

# Matlab 图像处理函数 汇总

IPT & Dipum Toolbox

## 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
- `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
- `trueimage` 调整图像的显示尺寸
- `warp` 将图像显示为纹理映射的表

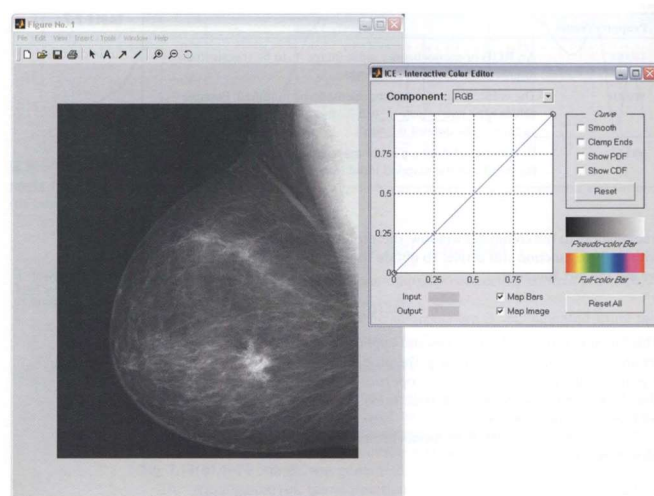
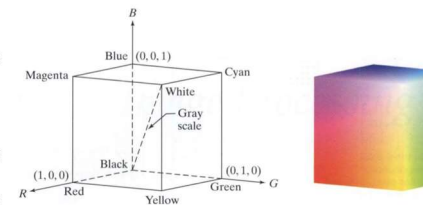


FIGURE 7.15 The typical opening windows of function `ice`. (Image courtesy of G.E. Medical Systems.)

## ICE 交互式彩色编辑

`ice`

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

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

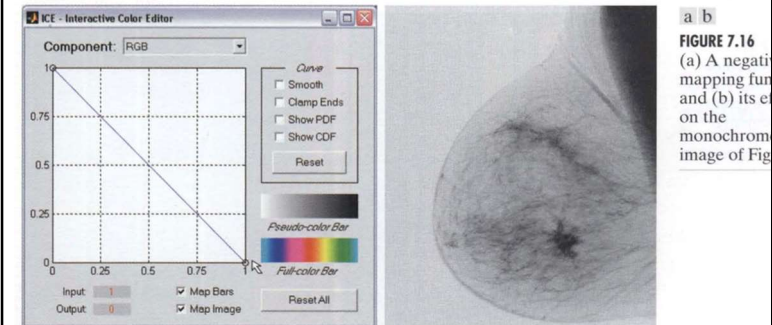
```
g = ice('image', f, 'space', 'hsi');
```

- `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

## 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 显示座标水平轴/垂直轴
- Component 颜色空间分量 (R, G, B, RGB) (H, S, I, HIS)

## 负片

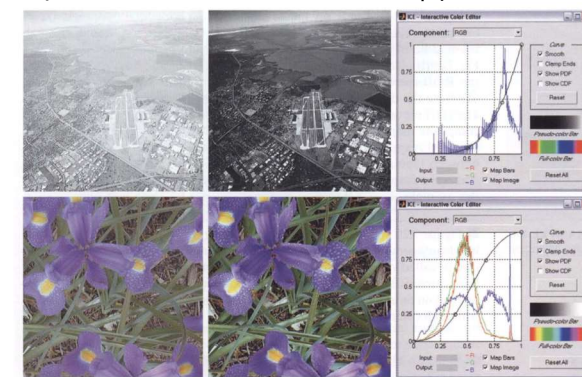


## 彩色的补色

**FIGURE 7.17**  
(a) A full color image, and (b) its negative (color complement).



## 单色和彩色对比度增强

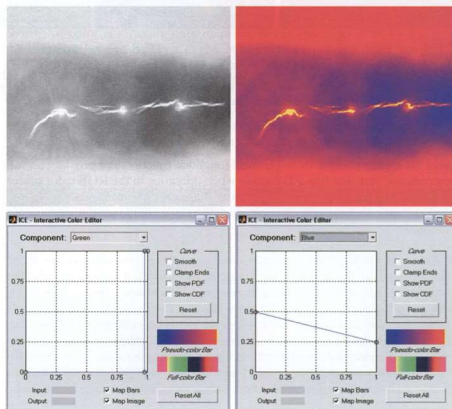


**FIGURE 7.18** Using function ice for monochrome and full color contrast enhancement: (a) and (d) are the input images, both of which have a "washed-out" appearance; (b) and (c) show the processed results; (e) and (f) are the ice displays. (Original monochrome image for this example courtesy of NASA.)

## 伪彩色映射

a b  
c d

FIGURE 7.19  
(a) X-ray of a defective weld;  
(b) a pseudo-color version of the weld; (c) and (d) mapping functions for the green and blue components.  
(Original image courtesy of X-TEK Systems, Ltd.)



色彩平衡  $f2 = \text{ice}('image', f1, 'space', 'CMY')$

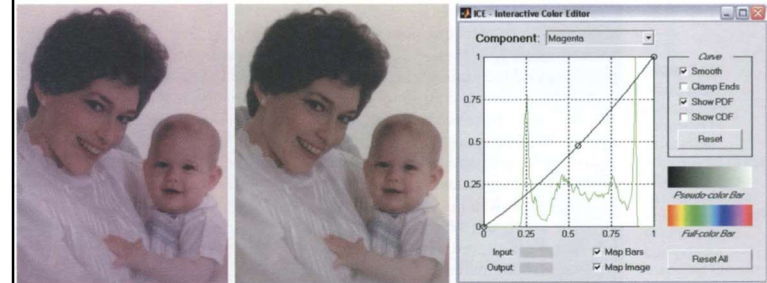


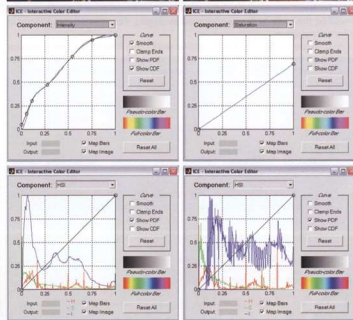
FIGURE 7.20 Using function `ice` for color balancing: (a) an image heavy in magenta; (b) the corrected image; and (c) the mapping function used to correct the imbalance.

## 基于映射的直方图



FIGURE 7.21 Histogram equalization followed by saturation adjustment in the HSI color space: (a) input image; (b) mapped result; (c) intensity component mapping function and cumulative distribution function; (d) saturation component mapping function; (e) input image's component histograms; and (f) mapped result's component histograms.

'space', 'hsi'



## 图像文件输入/输出

DICOM 医学数字成像和通信

- `Dicominfo` 从一条DICOM消息中读取元数据
- `Dicomread` 读一幅DICOM图像
- `Dicomwrite` 写一幅DICOM图像
- `Dicom-dict.txt` 包含DICOM数据字典的文本文件
- `Dicomuid` 产生DICOM唯一的识别器
- `Imfinfo` 返回关于图像的文件的文件的信息
- `Imread` 读图像文件
- `Imwrite` 写图像文件

## 图像算术

- **Imabsdiff** 计算两幅图像的绝对差
- **Imadd** 两幅图像相加或把常数加到图像上
- **Imcomplement** 图像求补
- **Imdivide** 两幅图像相除，或用常数除图像
- **imlincomb** 计算图像的线性组合
- **Immultiply** 两幅图像相乘或用常数乘图像
- **Imsubtract** 两幅图像相减，或从图像中减去常数

## imlincomb 计算图像的线性组合

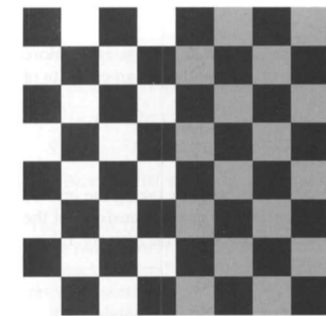
- $B = \text{imlincomb}(c1, A1, c2, A2, \dots, ck, Ak)$
- 相当于  $B = c1 * A1 + c2 * A2 + \dots + ck * Ak$
- $c$  是 double 类实标量， $A$  是有着相同类和大小的数组

## 几何变换

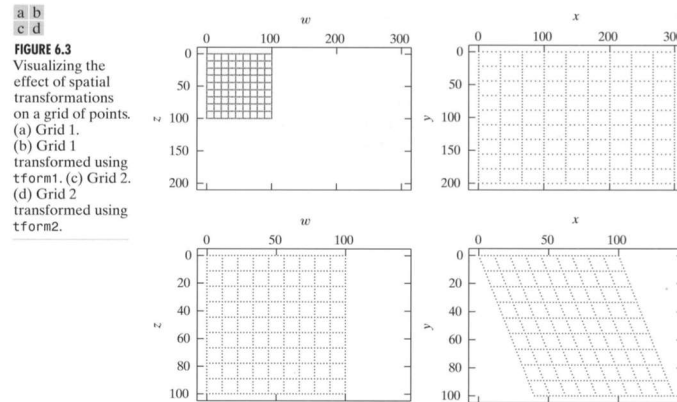
- **Checkerboard** 创建棋盘格图像
- **Findbounds** 求几何变换的输出范围
- **Flptform** 颠倒 TFORM 结构的输入/输出
- **Imcrop** 修剪图像
- **Imresize** 调整图像大小
- **Imrotate** 旋转图像
- **Imtransform** 对图像应用几何变换
- **Intline** 整数坐标线绘制算法
- **Makersampler** 创建重取样器结构
- **Maketform** 创建几何变换结构 (TFORM)
- **Pixeldup (DIPUM)** 在两个方向上复制图像的像素
- **Tformarray** 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

## Checkerboard 创建棋盘格图像

- $C = \text{checkerboard}(NP, M, N)$
- $NP$  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



## Maketform、Tformfwd、Vstformfwd



## 图像配准

- Cpstruct2pairs 将CPSTRUCT转换为有效的控制点对
- Cp2tform 由控制点对推断几何变换
- Cpcorr 使用互相关校准控制点位置C
- pselect 控制点选择工具
- Normxcorr2 归一化二维互相关

## 像素值及统计

- Corr2 计算二维相关系数
- Covmatrix (DIPUM) 计算向量族的协方差矩阵
- Imcontour 创建图像数据的轮廓线
- Imhist 显示图像数据的直方图
- Impixel 确定像素的彩色点
- Improfile 计算沿着线段的像素值横截面
- Mean2 计算矩阵元素的均值
- Pixval 显示关于像素的信息
- Regionprops 测量图像区域的属性
- Statmoments (DIPUM) 计算一幅图像直方图的统计中心距
- Std2 计算矩阵元素的标准偏差

## 图像分析（包括分割、描述和识别）

- Bayesgauss (DIPUM) 高斯模式的贝叶斯分类器
- Bound2eight (DIPUM) 将4连接边界转换为8连接边界
- Bound2four (DIPUM) 将8连接边界转换为4连接边界
- Bwboundaries 追踪区域边界
- Bwtraceboundary 追踪单个边界
- Bound2im (DIPUM) 将边界转换为图像B
- Boundaries (DIPUM) 追踪区域边界
- Bsubamp (DIPUM) 对边界二次取样
- Colorgrad (DIPUM) 计算一幅RGB图像的向量梯度
- Colorseq (DIPUM) 分割一幅彩色图像
- Connectpoly (DIPUM) 连接多边形的顶点
- Diameter (DIPUM) 测量图像区域的直径
- Edge (DIPUM) 在一幅亮度图像中寻找边缘
- Fchcode (DIPUM) 计算边界的freeman链码
- Frdescp (DIPUM) 计算傅里叶描绘子
- Graythresh 使用Ostu方法计算图像的全局阈值
- Hough (DIPUM) Hough变换
- Houghlines (DIPUM) 基于Hough变换提取线段
- Houghpeaks (DIPUM) 在Hough变换中检测峰值
- Houghpixels (DIPUM) 计算属于Hough变换bin的图像像素
- Ifrdescp (DIPUM) 计算逆傅里叶描绘子
- Imstack2vectors (DIPUM) 从图像堆栈提取向量
- Invmoments (DIPUM) 计算图像不变矩
- Mahalanobis (DIPUM) 计算Mahalanobis距离
- Minperpoly (DIPUM) 计算最小周长多边形
- Polyangles (DIPUM) 计算多边形内角
- Princomp (DIPUM) 得到主分量向量和相关量
- Qtdecomp 执行二叉树分解Qtgetblk得到二叉树分解中的块值
- Qtsetblk 在二叉树中设置块值
- Randvertex (DIPUM) 随机置换多边形顶点
- Regiongrow (DIPUM) 由区域生长来执行分割
- Signature (DIPUM) 计算边界的标记
- Specxture (DIPUM) 计算图像的谱纹理
- Splitmerge (DIPUM) 使用分离-合并算法分割图像
- Statxture (DIPUM) 计算图像中纹理的统计度量
- Strsimilarity (DIPUM) 两个串间的相似性度量
- X2majoraxis (DIPUM) 以区域的主轴排列坐标x

## Edge (DIPUM) 在一幅亮度图像中寻找边缘

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

Edge Detector	Description
Sobel	Finds edges using the Sobel approximation to the derivatives in Fig. 11.5(b)
Prewitt	Finds edges using the Prewitt approximation to the derivatives in Fig. 11.5(c).
Roberts	Finds edges using the Roberts approximation to the derivatives in Fig. 11.5(d).
Laplacian of a Gaussian (LoG)	Finds edges by looking for zero crossings after filtering $f(x, y)$ with a Laplacian of a Gaussian filter.
Zero crossings	Finds edges by looking for zero crossings after filtering $f(x, y)$ with a specified filter.
Canny	Finds edges by looking for local maxima of the gradient of $f(x, y)$ . The gradient is calculated using the derivative of a Gaussian filter. The method uses two thresholds to detect strong and weak edges, and includes the weak edges in the output only if they are connected to strong edges. Therefore, this method is more likely to detect true weak edges.

## 图像压缩

- `Compare` (DIPUM) 计算和显示两个矩阵间的误差
- `Entropy` (DIPUM) 计算矩阵的熵的一阶估计
- `Huff2mat` (DIPUM) 解码霍夫曼编码矩阵
- `Huffman` (DIPUM) 为符号源建立一个变长霍夫曼码
- `Im2jpeg` (DIPUM) 使用JPEG近似压缩一幅图像
- `Im2jpeg2k` (DIPUM) 使用JPEG2000近似压缩一幅图像
- `Imratio` (DIPUM) 计算两幅图像或变量中的比特率
- `Jpeg2im` (DIPUM) 解码IM2JPEG压缩的图像
- `Jpeg2k2im` (DIPUM) 解码IM2JPEG2K压缩的图像
- `Lpc2mat` (DIPUM) 解压缩一维有损预测编码矩阵
- `Mat2huff` (DIPUM) 霍夫曼编码矩阵
- `Mat2lpc` (DIPUM) 使用一维有损预测编码矩阵
- `Quantize` (DIPUM) 量化UINT8类矩阵的元素

## 图像增强

- `Adapthisteq` 自适应直方图量化
- `Decorrstretch` 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

## 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

## 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

## 线性二维滤波器设计

- Freqspace 确定二维频率响应间隔
- Freqz2 计算二维频率响应
- Fsamp2 使用频率取样设计二维FIR滤波器
- Ftrans2 使用频率变换设计二维FIR滤波器
- Fwind1 使用一维窗法设计二维滤波器
- Fwind2 使用二维窗法设计二维滤波器
- Hpfilt (DIPUM) 计算频率域高通滤波器
- Lpfilt (DIPUM) 计算频率域低通滤波器

## 图像去模糊 (复原)

- Deconvblind 使用盲去卷积去模糊图像
- Deconvlucy 使用Lucy-Richardson方法去模糊
- Deconvreg 使用规则化滤波器去模糊
- Deconvwnr 使用维纳滤波器去模糊
- Edgetaper 使用点扩散函数锐化边缘
- Otf2psf 光传递函数到点扩散函数
- Pst2otf 点扩散函数到光传递函数

## 图像变换

- Dct2 二维离散余弦变换
- Dctmtx 离散余弦变换矩阵
- Fan2para 将扇形束投影变换为并行射束
- Fanbeam 计算扇形射束变换
- Fft2 二维快速傅里叶变换
- Fftn N维快速傅里叶变换
- Fftshift 颠倒FFT输出的象限
- Idct2 二维逆离散余弦变换
- Ifanbeam 计算扇形射束逆变换
- Ifft2 二维快速傅里叶逆变换
- Ifftn N维快速傅里叶逆变换
- Iradon 计算逆Radon变换
- Para2fan 将并行射束投影变换为扇形射束
- Phantom 生成头部仿真模型的图像
- Radon 计算Radon变换

## 小波

- Wave2gray (DIPUM) 显示小波分解系数
- Waveback (DIPUM) 执行多灰度级二维快速小波逆变换
- Wavecopy (DIPUM) 存取小波分解结构的系数
- Wavecut (DIPUM) 在小波分解结构中置零系数
- Wavefast (DIPUM) 执行多灰度级二维快速小波变换
- Wavefilter (DIPUM) 构造小波分解和重构滤波器
- Wavepaste (DIPUM) 在小波分解结构中放置系数
- Wavework (DIPUM) 编辑小波分解结构
- Wavezero (DIPUM) 将小波细节系数设置为零

## 领域和块处理

- Bestblk 为块处理选择块大小
- Blkproc 为图像实现不同的块处理
- Col2im 将矩阵列重排为块
- Colfilt 按列邻域操作
- Im2col 将图像块重排为列
- Nlfilt 执行一般的滑动邻域操作

## 形态学操作（亮度和二值图像）

- Conndef 默认连通性
- Imbothat 执行底帽滤波
- Imclearborder 抑制与图像边框相连的亮结构
- Imclose 闭运算图像
- Imdilate 膨胀图像
- Imerosde 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
- Imreconstruct 形态学重构
- Imregionalmax 局部最大区域
- Imregionalmin 局部最小区域
- Imtophat 执行顶帽滤波
- Watershed 分水岭变换

## 形态学操作（二值图像）

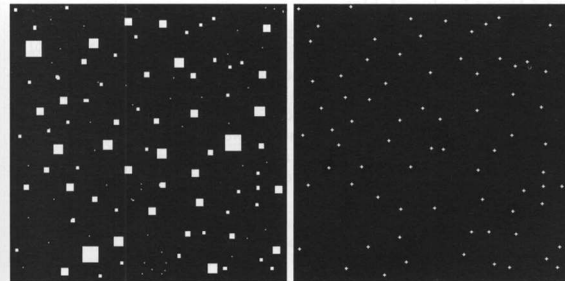
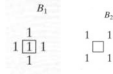
- Applylut 使用查表法执行邻域操作
- Bwarea 计算二值图像中的对象面积
- Bwareaopen 打开二值区域（删除小对象）
- Bwdist 计算二值图像的距离变换
- Bweuler 计算二值图像的欧拉数
- Bwhitmiss 二值击中不中操作
- Bwlabel 在二维图像中标记连接分量
- Bwlabeln 在N维二值图像中标记连接分量
- Bwmorph 对二值图像执行形态学操作
- Bwpack 打包二值图像
- Bwperim 确定二值图像中的对象的周长
- Bwselect 选择二值图像中的对象
- Bwulterode 最终腐蚀
- Bwunpack 解包二值图像
- Endpoints (DIPUM) 计算二值图像的端点
- Makelut 构建applylut使用的查找表



## Bwhitmiss

## 二值击中击不中

`g= bwhitmiss (f, B1, B2)`



**FIGURE 10.13**  
(a) Original image.  
(b) Result of applying the hit-or-miss transformation (the dots shown were enlarged to facilitate viewing).

## 结构元素 (STREL) 的创建和操作

- `Getheight` 得到strel的高度
- `Getneighbors` 得到strel邻域的偏移位置和高度
- `Getnhood` 得到strel邻域
- `Getsequence` 得到分解的strel序列
- `Isflat` 对平坦的strel返回值
- `Reflect` 以其中心反射strel
- `Strel` 创建形态学结构元素
- `Translate` 变换strel

## 基于区域的处理

- `Histroi (DIPUM)` 计算图像中的ROI的直方图
- `Poly2mask` 将ROI多边形转换为掩膜
- `Roicolor` 基于颜色选择ROI
- `Roifill` 在任意区域内平稳地内插
- `Roifilt2` 对ROI进行滤波
- `Roipoly` 选择多边形ROI

## 彩色映射处理

- `Brighten` 加亮或加暗彩色映射
- `Cmpermute` 在彩色映射中重排颜色
- `Cmunique` 寻找唯一的彩色映射颜色和相应的图像
- `Colormap` 设置或得到彩色查找表
- `Imapprox` 以很少的颜色近似被索引的图像
- `Rgbplot` 绘制RGB彩色映射分量

## 数组操作

- Circshift      循环地移位数组
- Dftuv (DIPUM)    计算网格数组
- Padarray      填充数组
- Paddedsize (DIPUM) 计算用于FFT的最小填充尺寸

## 图像类型和类型转换

- |                                  |                                      |
|----------------------------------|--------------------------------------|
| • Changeclass    改变一幅图像的类        | • Applyform    应用独立于设备的彩色空间变换        |
| • Dither    使用抖动转换图像             | • Hsv2rgb    将HSV值转换为RGB彩色空间         |
| • Gray2ind    将亮度图像转换为索引图像       | • Iccread    读ICC彩色配置文件              |
| • Grayslice    通过阈值处理从亮度图像创建索引图像 | • Lab2double    将L*a*b*彩色值转换为double类 |
| • Im2bw    通过阈值处理将图像转换为二值图像      | • Lab2uint16    将L*a*b*彩色值转换为uint16类 |
| • Im2double    将图像数组转换为双精度       | • Lab2uint8    将L*a*b*彩色值转换为uint8类   |
| • Im2java    将图像转换为Java图像        | • Makecform    创建独立于设备的彩色空间变换结构      |
| • Im2java2d    将图像转换为Java缓存的图像对象 | • Ntsc2rgb    将NTSC值转换为RGB彩色空间       |
| • Im2uint8    将图像数组转换为8比特无符号整数   | • Rgb2hsv    将RGB值转换为HSV彩色空间         |
| • Im2uint16    将图像数组转换为16比特无符号整数 | • Rgb2ntsc    将RGB值转换为NTSC彩色空间       |
| • Ind2gray    将索引图像转换为亮度图像       | • Rgb2ycbcr    将RGB值转换为YCBCR彩色空间     |
| • Ind2rgb    将索引图像转换为RGB图像       | • Ycbcr2rgb    将YCBCR值转换为RGB彩色空间     |
| • Label2rgb    将标记矩阵转换为RGB图像     | • Rgb2hsi (DIPUM)    将RGB值转换为HSI彩色空间 |
| • Mat2gray    将矩阵转换为亮度图像         | • Hsi2rgb (DIPUM)    将HSI值转换为RGB彩色空间 |
| • Rgb2gray    将RGB图像或彩色映射转换为灰度图像 | • Whitepoint    返回标准照明的XYZ值          |
| • Rgb2ind    将RGB图像转换为索引图像       | • Xyz2double    将XYZ彩色值转换为double类    |
|                                  | • Xyz2uint16    将XYZ彩色值转换为uint16类    |

## 彩色空间转换

## 其他函数

- Conwaylaws (DIPUM) 对单个像素应用
- Conway的遗传定律
- Manualhist (DIPUM) 交互地生成2模式直方图
- Twomodegauss (DIPUM) 生成一个2模式高斯函数
- Uintlut      基于查找表计算新数组值

## 工具箱参数

- Iptgetpref    获得图像处理工具箱参数的值
- Iptsetpref    设置图像处理工具箱参数的值