APPLICATIONS OF MATLAB IN ENGINEERING

Yan-Fu Kuo
Dept. of Bio-industrial Mechatronics Engineering
National Taiwan University

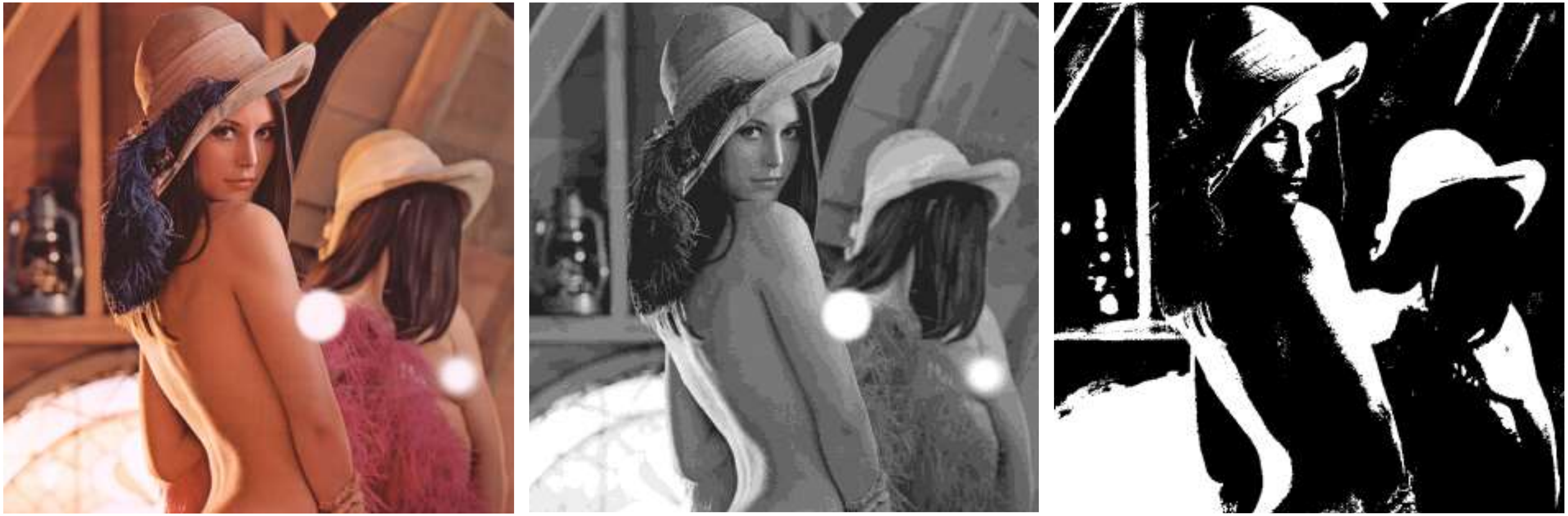
Fall 2015

Today:

- Introduction to digital image
- Read and show images
- Image arithmetic

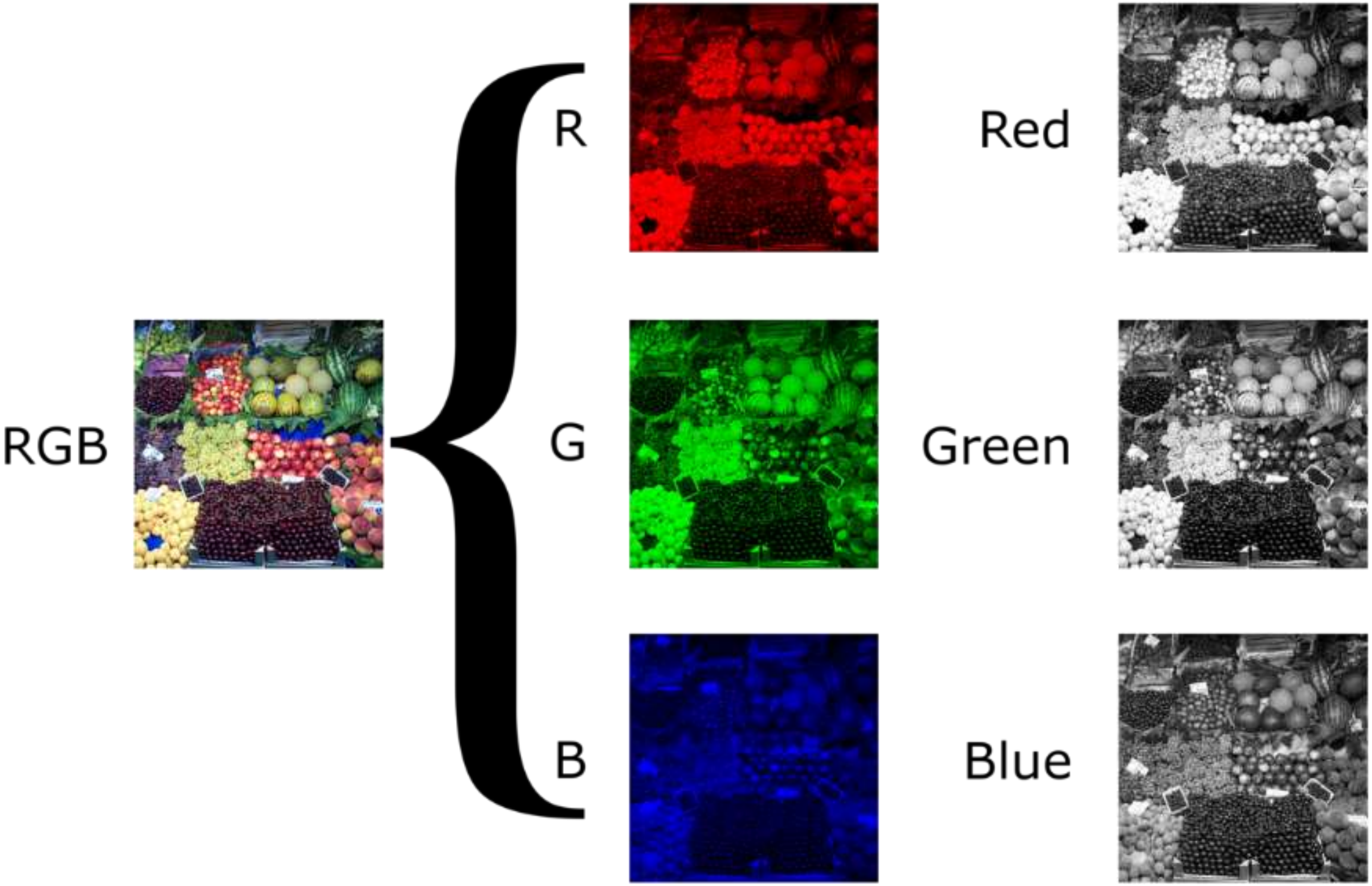
Digital Image and Its Acquisition

Types of Digital Image



- **Binary**: Each pixel is just black or white
- **Grayscale**: Each pixel is a shade of gray, normally from 0 (black) to 255 (white)
- **True color** or **RGB**: Each pixel has a particular color described by the amount of red, green and blue in it

Typical RGB Image



RGB

R

G

B

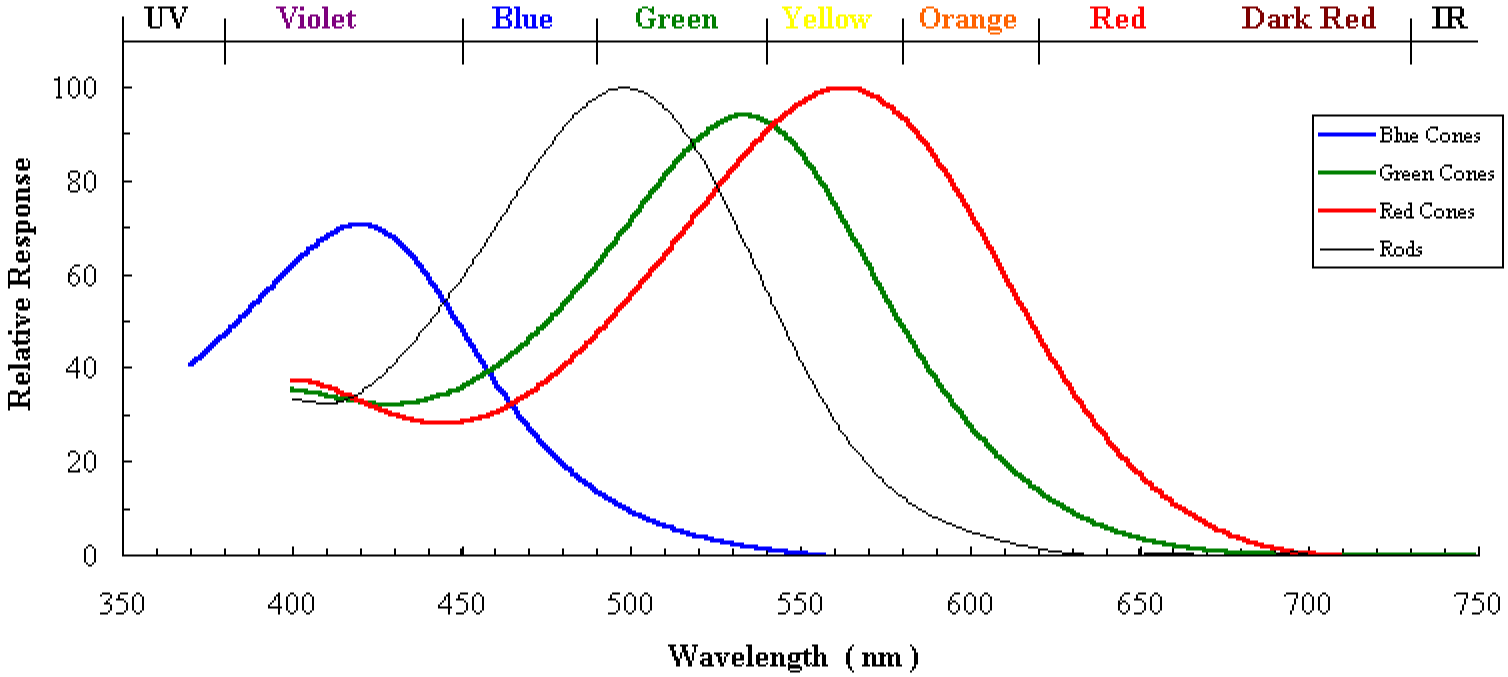
Red

Green

Blue

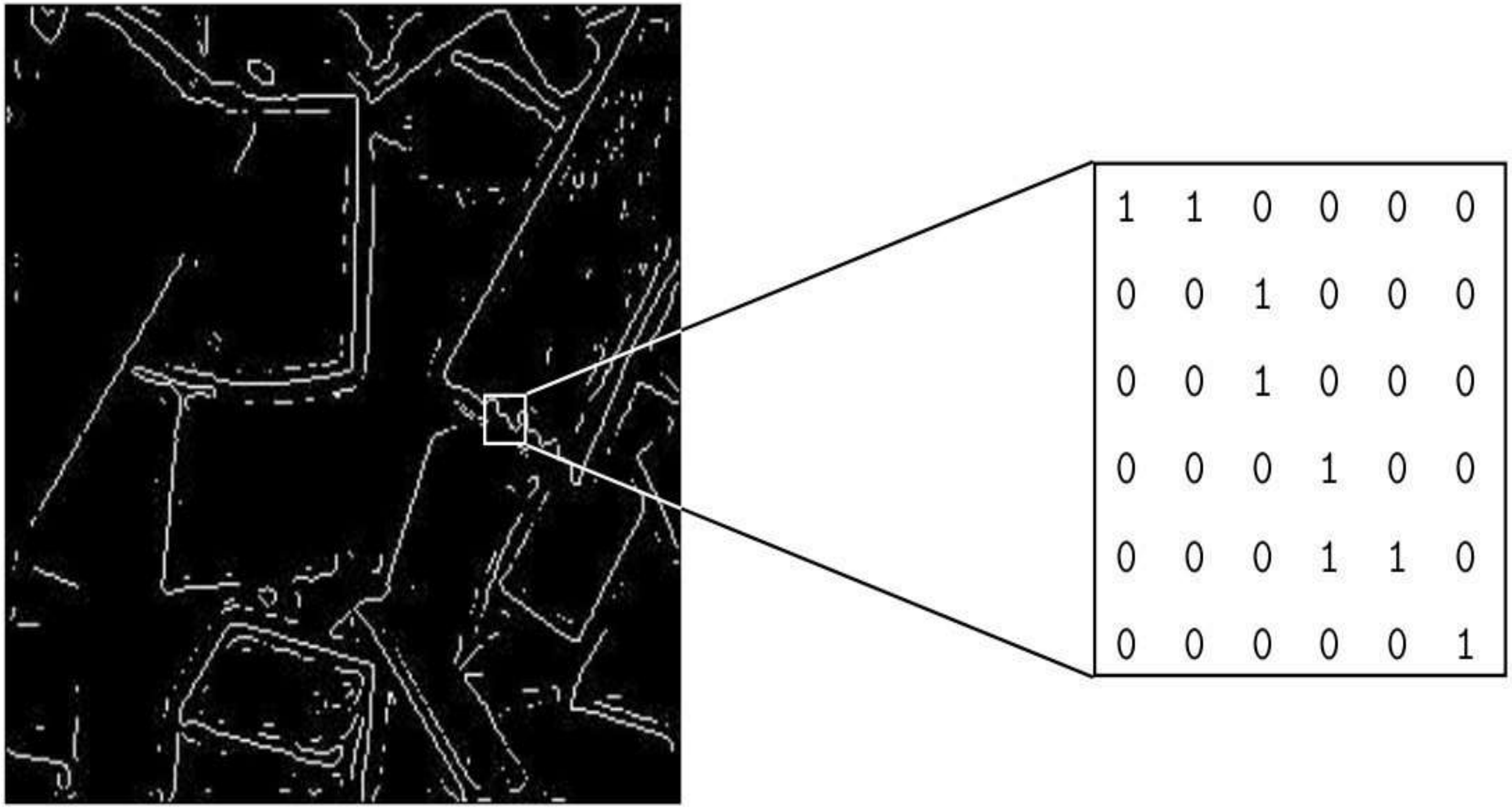
Why RGB?

- Three kinds of light-sensitive photoreceptor cells in the human eye (i.e., cone cells) respond most to red, green and blue



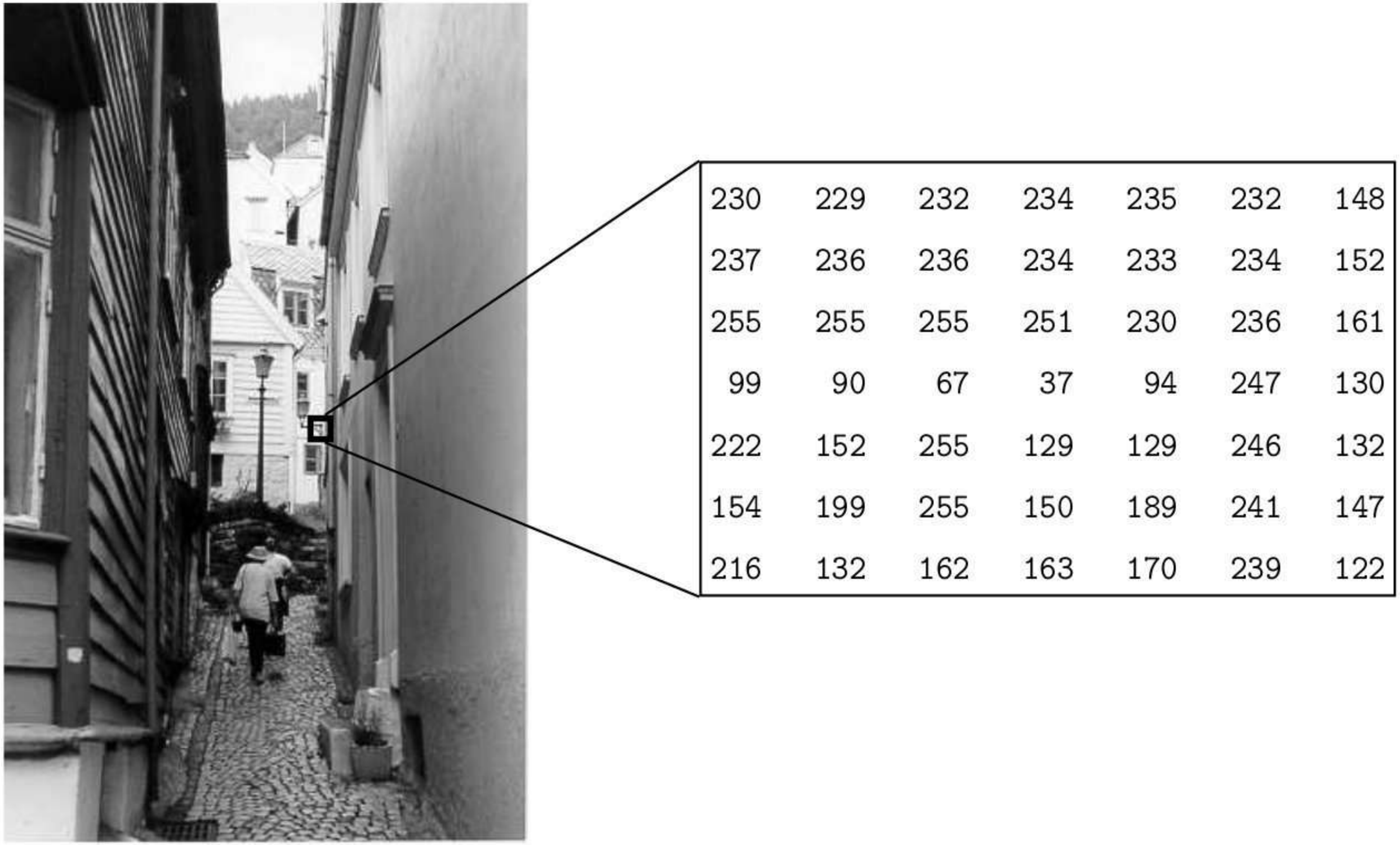
Elements of Images

Binary Image



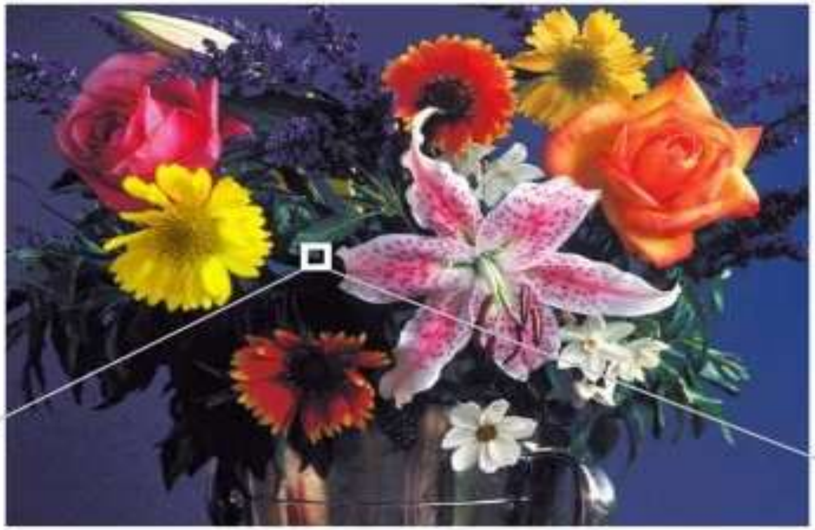
1	1	0	0	0	0
0	0	1	0	0	0
0	0	1	0	0	0
0	0	0	1	0	0
0	0	0	1	1	0
0	0	0	0	0	1

Greyscale Image



230	229	232	234	235	232	148
237	236	236	234	233	234	152
255	255	255	251	230	236	161
99	90	67	37	94	247	130
222	152	255	129	129	246	132
154	199	255	150	189	241	147
216	132	162	163	170	239	122

Color Image



49	55	56	57	52	53
58	60	60	58	55	57
58	58	54	53	55	56
83	78	72	69	68	69
88	91	91	84	83	82
69	76	83	78	76	75
61	69	73	78	76	76

Red

64	76	82	79	78	78
93	93	91	91	86	86
88	82	88	90	88	89
125	119	113	108	111	110
137	136	132	128	126	120
105	108	114	114	118	113
96	103	112	108	111	107

Green

66	80	77	80	87	77
81	93	96	99	86	85
83	83	91	94	92	88
135	128	126	112	107	106
141	129	129	117	115	101
95	99	109	108	112	109
84	93	107	101	105	102

Blue

Read and Show An Image

- Read an image: `imread()`
- Show an image: `imshow()`
- Example:

```
clear, close all
I = imread('pout.tif'); %read
imshow(I); %show
```

Image Variable in Workspace

```
 whos
```

```
Name        Size     Bytes   Class

   I     291x240     69840   uint8
```

- Image matrix:

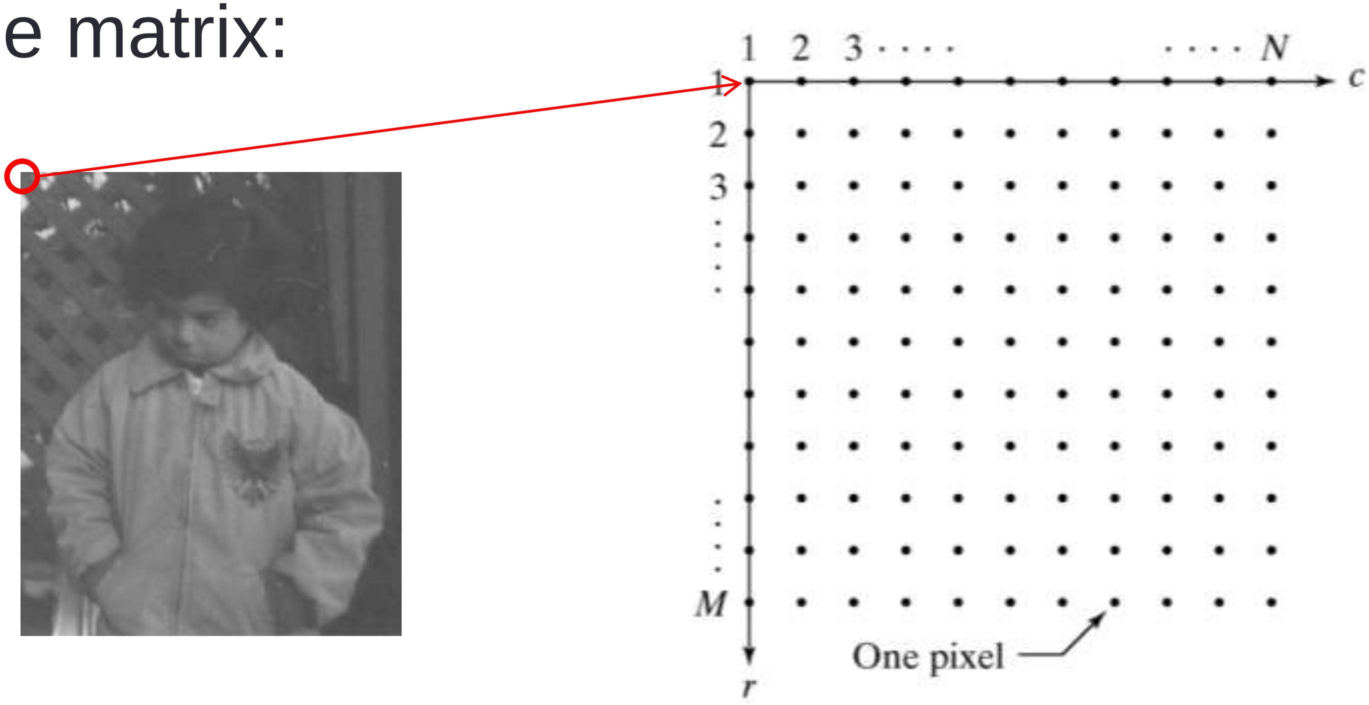
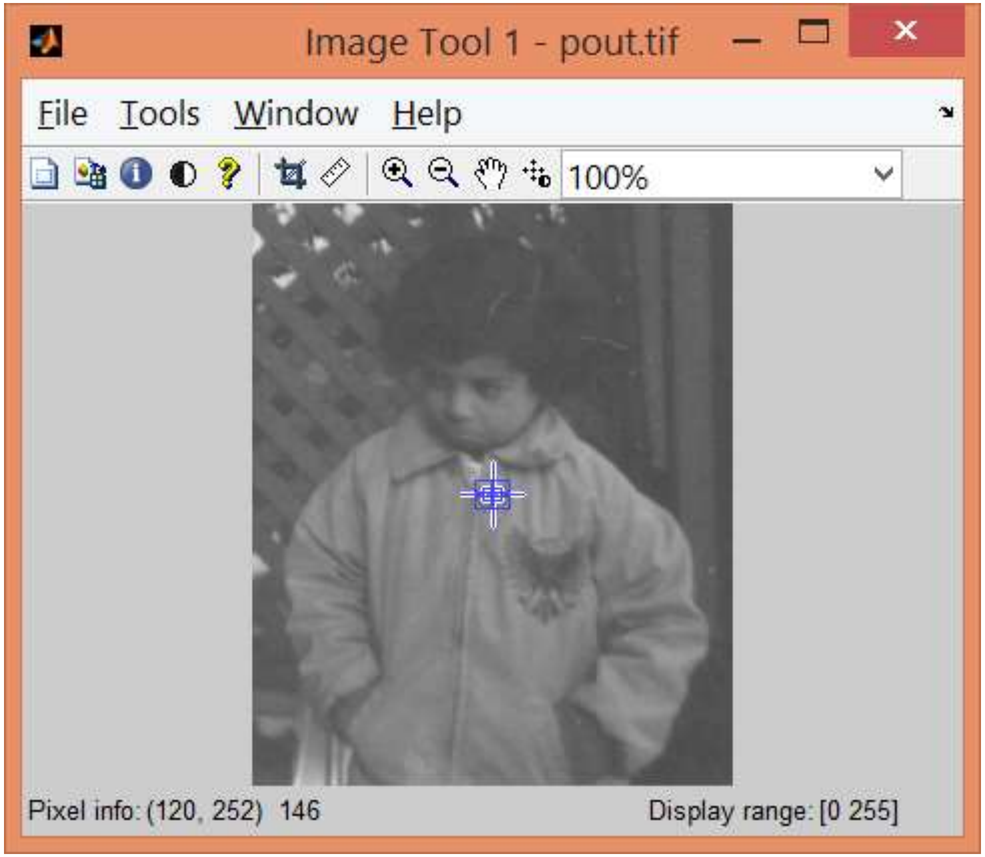


Image Info: `imageinfo('pout.tif')`

Filename	C:\Program Files\MATLAB\R2014a\toolbox\images\imdata\pout.tif
FileModDate	25-九月-2013 16:12:06
FileSize	69004
Format	tif
Width	240
Height	291
BitDepth	8
ColorType	grayscale
FormatSignature	[73 73 42 0]
ByteOrder	little-endian
BitsPerSample	8
SamplesPerPixel	1
RowsPerStrip	34
StripByteCounts	[1x9 double]
XResolution	72
YResolution	72
ResolutionUnit	Inch
MaxSampleValue	255
MinSampleValue	0

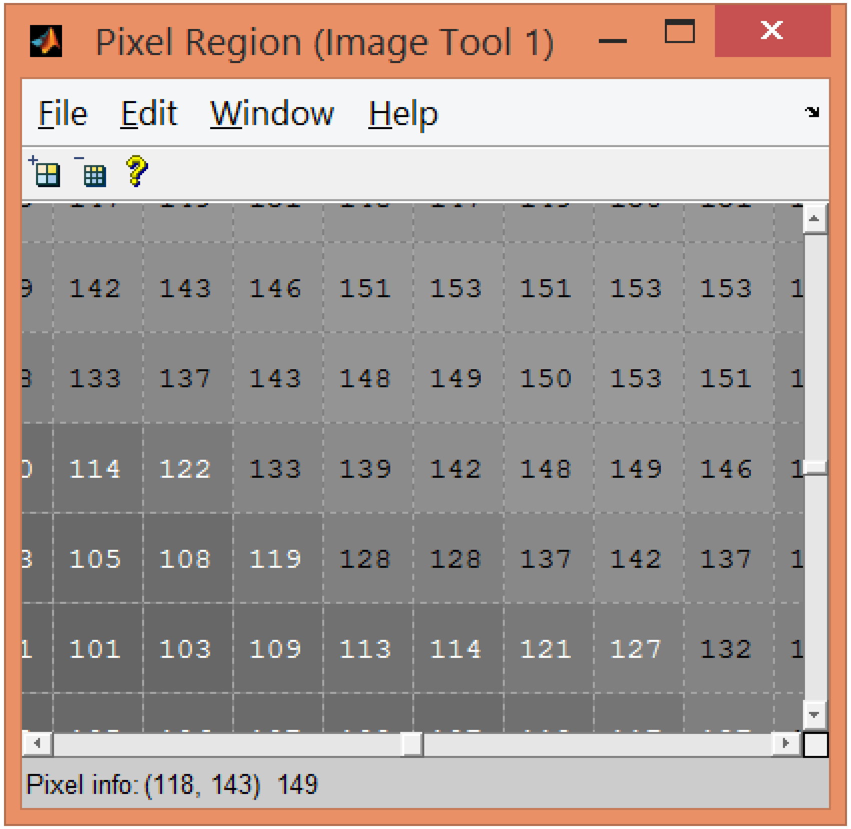
Image Viewer: `imtool('pout.tif')`

- Get pixel information in image viewer



Pixel info: (120, 252) 146

Display range: [0 255]



9	142	143	146	151	153	151	153	153	1
3	133	137	143	148	149	150	153	151	1
0	114	122	133	139	142	148	149	146	1
3	105	108	119	128	128	137	142	137	1
1	101	103	109	113	114	121	127	132	1

Pixel info: (118, 143) 149

Image Processing

- Any form of signal processing for which the input is an image

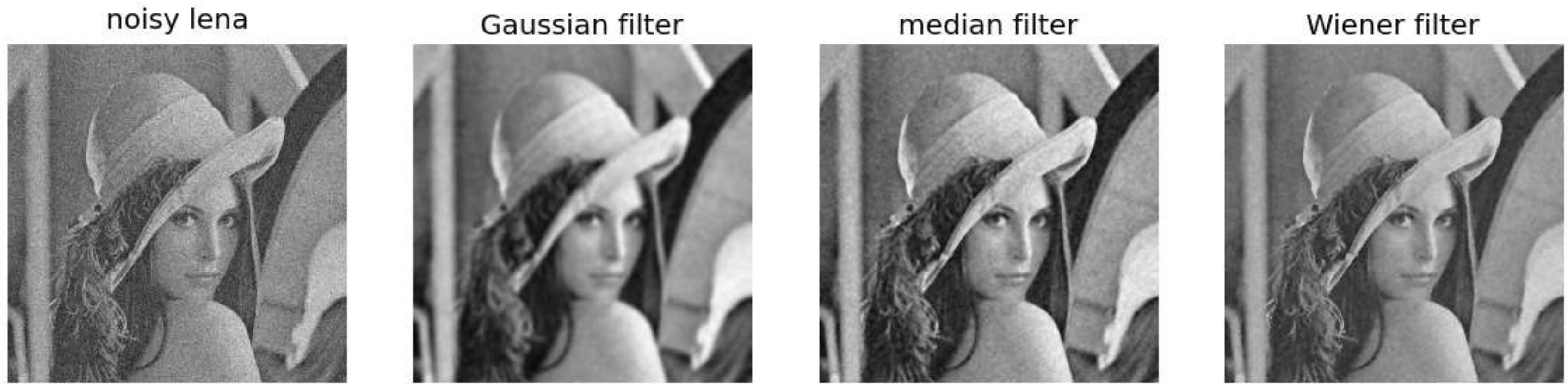
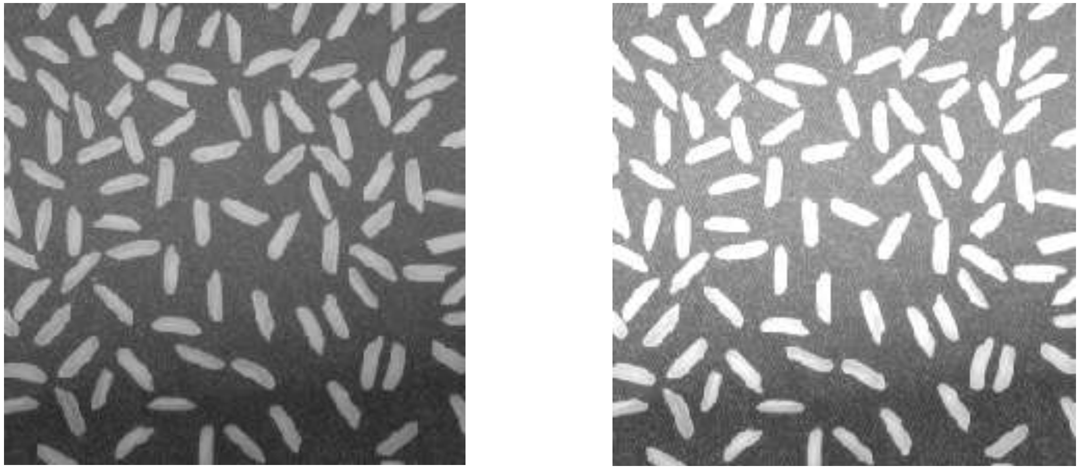


Image Arithmetic

imabsdiff	Absolute difference of two images
imadd	Add two images or add constant to image
imapplymatrix	Linear combination of color channels
imcomplement	Complement image
imdivide	Divide one image into another or divide image by constant
imlincomb	Linear combination of images
immultiply	Multiply two images or multiply image by constant
imsubtract	Subtract one image from another or subtract constant from image

Image Multiplication: `immultiply()`

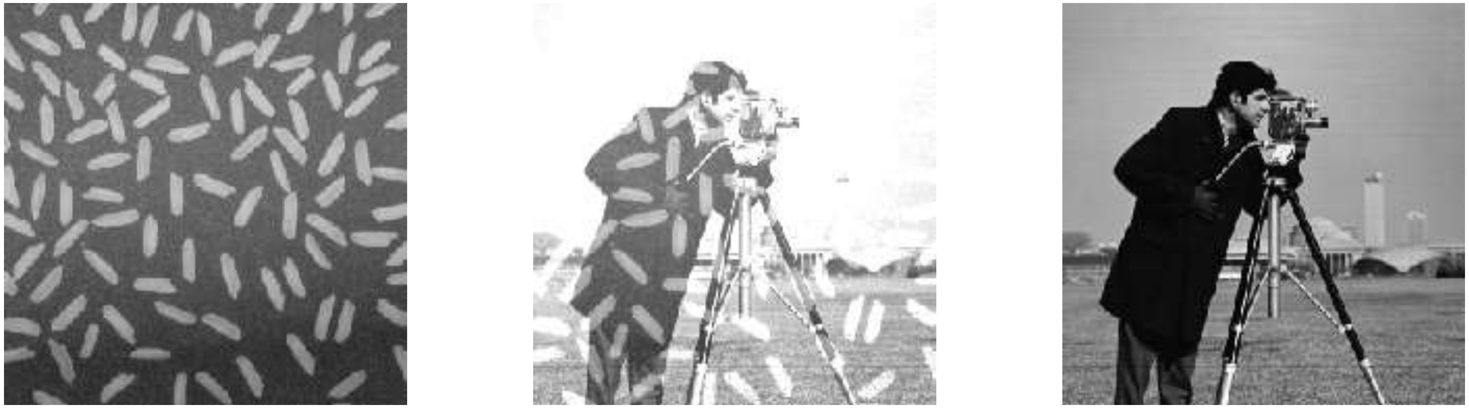
```
I=imread('rice.png');
subplot(1,2,1); imshow(I);
J=immultiply(I, 1.5);
subplot(1,2,2); imshow(J);
```



- How to reduce the brightness of the image?

Image Addition: `imadd()`

```
I=imread('rice.png');
J=imread('cameraman.tif'); K=imadd(I,J);
subplot(1,3,1); imshow(I);
subplot(1,3,2); imshow(K);
subplot(1,3,3); imshow(J);
```

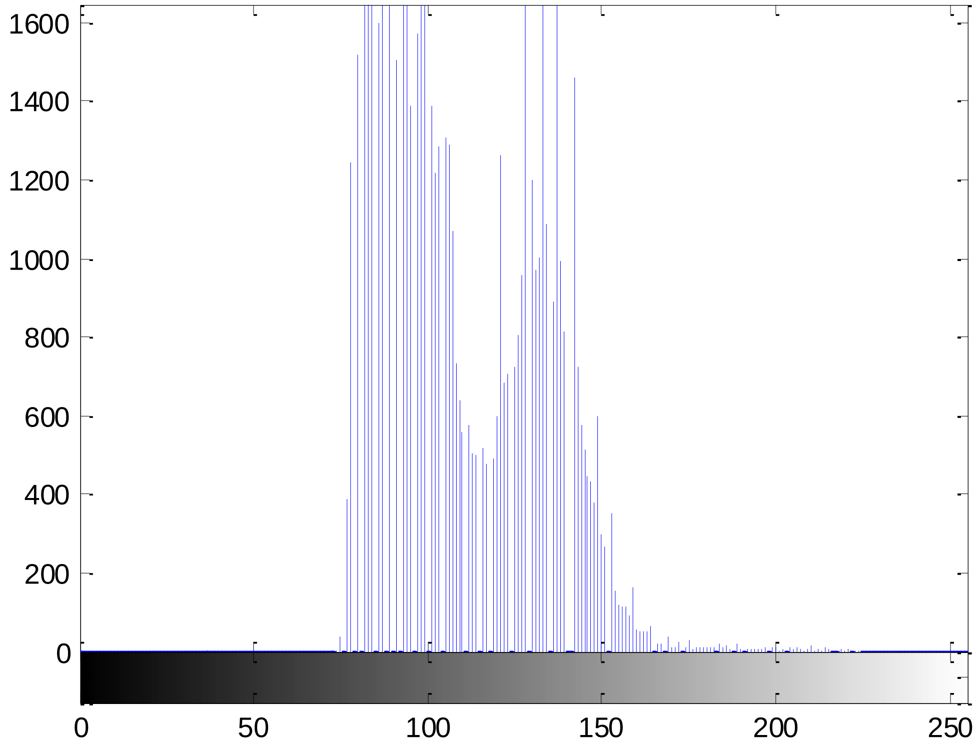


Practice

- Adjust the “brightness” and “contrast” of `rice.png` and display it on the screen

Image Histogram: `imhist()`

```
imhist(I)
```



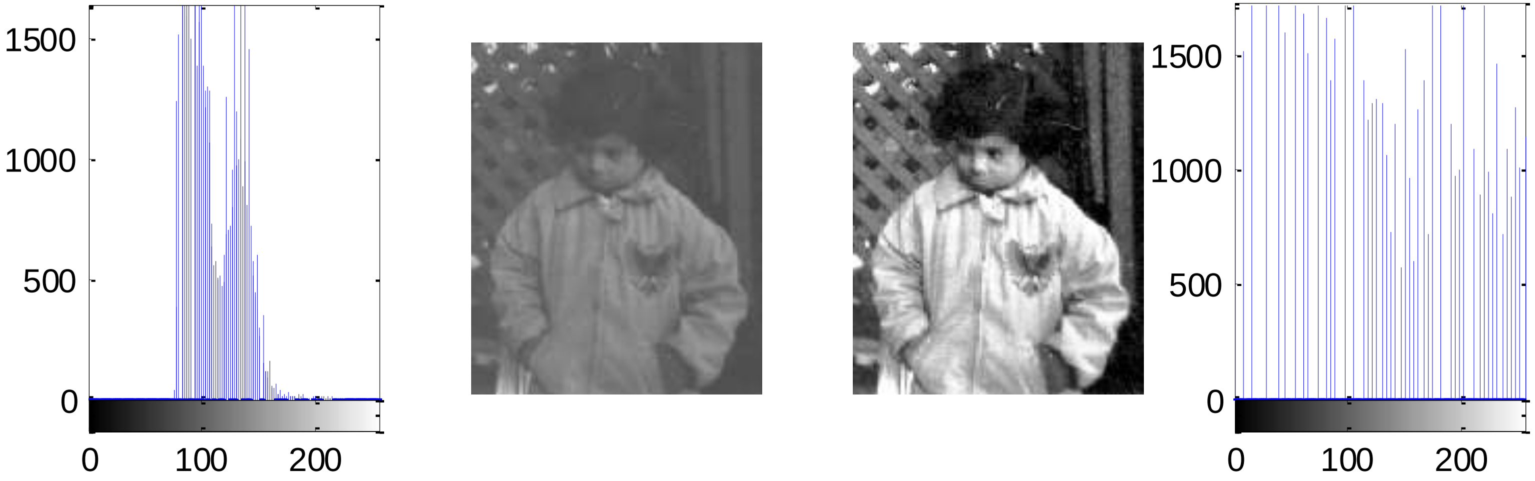
Practice

- Plot the histograms of the images before and after the “brightness” and “contrast” adjustment for `rice.png`

Histogram Equalization: `histeq()`

- Enhances the contrast of the image

```
I = imread('pout.tif'); I2 = histeq(I);
subplot(1,4,1); imhist(I);
subplot(1,4,2); imshow(I);
subplot(1,4,3); imshow(I2);
subplot(1,4,4); imhist(I2);
```

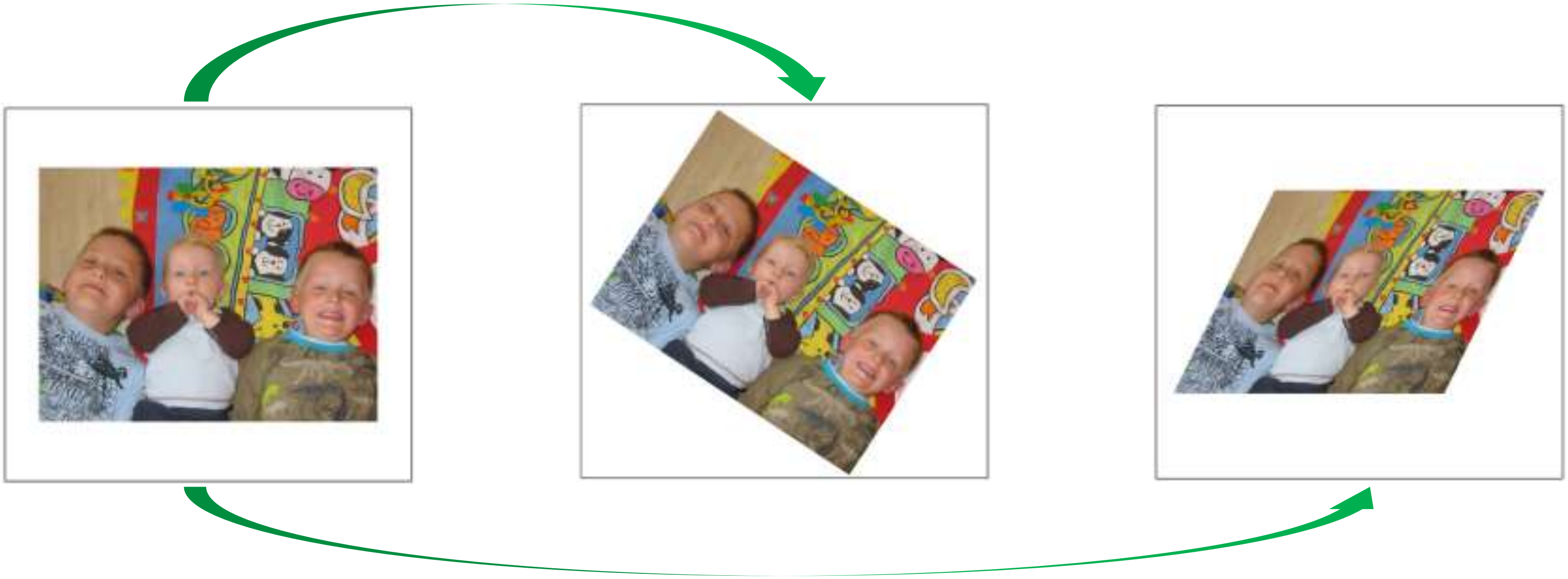


Practice

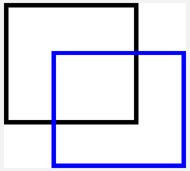
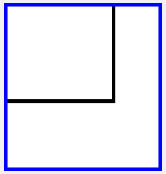
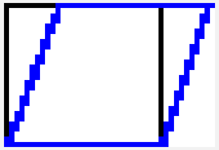
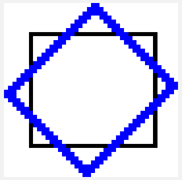
- Write your own equalization function, try it on `pout.tif`, and display it on the screen

Geometric Transformation

- Moving the coordinates (Not the gray-levels) of the pixels in an image



Geometric Transformation Matrices (2D)

Transform	Example	Transformation matrix
Translation		$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$
Scale `imresize()`		$\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$
Shear		$\begin{bmatrix} 1 & h_x & 0 \\ h_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
Rotation with θ (clock-wise) `imrotate()`		$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$

http://www.mathworks.com/help/images/performing-general-2-d-spatial-transformations.html

Image Rotation: `imrotate()`

```
I = imread('rice.png'); subplot(1,2,1);
imshow(I); J = imrotate(I, 35, 'bilinear');
subplot(1,2,2); imshow(J);
size(I)
size(J)
```

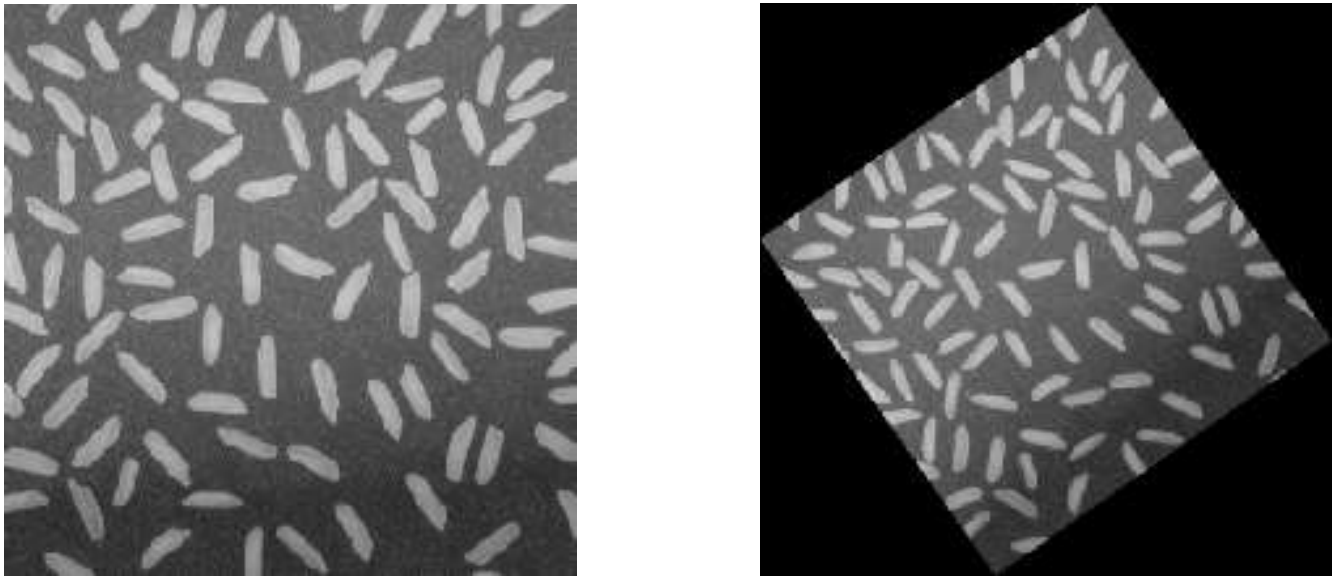
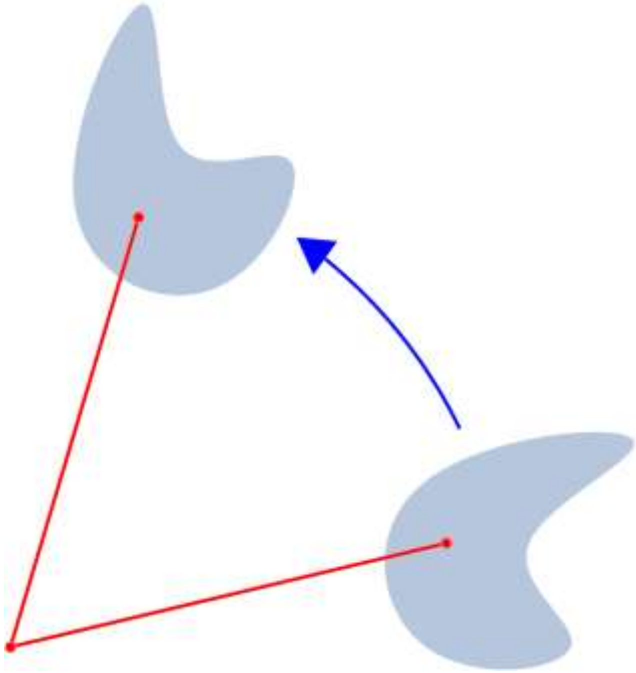


Image Rotation

- In two dimensions, rotation of a point (x, y) for an angle θ "counter-clockwise" can be written as:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



Write Image: `imwrite()`

- Format supported: 'bmp', 'gif', 'hdf', 'jpg', 'jpeg', 'jp2', 'jpx', 'pcx', 'pnm', 'ppm', 'ras', 'tif', 'tiff', 'xwd'
- Example:

```
imwrite(I, 'pout2.png');
```

End of Class