Lecture 19

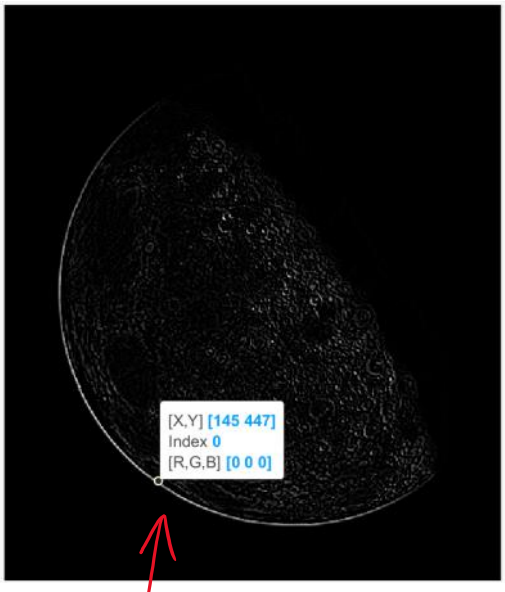
Image Sharpening Filters (cont'd)

```
>> I = imread('Fig0338(a)(blurry_moon).tif');
>> imshow(I)
>> h = fspecial('laplacian', 0)

h =

    0    1    0
    1   -4    1
    0    1    0

>> J1 = imfilter(I, h, 'replicate');
>> figure; imshow(J1, [])
```

(446, 145)

(447, 144) (447, 145) (447, 146)

(448, 145)




```
>> I(446,145)+ I(448,145)+I(447,144)+I(447,146)
ans =
  uint8
   255

>> I(446,145)+ I(448,145)+I(447,144)+I(447,146)-4*I(447,145)
ans =
  uint8
   0

>> double(I(446,145))+
double(I(448,145))+double(I(447,144))+double(I(447,146))-4*double(I(447,145))
ans =
   -26
```

Truncation issue!

Need to convert to double type!

im2double

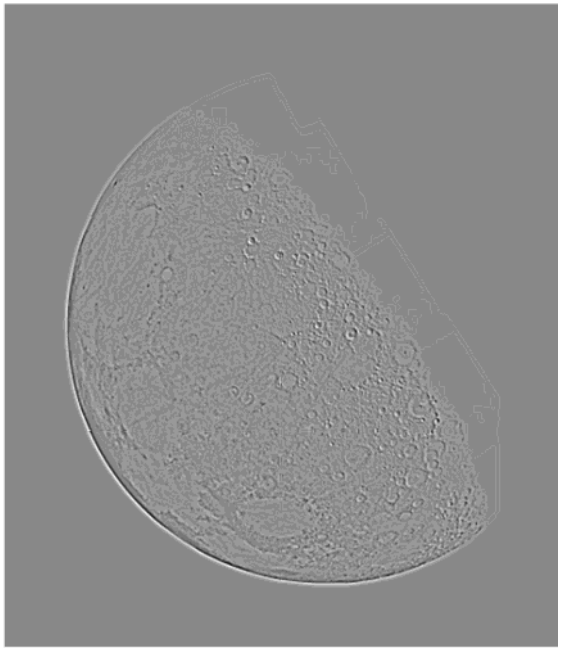
Convert image to double precision

I2 = im2double(I) converts the image I to double precision. I can be a grayscale intensity image, a truecolor image, or a binary image. im2double rescales the output from integer data types to the range **[0, 1]**.

```
>> ID = im2double(I);
>> max(I(:))
ans =
  uint8
   254

>> max(ID(:))
ans =
    0.9961
```

```
>> 254/255

ans =

    0.9961
```

```
>> J2 = imfilter(ID, h, 'replicate');
>> figure; imshow(J2, [])
```



```
>> J2(447,145)
ans =
   -0.1020

>> -26/255
ans =
   -0.1020
```

- $g(x,y) = f(x,y) - [\nabla^2 f(x,y)]$, if the center coefficient is negative.

```
>> G = ID - J2;
>> figure; montage({ID, G}, 'Size', [1 2]);
```

The Gradient for Image Sharpening

$$\nabla f \equiv \text{grad}(f) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix}$$

The magnitude of the vector is $M(x, y) = \text{mag}(\nabla f) = \sqrt{g_x^2 + g_y^2}$ called the *gradient image*, or simply as *gradient*.

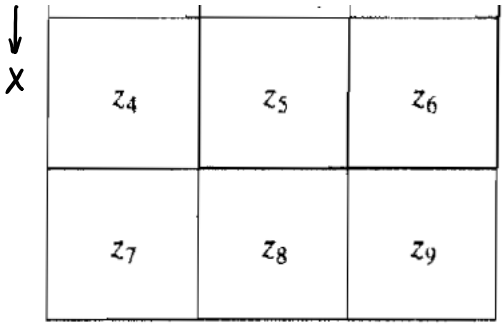
Sobel Operators

→ y

z_1	z_2	z_3

↓ x

$$g_x = \frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$



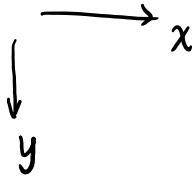
$$g_x = \frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

$$g_y = \frac{\partial f}{\partial y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

```
>> hy = -fspecial('sobel')

hy =

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

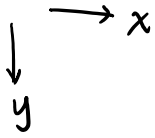
−1	−2	−1
0	0	0
1	2	1

Matlab



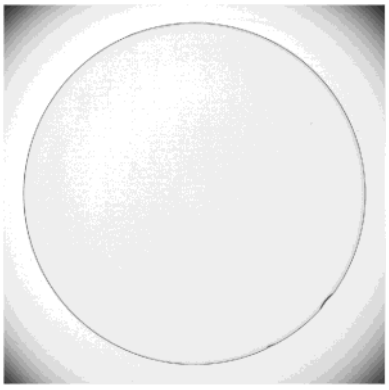
```
>> hx = hy'

hx =

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

−1	0	1
−2	0	2
−1	0	1



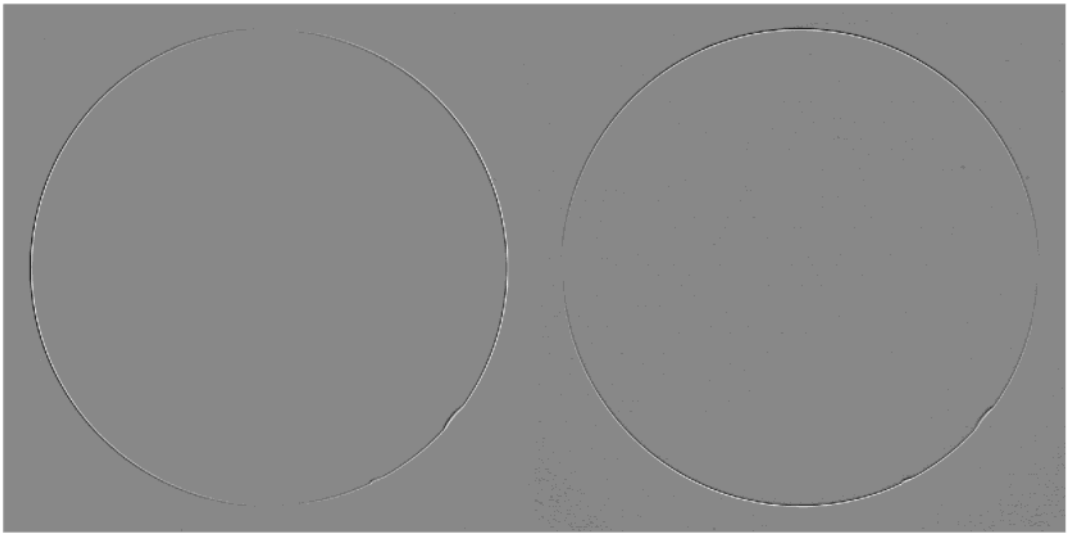
imgradientxy

Find directional gradients of 2-D images



```
>> I = imread('Fig0342(a)(contact_lens_original).tif');

>> [Gx, Gy] = imgradientxy(I, 'sobel');
>> figure; imshowpair(Gx, Gy, 'montage')
```

Gx Gy



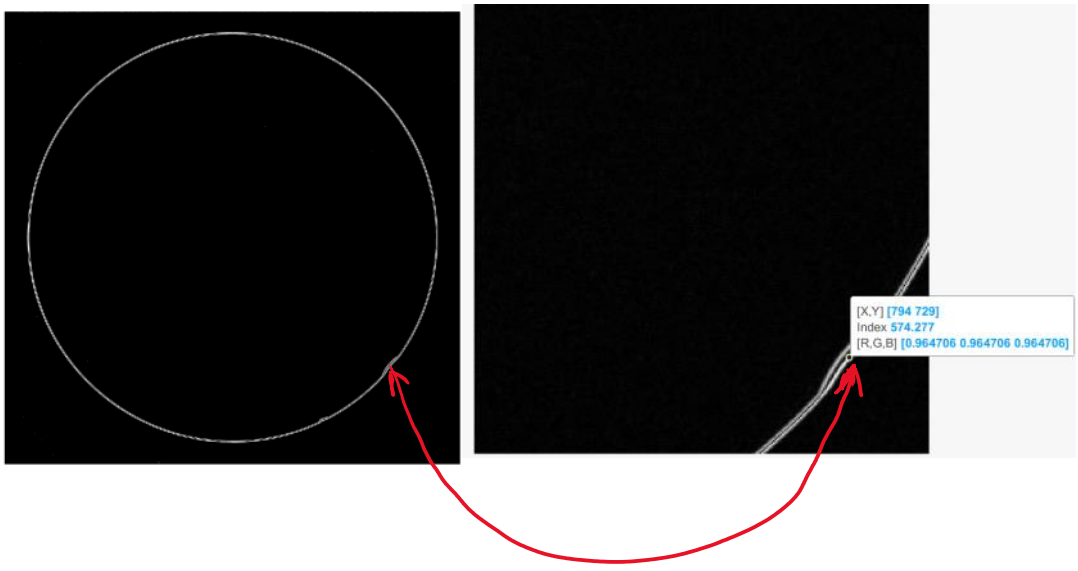
```
>> sqrt(Gx(729, 794)^2 +
Gy(729, 794)^2)

ans =

  574.2769
```

Imgradient

Find gradient **magnitude** and direction of 2-D image

```
sobelGradient = imgradient(I);

 imshow(sobelGradient, [ ]);
```

```
>> sobelGradient(729, 794)

ans =

  574.2769
```




```
>> [Gmag, Gdir] = imgradient(I);
>> figure; imagesc(Gdir); colorbar
```

Gdir contains angles in degrees within the range [-180, 180] measured counterclockwise from the positive *x*-axis.

```
>> Gdir(729, 794)
ans =
  -39.2052
>
```