

# Outline of the Lecture

- Image Processing Toolbox (IPT--Matlab).
- > Reading Images.
- Displaying Images

# Image Processing Toolbox (IPT) (Matlab)

- **IPT** is a collection of functions that extend the capability of the Matlab numeric computing environment and support a wide range of image processing including:
  - Spatial image transformation.
  - Morphological operations.
  - Neighborhood and block operations
  - Linear filtering and filter design.
  - Transform.
  - Image enhancement and analysis.
  - Deblurring.
  - Region of interest operations.

#### Coordinate Conventions:

- The result of **sampling** and **quantization** is a **matrix of real numbers**; two ways are used to represent the digital image:
  - First convention: this convention is frequently used in image processing books, in which,
    - ✓ The image origin is defined to be at (x, y) = (0,0), the next coordinate value along the first row of the image is (x, y) = (0,1).
    - $\checkmark x$  ranges from 0 to M 1 and y ranges from 0 to N 1 in integer increment.
  - Second convention: this convention is used in IPT Matlab toolbox, the notation (r, c) to indicate rows and columns.
    - ✓ The origin of the coordinate system is at (r, c) = (1, 1).
    - $\checkmark$  r ranges from 1 to M and c ranges from 1 to N in integer increment.
- IPT documentation refers to the coordinates as **pixel** coordinates or **spatial** coordinates.

## Matrix notation of the digital image:

- Representations of digital image functions:
  - 1. Image processing books matrix representation.



5		- OI	rigin	Ĕ.						 -101 - 101
	0	1	2	3 .						 N - 1
	0.4			+		+	-9-	*	÷.	+ • y
1	1 +									
33	2 🔶									(m)
	3 🕂	۰	۰	٠		٠	٠	٠		
										1.00
		÷								2.00
	+					٠	٠			
	*	æ	.0			٠	٠			3 <b>9</b> 0
	. ÷									
	: +-	Ð	٠	٠		٠	٠	٠		
M -	1 +			٠			2			
	ł		O	ne p	ixel		~			f(x, y)

# 2. Matlab matrix representation

	[ <i>f</i> (1,1)	f(1,2)	 f(1,N)ך
f(r,c) =	f(2,1)	f(2,2)	 f(2,N)
	$\frac{1}{f(M-1)}$	f(M, 2)	 f(M,N)
	L J (M, 1)	J (1VI, Z)	 J (141, 1V) J

- The two representations are **equal** except for the **shift** in origin.
- The notation f(p,q) denotes the element located in row p and column q.
- $1 \times N$  matrix is called a **row vector**.
- $M \times 1$  matrix is called a **column vector**.
- $1 \times 1$  matrix is **scalar**.

200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
200	200	280	298	200	208	268	280	290	286	200	200	200	88		80	88	288	208	208	200	288	280	290	286	288	208	200	288
200	290	200	200	200	200	200	200	290	03	10	-	10	60	80	63	00	60	80	60	60	200	200	290	200	200	200	200	200
200	200	200	200	200	205	200		80	02	100			90		180		60	10	60	60	80	00	200	200	208	200	200	200
290	200	200	299	290	200	89		190			100	-60		100	100	100	199	10				88	89	200	208	209	200	200
200	200	200	208	200	80	80	100		100		100	10	100	100	110	86	165	80	102		100	100	50	60	200	200	200	200
200	200	200	200	80	80		100	100	100	100		10	90		00	100	100	10		100	100	100		00	10	200	100	200
200	200	200	119	60				100	100		100			1.50	100	100	69.	80	100	100		100	100	100		80	200	200
290	200	200	119	100	100	100	100	100	100	100	100	100	100			80	les	1.	105	100	100	100	186	1	100	86	200	200
200	200	85							100			100	100	160	100	180			100			100			106	-	80	200
200	200	00								80.				100													00	200
200	200	00	107	100	100	100	100	100	02	10	100	100	100	100	100	100	190	80	30		100	100	100	100	100	306	90	200
250	80	88.	15	107	100	160	1.00	100	100	183	100	167	160	189	199	100	Tae.	10-1	100	108	Leo	186	146	155	188	100	46	200
200	60	60		100										100					100						100		80	200
200	80	63;	100	105	100	100	100	100		100	100	100	100	19	180	100	169	300	100	100		100	100	100	106	100	80	200
290	200	00	101	100	100	100	120		100	100	100	100		100			100	110	100	100	100	190		100		109	80	300
200	200	85	16								100	100															80	200
200	200	60.	100	100	200	100	100	100			100	100	100	120	100	100	100	100	100			100	100	100	101	60	80	200
200	200	60	80		100		100			60	3,00				100				100	90						60	200	200
200	200	200	89	100	100	100	100	100		100		- 10			100			10	0	100		100	100	100	10	86	200	200
200	200	200	200	20	100	100	150	100	100	100	100	100	80	80	60	80	66.	100	100	100	100	100	190	100	80	200	200	200
200	200	200	200	60	89										110	100					100			60	200	200	200	200
200	200	200	200	200	80	80			100			100				100	100					.001	-00	200	200	200	200	200
290	200	200	2011	2:00	200	200	har	10	100					100	100					80	22	121	200	200	200	200	200	200
200	200	200	200	200	2:00	200	200	200	00	80	- 80	80					60	80	-00	80	200	200	200	200	205	200	200	200



#### Dr. Qadri Hamarsheh

# Reading Images

• To read an image, use the **imread** command, whose syntax is:

```
I = imread ('filename')
```

• This command **reads** and **stores** the image in an **array I**. **'filename'-** string containing the **complete name (path)** of the image file including extension.

## Example:

- >> f= imread('chestxray.jpg');
- reads the jpg image "chestxray' into image array.
- >> I= imread(pout.tif');
  - reads and stores the image in an array I.

## whos command:

• whos command is used to get information about variable in the workspace.

#### >> whos

Name	Size	Bytes	Class	Attributes
Ι	291x240	69840	uint8	

#### size command:

- Function size gives the row and column dimensions of an image:
- >> size(f);

```
ans = 1024 1024
```

# >> [M , N] = size(f);

- This syntax returns the number of rows (M) and the number of columns (N) in the image.
- The following table summarizes some ways to get information about an image. These are not specific to the Image Processing Toolbox.

Command	Description
whos	To get information about size, type, and bytes, of all variables.
whos I	For information about an image stored in I.
size(I)	To get the size of the image stored in I.
class(I)	To get type of data stored in I

# Displaying Images

• To display an image, the **imshow** function is used, which has the basic syntax:

# a) imshow (f,G)

**f**- an image, **G**- the **number of intensity levels** used to display the image, if **G** is omitted, it defaults to **256** levels.

# b) imshow (f, [low high])

- ✓ Displays as **black** all values **less than or equal to low**.
- ✓ Displays as white all values greater than or equal to high.
- $\checkmark$  The values in between are displayed using the **default number of levels**.

# c) imshow (f, [ ])

#### Dr. Qadri Hamarsheh

✓ Sets variable low to the minimum value of array f and high to its maximum value. This form is useful for images with low dynamic range.

#### pixval command :

- **Pixval** is used to display the intensity values of individual pixel **interactively**: Moving the cursor, the coordinates of the cursor position and the corresponding intensity values are shown.
- **Pixval** displays **Red**, **Green**, **blue** components, when the image is color.
- If the left button on the mouse is **clicked** and then held pressed, **pixval** displays the **Euclidean distance** between the initial and current cursor location, the syntax is:

#### pixval

#### Example 1

```
>> f = imread ('rose_512.tif'); % read from disk an image
```

>> whos f % extract basic information about the image

```
>> imshow (f) % display the image.
```

• If another image, **g**, is displayed using **imshow**, Matlab replaces the image in the screen with the new image.

```
>> figure, imshow (g) % keep the first image and
```

%output a second image.

```
>> imshow (f), imshow (g) % display both images
```

# Example 2

- Suppose that we read an image **h** and find that using **imshow (h)** produces the image that has a **low dynamic range**, to correct:
- >> Imshow (h, [ ]) % improve the image h.
- There are a series of photos that come as part of the image processing toolkit. To get the list of images and credits, you can type:

# >> help imdemos

- If you want to view any of these photos, you can use the **imshow**, which opens a separate window displaying the image. For instance:
- >> imshow('football.jpg');
- >> imshow('coins.png');
- >> imshow('autumn.tif');
- >> imshow('board.tif');