Digital Image Processing

Lecture # 4

Digital Image Fundamentals - II
ALI JAVED
Lecturer
SOFTWARE ENGINEERING DEPARTMENT
U.E.T TAXILA

Email:: ali.javed@uettaxila.edu.pk

Office Room #: 7
Presentation Outline

- Image Formation Model
- Illumination Ranges
- Digital Image Representation
- Sampling and Quantization
- Spatial and Gray level Resolution
- Image Scaling
  - Image Zooming
  - Image Shrinking
- Basic relationship between pixels
  - Neighbors
  - Adjacency
  - Connectivity
  - Path
  - Regions and Boundaries
  - Distance between pixels
Object Visibility

- Object becomes visible when illuminating source strikes the objects and due to reflection our eyes can see the object because reflection reaches our eye after striking through object

- Scene Visibility = Reflection from the object, Light Source

- Image = reflectance, illumination
Image Formation Model

• For monochromatic image 2-D array: \( f(x, y) \)
• The \( f(x, y) \) is characterized by two components:
  – The amount of source illumination incident on the scene, i.e., \( i(x,y) \).
  – The amount of illumination reflected by the objects in the scene, i.e., reflectivity \( r(x, y) \).
• \( f(x, y) = i(x, y) \cdot r(x, y) \)
  where \( 0 < i(x, y) < \infty \) and \( 0 < r(x, y) < 1 \)
• Reflectivity function: \( r(x, y) \)
• For X-ray, transmissivity function
• The intensity of monochrome image is
  \[
  L_{\text{min}} \leq f(x,y) \leq L_{\text{max}} \quad \text{where} \quad L_{\text{min}} = i_{\text{min}} \cdot r_{\text{min}} \quad \text{and} \quad L_{\text{max}} = i_{\text{max}} \cdot r_{\text{max}}
  \]
• The interval \([L_{\text{min}}, L_{\text{max}}]\) is called the gray scale
• Common practice is to shift this interval to \([0 \text{ to } L-1]\), where \( l=0 \) is considered black and \( l=L-1 \) is considered white on the gray scale
• All intermediates are shades of gray varying from black to white
Some Typical Ranges of illumination

**Illumination**

- Lumen — A unit of light flow
- Lumen per square meter (lm/m²) — The metric unit of measure for illuminance of a surface
- On a clear day, the sun may produce in excess of 90,000 lm/m² of illumination on the surface of the Earth
- On a cloudy day, the sun may produce less than 10,000 lm/m² of illumination on the surface of the Earth
- On a clear evening, the moon yields about 0.1 lm/m² of illumination
- The typical illumination level in a commercial office is about 1000 lm/m²
Some Typical Ranges of Reflectance

- Reflectance
  - 0.01 for black velvet
  - 0.65 for stainless steel
  - 0.80 for flat-white wall paint
  - 0.90 for silver-plated metal
  - 0.93 for snow

Note::

Value range of reflectance **0 to 1**

0 means total absorption and 1 means total reflection
Digital Image Representation

- Digital image is represented as:

\[
\begin{bmatrix}
  f(0,0) & f(0,1) & \ldots & f(0, N-1) \\
  f(1,0) & f(1,1) & \ldots & f(1, N-1) \\
  \vdots & \vdots & \ddots & \vdots \\
  f(M-1,0) & f(M-1,1) & \ldots & f(M-1, N-1)
\end{bmatrix}
\]

N: No of Columns
M: No of Rows

\[\text{Origin} \quad 0 \quad 1 \quad 2 \quad 3 \ldots \quad \ldots \quad N-1\]

\[\text{Origin} \quad 0 \quad 1 \quad 2 \quad 3 \ldots\]

One pixel \( f(x, y) \)
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

Digitizing the coordinate values

Digitizing the amplitude values
**FIGURE 2.17** (a) Continuous image projected onto a sensor array. (b) Result of image sampling and quantization.
Spatial and Gray level Resolution

- **Spatial resolution**
  - 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**
  - The smallest discernible change in intensity or gray level
  - stated with *8 bits, 16 bits, etc.*
The Digitization process requires to determine $M$, $N$ and $L$

- $M$ and $N$ are Spatial Resolution
- $L = \text{Gray level Resolution}$
  - $L = 2^k$, 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 $\rightarrow$ Spatial Resolution
- Quantization $\rightarrow$ Gray level Resolution
## Spatial and Gray level Resolution

**TABLE 2.1**

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

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

Variation in Spatial Resolution from 1024 x 1024 to 32 x 32
FIGURE 2.20 (a) $1024 \times 1024$, 8-bit image. (b) $512 \times 512$ image resampled into $1024 \times 1024$ pixels by row and column duplication. (c) through (f) $256 \times 256$, $128 \times 128$, $64 \times 64$, and $32 \times 32$ images resampled into $1024 \times 1024$ pixels.
Spatial and Gray level Resolution

FIGURE 2.21
(a) 452 × 374, 256-level image. (b)–(d) Image displayed in 128, 64, and 32 gray levels, while keeping the spatial resolution constant.
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
Image Scaling (Zooming and Shrinking)

- Zooming (up scaling, resizing upward) requires two steps
  - The Creation of new pixel locations
  - Assignment of gray levels to new pixel locations
Image Scaling (Zooming and Shrinking)

- Zooming (up scaling, resizing upward) can be achieved by the following techniques:
  - Nearest neighbor Interpolation
  - Pixel Replication
  - Bilinear Interpolation
  - Bicubic Interpolation
Image Scaling (Zooming and Shrinking)

- **Nearest neighbor Interpolation**

  - Suppose that we have an image of size 500 x 500 and we want to enlarge it to 1.5 times 750 x 750 pixels.

  - For any zooming approach we have to create an imaginary grid of the size which is required over the original image. In that case we will have an imaginary grid of 750 x 750 over an original image.

  - Obviously the spacing in the grid would be less than one pixel because we fitting it over a smaller image. In order to perform gray level assignment for any point in the overlay, we look for the closest pixel in the original image and assign its gray level to new pixel in the grid.

  - When finished with all points in the grid, we can simply expand it to the originally specified size to obtain the zoomed image. **This method of gray level assignment is called nearest neighbor interpolation**
Image Scaling (Zooming and Shrinking)

- Pixel Replication

- Pixel replication is applicable when we want to increase the size of an image an integer number of times.

- For example to double the size of an image we can duplicate each column, this doubles the size of image in horizontal direction. Then we duplicate each row of the enlarged image to double the size in the vertical direction.

- The same procedure can be applied to enlarge the image by any integer number of times (triple, quadruple and so on).

- The gray level assignment of each pixel is predetermined by the fact that new locations are exact duplicate of old locations.
Image Scaling (Zooming and Shrinking)

- Bilinear Interpolation

  - Bilinear interpolation considers the closest 2x2 neighborhood of known pixel values surrounding the unknown pixel.

  - It then takes a weighted average of these 4 pixels to arrive at its final interpolated value. This results in much smoother looking images than nearest neighbor.

  - The diagram below is for a case when all known pixel distances are equal, so the interpolated value is simply their sum divided by four.

  - In case the distance varies then The closer pixels are given more weightage in the calculation
The key idea is to perform linear interpolation first in one direction, and then again in the other direction.

Suppose that we want to find the value of the unknown function $f$ at the point $P = (x, y)$.

It is assumed that we know the value of $f$ at the four points $Q_{11} = (x_1, y_1)$, $Q_{12} = (x_1, y_2)$, $Q_{21} = (x_2, y_1)$, and $Q_{22} = (x_2, y_2)$. 
We first do linear interpolation in the $x$-direction. This yields

$$f(R_1) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21})$$

where $R_1 = (x,y_1)$,

$$f(R_2) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22})$$

where $R_2 = (x,y_2)$.

$$f(P) \approx \frac{y_2 - y}{y_2 - y_1} f(R_1) + \frac{y - y_1}{y_2 - y_1} f(R_2).$$

We proceed by interpolating in the $y$-direction.
This gives us the desired estimate of $f(x, y)$.

\[
f(x, y) \approx f(Q_{11}) \left( \frac{x_2 - x}{(x_2 - x_1)(y_2 - y_1)} \right) (x_2 - x)(y_2 - y)
+ f(Q_{21}) \left( \frac{x - x_1}{(x_2 - x_1)(y_2 - y_1)} \right) (x - x_1)(y_2 - y)
+ f(Q_{12}) \left( \frac{x_2 - x}{(x_2 - x_1)(y_2 - y_1)} \right) (x_2 - x)(y - y_1)
+ f(Q_{22}) \left( \frac{x - x_1}{(x_2 - x_1)(y_2 - y_1)} \right) (x - x_1)(y - y_1).
\]
Image Scaling (Zooming and Shrinking)

- **Bicubic Interpolation**

  - Bicubic goes one step beyond bilinear by considering the closest 4x4 neighborhood of known pixels-- for a total of 16 pixels.

  - Since these are at various distances from the unknown pixel, closer pixels are given a higher weighting in the calculation.

  - Bicubic produces noticeably sharper images than the previous two methods, and is perhaps the ideal combination of processing time and output quality.

  - For this reason it is a standard in many image editing programs (including Adobe Photoshop), printer drivers and in-camera interpolation.
Increasing Spatial Resolution

**FIGURE 2.25** Top row: images zoomed from $128 \times 128$, $64 \times 64$, and $32 \times 32$ pixels to $1024 \times 1024$ pixels, using nearest neighbor gray-level interpolation. Bottom row: same sequence, but using bilinear interpolation.
Shrinking (Down scaling, resizing downward)

- Image shrinking is done in the similar manner as zooming with one difference as now the process of pixel replication is row column deletion. Now we can delete every second column and row for shrinking.
Basic Relationships between Pixels

- Neighborhood
- Connectivity
- Adjacency
- Paths
- Regions and boundaries
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 $(x, y+1)$

- **4 diagonal neighbors of $p$, denoted by $N_D(p)$:**
  - $(x-1, y-1)$, $(x+1, y+1)$, $(x+1,y-1)$, and $(x-1, y+1)$

- **8 neighbors of $p$, denoted $N_8(p)$**
  
  $N_8(p) = N_4(p) \cup N_D(p)$
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 $N_4(p) \cap N_4(p)$ has no pixels whose values are from $V$

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

- Let $S$ represent a subset of pixels in an image. Two pixels $p$ with coordinates $(x_0, y_0)$ and $q$ with coordinates $(x_n, y_n)$ are said to be connected in $S$ if there exists a path between them consisting entirely of pixels in $S$

  e.g. $(x_0, y_0), (x_1, y_1), ..., (x_n, y_n)$

  Where

  \[ \forall i, 0 \leq i \leq n, (x_i, y_i) \in S \]

- 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
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 \((x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)\)

Where \((x_0, y_0) = (x, y)\), \((x_n, y_n) = (s, t)\)

\((x_i, y_i)\) 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.
Examples: Adjacency and Path

\[ V = \{1, 2\} \]

\[
\begin{array}{ccc}
0 & 1 & 1 \\
0 & 2 & 0 \\
0 & 0 & 1 \\
\end{array}
\]

\[
\begin{array}{ccc}
0 & 1 & 1 \\
0 & 2 & 0 \\
0 & 0 & 1 \\
\end{array}
\]
Examples: Adjacency and Path

\[ V = \{1, 2\} \]

\[
\begin{array}{ccc}
0 & 1 & 1 \\
0 & 2 & 0 \\
0 & 0 & 1 \\
\end{array}
\quad \begin{array}{ccc}
0 & 1 & 1 \\
0 & 2 & 0 \\
0 & 0 & 1 \\
\end{array}
\quad \begin{array}{ccc}
0 & 1 & 1 \\
0 & 2 & 0 \\
0 & 0 & 1 \\
\end{array}
\]

8-adjacent
Examples: Adjacency and Path

\[ V = \{1, 2\} \]

\[
\begin{array}{ccc}
0 & 1 & 1 \\
0 & 2 & 0 \\
0 & 0 & 1 \\
\end{array}
\] 8-adjacent

\[
\begin{array}{ccc}
0 & 1 & 1 \\
0 & 2 & 0 \\
0 & 0 & 1 \\
\end{array}
\] m-adjacent
Examples: Adjacency and Path

\[ V = \{1, 2\} \]

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>1</th>
<th></th>
<th>0</th>
<th>1</th>
<th>1</th>
<th></th>
<th>0</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td></td>
<td>0</td>
<td>2</td>
<td>0</td>
<td></td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

8-adjacent

m-adjacent

The 8-path from (1,3) to (3,3):
(i) (1,3), (1,2), (2,2), (3,3)
(ii) (1,3), (2,2), (3,3)

The m-path from (1,3) to (3,3):
(1,3), (1,2), (2,2), (3,3)
Region and Boundary

- Let $R$ represent a subset of pixels in an image
- We call $R$ a region of the image if $R$ is a connected set
- We can specify the region by using 4-adjacency and 8-adjacency
- Region $= \{ \text{Set of all pixels which fulfill adjacency criteria} \}$

**Boundary (or border)**

- The *boundary* of the region $R$ is the set of pixels in the region that have one or more neighbors that are not in $R$.
- If $R$ happens to be an entire image, then its boundary is defined as the set of pixels in the first and last rows and columns of the image.

**Foreground and background**

- An image contains $K$ disjoint regions, $R_k, k = 1, 2, \ldots, K$. Let $R_u$ denote the union of all the $K$ regions, and let $(R_u)^c$ denote its complement
  - All the points in $R_u$ is called foreground;
  - All the points in $(R_u)^c$ is called background.
Connected Component Labeling

- Ability to assign different labels to various disjoint connected components of an image.

- Connected component labeling is a fundamental step in automated image analysis (Shape, Area, Boundary).

```
1 1 0 1 1 1 0 1
1 1 0 1 0 1 0 1
1 1 1 1 0 0 0 1
0 0 0 0 0 0 0 1
1 1 1 1 0 1 0 1
0 0 0 1 0 1 0 1
1 1 0 1 0 0 0 1
1 1 0 1 0 1 1 1
```

```
1 1 0 1 1 1 0 2
1 1 0 1 0 1 0 2
1 1 1 1 0 0 0 2
0 0 0 0 0 0 0 2
3 3 3 3 0 4 0 2
0 0 0 3 0 4 0 2
5 5 0 3 0 0 0 2
5 5 0 3 0 2 2 2
```

a) binary image
b) connected components labeling

c) binary image and labeling, expanded for viewing
Distance measures

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

$$D(p, q) \geq 0 \quad (D(p, q) = 0 \text{ if } p=q)$$
$$D(p, q) = D(q, p)$$
$$D(p, z) \leq (D(p, q) + D(q, z))$$

- Three different ways to calculate distance depending upon the traversing criteria of pixels are:

- **Euclidean distance**
- **City-block distance or D4 distance.**
- **D8 distance or chessboard distance.**
Distance measures

- **Euclidean distance**

  The Euclidean distance between p and q is defined as
  \[
  D_e(p, q) = [(x - s)^2 + (y - t)^2]^{1/2}
  \]

- **City-block distance or D4 distance**

  The Euclidean distance between p and q is defined as
  \[
  D_4(p, q) = |x - s| + |y - t|
  \]

  The pixels having a distance D4 from (x, y) less than or equal to some value r from a diamond centered at (x, y). E.g. the pixels with D4 distance <= 2 from (x, y) (the center point) form the following contours of constant distance:

  \[
  \begin{array}{ccc}
  2 & 2 & 2 \\
  2 & 1 & 2 \\
  2 & 1 & 0 & 1 & 2 \\
  2 & 1 & 2 \\
  2 & 2 \\
  \end{array}
  \]
Distance measures

- D8 distance or chessboard distance.

\[ D_8(p, q) = \max(|x - s|, |y - t|) \]

The pixels having a D8 distance from (x, y) less than or equal to some value \( r \) from a squared centered at (x, y). E.g. the pixels with D8 distance \( \leq 2 \) from (x, y) (the center point) form the following contours of constant distance.

The pixels with D8 = 1 are the 8-neighbors of (x, y):

```
2 2 2 2 2 2
2 1 1 1 1 2
2 1 0 1 2
2 1 1 1 1 2
2 2 2 2 2 2
```
Distance measures

- Consider the following arrangement of pixels and assume that p, p2 and p4 have value 1 and p1 and p3 have value 0 or 1:

```
      p3----p4
     |
     |
p1       p2
     |
     |
p
```

- Suppose we have V={1} [Adjacency criteria]

- If p1 and p3 are zero, the length of the shortest m-path (Dm distance) between p and p4 is 2

- If p1=1 then p2 and p will no longer be m-adjacent and the length of the shortest m-path becomes 3 [Path will be p p1 p2 p4]

- If p3=1 then the length of the shortest m-path will also be

- Finally if both p1 and p3 are 1 then the length of the shortest m-path between p and p4 is 4 [Path will be p p1 p2 p3 p4]
Image Operation on pixel basis

- As we know images are represented in the form of matrices with Rows and Columns arrangements

- Any operation either arithmetic or logical is carried out between corresponding pixels in the images

- For example if you want to add two images \( a \) and \( b \) then the resultant image will be formed in a way that the first element will be formed by the addition of the first pixel in \( a \) with the first pixel in \( b \) and so on for each pixel

- Similarly subtraction, multiplication and division are carried out in a similar manner

- The logical operators **AND, OR** and **NOT** are applied on only one image at a time
Any question