Digital Image Fundamentals

- **Image Sampling and Quantization,**
- **Representing Digital Images,**
- **Spatial and Gray-level Resolution,**
- **Zooming and Shrinking Digital Images,**
- **Some Basic Relationships between Pixels,**
- **Introduction to the Basic Mathematical Tools Used in Digital Image Processing**

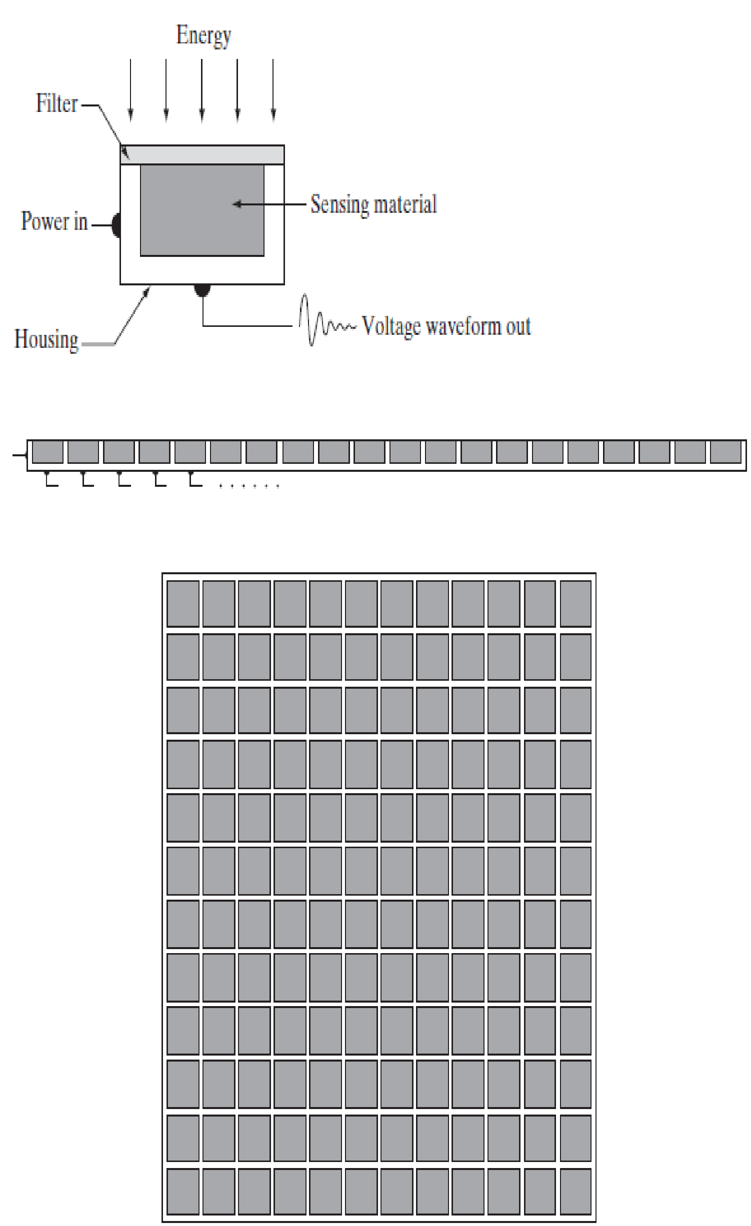
PRE-REQUISITES

- **Image Acquisition and Sensing methods**

Image sampling & quantization

- Basic concepts in Samplig and Quantization
- Representation Digital Images
- Spatial and Intensity Resolution
- Image Interpolation

Image sensing & acquision and



a
b
c

FIGURE 2.12 (a) Single imaging sensor. (b) Line sensor. (c) Array sensor.

Most of the images in which we are interested are generated by the combination of an "illumination" source and the reflection or absorption of energy from that source by the elements of the "scene" being imaged. A Visible light source illuminates a common everyday 3-D (three-dimensional) scene. For example,
the illumination may originate from a source of electromagnetic energy such as radar, infrared, or X-ray system.

Image sampling & quantization

- Basic concepts in Samplig and Quantization
- Representation Digital Images
- Spatial and Intensity Resolution
- Image Interpolation

Image sampling & quantization

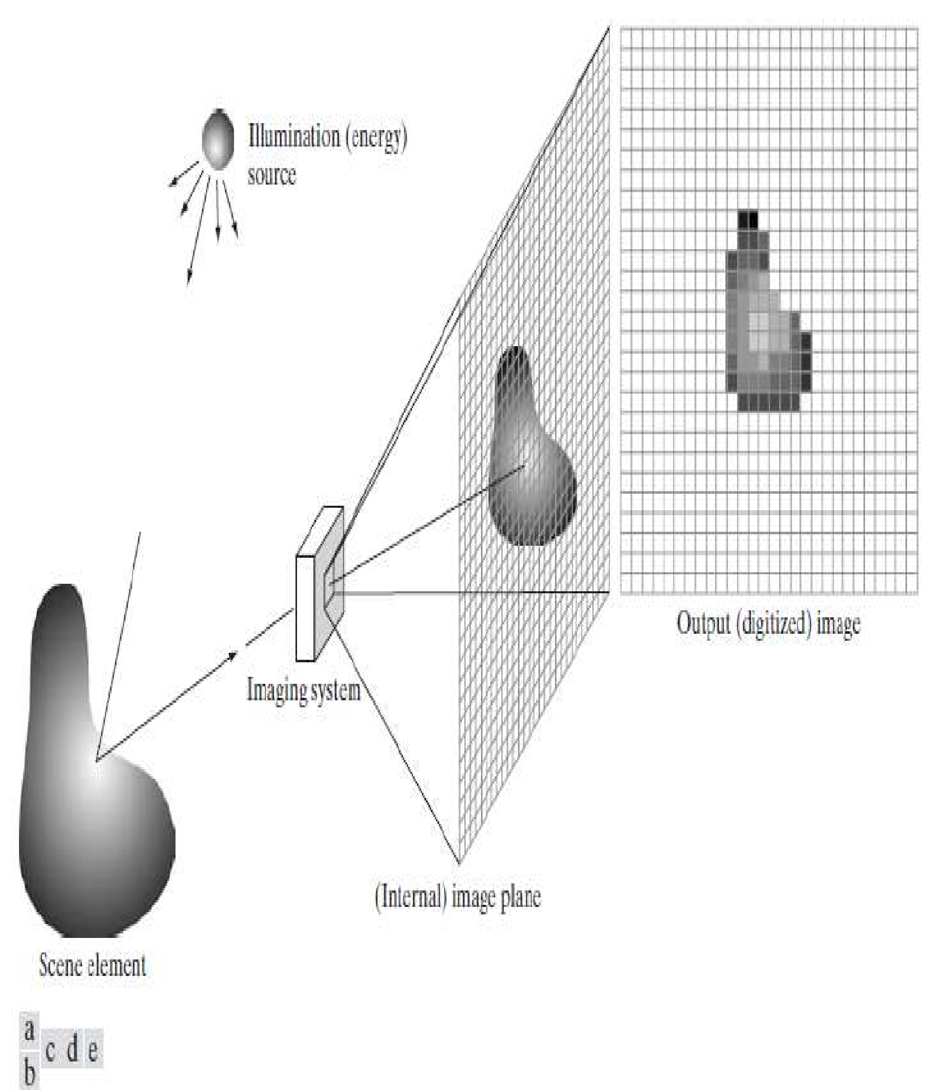
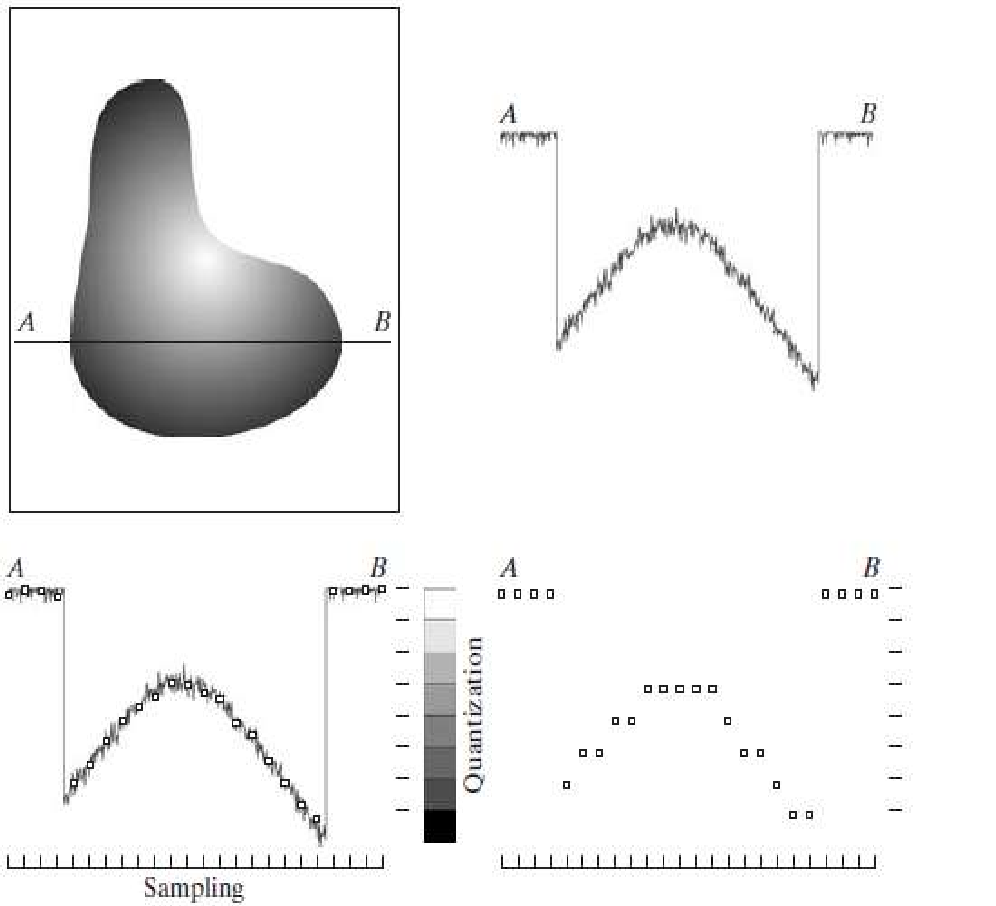


FIGURE 2.15 An example of the digital image acquisition process. (a) Energy ("illumination") source. (b) An element of a scene. (c) Imaging system. (d) Projection of the scene onto the image plane. (e) Digitized image.

we denote images by two-dimensional functions of the form . The value or amplitude of *f at spatial coordinates* is a positive scalar quantity whose physical meaning is determined by the source of the image.When an image is generated from a physical process, its intensity values are proportional to energy radiated by a physical source. (e.g., electromagnetic waves). As a consequence, f(x, y) must be nonzero

Image sampling & quantization



a b
c d

FIGURE 2.16 Generating a digital image. (a) Continuous image. (b) A scan line from *A* to *B* in the continuous image, used to illustrate the concepts of sampling and quantization. (c) Sampling and quantization. (d) Digital scan line.

Converting the Image acquired into the Digital Format.
Continuos image intensities into the Bits representation.
Image shows a continuous image *f that we want to convert to digital* form. An image may be continuous with respect to the *x- and y-coordinates,* and also in amplitude. To convert it to digital form, we have to sample the function in both coordinates and in amplitude.
Digitizing the coordinate values is called *Sampling.*
Digitizing the amplitude values is called Quantization.

Image sampling & quantization

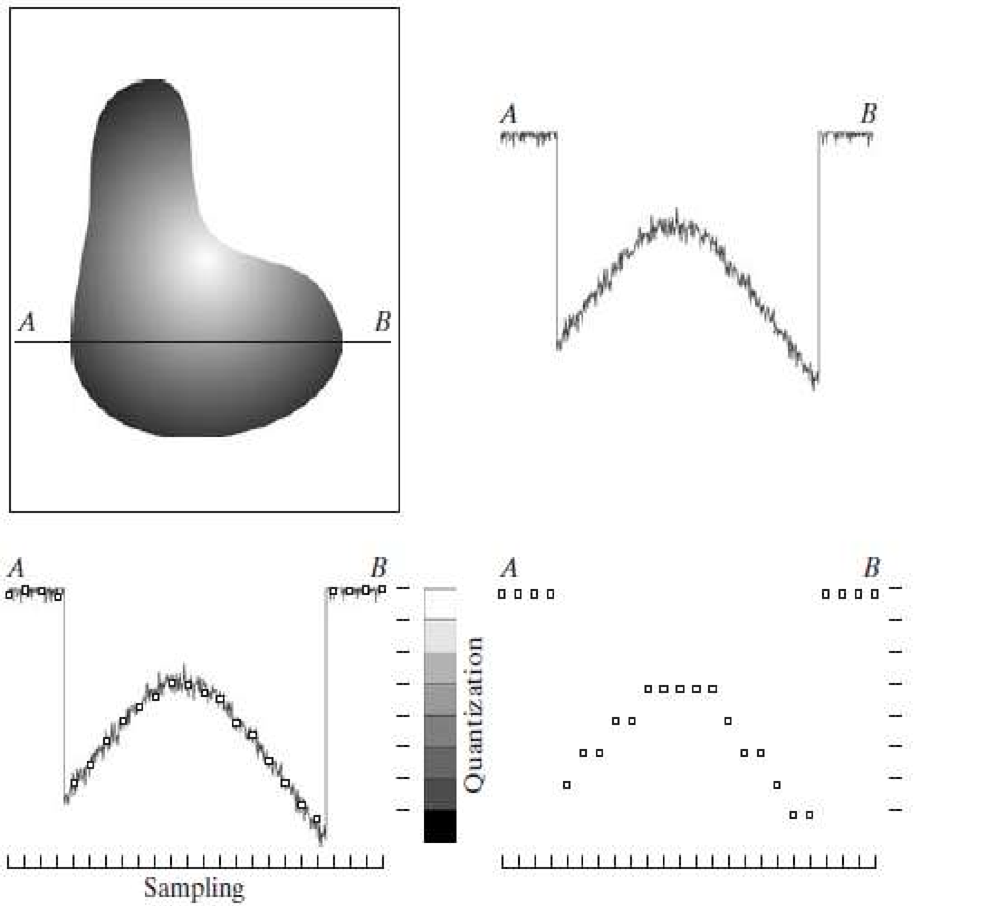
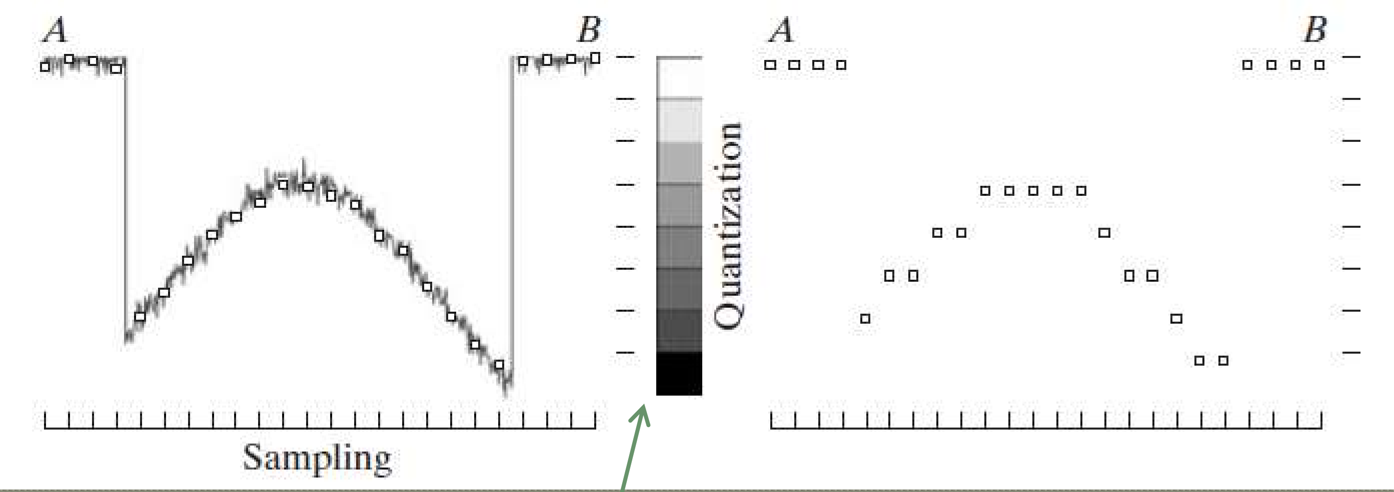


FIGURE 2.16 Generating a digital image. (a) Continuous image. (b) A scan line from *A* to *B* in the continuous image, used to illustrate the concepts of sampling and quantization. (c) Sampling and quantization. (d) Digital scan line.

Fig. 2.16(b) is a plot of amplitude (intensity level) values of the continuous image along the line segment *AB in Fig. 2.16(a).* The random variations are due to image noise.

*Fig. 2.16(c) shows: E*qually spaced samples along line *AB.*
The spatial location of each sample is indicated by a vertical tick mark in the bottom part of the figure.The samples are shown as small white squares superimposed on the function. The set of these discrete locations gives the sampled function.

Image sampling & quantization



To form a digital function, the intensity values also must be converted (*quantized*) *into discrete quantities.*

The right side of Fig. 2.16(c) shows the intensity scale divided into eight discrete intervals, ranging from black to white.The vertical tick marks indicate the specific value assigned to each of the eight intensity intervals.The continuous intensity levels are quantized by assigning one of the eight values to each sample.The assignment is made depending on the vertical proximity of a sample to a vertical tick mark. The digital samples resulting from both sampling and quantization are shown in Fig. 2.16(d). **Starting at the top of the image and carrying out this procedure line by line produces a two-dimensional digital image**. It is implied in Fig. 2.16 that, in addition to the number of discrete levels used, **the accuracy achieved in quantization is highly dependent on the noise content of the sampled signal.**

Digital Image Representation using x,y and f(x,y)

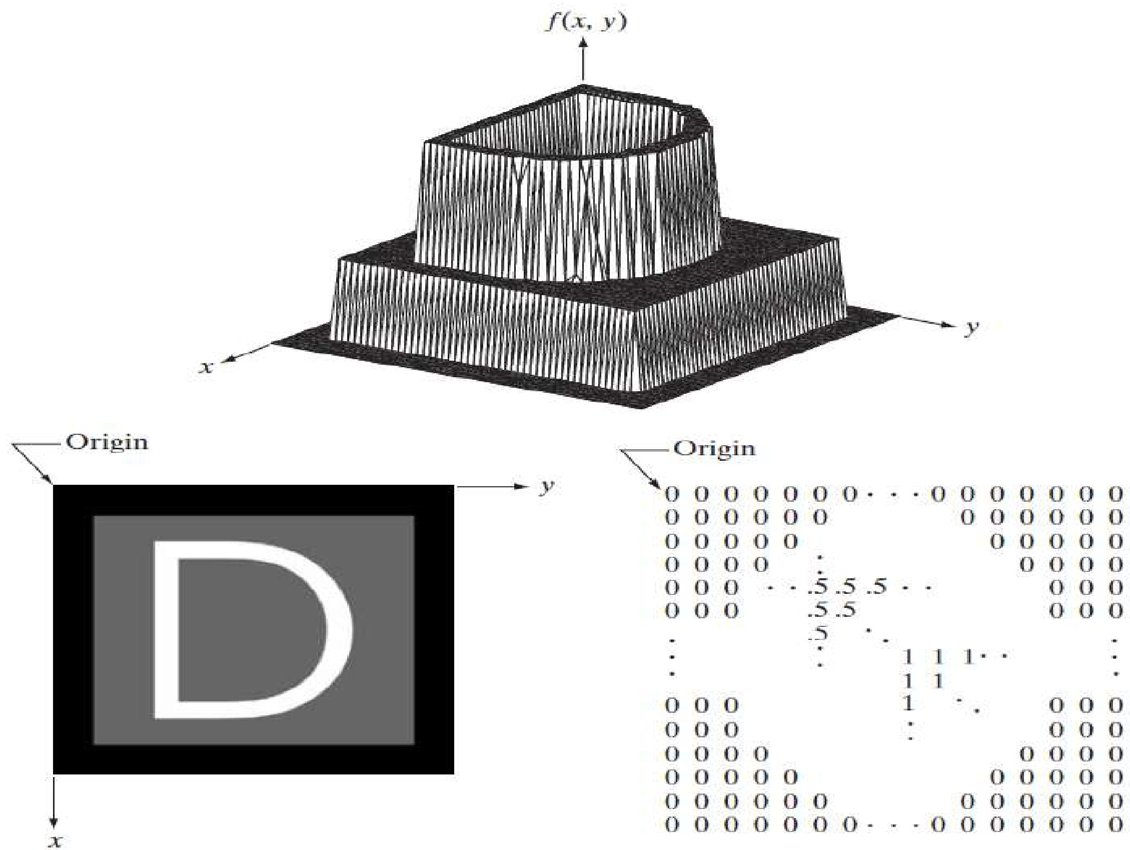
x,y: Co-ordinates, f(x,y) : Intensities Looks Like a Matrix

Dark/Pitch Black area Pixel Intensity = 0

Gray scale area Pixel Intensity = 0.5

Full White area Pixel Intensity = 1

Total of 3 Intensity Levels used to draw the Picture D



a
b c

FIGURE 2.18 (a) Image plotted as a surface. (b) Image displayed as a visual intensity array. (c) Image shown as a 2-D numerical array (0, .5, and 1 represent black, gray, and white, respectively).

Digital Image Representation using x,y and f(x,y)

Using Matrix Values as Representation to store an Image

sults at a glance. Numerical arrays are used for processing and algorithm development. In equation form, we write the representation of an $M \times N$ numerical array as

$$f(x, y) = \begin{bmatrix} f(0, 0) & f(0, 1) & \cdots & f(0, N-1) \\ f(1, 0) & f(1, 1) & \cdots & f(1, N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1, 0) & f(M-1, 1) & \cdots & f(M-1, N-1) \end{bmatrix} \quad (2.4\text{-}1)$$

Both sides of this equation are equivalent ways of expressing a digital image quantitatively. The right side is a matrix of real numbers. Each element of this matrix is called an *image element*, *picture element*, *pixel*, or *pel*. The terms *image* and *pixel* are used throughout the book to denote a digital image and its elements.

In some discussions it is advantageous to use a more traditional matrix notation to denote a digital image and its elements:

$$\mathbf{A} = \begin{bmatrix} a_{0,0} & a_{0,1} & \cdots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdots & a_{1,N-1} \\ \vdots & \vdots & & \vdots \\ a_{M-1,0} & a_{M-1,1} & \cdots & a_{M-1,N-1} \end{bmatrix} \quad (2.4\text{-}2)$$

Image Interpolation

Interpolation is a basic tool used extensively in tasks such as

- Zooming, (Image size changes)
- Shrinking, (Image size changes)
- Rotating,
- Geometric corrections.

Interpolation is the process of using known Pixels data to estimate values at unknown Pixels locations

Image Interpolation

- Zooming, (Image size changes)
- Shrinking, (Image size changes)

- ZOOMING: https://www.youtube.com/watch?v=8bTDssnJyZc
- An image of 500x500 to 1.5 time => 750x750???

To visualize zooming is to create an imaginary grid with the same pixel spacing as the original, and then shrink it so that it fits exactly over the original image. Obviously, the pixel spacing in the shrunken 750x750 grid will be less than the pixel spacing in the original image. To perform intensity-level assignment for any point in the overlay, we look for its closest pixel in the original image and assign the intensity of that pixel to the new pixel in the750x750 grid. When we are finished assigning intensities to all the points in the overlay grid, we expand it to the original specified size to obtain the zoomed image.

Image Interpolation

- Zooming, (Image size changes)
- Shrinking, (Image size changes)

ZOOMING:

1. Nearest Neighbor Interpolation
2. Bi-Linear Interpolatio
3. BI-Cubical Interpolation

2x2 Enlarge 5 Times => 6x6

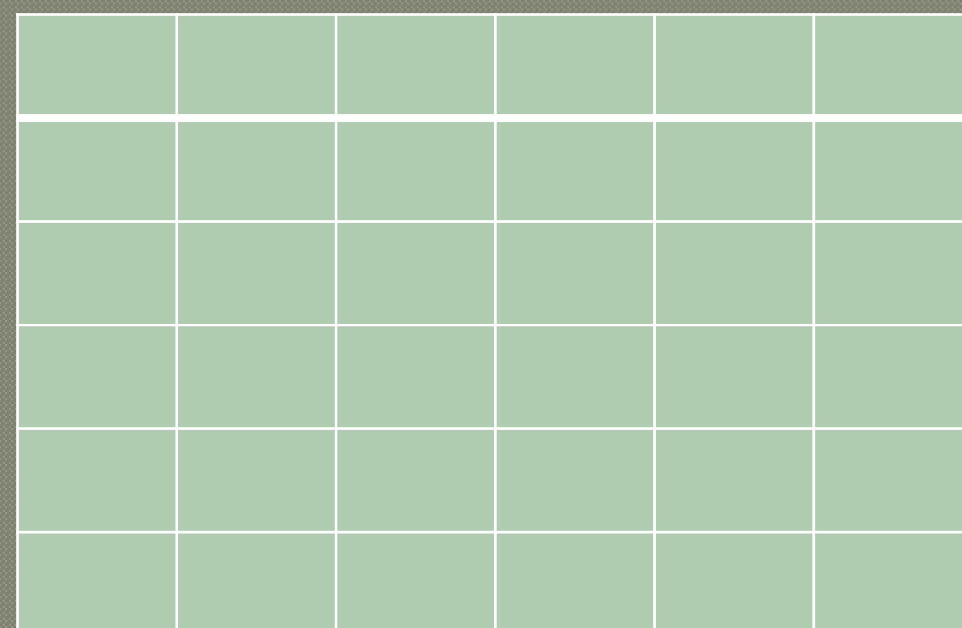


Image Interpolation

- Zooming, (Image size changes)
- Shrinking, (Image size changes)

ZOOMING: Nearest Neighbor Interpolation

- An image of **2x2 Enlarge 5 Times => 6x6**

10	40
30	20

10	10	10	40	40	40
10	10	10	40	40	40
10	10	10	40	40	40
30	30	30	20	20	20
30	30	30	20	20	20
30	30	30	20	20	20

Nearest Neighborhood Interpolation

Suppose that an image of size pixels has to be enlarged 1.5 times to pixels. A simple way to visualize zooming is to create an imaginary grid with the same pixel spacing as the original, and then shrink it so that it fits exactly over the original image. Obviously, the pixel spacing in the shrunken grid will be less than the pixel spacing in the original image. To perform intensity-level assignment for any point in the overlay, we look for its closest pixel in the original image and assign the intensity of that pixel to the new pixel in the grid. When we are finished assigning intensities to all the points in the overlay grid, we expand it to the original specified size to obtain the zoomed image. 750 * 750

Linear Image Interpolation

- ZOOMING:
- An image of 2X2 to 5 time => 6x6
- we use the Linearly map the intensity at a given location.

2x2

10	40
30	20

6x6

10	p	(q) 22	r	s	40
30					20

$q = (d1 * 40 + d2 * 10)/5$

$d1 = 2/5$

$d2 = 3/5$

$q = [(2*40)/5+(3*10)/5]$

$q = 22$

Linear Image Interpolation

- ZOOMING:
- An image of 2X2 to 5 time => 6x6
- we use the Linearly map the intensity at a given location.

p = d1* 40 + d2 * 10

d1 = 1/5

d2 = 4/5

p = [(1*40)/5+(4*10)/5]

p = 16

r = d1* 40 + d2 * 10

d1 = 3/5

d2 = 2/5

r = [3*40)/5+(2*10)/5]

r = 28

s = d1* 40 + d2 * 10

d1 = 4/5

d2 = 1/5

s = [4*40)/5+(1*10)/5]

s = 34

Linear Image Interpolation

- ZOOMING:
- An image of 2X2 to 5 time => 6x6
- we use the Linearly map the intensity at a given location.

2x2

10	40
30	20

6x6

10	16	22	28	34	40
30					20

$q = d1 * 40 + d2 * 10$

$d1 = 2/5$

$d2 = 3/5$

$q = [(2*40)/5+(3*10)/5]$

$q = 22$

Bi-Linear Image Interpolation

- ZOOMING:
- An image of 2X2 to 5 time => 6x6
- we use the Bi-Linearly map the intensity at a given location. Do it Horizontal and Vertical directions

2x2

10	40
30	20

10	16	b?	28	34	40
		C?			
30		a?			20

6x6

What is the Intensity of pixel marked C?

Determine the Values of PIXELs b & a,

determine using Linear image interpolation in Vertical axis.

a = (30*3)/5+(20*2)/5 = (90+40)/5=26

b= (10*3)/5+(40*2)/5=(30+80)/5=22

C=(22*4)/5+(26*1)/5=(88+26)/5=22.8=23

Bi-Cubic Interpolation

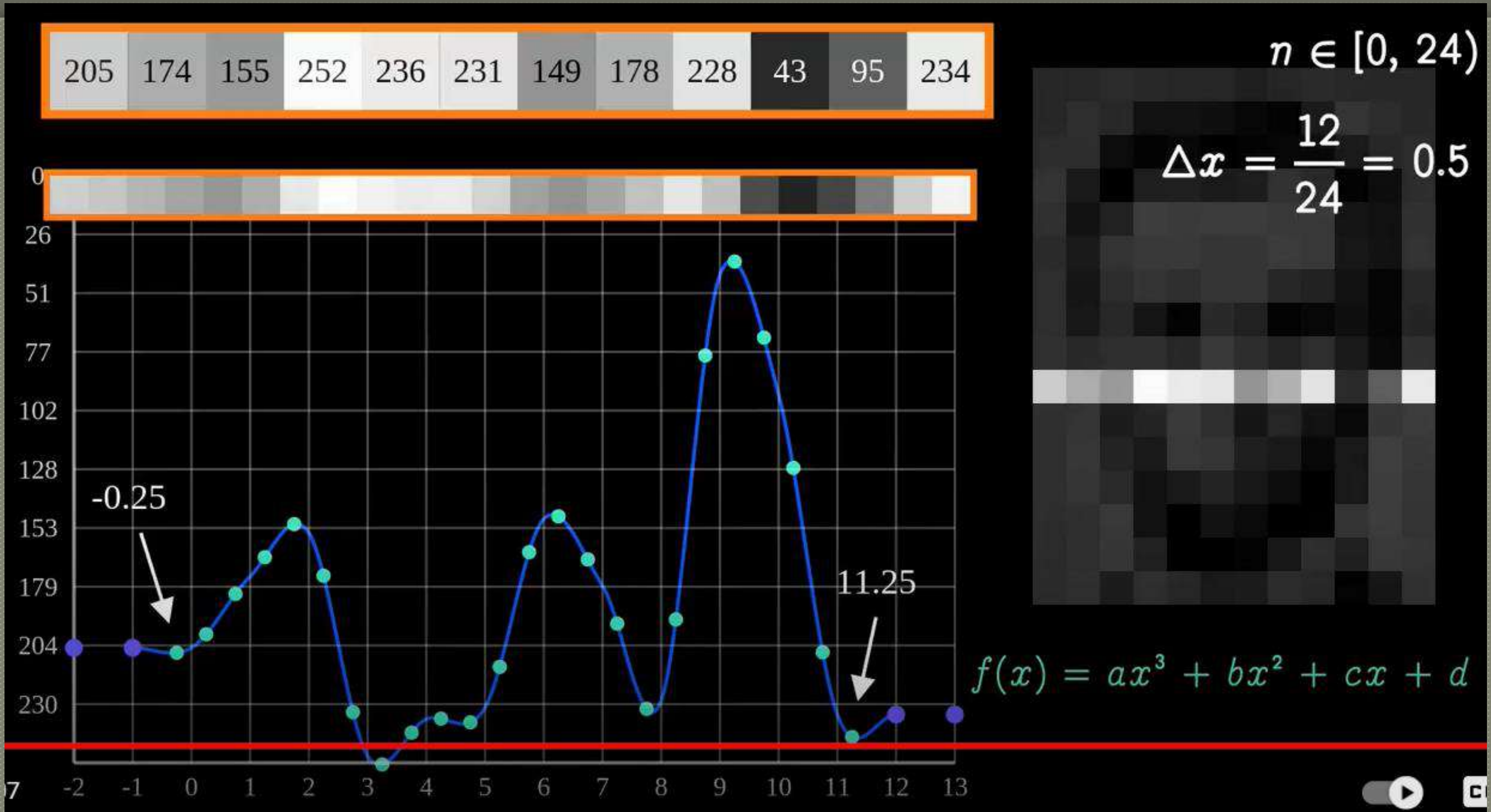
For Better understandig pictorially Follow this Youtube Link: https://www.youtube.com/watch?v=syH8ASkotFg

We use 3rd degree Quadratic Equations to determine the Co-efficients of Eq: f(x). Below chart X-axis: Spatial resolution, Y-axis: Intensity resolutions. Above organge box is One row vals

Bi-Cubic Interpolation

[For Better understandig pictorially Follow this Youtube Link: https://www.youtube.com/watch?v=syH8ASkotFg](https://www.youtube.com/watch?v=syH8ASkotFg)

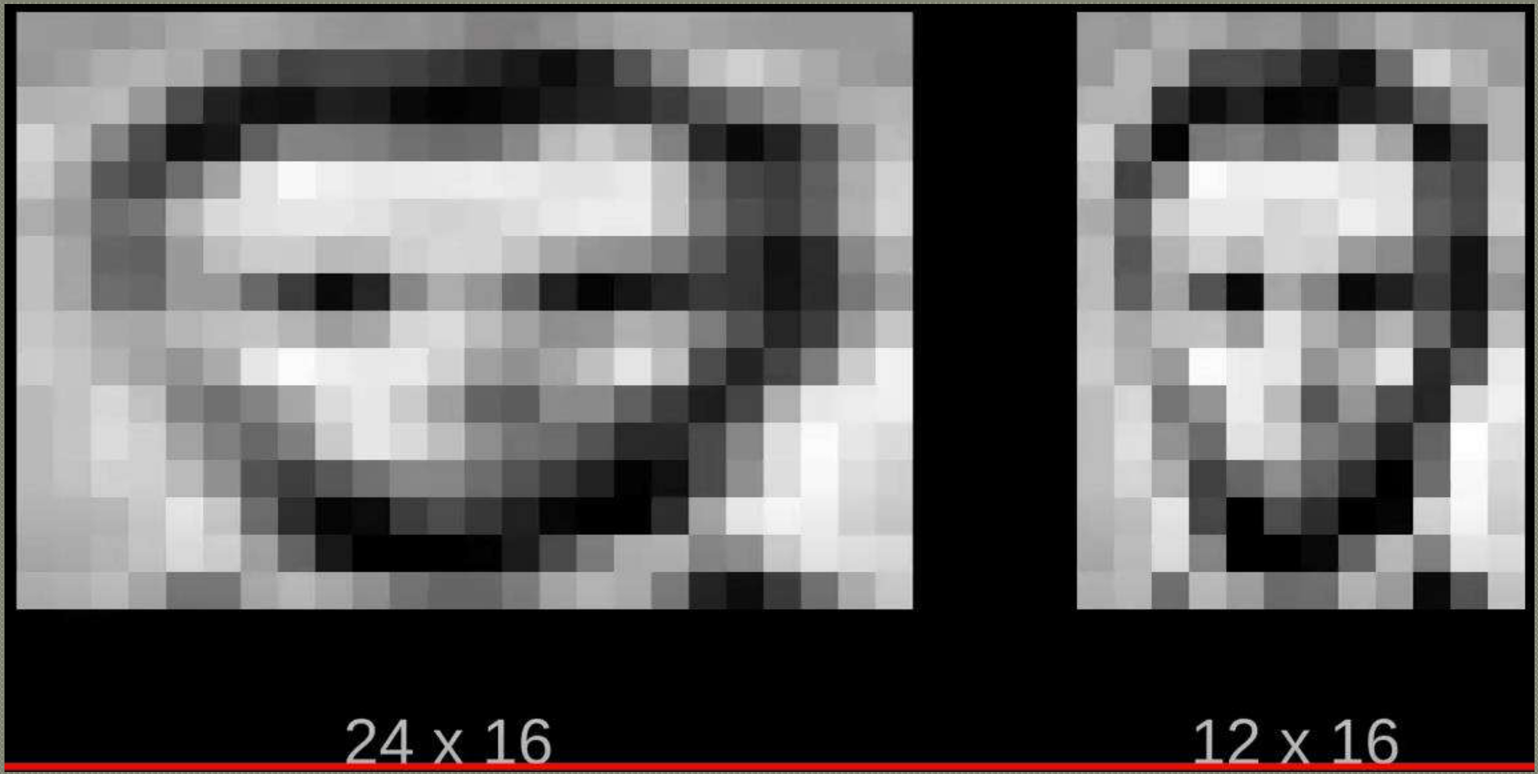
When we zoom increase the number of points to represent and the new pixels value are marked as SPLINE and green Dots.



Bi-Cubic Interpolation

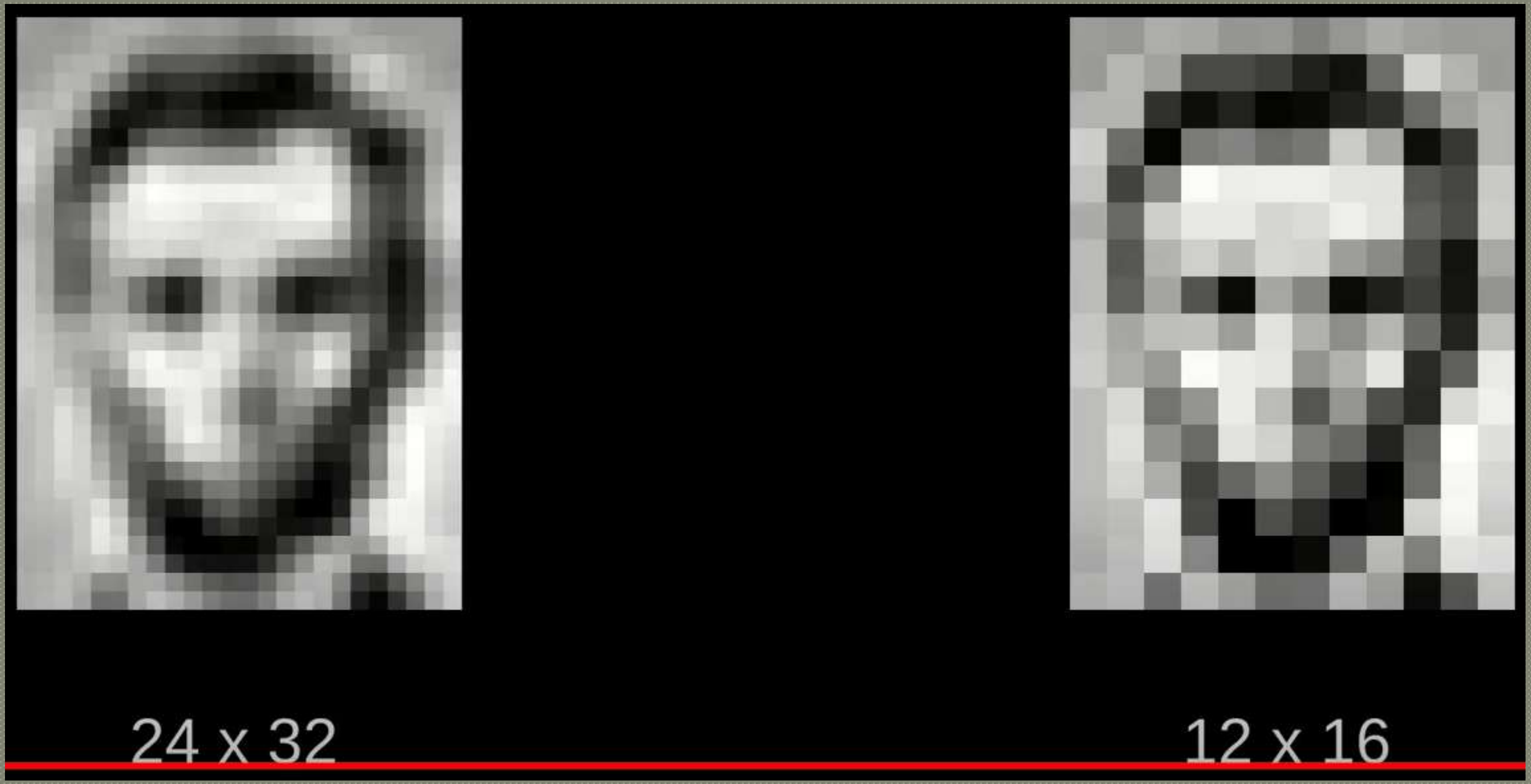
Applying spline operations with the increased Pixes (zoomed) on Row Wise, this is the Result.

.

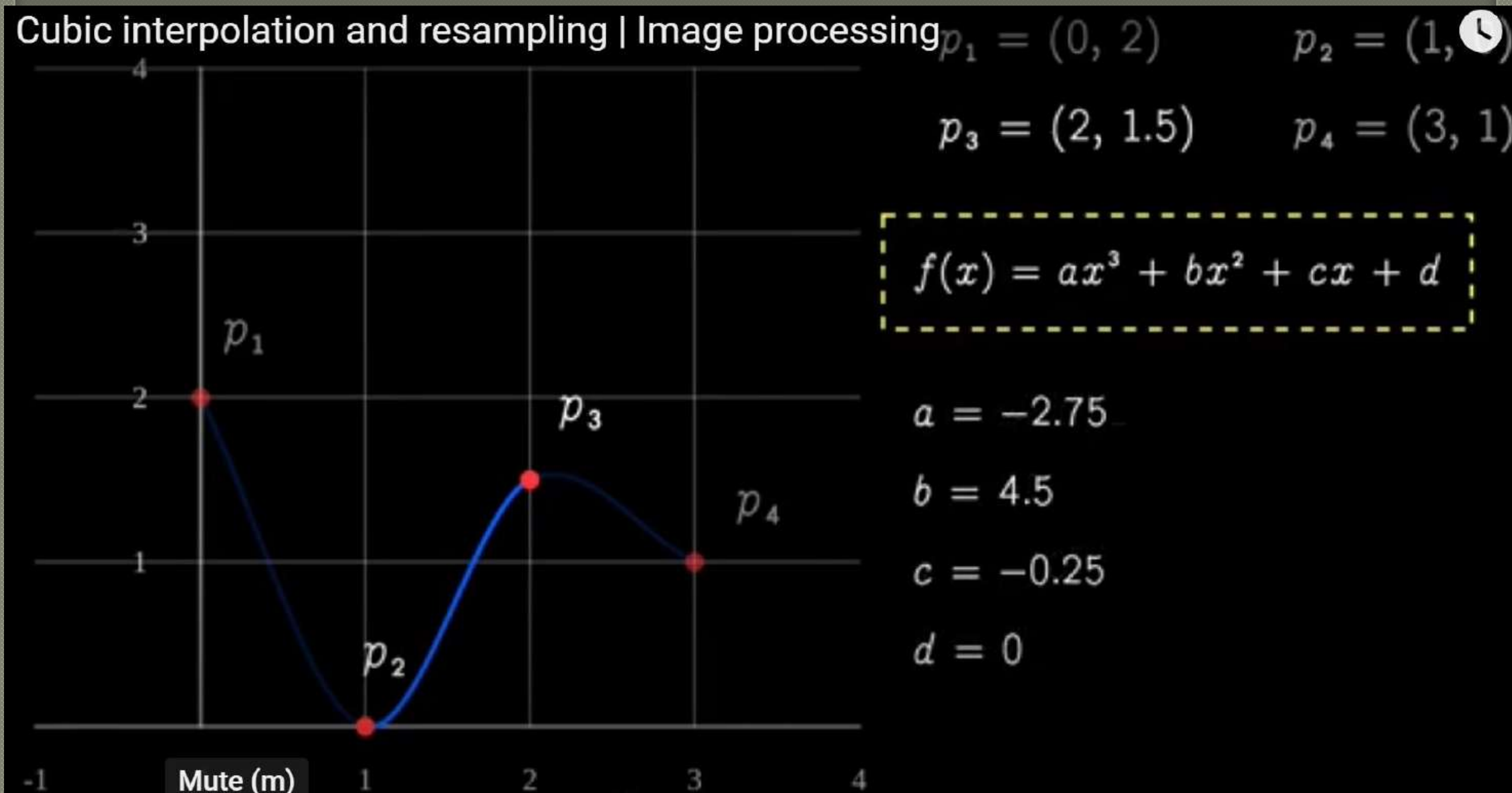


Bi-Cubic Interpolation

Applying spline operations on Row Wise apply on Column Wise, this is the Result. See the Image on the Left is better than the Right. Number of Pixels left is double the size of original pixels.



Bi-Cubic interpolation



Bi-Cubic Interpolation

Can we apply zooming with increased Pixels on BiCubic interpolation? What will happen?

$f(x) = ax^3 + bx^2 + cx + d$

$f'(x) = 3ax^2 + 2bx + c$

$f(0) = d \qquad f(1) = a + b + c + d$

$f(2) = c \qquad f(3) = 3a + 2b + c$

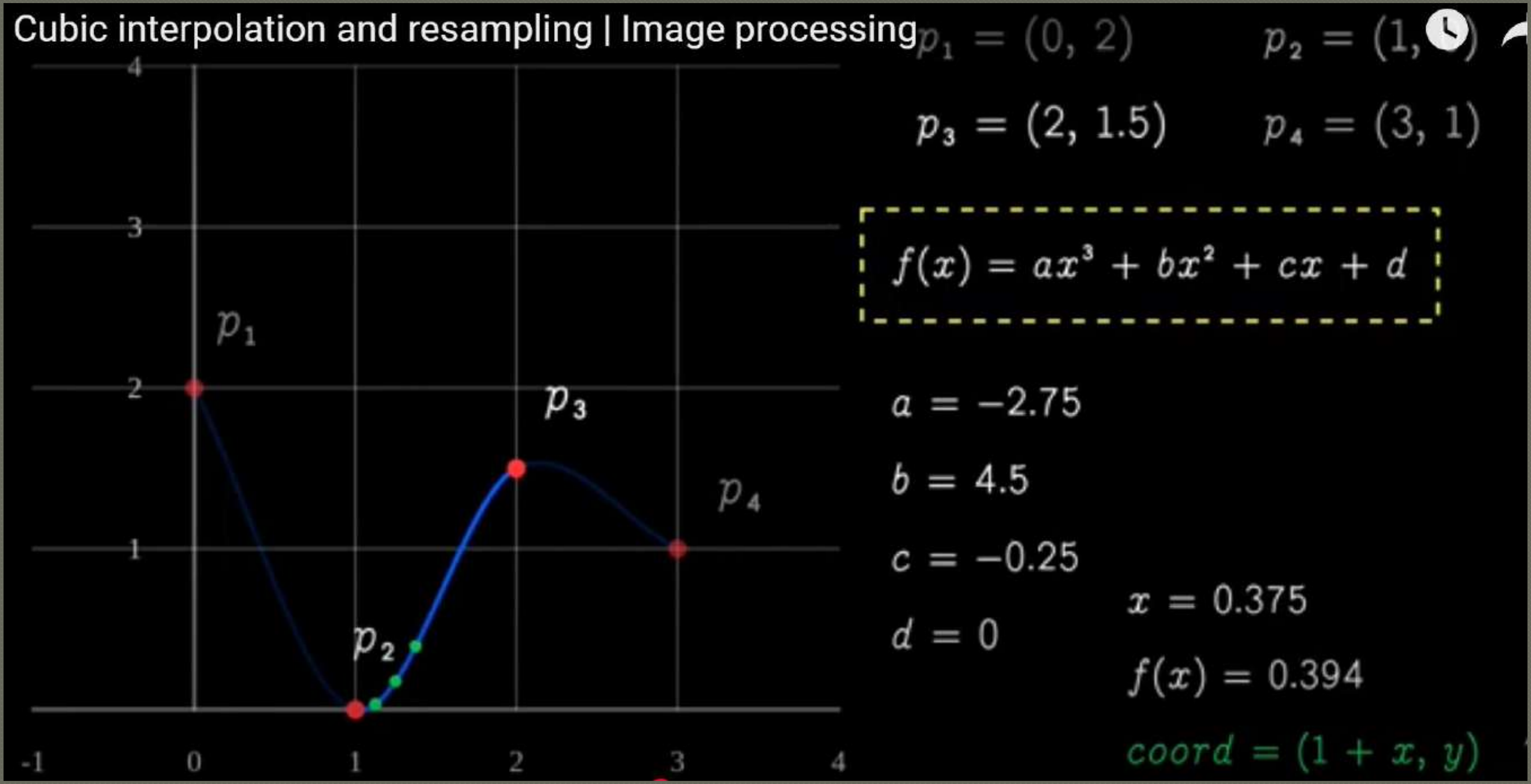
$a = 2\,f(0) - 2\,f(1) + f'(0) + f'(1)$

$b = -3\,f(0) + 3\,f(1) - 2\,f'(0) - f'(1)$

$c = f'(0)$

$d = f(0)$

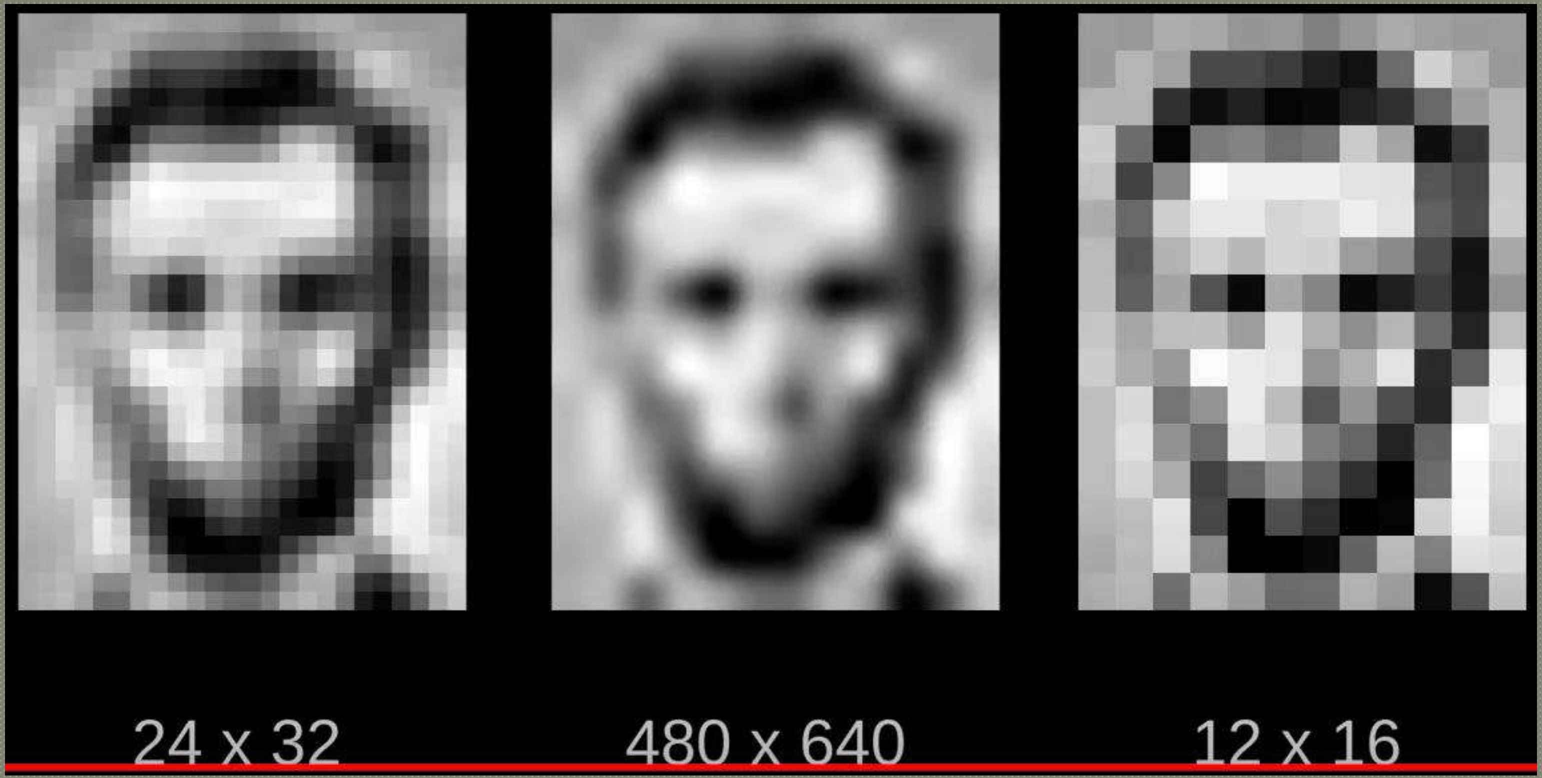
Bi-cubic interpolation



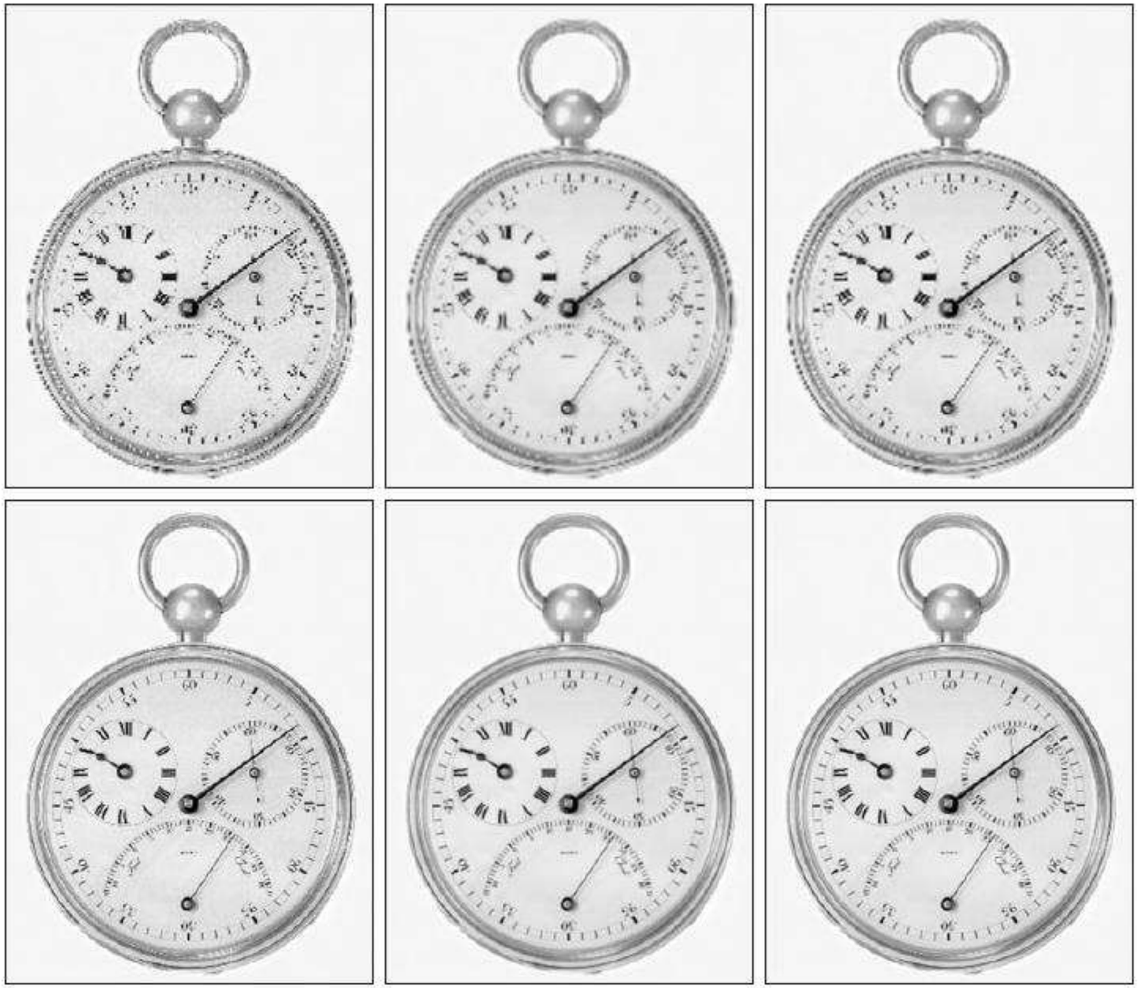
Bi-Cubic Interpolation

Applying Bi Cubic Interpolation with higher zooming gives even better results as shown the Middle Picture zoomed to 480X640 Pixes.

.



Compare Image Quality after ZOOM with various Interpolation methods



a b c
d e f

FIGURE 2.24 (a) Image reduced to 72 dpi and zoomed back to its original size (3692 × 2812 pixels) using nearest neighbor interpolation. This figure is the same as Fig. 2.20(d). (b) Image shrunk and zoomed using bilinear interpolation. (c) Same as (b) but using bicubic interpolation. (d)–(f) Same sequence, but shrinking down to 150 dpi instead of 72 dpi [Fig. 2.24(d) is the same as Fig. 2.20(c)]. Compare Figs. 2.24(e) and (f), especially the latter, with the original image in Fig. 2.20(a).

Basic Relationships between Pixels
https://www.youtube.com/watch?v=AjFURMXJTbw

Neighbors of Pixel p(x,y): 4 Adjacency

Horizontal Neighbor

Veriticl Neighbor pixels

4-adjacency:Two pixels p and q with values from V are 4-adjacent if q is in the set N4P(x,y)

4 Neighbors of Pixel: $N_4P(x,y)$

		p(x,y-1)		
	p(x-1,y)	**p(x,y)**	p(x+1,y)	
		p(x,y+1)		

Basic Relationships between Pixels

Neighbors of Pixel p(x,y): 8 Adjacency

Diagnol Neighbors = $N_D P(x,y)$

8-*adjacency: Two pixels p and q with values from V are 8-adjacent if q is in* the set $N_8P(x,y)$

$N_8P(x,y) = N_4P(x,y) + N_DP(x,)$

8 Neighbors = 4 Neighbors, 4 Diagnol Neighbors

	p(x-1,y-1)	**p(x,y-1)**	**p(x+1,y-1)**	
	p(x-1,y)	p(x,y)	**p(x+1,y)**	
	p(x-1,y-1)	**p(x,y+1)**	**p(x+1,y+1)**	

Basic Relationships between Pixels

Pixels relationships Problems:

Binary Image {1,0}, having V={1} Find the 4 Adjacency and Diagnal Adj, 8 Adjacency of P(1,2)

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

8 Adjacency=(0,1), (0,3), (2,2)

D Adjacency==(0,1), (0,3)

4 Adjacency=(2,2)

Basic Relationships between Pixels

Pixels relationships Problems:

Gray Scale Image [0..255], selected Set V={0,1,2,3,4,5,6,7,8,9,10}

Find the 4 Adj, 8Adj of P(1,2)=2

8 Adjacency=(0,1), (0,3), (2,2)

54	10	100	5
81	150	2	34
201	200	3	45
7?	70	137	56

4 Adjacency=(2,2)

P(3,0)=7, is not in Any Adjacency of P(1,2)=2

Basic Relationships between Pixels

Pixels Relationships: mixed Adjacency (m Adj)

It is introduced to remove the ambiguity that arise due to 8 Adjacency.

Connectivity Between P(2,2) to P(0,2)

0	1	1
0	1	0
0	0	1

8 Adj of P(1,1)

4 Adj of P(1,1)

4 Adj of P(1,0)

To Reach P(0,2) should we take 8 Adj or 4 Adj?

Straight Connectivity is given High priority than Diagnol Connectivity. P(2,2)=1 is Connected to P(0,2) Via P(0,1), P(1,1)

Connected Set = {P(0,2), P(0,1), P(1,1), P(2,2)}

m Adj

- *m-adjacency (mixed adjacency).Two pixels p and q with values from V are*
- *m-adjacent if*
- **(i)** ***q is in N_4P or,***
- **(ii)** ***q is in N_DP and the set $N_4P \cap N_DP$ has no pixels whose values are from V***

Connectivity

- *Connectivity between Pixels is a FUNdamentals concept that simplifies the definition of numerous Digital Concepts such as REGIONS & BOUNDARIES.*
- *To establish if Two Pixels are Connected, it must be determined if they are Neighbors and their Intensity levels satisfy a specified Criteria.*

Basic Relationships between Pixels

Pixels relationships Problems IN A BINARY IMAGE:

Pixes CONNECTIVITY -- REGIONS

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

CONNECTED SET 1

IF SET HAS CONNECTED SET THEN CALLED REGION 1

CONNECTED SET 2

REGION 2

Pixels relationships Problems IN A BINARY IMAGE :

Pixels CONNECTIVITY – REGIONS – Example

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

Region of Values -1
But also a
Connected SET

CONNECTED SET
of Values -0
But also a
Connected SET

Basic Relationships between Pixels

Pixels relationships Problems IN A BINARY IMAGE :

Pixels BOUNDARY

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

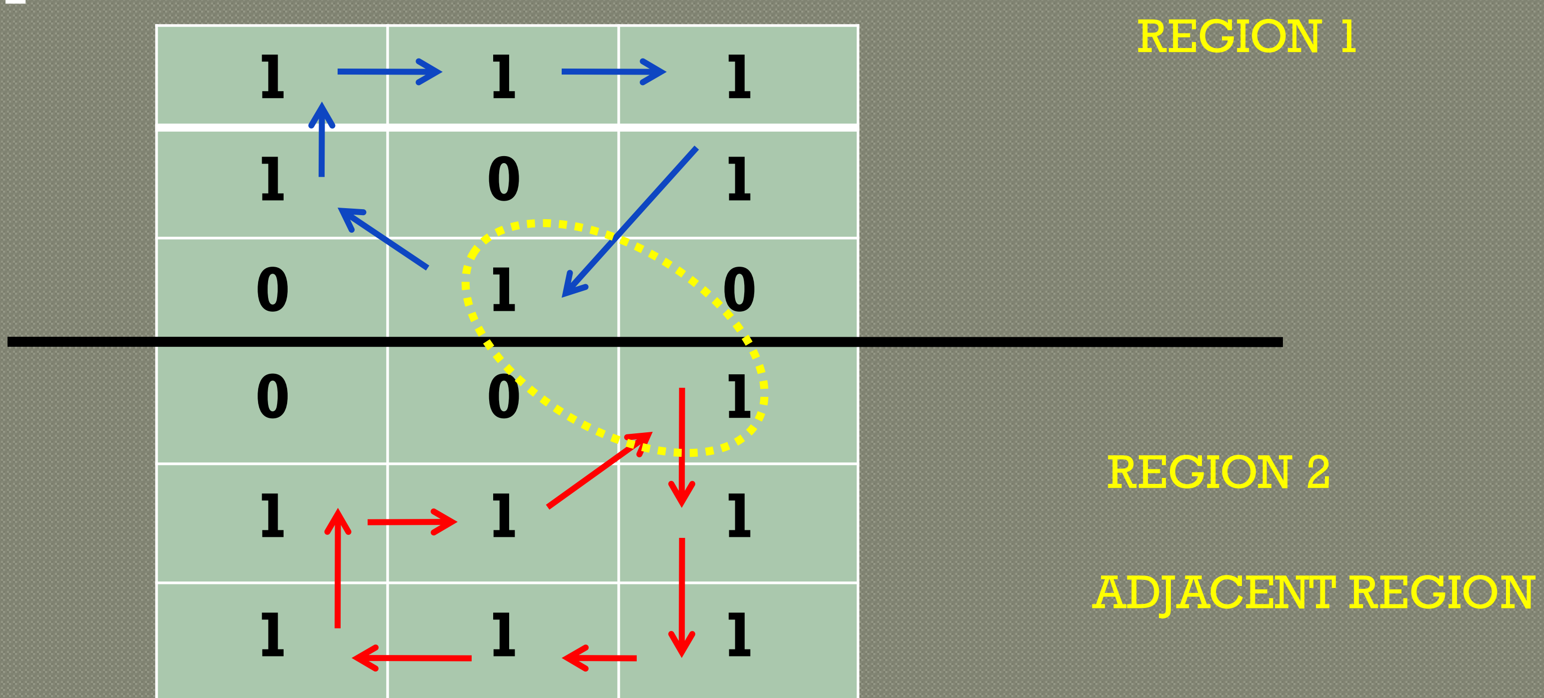
INNER BOUNDARY

OUTER BOUNDARY

Basic Relationships between Pixels

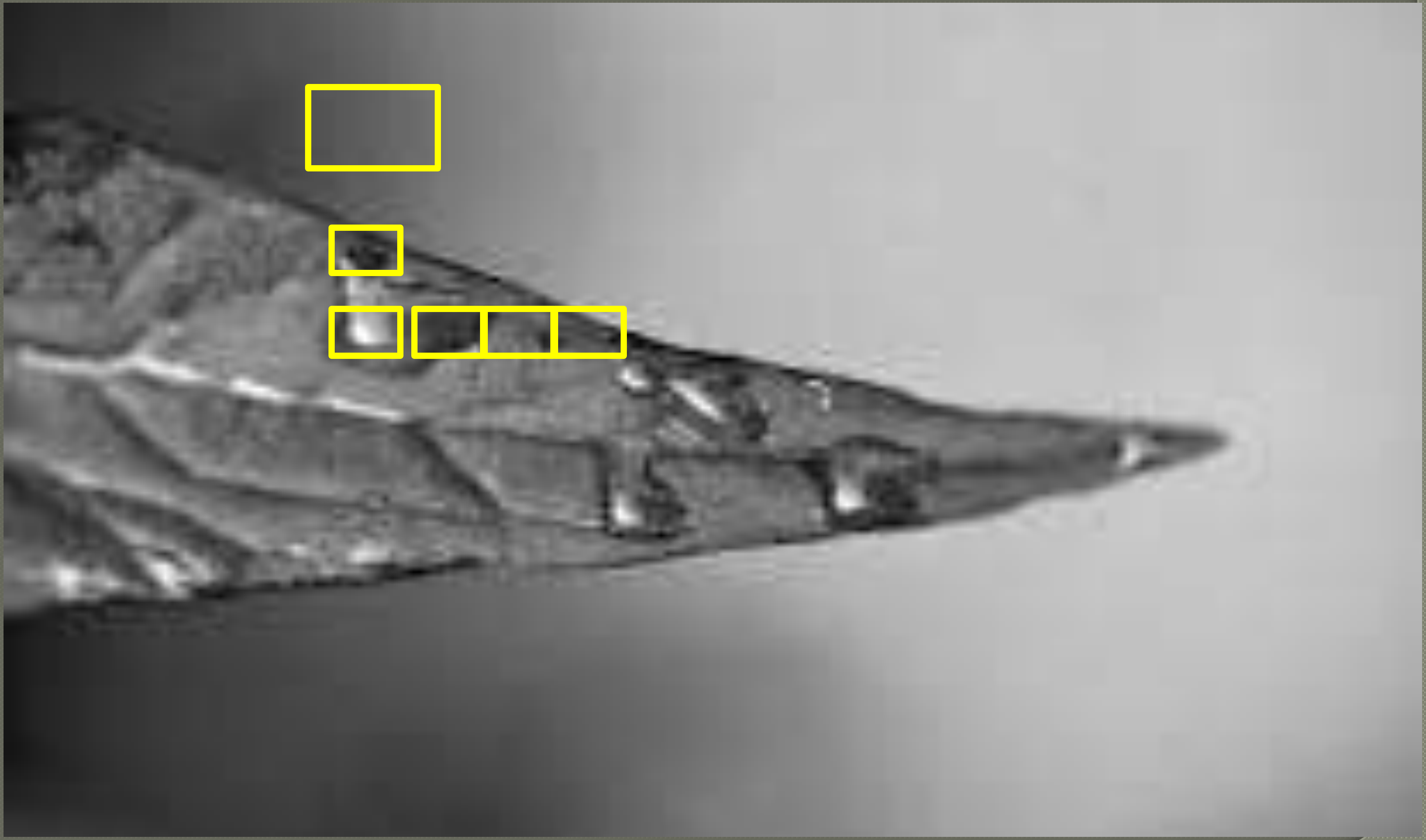
Pixels relationships Problems IN A BINARY IMAGE:

ADJACENT REGIONS WITH 8-ADJACENCY
BUT (R1 Û R2) is not connected set because 4 adj not applicable



Connected Regions?
Adjacent Regions?
Disjoint Regions?

BORDERS?



Connected Regions?
Adjacent Regions?
Disjoint Regions?

Foreground Pixels
Background Pixels

Edge: Are formed from pixels with derivative values that exceed a preset Threshold Pixel Value.

PIXEL Distance Calculations

For pixels p, q, and z, with coordinates (x, y), (s, t), and (v, w), respectively, D is a *distance function* or *metric* if

(a) $D(p, q) \geq 0 \quad (D(p, q) = 0 \quad \text{iff} \quad p = q)$,

(b) $D(p, q) = D(q, p)$, and

(c) $D(p, z) \leq D(p, q) + D(q, z)$.

- Distance measures?

1. Euclidean Distance
2. City-Block Distance
3. Chess Board Distance

PIXEL Distance Calculations

For pixels p, q, and z, with coordinates (x, y), (s, t), and (v, w), respectively, D is a *distance function* or *metric* if

(a) $D(p, q) \geq 0 \quad (D(p, q) = 0 \quad \text{iff} \quad p = q)$,
(b) $D(p, q) = D(q, p)$, and
(c) $D(p, z) \leq D(p, q) + D(q, z)$.

Euclidean Distance between p and q is defined as
Formula: $D_e(p,q) = [(x-s)^2 + (y-t)^2]^{1/2}$
Using Pythogorous Theorem $AC = \sqrt{AB^2+BC^2}$

D_4 Distance (City-Based Distance) between p and q is defined as $D_4(p,q) = |x-s| + |y-t|$

Chess Board Distance = $D_8(p,q) = \max(|x-s|,|y-t|)$

D_m (m-Path distance) for a given Set V{} = Shortest Path between p, q.

PIXEL Distance Calculations

p(0,0)

2	3	2	6	1
6	1	3	6	2
5	4	2	3	5
2	4	4	5	2
4	5	2	3	6

q(4,4)

Eucledean Distance between p,q = $\sqrt{(4-0)^2+(4-0)^2} = \sqrt{32}$

D_4 Distance = $|4-0| + |4-0| = 8$

D_8 Distance = $\max(|4-0| + |4-0|) = 4$

Interested set of Image Pixels Intensitties V={2,3,5,6}

D_m Distance = Shortest of {2,3,3,3,2,6}, {2,6,5,2,5,2,3,6}

PIXEL Distance Calculations

p(0,0)

2	3	2	6	1
6	1	3	6	2
5	4	2 (2,2)	3	5
2	4	4	5	2
4	5	2	3	6

q(4,4)

All pixels of having D_4 Distance of <= 1from P(2,2) ?
All pixels of having D_8 Distance of <= 2 from P(2,2) ?

All pixels of having D_4 Distance of <= 2 from $P_{(2,2)}$?

p(0,0)

2	3	2	6	1
6	1	3	6	2
5	4	2 (2,2)	3	5
2	4	4	5	2
4	5	2	3	6

q(4,4)

All pixels of having D_4 Distance of <= 2 from (2,2) ?

D_4 Distance <=1?

All the Pixels in the Diamond shape form the Contors of the Constant Distance

All pixels of having D_8 Distance of <= 2 from (2,2) ?

p(0,0)

2	3	2	6	1
6	1	3	6	2
5	4	2 (2,2)	3	5
2	4	4	5	2
4	5	2	3	6

q(4,4)

All the Pixels in the Square shape form the Contors of the Constant Distance.

MATHEMATICAL OPERATIONS ON PIXELS or IMAGES

Array Operation = Multiplication

$$\begin{bmatrix} a11 & a12 \\ a21 & a22 \end{bmatrix} * \begin{bmatrix} b11 & b12 \\ b21 & b22 \end{bmatrix} = \begin{bmatrix} a11*b11 & a12*b12 \\ a21*b21 & a22*b22 \end{bmatrix}$$

MATHEMATICAL OPERATIONS ON PIXELS or IMAGES

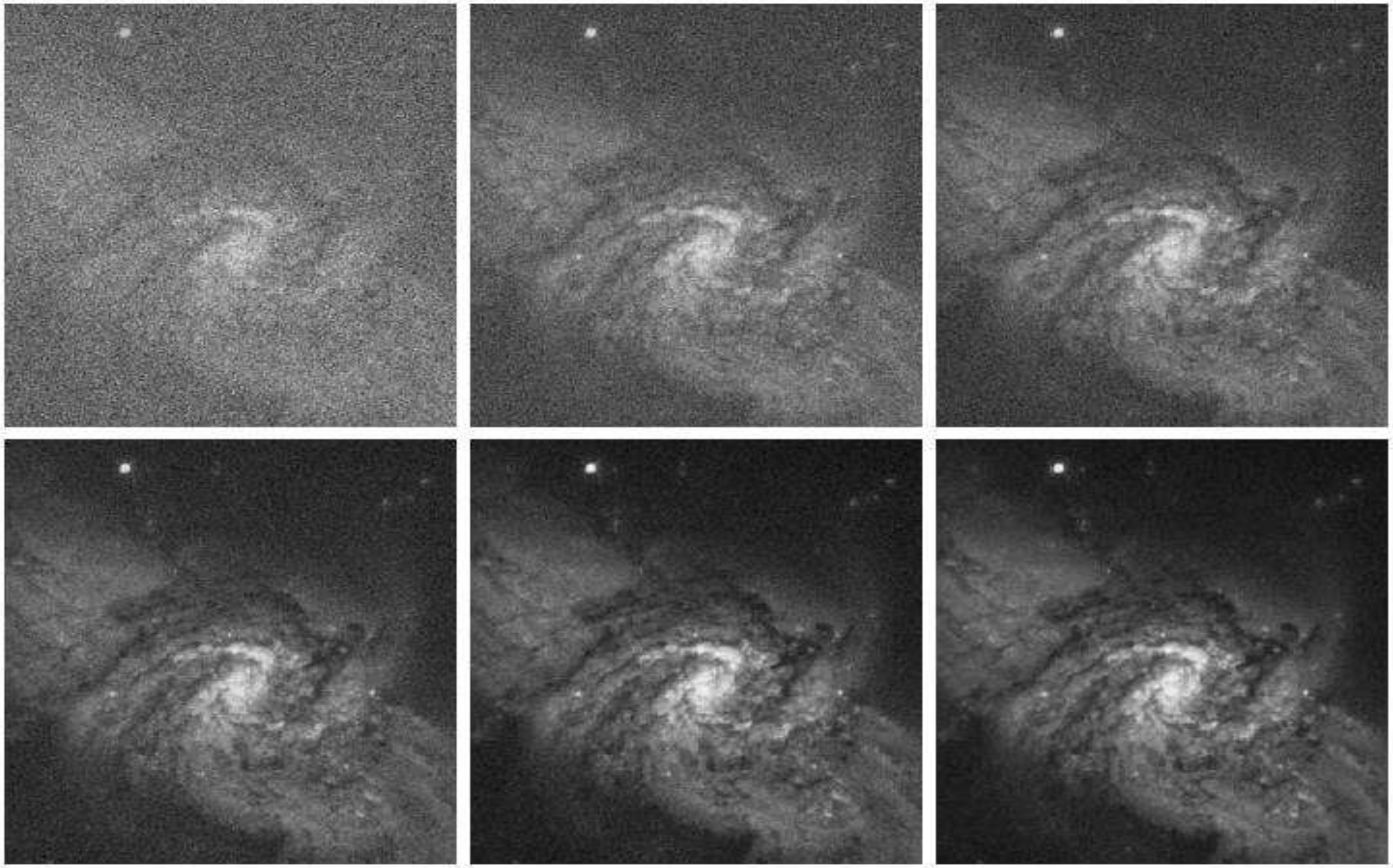
Image Addition:

$$\begin{bmatrix} a11 & a12 \\ a21 & a22 \end{bmatrix} + \begin{bmatrix} b11 & b12 \\ b21 & b22 \end{bmatrix} = \begin{bmatrix} a11+b11 & a12+b12 \\ a21+b21 & a22+b22 \end{bmatrix}$$

Use of Image Addition?

1. Addition of Noise images to reduce Noise
2. Image Averaging in the Fileld of Astronomy

MATHEMATICAL OPERATIONS ON PIXELS or IMAGES



a b c
d e f

FIGURE 2.26 (a) Image of Galaxy Pair NGC 3314 corrupted by additive Gaussian noise. (b)–(f) Results of averaging 5, 10, 20, 50, and 100 noisy images, respectively. (Original image courtesy of NASA.)

MATHEMATICAL OPERATIONS ON PIXELS or IMAGES

Image Addition:

An important application of image averaging is in the field of astronomy, where imaging under very low light levels frequently causes sensor noise to render single images virtually useless for analysis.

Figure 2.26(a) in previous SLIDE shows an 8-bit image in which corruption was simulated by adding to it Gaussian noise with zero mean and a standard deviation of 64 intensity levels.This image, typical of noisy images taken under low light conditions, is useless for all practical purposes.

Figures 2.26(b) through (f) show the results of averaging 5, 10, 20, 50, and 100 images, respectively. We see that the result in Fig. 2.26(e), obtained with is reasonably clean. The image Fig. 2.26(f), resulting from averaging 100 noisy images, is only a slight improvement over the image in Fig. 2.26(e).

Image Addition: (PAPER Writing Purpose)

Let g(x,y) denote a corrupted image formed by the addition of noise,n(x,y) to a noiseless image f(x,y) ; that is, $g(x,y) = f(x,y) + ή(x,y)$ ← (2.6-4)
where the assumption is that at every pair of coordinates *(x, y) the noise is uncorrelated†* and has zero average value. The objective of the following procedure is to reduce the noise content by adding a set of noisy images, $\{g_i(x,yi)\}$. This is a technique used frequently for image enhancement. If the noise satisfies the constraints just stated, it can be shown (Problem 2.20) that if an image is formed by averaging *K different noisy images,*

$g(x, y) = 1/K \sum^{k}_{i=0} g_i(x,y)$ (2.6-5)

then it follows that

$E\{\acute{g}(x, y)\} = f(x, y)$ (2.6-6)

and

$\sigma^{2/g(x,y)} = 1/K\, \sigma^2_{g(x,y)}$ (2.6-7)

where is the expected value of and and are the variances of and respectively, all at coordinates *(x, y).The standard deviation* (square root of the variance) at any point in the average image is

$\sigma^{2/g(x,y)} = 1/\sqrt{K}\, \sigma^2_{g(x,y)}$ (2.6-8)

As *K increases, Eqs. (2.6-7) and (2.6-8) indicate that the variability (as measured* by the variance or the standard deviation) of the pixel values at each location decreases. Because this means that approaches as the number of noisy images used in the averaging process increases. In practice, the images must be *registered (aligned) in order to* avoid the introduction of blurring and other artifacts in the output image.

Image Subtraction:

$$\begin{bmatrix} a11 & a12 \\ a21 & a22 \end{bmatrix} - \begin{bmatrix} b11 & b12 \\ b21 & b22 \end{bmatrix} = \begin{bmatrix} a11-b11 & a12-b12 \\ a21-b21 & a22-b22 \end{bmatrix}$$

Use of Image Subtraction?

1. Image subtraction is used in the enhancement of *differences between images.*
2. *Mask mode image enhancement in Medical Images*

MATHEMATICAL OPERATIONS ON PIXELS or IMAGES

Image Subtraction:

The image in Fig. 2.27(b) was obtained by setting to zero the least-significant bit of every pixel in Fig. 2.27(a). Visually, these images are indistinguishable. Fig. 2.27(c) shows, subtracting one image from the other clearly shows their differences. Black (0) values in this difference image indicate locations where there is no difference between the images in Figs. 2.27(a) and (b).

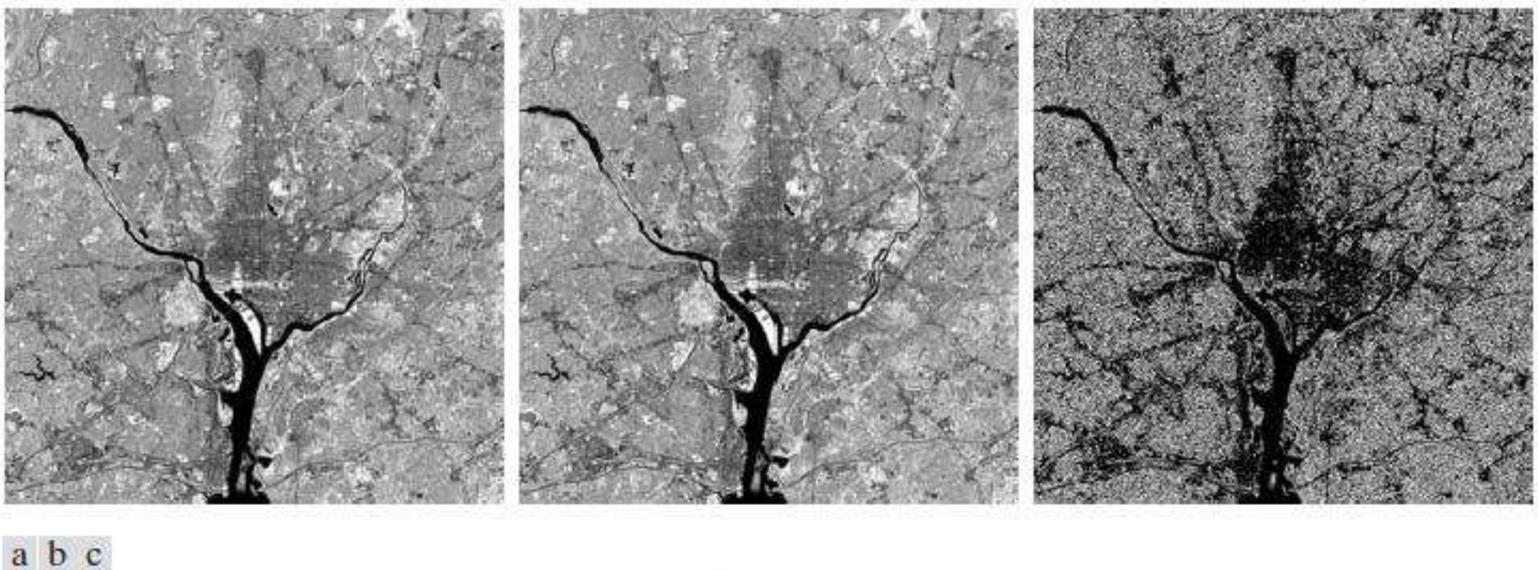
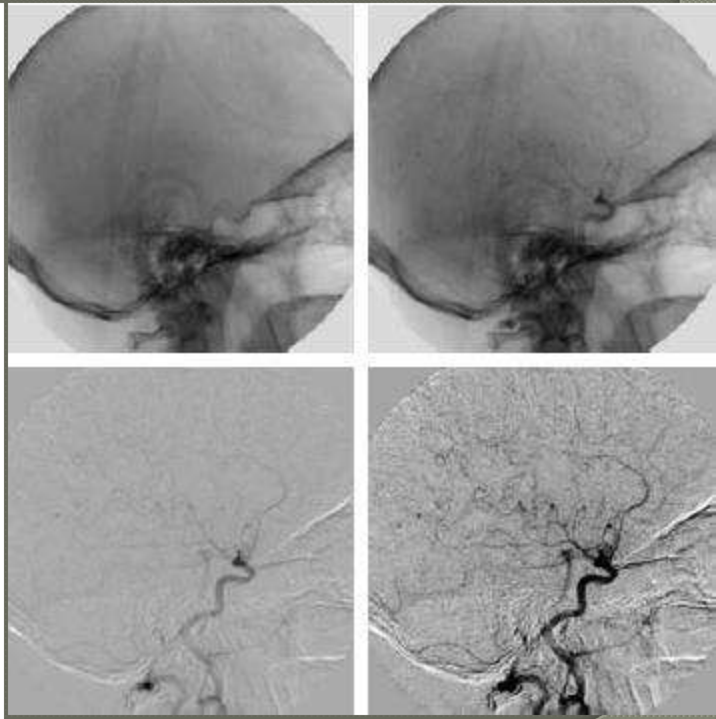


FIGURE 2.27 (a) Infrared image of the Washington, D.C. area. (b) Image obtained by setting to zero the least significant bit of every pixel in (a). (c) Difference of the two images, scaled to the range [0, 255] for clarity.

MATHEMATICAL OPERATIONS ON PIXELS or IMAGES

The *mask, is an X-ray image of a region of a patient's body* captured by an intensified TV camera (instead of traditional X-ray film) located opposite an X-ray source. The procedure consists of injecting an X-ray contrast medium into the patient's blood stream, taking a series of images called *live images of the same anatomical* region as taken early and subtracting the mask from the series of incoming live images after injection of the contrast medium. The net effect of subtracting the mask from each sample live image is that the areas that are different between and appear in the output image, as enhanced detail.

Image Subtraction: (*mask mode radiography)*



a b
c d

FIGURE 2.28 Digital subtraction angiography. (a) Mask image. (b) A live image. (c) Difference between (a) and (b). (d) Enhanced difference image. (Figures (a) and (b) courtesy of The Image Sciences Institute, University Medical Center, Utrecht, The Netherlands.)

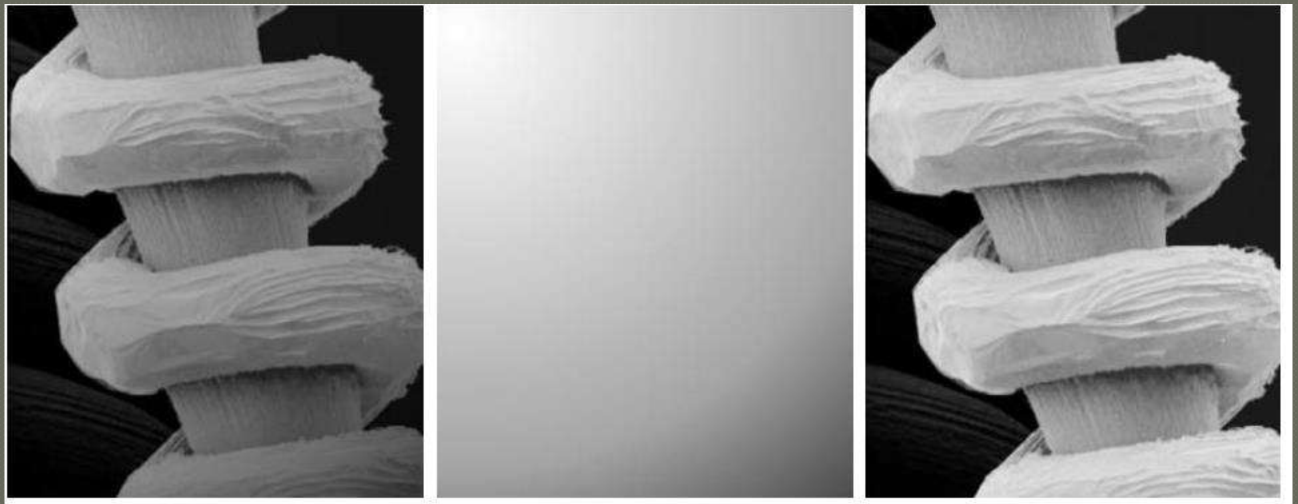
Image Multiplication/Division:

$$\begin{bmatrix} a11 & a12 \\ a21 & a22 \end{bmatrix} * \begin{bmatrix} b11 & b12 \\ b21 & b22 \end{bmatrix} = \begin{bmatrix} a11*b11 & a12*b12 \\ a21*b21 & a22*b22 \end{bmatrix}$$

Use of Image Multiplication?

1. Shading Correction.
2. *Masking (or) Region of Interest operations*

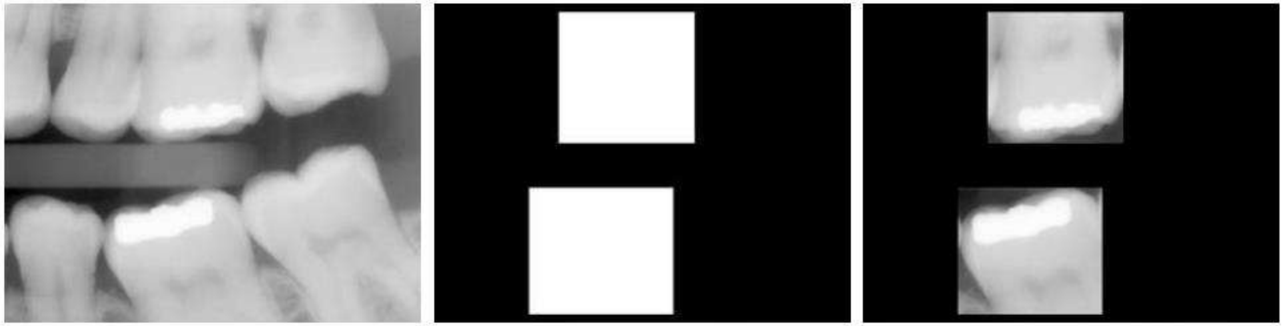
MATHEMATICAL OPERATIONS ON PIXELS or IMAGES



a b c

FIGURE 2.29 Shading correction. (a) Shaded SEM image of a tungsten filament and support, magnified approximately 130 times. (b) The shading pattern. (c) Product of (a) by the reciprocal of (b). (Original image courtesy of Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

MATHEMATICAL OPERATIONS ON PIXELS or IMAGES



a b c

FIGURE 2.30 (a) Digital dental X-ray image. (b) ROI mask for isolating teeth with fillings (white corresponds to 1 and black corresponds to 0). (c) Product of (a) and (b).

LOGICAL OPERATIONS ON PIXES (AND/OR/NOT/XNOR/XOR)

For Binary Images: Intensity values are 1 or 0
Logical operations are applicable easily

a b c
d e

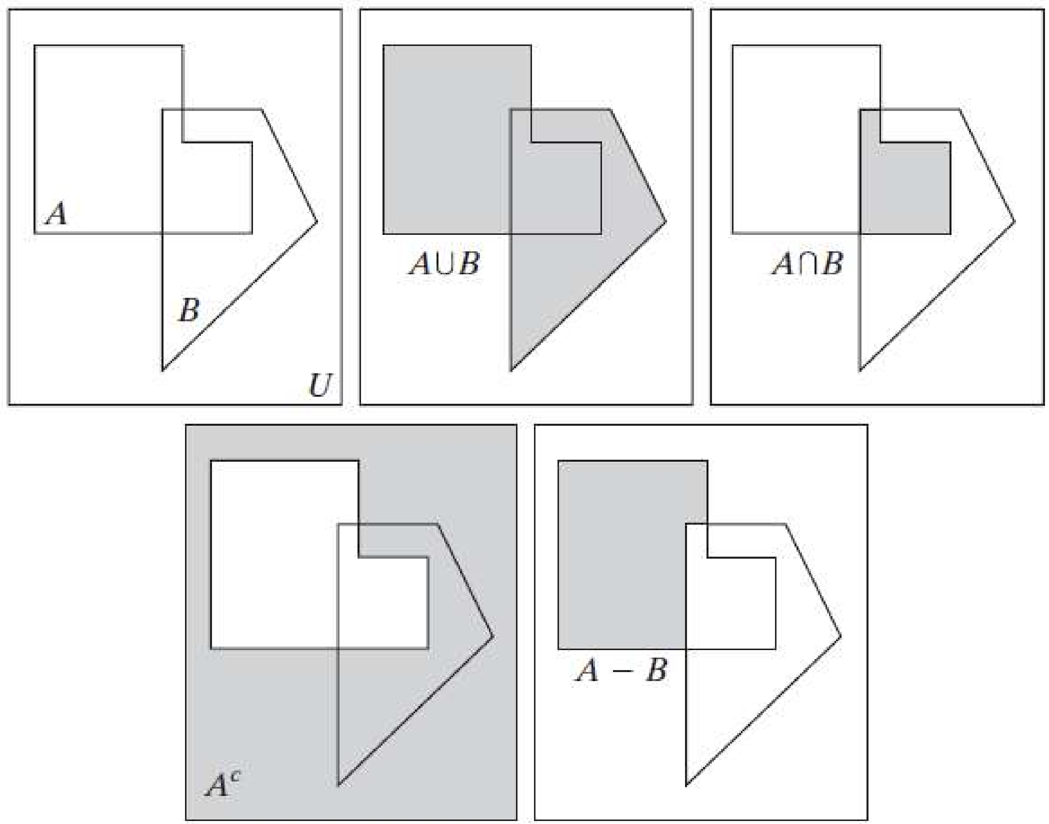


FIGURE 2.31 (a) Two sets of coordinates, A and B, in 2-D space. (b) The union of A and B. (c) The intersection of A and B. (d) The complement of A. (e) The difference between A and B. In (b)–(e) the shaded areas represent the members of the set operation indicated.

LOGICAL OPERATIONS ON PIXES (AND/OR/NOT/XNOT/XOR)

For Gray Scale Images How does Logical Operations can be applied?

LOGICAL OPERATIONS ON PIXES (AND/OR/NOT/XNOT/XOR)

For Gray Scale Images How does Logical Operations can be applied?

PLEASE THINK

LOGICAL OPERATIONS ON PIXES (AND/OR/NOT/XNOT/XOR)

For Gray Scale Images How does Logical Operations can be applied?

gray-scale image be represented by a set *A whose* elements are triplets of the form f(x,y,z) , where *x and y are spatial coordinates* and *z denotes intensity.*

$A^c = \{(x, y, K - z) \mid (x, y, z) \varepsilon A\}$, which simply denotes the set of pixels of *A whose intensities have been subtracted* from a constant *K.* where K *is the number of* intensity bits used to represent *z.*

LOGICAL OPERATIONS ON PIXES (AND/OR/NOT/XNOT/XOR)

For Gray Scale Images How does Logical Operations can be applied? (NOT Operation)

For Example: Let *A denote the 8-bit gray-scale image in* Fig. 2.32(a), and suppose that we want to form the negative of *A using set* operations. We simply form the set $A^c = \{(x, y, 255 - z) \mid (x, y, z) \varepsilon A\}$

LOGICAL OPERATIONS ON PIXES (AND/OR/NOT/XNOT/XOR)

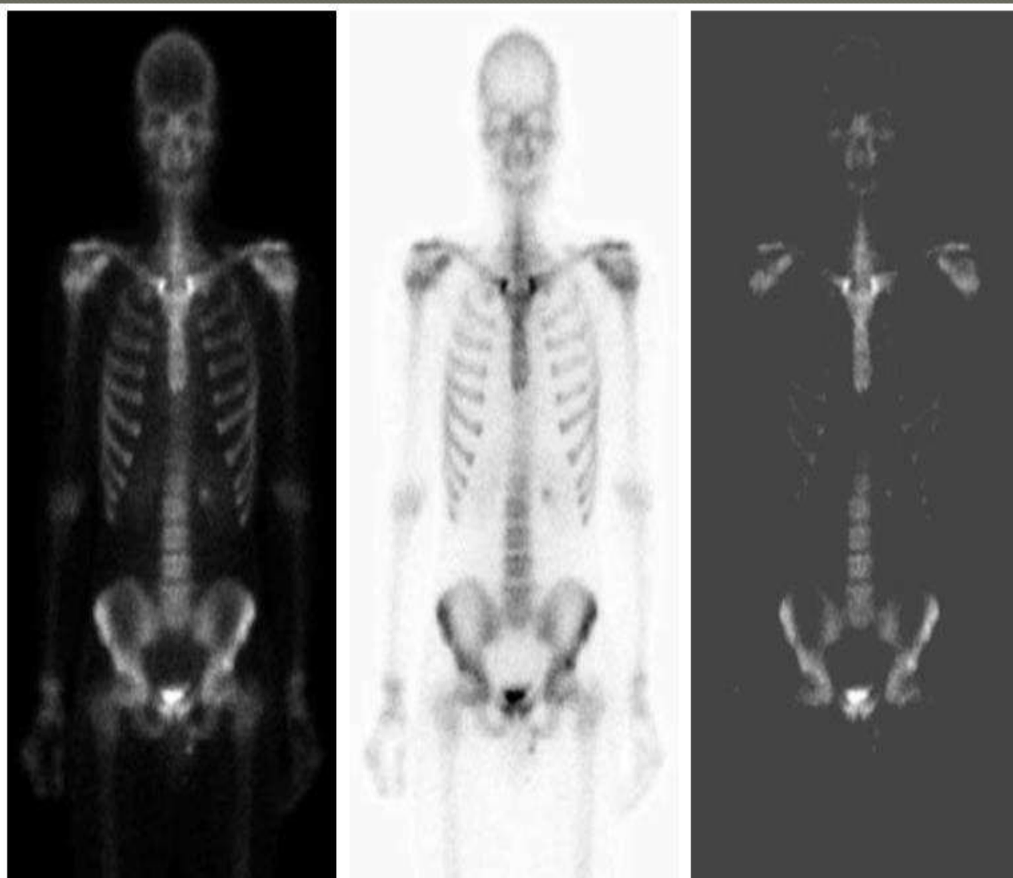
For Gray Scale Images How does Logical Operations can be applied? (UNION Operation)

For Example: The union of two gray-scale sets *A and B may be defined as the set* $A \cup B = \{\max(a,b) \mid a \in A, b \in B\}$

The union of two gray-scale sets (images) is an array formed from the maximum intensity between pairs of spatially corresponding elements

LOGICAL OPERATIONS ON PIXES (AND/OR/NOT/XNOT/XOR)

For Example: The union of two gray-scale sets *A and B may be defined as the set* A U B = {max(a,b) | aεA, bεB } have value which is a mid-gray value.



a b c

FIGURE 2.32 Set operations involving gray-scale images. (a) Original image. (b) Image negative obtained using set complementation. (c) The union of (a) and a constant image. (Original image courtesy of G.E. Medical Systems.)

Suppose that *A represents* the image in Fig. 2.32(a), and let *B denote a Negative obtained using Set complementation of A Figure 2.32(c) shows the result of performing* the set union, of A U B.

LOGICAL OPERATIONS ON BINARY IMAGES

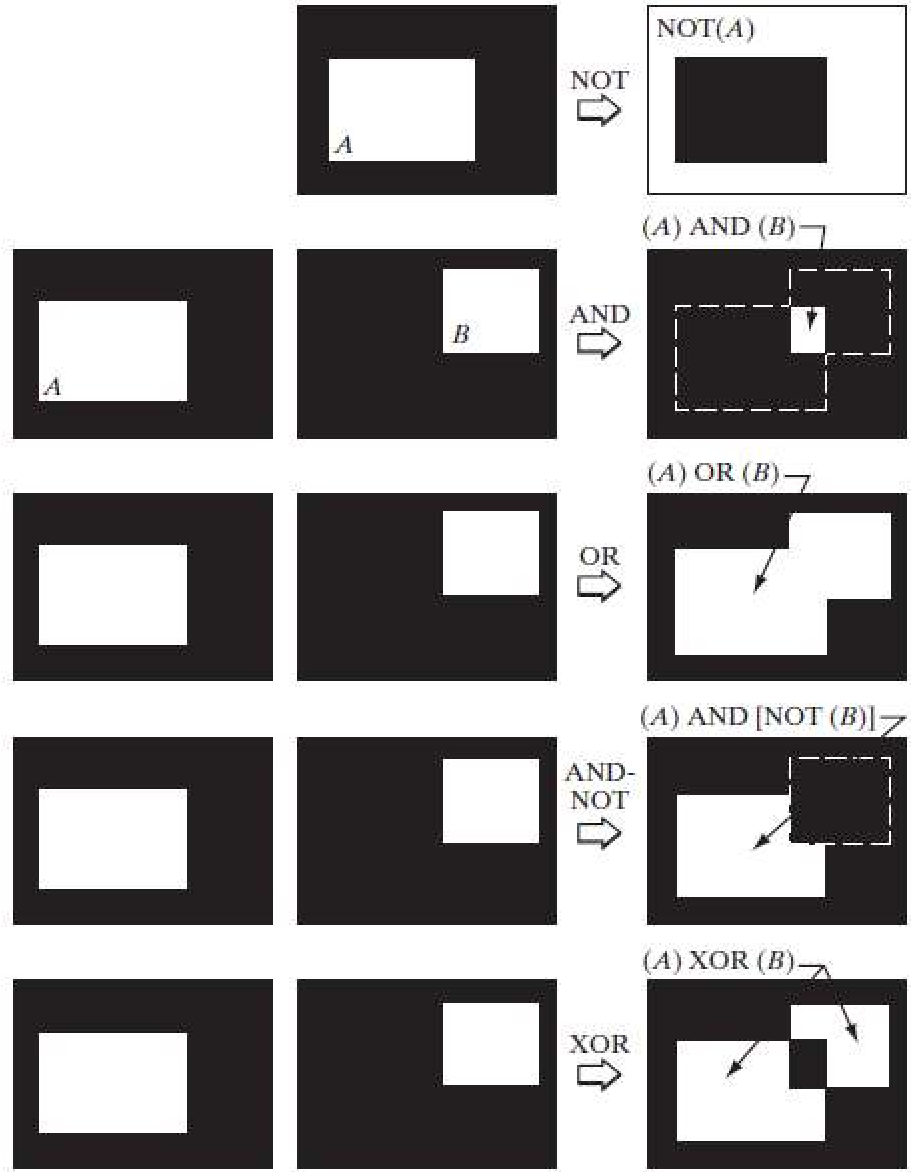


FIGURE 2.33 Illustration of logical operations involving foreground (white) pixels. Black represents binary 0s and white binary 1s. The dashed lines are shown for reference only. They are not part of the result.

SPATIAL OPERATIONS ON PIXELS

1.) Single Pixel Operation (1 Pixel)–
2.) Neighborhood Operation (1 Pixel & neighbors)
3.) Geometric Spatial transformations (1 Pixel with spatial and Intensity change)

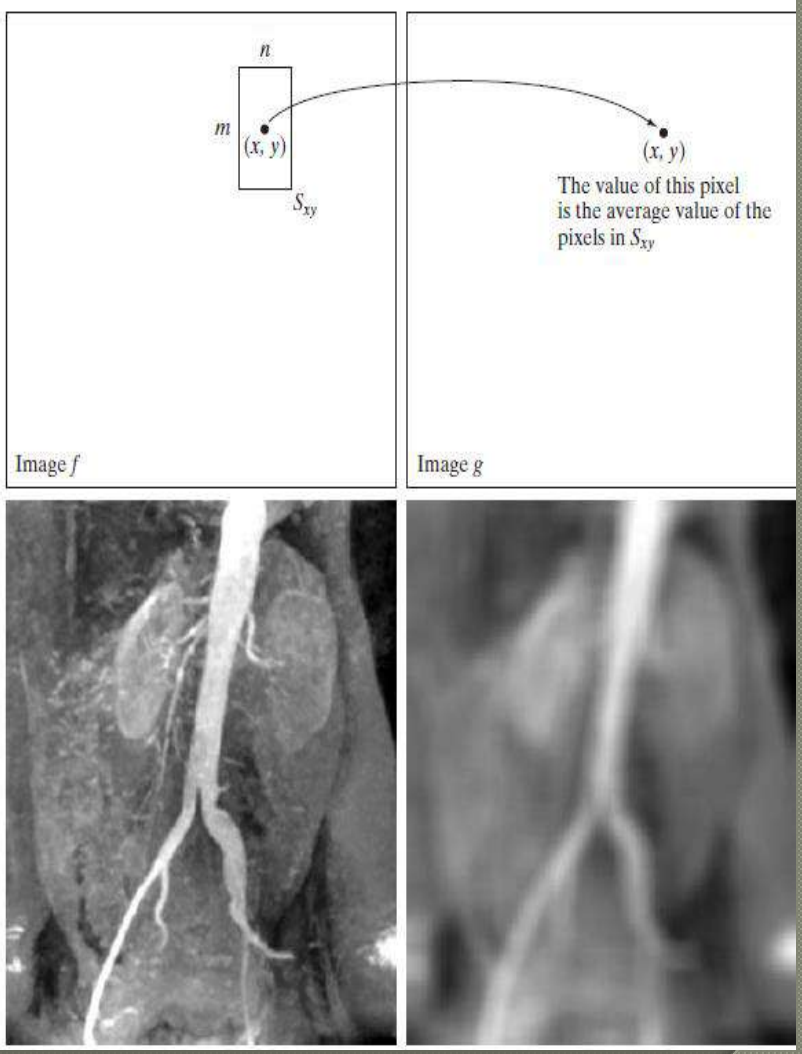
Single Pixel Operation: To alter the values of its individual pixels based on their intensity.
Ex: Get a Negative of the Original Image

SPATIAL OPERATIONS ON PIXELS

Neighborhood Operation: Let S_{xy} denote the set of coordinates of a neighborhood centered on an arbitrary point *(x, y) in an image, f. Neighborhood processing generates a corresponding* pixel at the same coordinates in an output (processed) image, *g, such* that the value of that pixel is determined by a specified operation involving the pixels in the input image with coordinates in S_{xy}.

a b
c d

FIGURE 2.35 Local averaging using neighborhood processing. The procedure is illustrated in (a) and (b) for a rectangular neighborhood. (c) The aortic angiogram discussed in Section 1.3.2. (d) The result of using Eq. (2.6-21) with $m = n = 41$. The images are of size 790×686 pixels.



SPATIAL OPERATIONS ON PIXELS

Geometric Spatial transformations:

A geometric transformation consists of two basic operations:

(1) a spatial transformation of coordinates and

(2) Intensity interpolation that assigns intensity values to the spatially transformed pixels

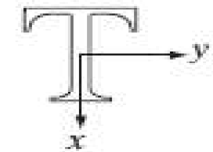
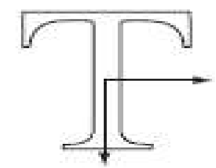
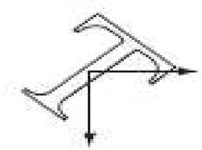
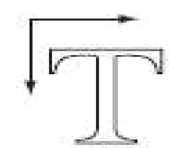
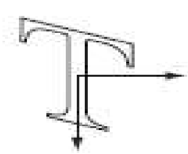
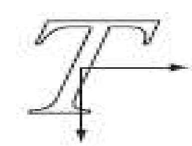
(3) Eg: Given Below Matrix Multiplication Equation used

$$[x \;\; y \;\; 1] = [v \;\; w \;\; 1]\,\mathbf{T} = [v \;\; w \;\; 1]\begin{bmatrix} t_{11} & t_{12} & 0 \\ t_{21} & t_{22} & 0 \\ t_{31} & t_{32} & 1 \end{bmatrix} \qquad (2.6\text{-}23)$$

This transformation can scale, rotate, translate, or sheer a set of coordinate points, depending on the value chosen for the elements of matrix **T**

GEOMETRIC SPATIAL OPERATIONS ON PIXELS

TABLE 2.2
Affine transformations based on Eq. (2.6-23).

Transformation Name	Affine Matrix, T	Coordinate Equations	Example
Identity	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v$ $y = w$	
Scaling	$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = c_x v$ $y = c_y w$	
Rotation	$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v\cos\theta - w\sin\theta$ $y = v\cos\theta + w\sin\theta$	
Translation	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{bmatrix}$	$x = v + t_x$ $y = w + t_y$	
Shear (vertical)	$\begin{bmatrix} 1 & 0 & 0 \\ s_v & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v + s_v w$ $y = w$	
Shear (horizontal)	$\begin{bmatrix} 1 & s_h & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v$ $y = s_h v + w$	

SPATIAL OPERATIONS ON PIXELS

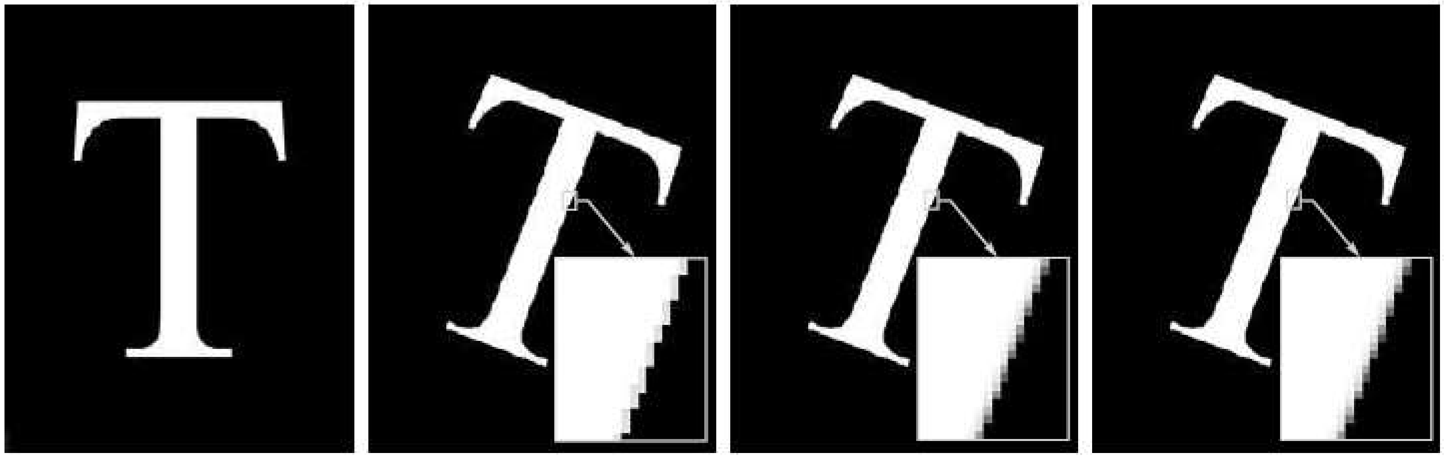
Geometric Spatial transformations:

Methods of AFFINE TRANFORMATIONS

1.) Forward Mapping

2.) Inverse Mapping : $(v, w) = T^{-1}(x, y)$.

Rest of pixels we do Interpolation method to fill.



a b c d

FIGURE 2.36 (a) A 300 dpi image of the letter T. (b) Image rotated 21° using nearest neighbor interpolation to assign intensity values to the spatially transformed pixels. (c) Image rotated 21° using bilinear interpolation. (d) Image rotated 21° using bicubic interpolation. The enlarged sections show edge detail for the three interpolation approaches.