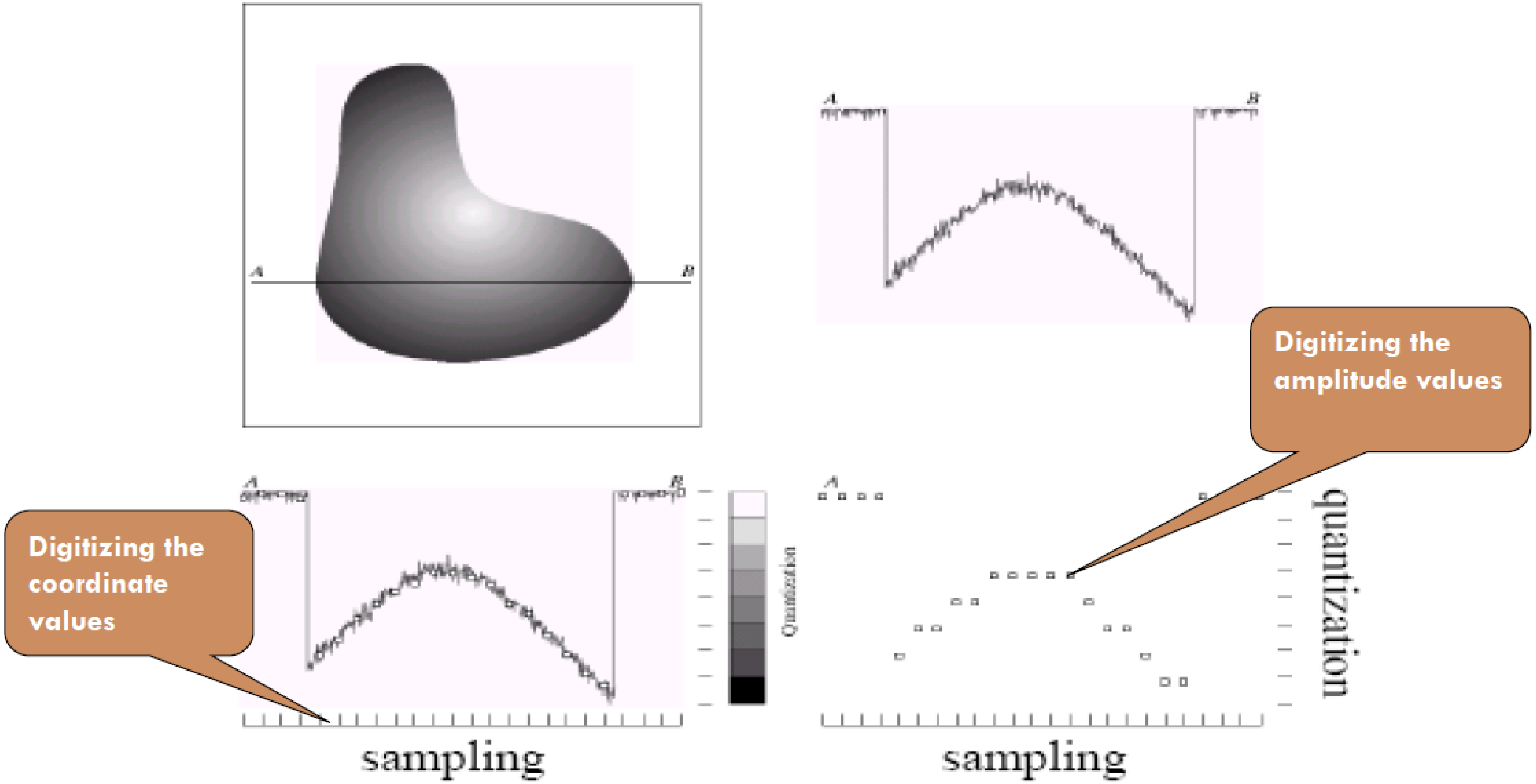
Digital Image Processing

Image Fundamentals – Chapter 2

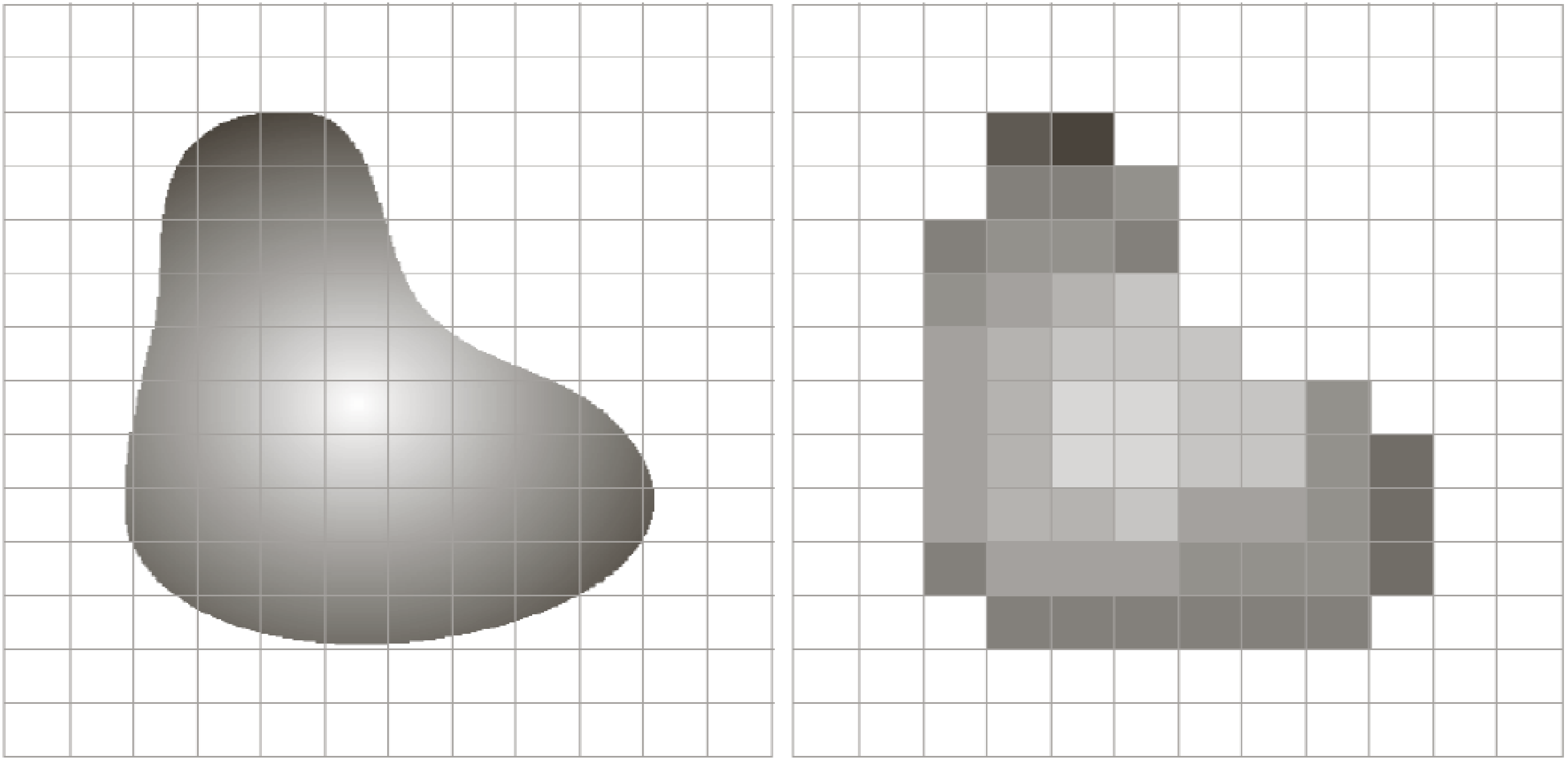
Sampling and Quantization

- Sampling: Digitization of the spatial coordinates (*x*,*y*)
- Quantization: Digitization in amplitude (also called *gray level quantization*)
- 8 bit quantization: 2^8 =256 gray levels (0: black, 255: white)
- Binary (1 bit quantization):2 gray levels (0: black, 1: white)
- Commonly used number of samples (resolution)
- Digital still cameras: 640x480, 1024x1024,up to 4064 x 2704
- Digital video cameras: 640x480 at 30 frames/second and higher

Sampling and Quantization



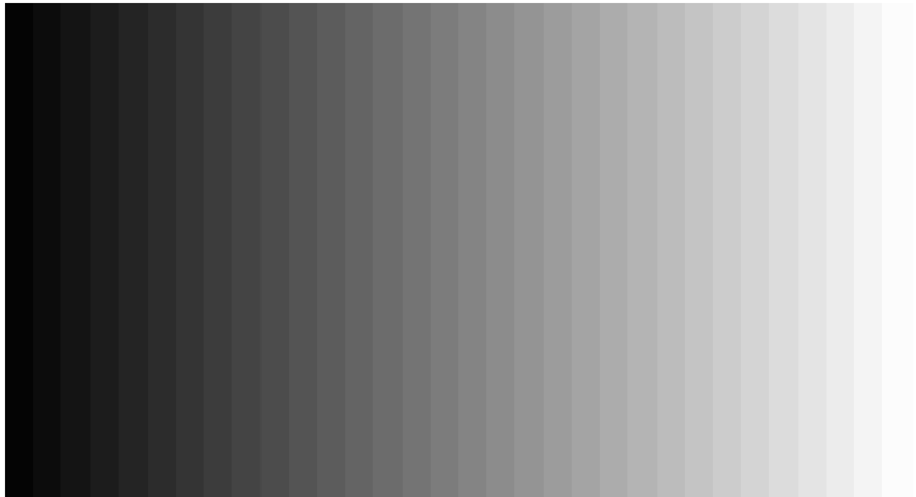
Sampling and Quantization



a b

Spatial and Gray level Resolution

- **Spatial resolution** — Resolution in Space
  - A measure of the smallest discernible detail in an image stated with *dots (pixels) per unit distance, dots per inch (dpi)*
  - *No. of pixels specifies the spatial resolution*
- **Intensity or Gray level resolution** — Resolution in Gray values
  - The smallest discernible change in intensity or gray level stated with *8 bits, 16 bits, etc.*



Spatial and Gray level Resolution

- The Digitization process requires to determine M, N and L

- M and N are Spatial Resolution
- L = Gray level Resolution
  - $L = 2^k$, where L represents Gray level
- Image Storage = Spatial Resolution * Gray level Resolution
- The no. of bits required to store the image is::
  - $b = M \times N \times k$ or $b = N^2 \times k$
- Sampling --→ Spatial Resolution
- Quantization --→ Gray level Resolution

Spatial and Gray level Resolution

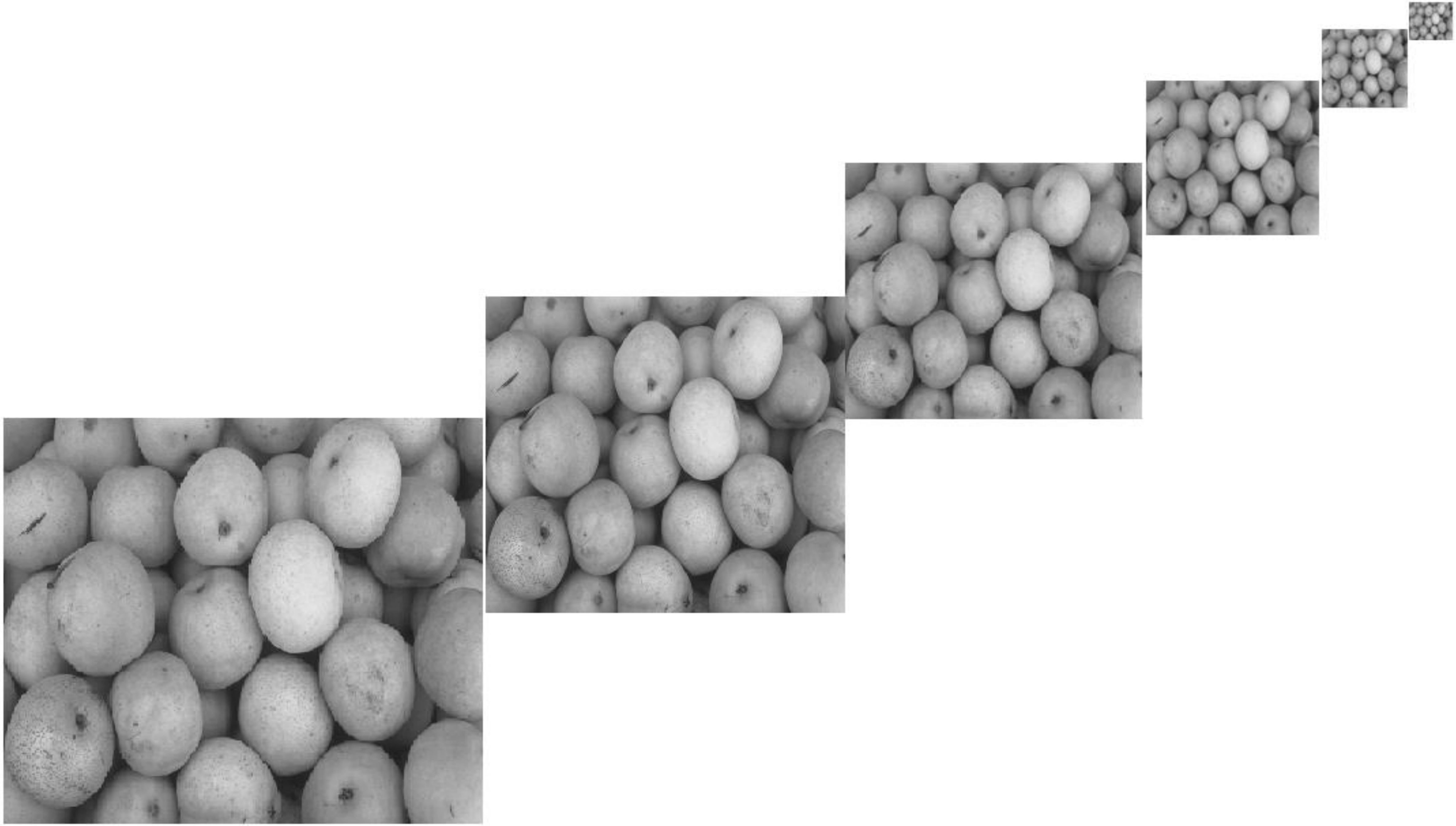
TABLE 2.1

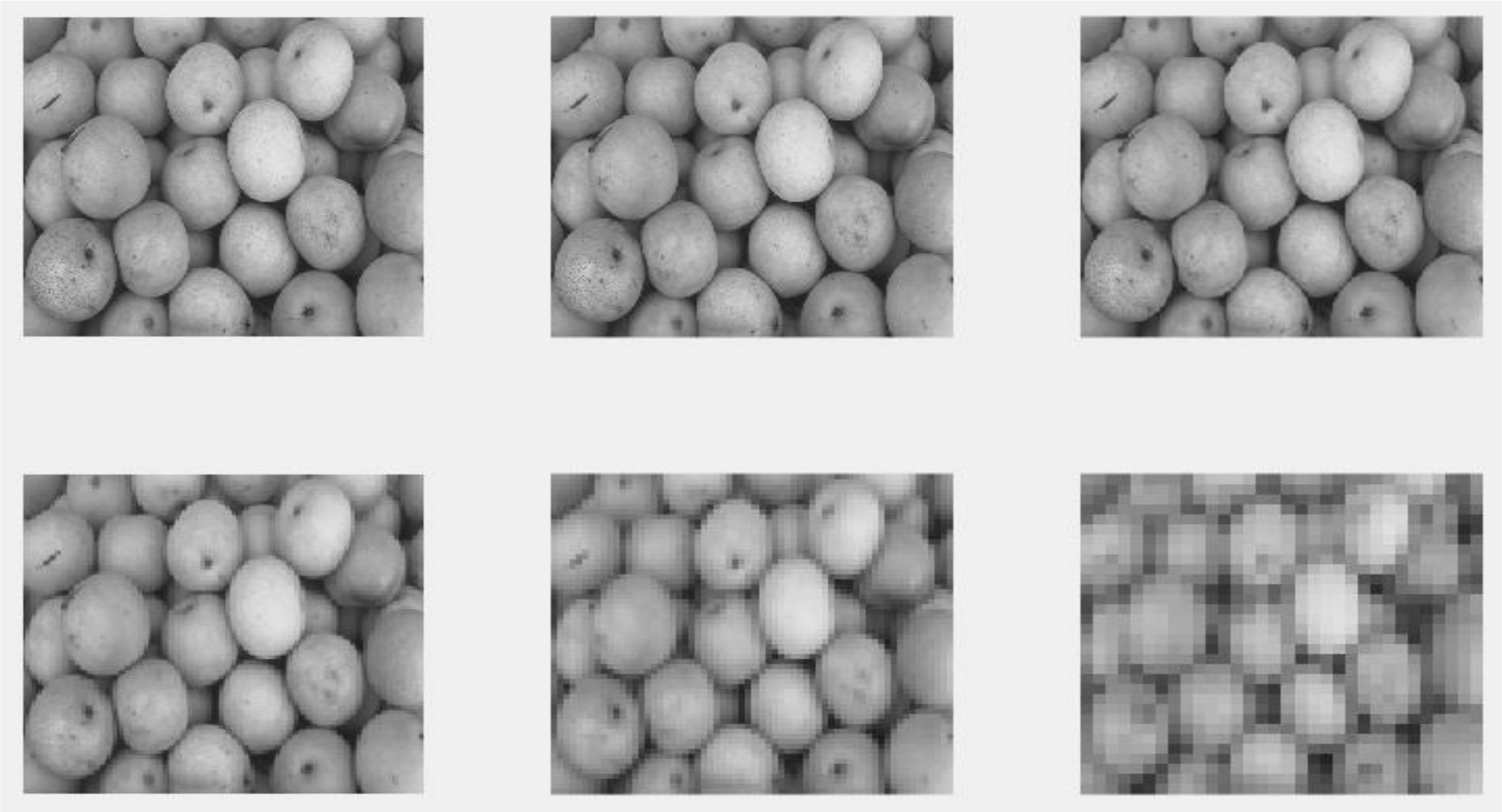
Number of storage bits for various values of N and k.

N/k	1 (L = 2)	2 (L = 4)	3 (L = 8)	4 (L = 16)	5 (L = 32)	6 (L = 64)	7 (L = 128)	8 (L = 256)
32	1,024	2,048	3,072	4,096	5,120	6,144	7,168	8,192
64	4,096	8,192	12,288	16,384	20,480	24,576	28,672	32,768
128	16,384	32,768	49,152	65,536	81,920	98,304	114,688	131,072
256	65,536	131,072	196,608	262,144	327,680	393,216	458,752	524,288
512	262,144	524,288	786,432	1,048,576	1,310,720	1,572,864	1,835,008	2,097,152
1024	1,048,576	2,097,152	3,145,728	4,194,304	5,242,880	6,291,456	7,340,032	8,388,608
2048	4,194,304	8,388,608	12,582,912	16,777,216	20,971,520	25,165,824	29,369,128	33,554,432
4096	16,777,216	33,554,432	50,331,648	67,108,864	83,886,080	100,663,296	117,440,512	134,217,728
8192	67,108,864	134,217,728	201,326,592	268,435,456	335,544,320	402,653,184	469,762,048	536,870,912

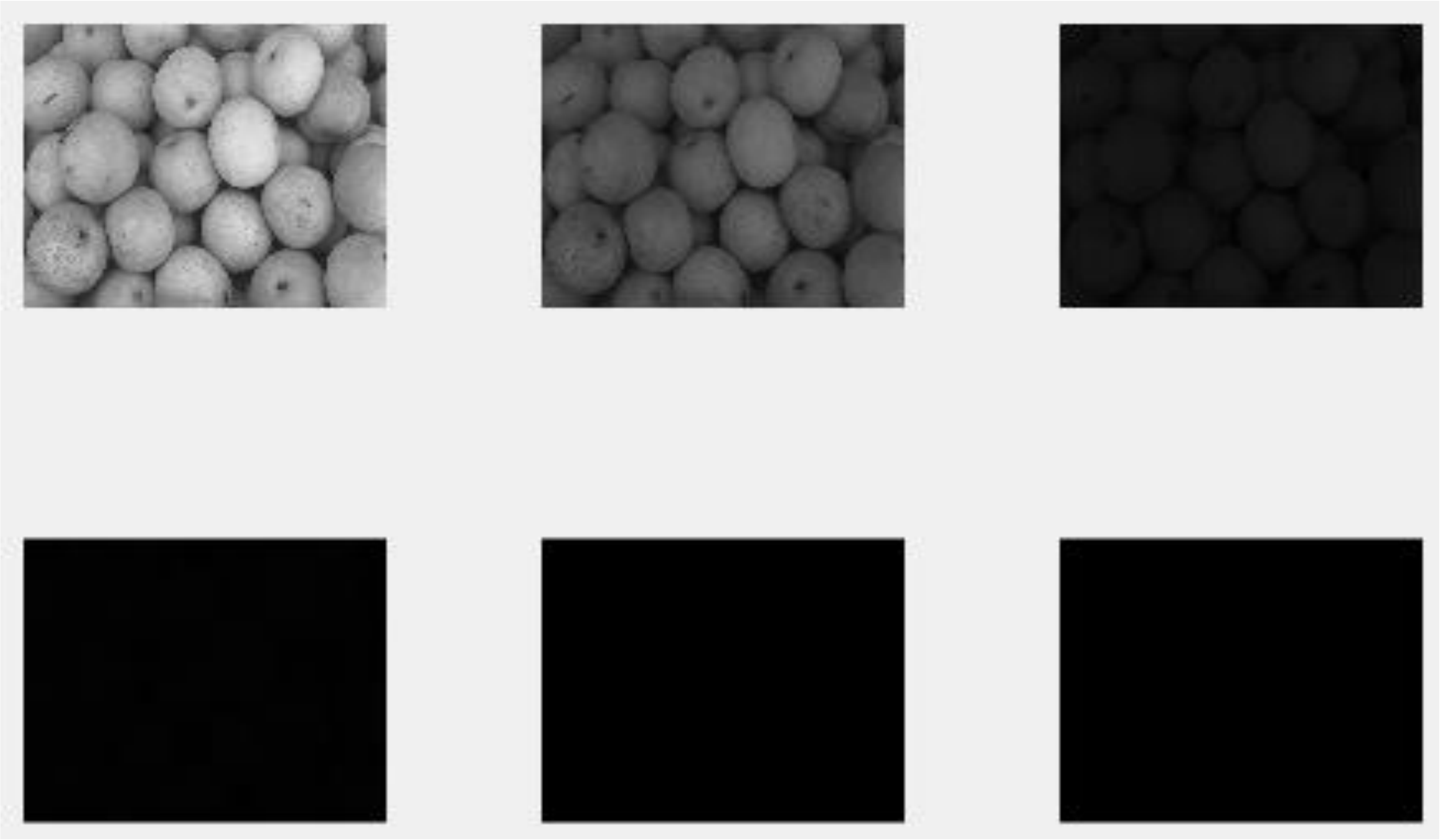
Spatial and Gray level Resolution

Variation in Spatial Resolution from 1024 x 1024 to 32 x 32





Quantization



Spatial and Gray level Resolution

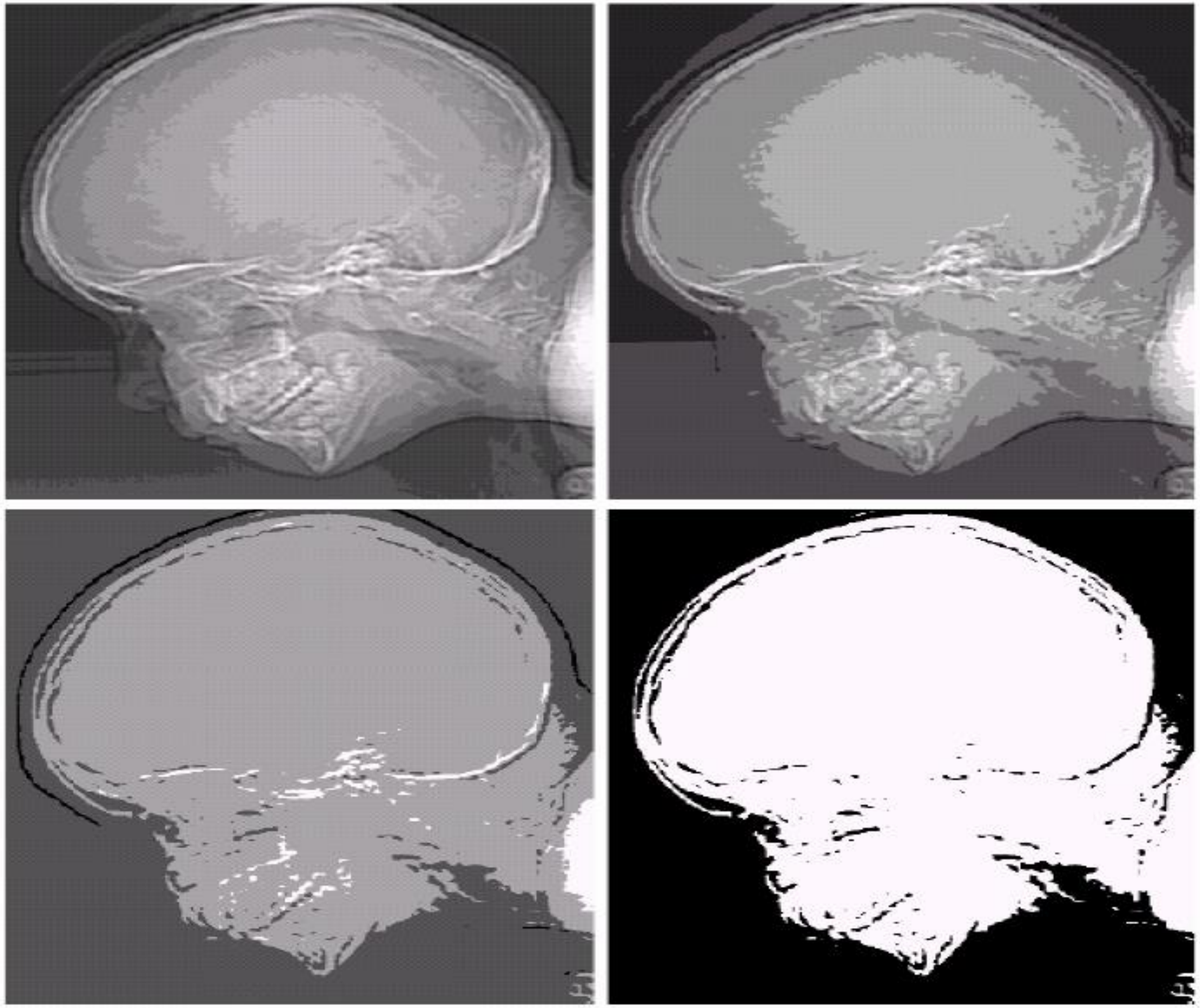
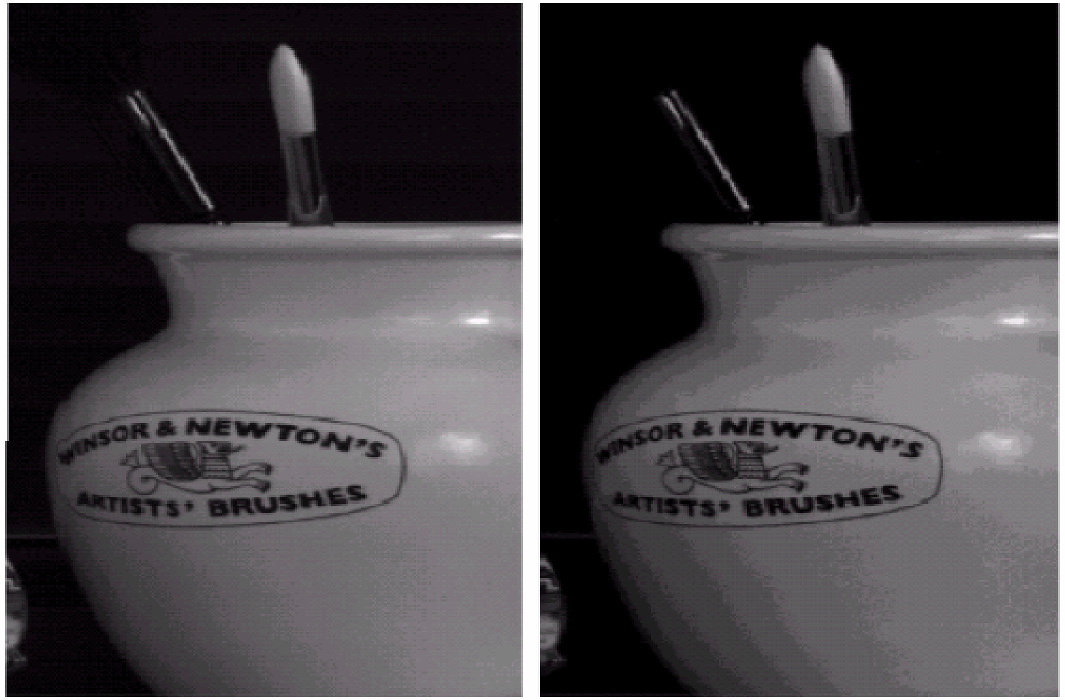


FIGURE 2.21 *(Continued)* (e)–(h) Image displayed in 16, 8, 4, and 2 gray levels. (Original courtesy of Dr. David R. Pickens, Department of Radiology & Radiological Sciences, Vanderbilt University Medical Center.)

Spatial and Gray level Resolution

- The representation of an image with insufficient number of gray levels produces false edges or boundaries in an image, a phenomenon known as False Contouring or Contouring defect

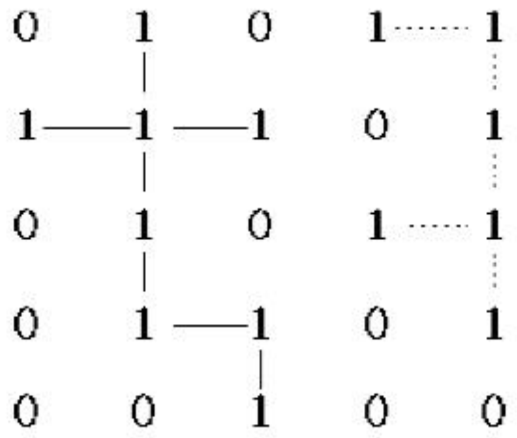
- False Contouring is quite visible in images displayed using 16 or less gray levels as shown in the images of the skeleton in the previous slide



Contouring defect

Basic Relationship between Pixels

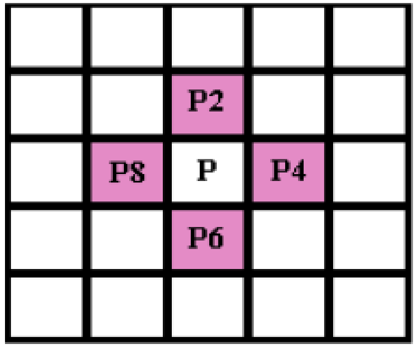
- Neighborhood
- Connectivity
- Adjacency
- Paths
- Regions and boundaries



Pixels Neighborhood

- Neighbors of a pixel p at coordinates (x,y)
  - 4-neighbors of p, denoted by $N_4(p)$:

    (x-1, y), (x+1, y), (x,y-1), and (x, y+1)



	1	
0	1	0
	0	

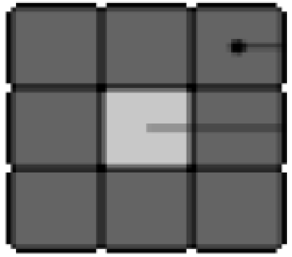
  - 4 diagonal neighbors of p, denoted by $N_D(p)$:

    (x-1, y-1), (x+1, y+1), (x+1,y-1), and (x-1, y+1)

0		1
	1	
0		1

  - 8 neighbors of p, denoted $N_8(p)$

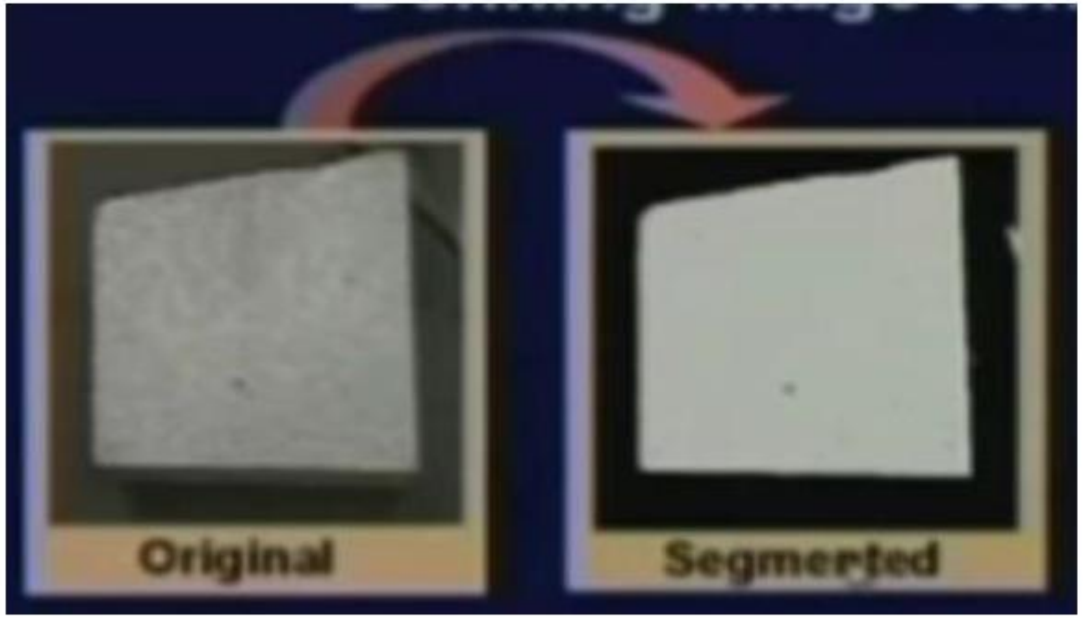
    $N_8(p) = N_4(p) \cup N_D(p)$



0	1	1
0	1	0
0	0	1

Connectivity

- Connectivity between pixels is a very important concept.
- It is very useful for
  - ✓ Establishing object boundaries
  - ✓ Defining image components/regions etc
- If F (x, y) > Th
- (x, y) € object
- Else
- (x, y) € background



Connectivity & Adjacency

- Two pixels are said to be connected if they are adjacent in some sense
  - They are neighbors
  - There intensity values are the similar [gray levels are equal]
    - For example in a binary image with values 0 and 1, two pixels may be 4-neighbors, but they are said to be connected only if they have the same value
- Let V be the set of gray level values used to define adjacency
- In a binary image, V={1} if we are referring to adjacency of pixels with value 1
- In gray scale image the idea is same but set V contains more elements. For example, in the adjacency of pixels with the range of possible gray level values 0 to 255, set V could be any subset of these 256 values

Connectivity & Adjacency

- Let V be the set of intensity values

- **4-adjacency:** Two pixels p and q with values from V are 4-adjacent if q is in the set $N_4(p)$

- **8-adjacency:** Two pixels p and q with values from V are 8-adjacent if q is in the set $N_8(p)$

- **m-adjacency:** Two pixels p and q with values from V are m-adjacent if

  (i) q is in the set $N_4(p)$, or

  (ii) q is in the set $N_D(p)$ and the set $N_4(p) \cap N_4(q)$ has no pixels whose values are from V

  $$q \in N_D(p) \text{ and } N_4(p) \cap N_4(q) = \emptyset$$

- **Mixed adjacency** is a modification of 8-Adjacency. It is introduced to eliminate the ambiguities that often arise when 8-Adjacency is used

Path

- A (digital) path (or curve) from pixel p with coordinates (x, y) to pixel q with coordinates (s, t) is a sequence of distinct pixels with coordinates (x_o, y_o), (x_1, y_1), …, (x_n, y_n)

  Where (xo, yo) = (x , y) , (xn, yn) = (s,t)

  (x_i, y_i) and (x_{i-1}, y_{i-1}) are adjacent for $1 \leq i \leq n$.

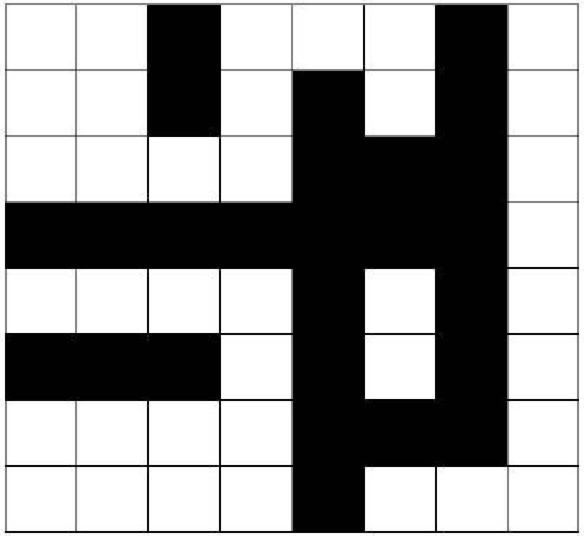
- Here *n* is the *length* of the path.

- If $(x_0, y_0) = (x_n, y_n)$, the path is *closed* path.

- We can define 4-, 8-, and m-paths based on the type of adjacency used.

Connectivity & Adjacency

- Let S represent a subset of pixels in an image. Two pixels p with coordinates (x0, y0) and q with coordinates (xn, yn) are said to be **connected in S** if there exists a path between them consisting entirely of pixels in S



  e.g (x0, y0), (x1, y1), …, (xn, yn)

- Where

$$\forall i, 0 \le i \le n, (x_i, y_i) \in S$$

- For any pixel p in S, the set of pixels that are connected to it in S is called a **connected component** of S

- If it only has one connected component, then set S is called **Connected set**

Examples: Adjacency and Path

V = {1, 2}

0	1	1	0	1	1	0	1	1
0	2	0	0	2	0	0	2	0
0	0	1	0	0	1	0	0	1

Examples: Adjacency and Path

V = {1, 2}

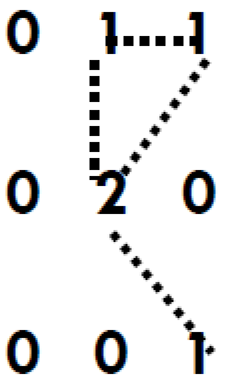
0	1	1		0	1	1		0	1	1
0	2	0		0	2	0		0	2	0
0	0	1		0	0	1		0	0	1

8-adjacent

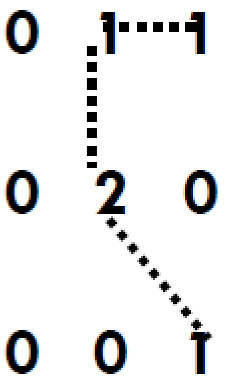
Examples: Adjacency and Path

$V = \{1, 2\}$

0	1	1
0	2	0
0	0	1



8-adjacent

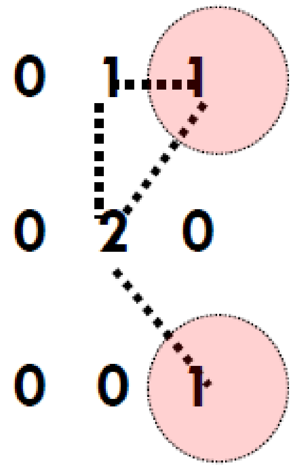


m-adjacent

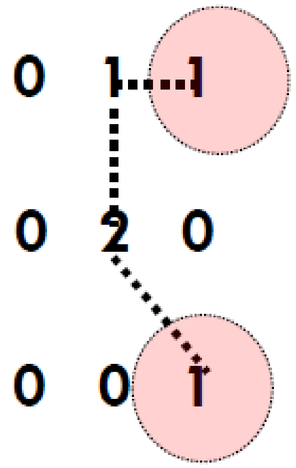
Examples: Adjacency and Path

V = {1, 2}

0	1	1
0	2	0
0	0	1



8-adjacent



m-adjacent

The 8-path from (1,3) to (3,3):

(i) (1,3), (1,2), (2,2), (3,3)

The m-path from (1,3) to (3,3):

(1,3), (1,2), (2,2), (3,3)

Image Operations in Matlab

imread() – reading an image with different postfixes

imresize() – resizing an image to any given size

figure – opening a new graphical window

subplot(#of row, # of col, location) – showing different plots/images in one graphical window

imshow() – displaying an image

Image Operations in Matlab

generating figures of slide 9

```
im=rgb2gray(imread('pears.png'));
im1=imresize(im, [1024 1024]);
im2=imresize(im1, [1024 1024]/2);
im3=imresize(im1, [1024 1024]/4);
im4=imresize(im1, [1024 1024]/8);
im5=imresize(im1, [1024 1024]/16);
im6=imresize(im1, [1024 1024]/32);

figure;imshow(im1)
figure;imshow(im2)
figure;imshow(im3)
figure;imshow(im4)
figure;imshow(im5)
Figure;imshow(im6)
```

Image Operations in Matlab

generating figure of slide 10

```
figure;
subplot(2,3,1);imshow(im1);subplot(2,3,2);imshow(im2)
subplot(2,3,3);imshow(im3);subplot(2,3,4);imshow(im4)
subplot(2,3,5);imshow(im5);subplot(2,3,6);imshow(im6)
```

Image Operations in Matlab

generating figure of slide 11

- im = rgb2gray(imread('pears.png'));
- im1 = imresize(im, [1024 1024]);
- im2 = gray2ind(im1, 2^7);
- im3 = gray2ind(im2, 2^6);
- im4 = gray2ind(im3, 2^5);
- im5 = gray2ind(im4, 2^4);
- im6 = gray2ind(im5, 2^3);
- im7 = gray2ind(im6, 2^2);
- im8 = gray2ind(im7, 2^1);

Image Operations in Matlab

- Creating mirror image
- a=imread('pout.tif');
- [r,c]=size(a);
- for i=1:1:r
    - k=1;
- for j=c:-1:1
    - temp=a(i,k);
    - result(i,k)=a(i,j);
    - result(i,j)=temp;
    - k=k+1;
- end
- end
- subplot(1,2,1),imshow(a)
- subplot(1,2,2),imshow(result)

Output for Slide 29

- Mirror Image Generation

Assignment

- Submission date 10th may
- How to place labels in the subplot. Practice code given in slide 29 with label according o gray level, starting from 8bit image to 1bit
- Write a MATLAB code that reads a gray scale image and generates the flipped image of original image. **Your output should be like the one given below**