图像操作

**Demo0**

%通过仿射水平剪切图像

I=imread('rice.png');

imshow(I);

tform=affine2d([1 0 0;0.5 1 0;0 0 1]);

J=imwarp(I,tform);

figure,imshow(J);

**demo-0.5**

%通过仿射旋转图像

I=imread('rice.png');

imshow(I);

theta=10;

tform1=affine2d([cosd(theta) -sind(theta) 0;...

sind(theta) cosd(theta) 0;...

0 0 1]);

JJ=imwarp(I,tform1);

figure,imshow(JJ);

**demo1**

I=imread('rice.jpg');

I2=colfilt(I,[7 7],'sliding','mean');

imshow(I)

figure,imshow(I2,[]);

**demo2**

I=imread('saturn.png');

I1=rgb2gray(I);

imshow(I1);

I2=colfilt(I1,[8 8],'distinct',@(x)ones(64,1)\*mean(x));

figure,imshow(I2,[]);

**demo3-mosaic- blockproc**

%Set the pixels in each 32-by-32 block to the standard deviation of the elements in that block:

fun = @(block\_struct)std2(block\_struct.data) \* ones(size(block\_struct.data));

I=imread('moon.tif');

I2 = blockproc(I,[16 16],fun);

figure;

imshow(I);

figure;

imshow(I2,[]);

**demo4-resize-blockproc**

%Generate an image thumbnail:

fun = @(block\_struct) imresize(block\_struct.data,0.15);

I = imread('pears.png');

I2 = blockproc(I,[100 100],fun);

figure;

imshow(I);

figure;

imshow(I2);

I3=imresize(I,0.15);%compare with imresize directly

figure,imshow(I3);

**demo5-exchangeRGB-blockproc**

%Switch the red and green bands of an RGB image

I = imread('peppers.png');

fun = @(block\_struct) block\_struct.data(:,:,[2 1 3]);

I2=blockproc(I,[100 100],fun);

figure;

imshow(I);

figure;

imshow(I2);

**demo5.5 myrgb2gray -blockproc**

I=imread('flamingos.jpg');

fun=@(block\_struct)(block\_struct.data(:,:,1)\*0.299+...

block\_struct.data(:,:,2)\*0.587+...

block\_struct.data(:,:,3)\*0.114);

[M,N,P]=size(I);

siz=bestblk([M,N],50);

B=blockproc(I,siz,fun);

imshow(B);

**demo6-im2col**

A=pascal(6)

B=im2col(A,[2 2],'distinct')

C=im2col(A,[2 2],'sliding')

**Demo7-col2im**

A=pascal(6)

B=im2col(A,[2 2],'distinct')

C=im2col(A,[2 2],'sliding')

A1=col2im(B,[2 2],[6 6],'distinct')

A2=col2im(C(1,:),[2 2],[6 6],'sliding') %the first parameter must be row vector

**Demo8-roiploy**

I = imread('eight.tif');

c = [222 272 300 270 221 194];

r = [21 21 75 121 121 75];

BW = roipoly(I,c,r);

figure, imshow(I)

figure, imshow(BW)

**Demo9-roicolor**

load clown

BW = roicolor(X,10,20);

imshow(X,map)

figure,imshow(BW)

**Demo10-roifilt2**

I = imread('eight.tif');

c = [222 272 300 270 221 194];

r = [21 21 75 121 121 75];

BW = roipoly(I,c,r);

H = fspecial('unsharp');

J = roifilt2(H,I,BW);

figure, imshow(I), figure, imshow(J)

**demo11-regionfill**

%Read grayscale image into the workspace.

I = imread('eight.tif');

x = [222 272 300 270 221 194];

y = [21 21 75 121 121 75];

%Fill the polygon, using the regionfill function.

J = regionfill(I,x,y);

%Display the original image and the filled image side-by-side.

figure

subplot(1,2,1)

imshow(I)

title('Original image')

subplot(1,2,2)

imshow(J)

title('Image with one less coin')

**demo12-高光谱图像的显示**

%显示一条近红外光谱

for i=1:100

plot(squeeze(salinas(100,i,:)));

hold on

end

显示一个平面

imshow(salinas(:,:,101),[]);