果

取绝对值：对边缘进行有方向的提取

无方向一阶锐化：Roberts 算子（交叉微分算子）

Sobel 算子 dx dy 两个方向 均方根

Priwitt 锐化算法

二阶微分锐化算子：拉普拉斯算子

图像增强：4->5

加入对数环节：整体除以 4 Wallis 算法 对暗区细节进行比较好的锐化

$$H_1 = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

【第七章：图像分割】/二值化处理

p-参数法

均匀性度量法 类内

聚类方法 类间

ADI 累积差异图像 去除背景干扰

【第八章：二值图像分析】

四/八连接 连通域

贴标签

腐蚀 注意目标是黑色还是白色

膨胀

【第九章：彩色图像】

颜色系统

XYZ Lab RGB HIS CMYK（吸收红/绿/蓝）

YUV YCbCr

灰度级修正、平滑滤波、图像锐化

彩色补偿 颜色扩散

三基色不平衡

【第十章：图像的频域变换】

FFFT 分组

【第十一章：编码技术】

数据冗余 有损压缩、无损压缩

计算压缩率

行程编码=>确定数据位的位数

熵编码/霍夫曼编码

DCCT 编码：有损编码，取整

【图像缩放技术】

最邻近插值法

双线性内插法 计算双线性内插结果

立方卷积差值法

【形态学图像操作】

开运算：腐蚀+膨胀

闭运算：膨胀+腐蚀

开运算：断开窄连接、消除小凸起

闭运算：连接小断点、消除小孔洞、填补空隙

重复做开/闭运算等于做一次开/闭运算。如果想要改变效果，多次腐蚀/膨胀

击中击不中运算：W、W-X A A 的补集

区域填充

连通分量提取

骨架

【实验部分】

图像显示：

```
I=double(imread('lena.png'));  
imshow(I);
```

=====

```
imshow(uint8(I));#方法一 将数据转化到 Uint8 类型  
imshow (I/255); #方法二 将数据值转化到 0 与 1 之间  
imshow(I,[]);#方法三 []的目的是自动调整数据的范围以便于显示
```

imshow 对 double 和 uint8 的显示范围不一样

(uint8(I))按 256 级灰度值显示图像。0 表示黑色，255 表示白色。而大于 255 的值均变成 255，即白色；

```
BW = im2bw(I, level)
```

```
BW = im2bw(X, map, level)
```

```
BW = im2bw(RGB, level)
```

其中 level 就是设置阈值的。level 取值范围[0, 1]。

实现图像矩阵的[归一化](#)操作。所谓"归一化"就是使矩阵的每个元素的值都在 [0 和 1 之间](#)。

uint8 类型的图像算术运算前必须转换成 double 类型

```
ones (3) eye (3) .
```

空间域的横线到频域中就变成了竖线 灰度级变换

时域卷积在频域上直接矩阵运算即可，**不需要卷积**，从而运算量低

高斯滤波也是一种线性滤波器

`imfilter` 可进行多维图像（RGB 等）进行空间滤波，且可选参数较多

`filter2` 只能对二维图像（灰度图）进行空间滤波

两个函数结果类型不一样，只需要在 `I1=filter2(h,I)`后面加上 `I1=uint8(I1)`进行类型转换，结果就是一样的。

【matlab 中的一些函数】

Matlab 一些函数

--from wyh

- `imread`, `size`, `imshow`, `figure`, `imwrite`, `imfinfo`, `im2uint8`, `mat2gray`, `im2double`, `im2bw`, `imadd`, `imsubtract`, `immultiply`, `imdivide`, `imabsdiff`, `imcomplement`
- `imadjust`, `imhist`, `histeq`, `imfilter`, `filter2`, `colfilt`, `padarray`, `fspecial`, `ordfilt2`, `medfilt2`
- `paddedsz`, `fft2`, `ifft2`, `conv2`, `fftshift`, `lpfilter`, `imnoise`, `edge`
- `bwlabel`, `imdilate`, `imerode`, `imopen`, `imclose`

```
f=imread('grey.jpg');
[M, N] = size(f);
figure%; %新建一个窗口，所有参数采用默认
imshow(f);%重复使用只能产生一个窗口
figure(s); %s 为参数，s 为数据时要大于 0，否则报错
imwrite(f, 'grey3.tif');
imfinfo('grey.jpg'); %不能imfinfo(f);
```

Name	Converts Input to:	Valid Input Image Data Classes
<code>im2uint8</code>	<code>uint8</code>	<code>logical</code> , <code>uint8</code> , <code>uint16</code> , and <code>double</code>
<code>im2uint16</code>	<code>uint16</code>	<code>logical</code> , <code>uint8</code> , <code>uint16</code> , and <code>double</code>
<code>mat2gray</code>	<code>double</code> (in range [0, 1])	<code>double</code>
<code>im2double</code>	<code>double</code>	<code>logical</code> , <code>uint8</code> , <code>uint16</code> , and <code>double</code>
<code>im2bw</code>	<code>logical</code>	<code>uint8</code> , <code>uint16</code> , and <code>double</code>

```
[g,gmap]=rgb2ind(p,256); %彩色图像转换为索引色图像
h=imadd(f,f); %必须相同大小 结果变白
h=imsubtract(f,f); %必须相同大小 结果全黑
h=immultiply(f,f); %必须相同大小 结果变白
h=imdivide(f,f); %必须相同大小 结果全黑（肉眼）
h=imabsdiff(f,f); %必须相同大小 结果全黑
h=imcomplement(f); %对图像取反（负片）
```

```
J=imadjust( I, [low_in;high_in], [low_out;high_out]);
%[low_in;high_in]为原图像中要变换的灰度范围，[low_out;high_out]为变换后的灰度范围。该 gamma 参数为映射的方式，默认值为 1，即线性映射。当 gamma 不等于 1 时为非线性映射。
g1 = imadjust(f,[0 1],[1 0]); %负片
```


h = imhist(f, b); %b 为收集箱的个数（灰度级的个数）默认值=256 比如 b=2, 两块: [0,127]和[128,255]

imhist(f); %直接显示即可 imshow(h) 不能出现正确图像

axis tight; %加坐标轴

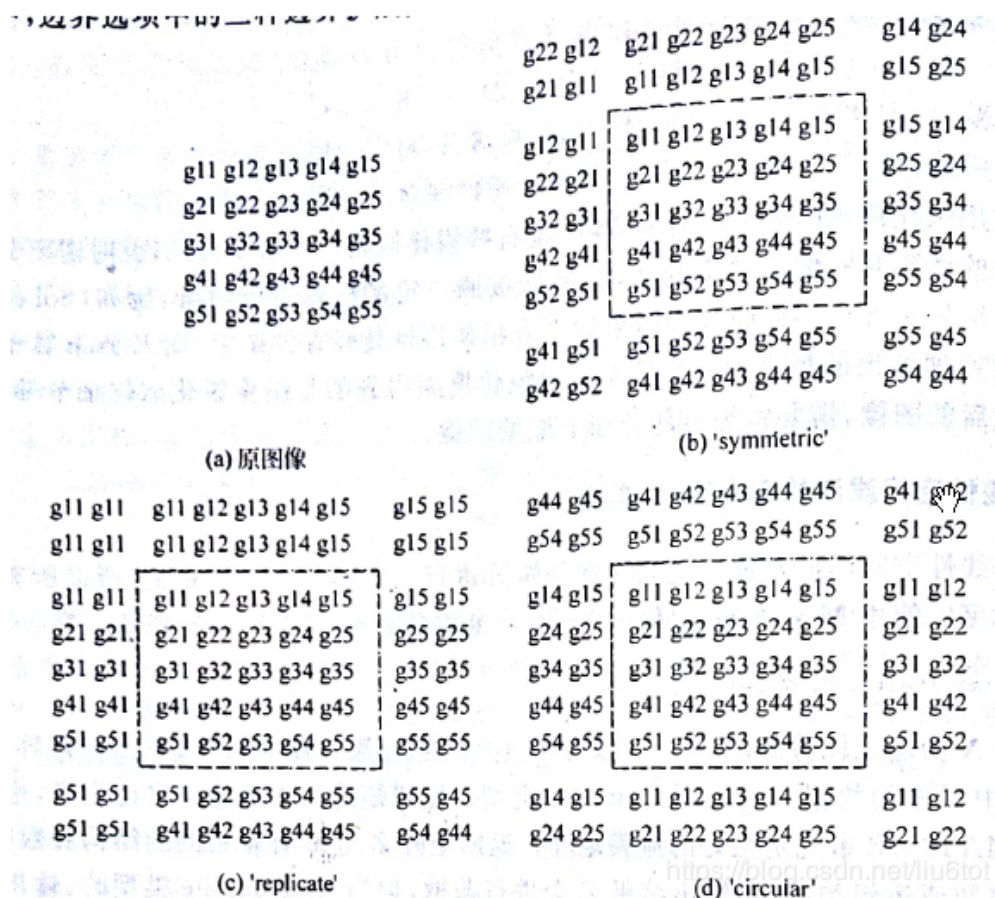
p = imhist(f, b) / numel(f); %归一化直方图 分母为像素个数

p=histeq(f); %直方图均衡化 imhist(p);

g=imfilter(f,w,filtering_mode,boundary_options,size_optinos);

%f 是输入图像, w 为滤波模板, g 为滤波结果

选项		说明
滤波模式	'corr'	滤波通过使用相关来完成, 这是默认值
filtering_mode	'conv'	滤波通过使用卷积来完成
边界选项	P	输入图像的边界通过值 P 填充来扩展; P 的默认值是 0
boundary_options	'replicate'	图像大小通过复制外边界的值来扩展
	'symmetric'	图像大小通过沿自身的边界进行镜像映射扩展
	'circular'	图像大小通过将图像作为二维周期函数的一个周期来扩展
大小选项	'full'	输出图像的大小与被扩展（填充）图像的大小相同
size_optinos	'same'	输出图像的大小与输入图像的大小相同, 这可通过将滤波模板的中心点的偏移限制为原始图像中包含的点来实现, 这是默认值



filter2(B,X); %B 为滤波器,X 为要滤波的数据,这里将 B 放在 X 上,一个一个移动进行模板滤波.

```

colfilt(f,[m,n],'sliding',@fun,parameters);%非线性滤波
padarray(f,[r c],method,direction);%预先填充
w = fspecial('type', parameters); %标准线性空间滤波器
g = ordfilt2(f, order, domain) ; %统计排序滤波器（最大值 最小值 中值）
g = ordfilt2(f, median(1:m*n), ones(m, n)); %中值滤波器
g = medfilt2(f, [m n], padopt); %中值滤波器
pq=paddedsize(size(g)); %含有两个元素 好像是长宽都乘2???
F = fft2(f); %二维快速傅里叶变换
i=ifft2(F); %傅里叶反变换
conv2(x,y); %二维卷积
ff=fftshift(fft2(f)); imshow(log(1+abs(ff)), [ ]);%作用：将零频点移到频谱的中间
Hp = lpfilter('gaussian', PQ(1), PQ(2), sig); %计算频率域低通滤波器
y = imnoise(X,'salt & pepper',p); %椒盐噪声
y = imnoise(X,'gaussian',m,v); %高斯白噪声
h=edge(f); %采用灰度或一个二值化图像I作为它的输入，并返回一个与I相同大小的二值化图像BW，在函数检测到边缘的地方为1，其他地方为0。

```

h=bwlabel(f); %返回一个和BW大小相同的L矩阵，包含了标记了BW中每个连通区域的类别标签，这些标签的值为1、2、num（连通区域的个数）。n的值为4或8，表示是按4连通寻找区域，还是8连通寻找，默认为8。

h=imdilate(f,SE); %膨胀操作 SE为结构元素 结构元原点位置都是默认的中点

strel—structuring element 运用各种形状和大小构造元素，基本语法为

SE = strel(shape, parameters)

shape 是指定希望形状的字符串，**parameters** 是指定形状信息的一系列参数

SE = strel('rectangle', MN)

创建一个平坦的矩形结构，**MN** 指定大小

SE = strel('square', W)

创建一个方形的结构元素，边长为**N**个像素

SE = strel('disk', R, N)

创建一个平坦的圆形结构元素，半径为**R**。

N 必须是 0，4，6 或 8。当 **N** 比 0 大时，圆形结构元素被 **N** 个周期线型（**periodic-line**）结构元素序列近似，当 **N** 等于 0 时，没有近似，结构元素包含所有的小于从原点到 **R** 的像素。

h=imerode(f,SE); %腐蚀操作 SE为结构元素

h=imerode(f,SE); %开运算

h=imclose(f,SE); %闭运算

图像显示

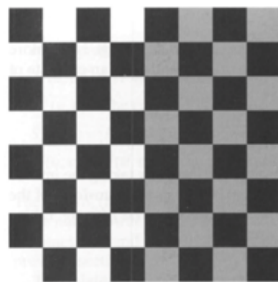
- `colorbar` 显示彩条
- `getimage` 由坐标轴得到图像数据
- `ice (DIPUM)` 交互彩色编辑
- `image` 创建和显示图像对象
- `imagesc` 缩放数据并显示为图像
- `immovie` 由多帧图像制作电影
- `imshow` 显示图像
- `imview` 在Image Viewer中显示图像
- `montage` 将多个图像帧显示为矩阵蒙太奇
- `movie` 播放录制的电影帧
- `rgbcube` 显示一个彩色RGB立方体
- `subimage` 在单个图形中显示多幅图像
- `trueimage` 调整图像的显示尺寸
- `warp` 将图像显示为纹理映射的表面

ICE界面

- Smooth 三次样条（平滑曲线）插值
- Clamp Ends 在三次样条插值中强制开始和结束曲线滚降为0
- Show PDF 显示概率密度函数
- Show CDF 显示累计分布函数
- Map Image 图像映射
- Map Bars 启用伪彩色和全彩色条映射
- Reset 重置当前映射函数及参数
- Reset All 重置所有
- Input/Output 显示坐标水平轴/垂直轴
- Component 颜色空间分量 (R, G, B, RGB) (H, S, I, HIS)

Checkerboard 创建棋盘格图像

- `C = checkerboard (NP, M, N)`
- NP 正方形一边的像素数，默认10
- M 行数，默认8
- N 列数，默认8
- 测试板的左半白色，右半灰色



- `[g, t] = edge(f, 'method', parameters)`

图像噪声

- `Imnoise` 给一幅图像添加噪声
- `Imnoise2 (DIPUM)` 使用指定的PDF生成一个随机数数组
- `Imnoise3 (DIPUM)` 生成周期噪声

ICE 交互式彩色编辑

ice

```
g = ice('image', f);
```

```
g = ice('image', f, 'wait', 'off');
```

```
g = ice('image', f, 'space', 'hsi');
```

- ice的输出总是RGB，输入单色或者RGB
- 鼠标左键：移动控制点
- 左键+Shift：加入控制点
- 左键+Ctrl：删除控制点

几何变换

- `Checkerboard` 创建棋盘格图像
- `Findbounds` 求几何变换的输出范围
- `Flipform` 颠倒TFORM结构的输入/输出
- `Imcrop` 修剪图像
- `Imresize` 调整图像大小
- `Imrotate` 旋转图像
- `Imtransform` 对图像应用几何变换
- `Intline` 整数坐标线绘制算法
- `Makersampler` 创建重取样器结构
- `Maketform` 创建几何变换结构 (TFORM)
- `Pixeldup (DIPUM)` 在两个方向上复制图像的像素
- `Tformarray` 对N-D数组应用几何变换
- `Tformfwd` 应用正向几何变换
- `Tforminv` 应用反向几何变换
- `Vstformfwd (DIPUM)` 可视化正向几何变换

图像增强

- `Adapthisteq` 自适应直方图量化
- `Decorstretch` 对多通道图像应用去相关拉伸
- `Gscale (DIPUM)` 按比例调整输入图像的亮度
- `Histeq` 使用直方图均衡化来增强对比度
- `Intrans (DIPUM)` 执行亮度变换
- `Imadjust` 调整图像亮度值或彩色映射
- `Stretchlim` 寻找对比度拉伸图像的限制

线性和非线性空间滤波

- [Adpmedian](#) (DIPUM) 执行自适应中值滤波
 - [Convmtx2](#) 计算二维卷积矩阵
 - [Dftcorr](#) (DIPUM) 执行频率域相关
 - [Dftfilt](#) (DIPUM) 执行频率域滤波
 - [Fspecial](#) 创建预定义滤波器
 - [Medfilt2](#) 执行二维中值滤波
 - [Imfilter](#) 滤波二维和N维图像
 - [Ordfilter2](#) 执行二维顺序统计滤波
 - [Spfilt](#) (DIPUM) 执行线性和非线性空间滤波
 - [Wiener2](#) 执行二维去噪滤波
-
- [Hpfilt](#) (DIPUM) 计算频率域高通滤波器
 - [Lpfilt](#) (DIPUM) 计算频率域低通滤波器
-
- [Imclose](#) 闭运算图像
 - [Imdilate](#) 膨胀图像
 - [Imerode](#) 腐蚀图像
 - [Imextendedmax](#) 最大扩展变换
 - [Imextendedmin](#) 最小扩展变换
 - [Imfill](#) 填充图像区域和孔洞
 - [Imhmax](#) H最大变换
 - [Imhmin](#) H最小变换
 - [Imimposemin](#) 强制最小
 - [Imopen](#) 开运算图像
-
- [Bwhitmiss](#) 二值击中不中操作

图像类型和类型转换

- [Changeclass](#) 改变一幅图像的类
- [Dither](#) 使用抖动转换图像
- [Gray2ind](#) 将亮度图像转换为索引图像
- [Grayslice](#) 通过阈值处理从亮度图像创建索引图像
- [Im2bw](#) 通过阈值处理将图像转换为二值图像
- [Im2double](#) 将图像数组转换为双精度
- [Im2java](#) 将图像转换为Java图像
- [Im2java2d](#) 将图像转换为Java缓存的图像对象
- [Im2uint8](#) 将图像数组转换为8比特无符号整数
- [Im2uint16](#) 将图像数组转换为16比特无符号整数
- [Ind2gray](#) 将索引图像转换为亮度图像
- [Ind2rgb](#) 将索引图像转换为RGB图像
- [Label2rgb](#) 将标记矩阵转换为RGB图像
- [Mat2gray](#) 将矩阵转换为亮度图像
- [Rgb2gray](#) 将RGB图像或彩色映射转换为灰度图像
- [Rgb2ind](#) 将RGB图像转换为索引图像