Islamic University Of Gaza
Faculty of Engineering
Computer Department
Eng. Ahmed M. Ayash

Digital Image Processing
Discussion
Chapter 2
Date: 24/02/2013

CHAPTER 2

Digital Image Fundamentals

Part 2

1. Theoretical

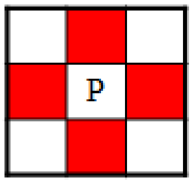
➢ Basic Relationships Between Pixels

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

Neighbors of a Pixel

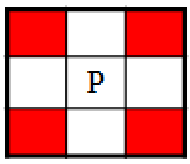
✖ The 4- neighbors of pixel p are: **N_4(p)**
Any pixel p(x,y) has two vertical and two horizontal neighbors, given by:

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



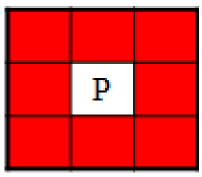
✖ The 4- diagonal neighbors are: **N_D(p)**
given by:

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



✖ The 8-neighbors are : **N_8(p)**
8-neighbors of a pixel pare its vertical, horizontal and 4 diagonal neighbors denoted by N_8(p)

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



Connectivity

Two pixels are said to be connected if they are adjacent in some sense.

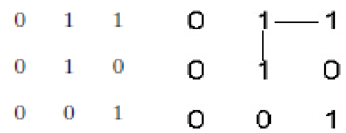
- They are neighbors (N_4, N_D, N_8) and
- Their intensity values (gray levels) are similar.

Adjacency

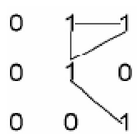
Let V be the set of intensity used to define adjacency; e.g. V={1} in a binary image or V={100,101,102,…,120} inn a gray-scale image.

We consider three types of adjacency:

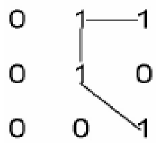
- **4-adjacency:**
  Two pixels p and q with values from V are 4-adjacent if q is in the set $N_4(p)$.



- **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** (mixed adjacency):
  Two pixels p and q with values from V are m-adjacent if:
  (i) q is in $N_4(p)$,or
  (ii) q is in $N_D(p)$ and $N_4(p) \cap N_4(q)$ is empty (has no pixels whose values are from V)



Two image subsets S1 and S2 are adjacent if some pixel in S1 is adjacent to some pixel in S2.

Question1:

Consider the two image subsets, $S1$ and $S2$, shown in the following figure. For V={1}, determine whether these two subsets are (a) 4-adjacent, (b) 8-adjacent, or (c) m-adjacent.

		S_1				S_2			
0	0	0	0	0	0	0	1	1	0
1	0	0	1	0	0	1	0	0	1
1	0	0	1	0	1	1	0	0	0
0	0	1	1	1	0	0	0	0	0
0	0	1	1	1	0	0	1	1	1

Solution:

Let p and q be as shown in Fig. Then:

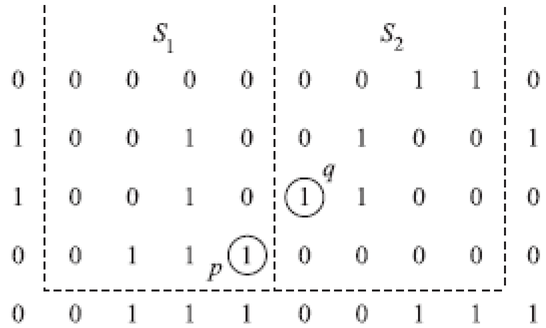
(a) $S1$ and $S2$ are **not 4-connected** because q is not in the set $N_4(p)$;

(b) $S1$ and $S2$ are **8-connected** because q is in the set $N_8(p)$;

(c) $S1$ and $S2$ are ***m*-connected** because

(i) q is in $N_D(p)$, and

(ii) the set $N_4(p) \cap N_4(q)$ is empty.



Paths

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

$$\boxed{(x_0, y_0), (x_1, y_1), \ldots\ldots, (x_n, y_n)}$$

- where $(x_0, y_0) = (x, y)$, $(x_n, y_n) = (s, t)$,
- and pixels (x_i, y_i) and (x_{i-1}, y_{i-1}) are adjacent for $1 \leq i \leq n$.
- In this case, n is the length of the path.
- If $(x_0, y_0) = (x_n, y_n)$ the path is a closed path.
- The path can be defined 4-,8-,m-paths depending on adjacency type.

Let S be a subset of pixels in an image. Two pixels p and q are said to be **connected** in S if there exists a path between them consisting entirely of pixels 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 a **connected set**.

Question2:

Consider the image segment shown.

Let V={0, 1} and compute the lengths of the shortest 4-, 8-, and m-path between p and q. If a particular path does not exist between these two points, explain why.

3	1	2	1 (q)
2	2	0	2
1	2	1	1
(p) 1	0	1	2

Solution:

- When $V = \{0,1\}$, *4*-path does not exist between *p* and *q* because it is impossible to get from *p* to *q* by traveling along points that are both 4-adjacent and also have values from *V* . Fig. *a* shows this condition; it is not possible to get to *q*.
- The shortest *8*-path is shown in Fig. *b* its length is **4**.
- The length of the shortest *m*- path (shown dashed) is **5**.
- Both of these shortest paths are unique in this case.

(a) (b)

Regions and boundaries

- Let R be a subset of pixels in an image. We call R a **region** of the image if R is a **connected set.**
- The **boundary** (also called border or contour) of a region R is the set of pixels in the region that have one or more neighbors that are not in R.

Distance Measures

Given pixels p, q and z with coordinates (x, y), (s, t), (u, v) respectively, the distance function D has following properties:

- $D(p, q) \geq 0$ [$D(p, q) = 0$, iff $p = q$]
- $D(p, q) = D(q, p)$
- $D(p, z) \leq D(p, q) + D(q, z)$

- The following are the different Distance measures:

a. Euclidean Distance:

$$\boxed{\mathbf{D_e(p, q) = [(x-s)^2 + (y-t)^2]^{1/2}}}$$

- The pixels having a distance less than or equal to some value *r* from (x, y) are the points contained in a **disk** of radius r centered at (x, y).

b. City Block Distance:

$$\boxed{\mathbf{D_4(p, q) = |x-s| + |y-t|}}$$

- The pixels having a D_4 distance from (x, y) less than or equal to some value r form a **diamond** centered at (x, y). For example, the pixels with D_4 distance ≤ 2 from (x, y) (the center point) form the following contours of constant distance:

$$\begin{array}{ccccc} & & 2 & & \\ & 2 & 1 & 2 & \\ 2 & 1 & 0 & 1 & 2 \\ & 2 & 1 & 2 & \\ & & 2 & & \end{array}$$

- The pixels with D_4=1 are the 4-neighbors of (x, y).

c. Chess Board Distance:

$$\boxed{\mathbf{D_8(p, q) = max(|x\text{-}s|, |y\text{-}t|)}}$$

- The pixels with D_8 distance from (x, y) less than or equal to some value r form a **square** centered at (x, y). For example, the pixels with D_8 distance ≤ 2 from (x, y) (the center point) form the following contours of constant distance:

$$\begin{array}{ccccc} 2 & 2 & 2 & 2 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 2 & 1 & 0 & 1 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 2 & 2 & 2 & 2 & 2 \end{array}$$

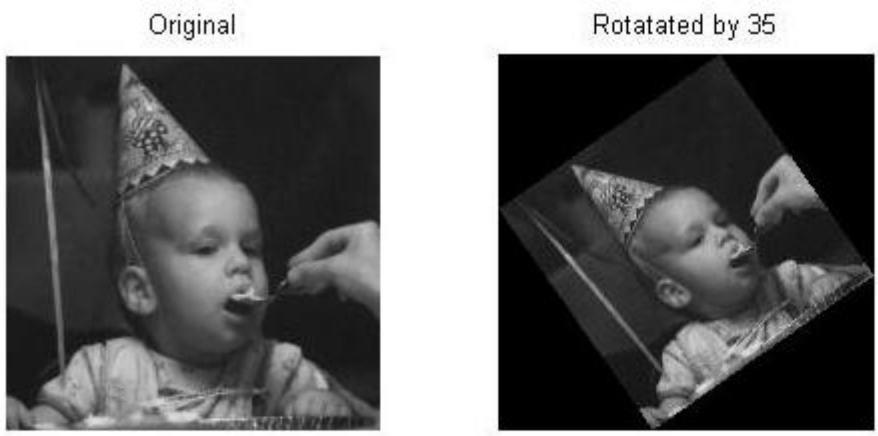
- The pixels with D_8=1 are the 8-neighbors of (x, y).

2. Practical

Example1: Image rotate

```
%ex1.m
clear all;
I = imread('1.jpg');
J = imrotate(I,35); %rotates an image 35° counterclockwise
subplot(121),imshow(I),title('Original');
subplot(122),imshow(J),title('Rotatated by 35');
```

Output:



Example2: Cropping an Image

```
%ex2.m
clear all;
I = imread('1.jpg');
J = imrotate(I,35);
J2= imcrop(J);
subplot(131),imshow(I),title('Original');
subplot(132),imshow(J),title('Rotatated by 35');
subplot(133),imshow(J2),title('Rotatated by 35 croped');
```

Output:

3. Homework:

1. Consider the image segment shown.
   Let V= {1, 2} and compute the lengths of the shortest 4-, 8-, and m-path between p and q. If a particular path does not exist between these two points, explain why.

	3	1	2	1	(q)
	2	2	0	2	
	1	2	1	1	
(p)	1	0	1	2	

2. Write a Matlab code to rotate an image by 180° (do not use the imrotate function, you should implement it)
   Hint (Use ***end*** function)

3. Given the two images below, perform an enhancement operation to get Fig (1).
   Hint (Use a **logical** operation)

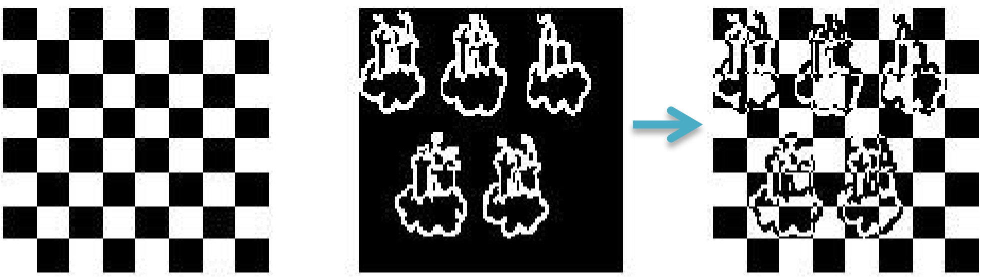


Fig (1)