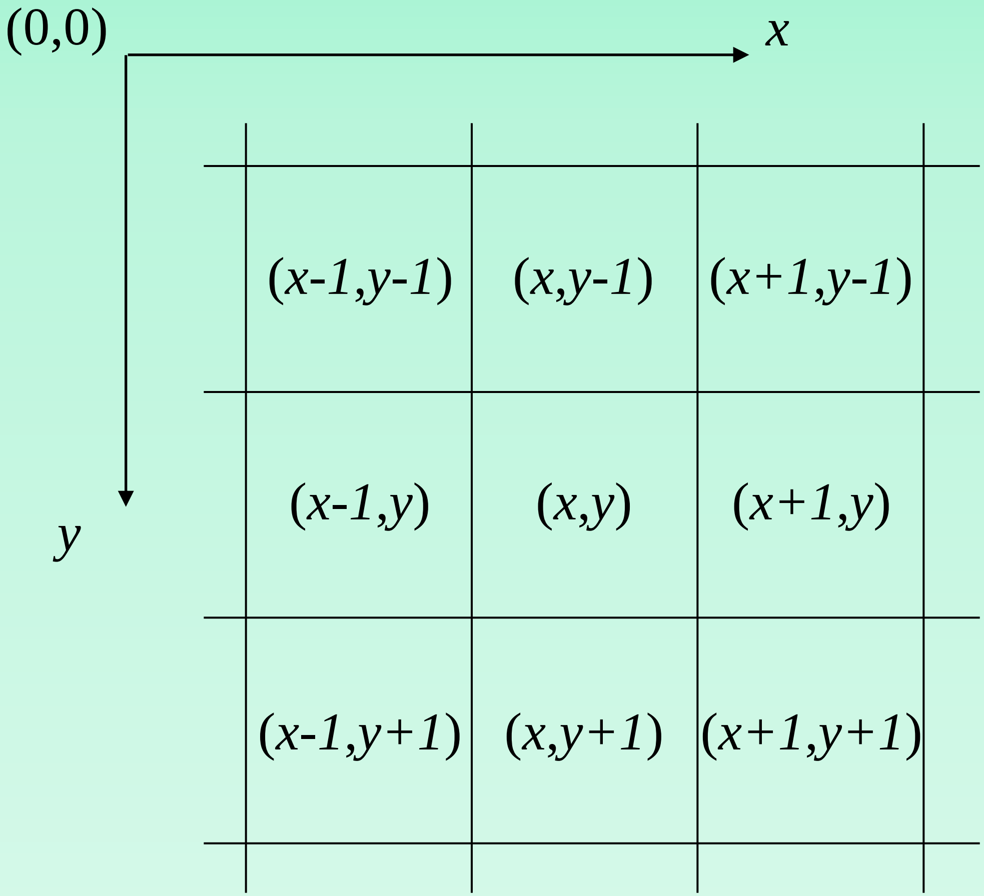
Digital Image Processing

Relationships between pixels

Basic Relationship of Pixels



Conventional indexing method

Neighbors of a Pixel

- we consider several important relationships between pixels in a digital image.
- A pixel *p* at coordinates (*x*,*y*) has four *horizontal* and *vertical* neighbors whose coordinates are given by:

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

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

This set of pixels, called the 4-*neighbors* or *p*, is denoted by $N_4(p)$. Each pixel is one unit distance from (x,y) and some of the neighbors of p lie outside the digital image if (x,y) is on the border of the image.

Neighbors of a Pixel

- The four *diagonal* neighbors of p have coordinates:

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

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

and are denoted by $N_D(p)$.

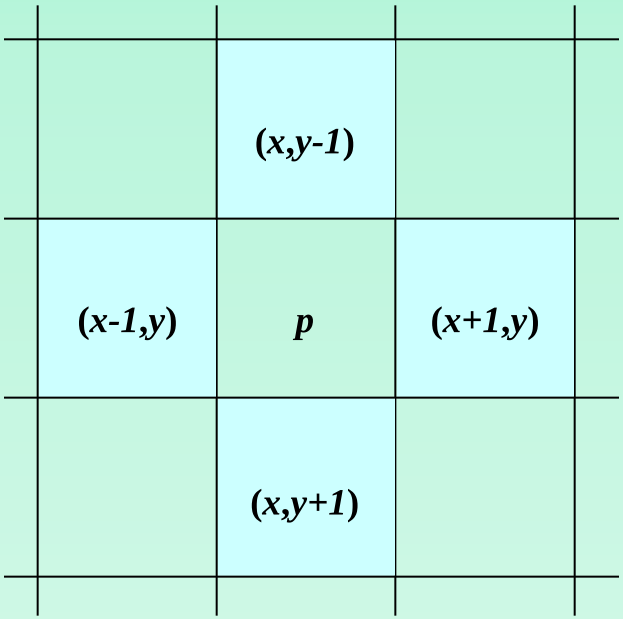
These points, together with the 4-neighbors, are called the 8-neighbors of p, denoted by $N_8(p)$.

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

As before, some of the points in $N_D(p)$ and $N_8(p)$ fall outside the image if (x,y) is on the border of the image.

Neighbors of a Pixel

Neighborhood relation is used to tell adjacent pixels. It is useful for analyzing regions.



4-neighbors of *p*:

$$N_4(p) = \left\{ \begin{array}{c} (x-1,y) \\ (x+1,y) \\ (x,y-1) \\ (x,y+1) \end{array} \right\}$$

4-neighborhood relation considers only vertical and horizontal neighbors.

Neighbors of a Pixel (cont.)

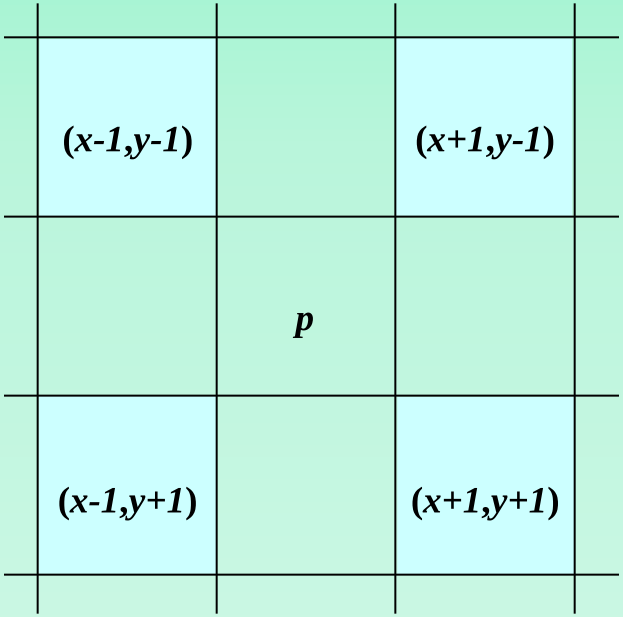
(*x*-*1*,*y*-*1*)	**(*x*,*y*-*1*)**	**(*x*+*1*,*y*-*1*)**
(*x*-*1*,*y*)	***p***	**(*x*+*1*,*y*)**
(*x*-*1*,*y*+*1*)	**(*x*,*y*+*1*)**	**(*x*+*1*,*y*+*1*)**

8-neighbors of *p*:

$$N_8(p) = \left\{ \begin{pmatrix} (x-1,y-1) \\ (x,y-1) \\ (x+1,y-1) \\ (x-1,y) \\ (x+1,y) \\ (x-1,y+1) \\ (x,y+1) \\ (x+1,y+1) \end{pmatrix} \right\}$$

8-neighborhood relation considers all neighbor pixels.

Neighbors of a Pixel (cont.)

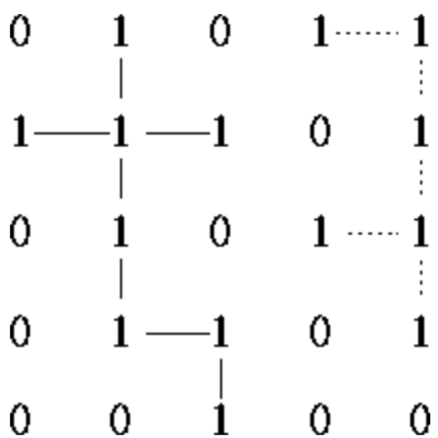


Diagonal neighbors of *p*:

$$N_D(p) = \begin{Bmatrix} (x-1,y-1) \\ (x+1,y-1) \\ (x-1,y+1) \\ (x+1,y+1) \end{Bmatrix}$$

Diagonal -neighborhood relation considers only diagonal neighbor pixels.

An example of a binary image with two connected components which are based on *4*-connectivity



Connectivity

Connectivity is adapted from neighborhood relation. Two pixels are connected if they are in the same class (i.e. the same color or the same range of intensity) and they are neighbors of one another.

- 4-connectivity: p and q are 4-connected if $q \in N_4(p)$

- 8-connectivity: p and q are 8-connected if $q \in N_8(p)$

- mixed-connectivity (m-connectivity):

  m-*connectivity*. p and q are m-connected

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

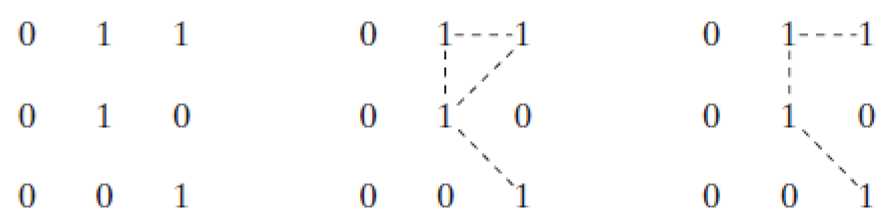
  (ii) q is in the set $N_D(p)$ and the intersection of $N_4(p)$ and $N_4(q)$ **has no pixels whose value from V.** That is, there are no pixels whose values are from V and which are 4-neighbors of both p and q.

Types of Adjacency

1. **4-adjacency:** Two pixels p and q with values from V are 4-adjacent if q is in the set $N_4(p)$.
2. **8-adjacency:** Two pixels p and q with values from V are 8-adjacent if q is in the set $N_8(p)$.
3. **m-adjacency =(mixed)**

Types of Adjacency

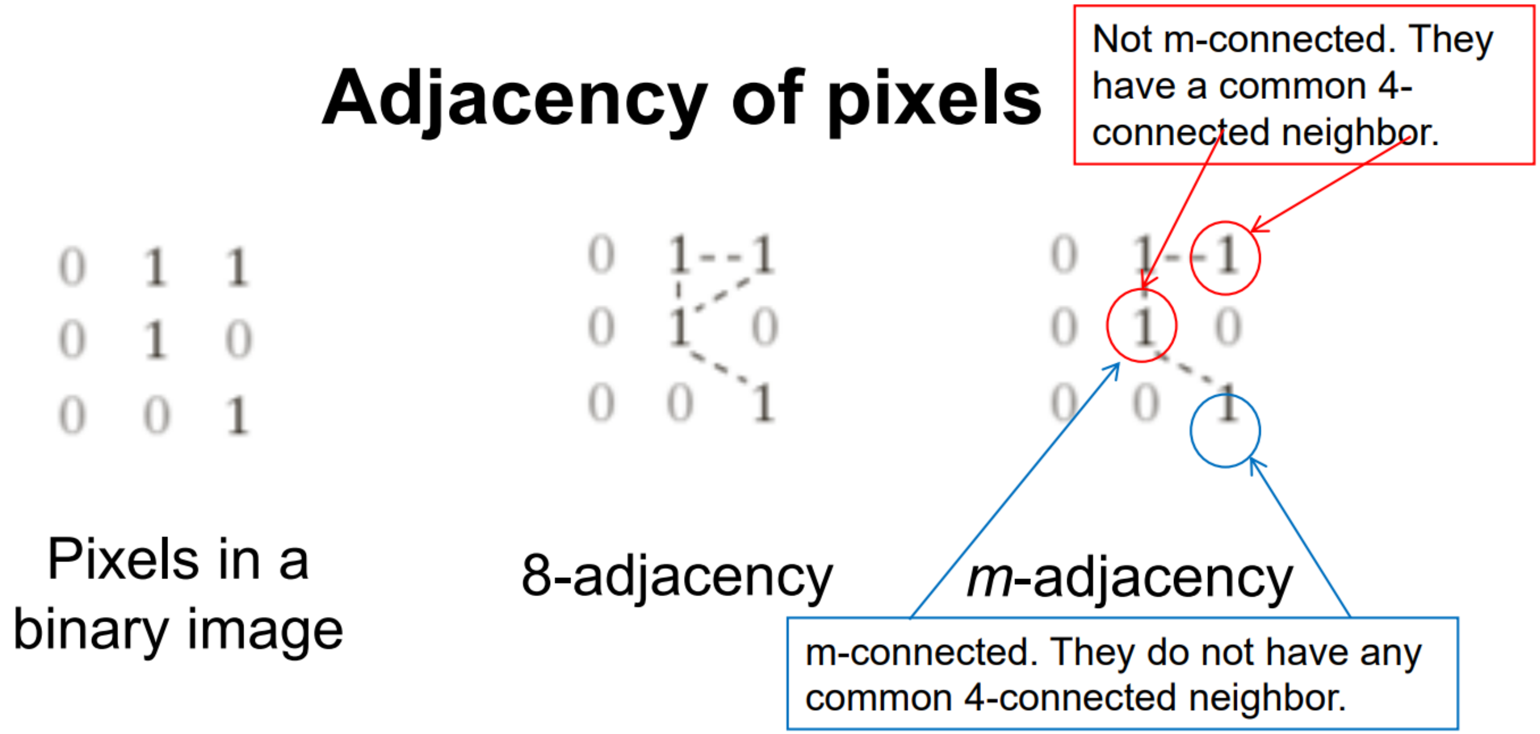
- Mixed adjacency is a modification of 8-adjacency. It is introduced to eliminate the ambiguities that often arise when 8-adjacency is used.
- For example:



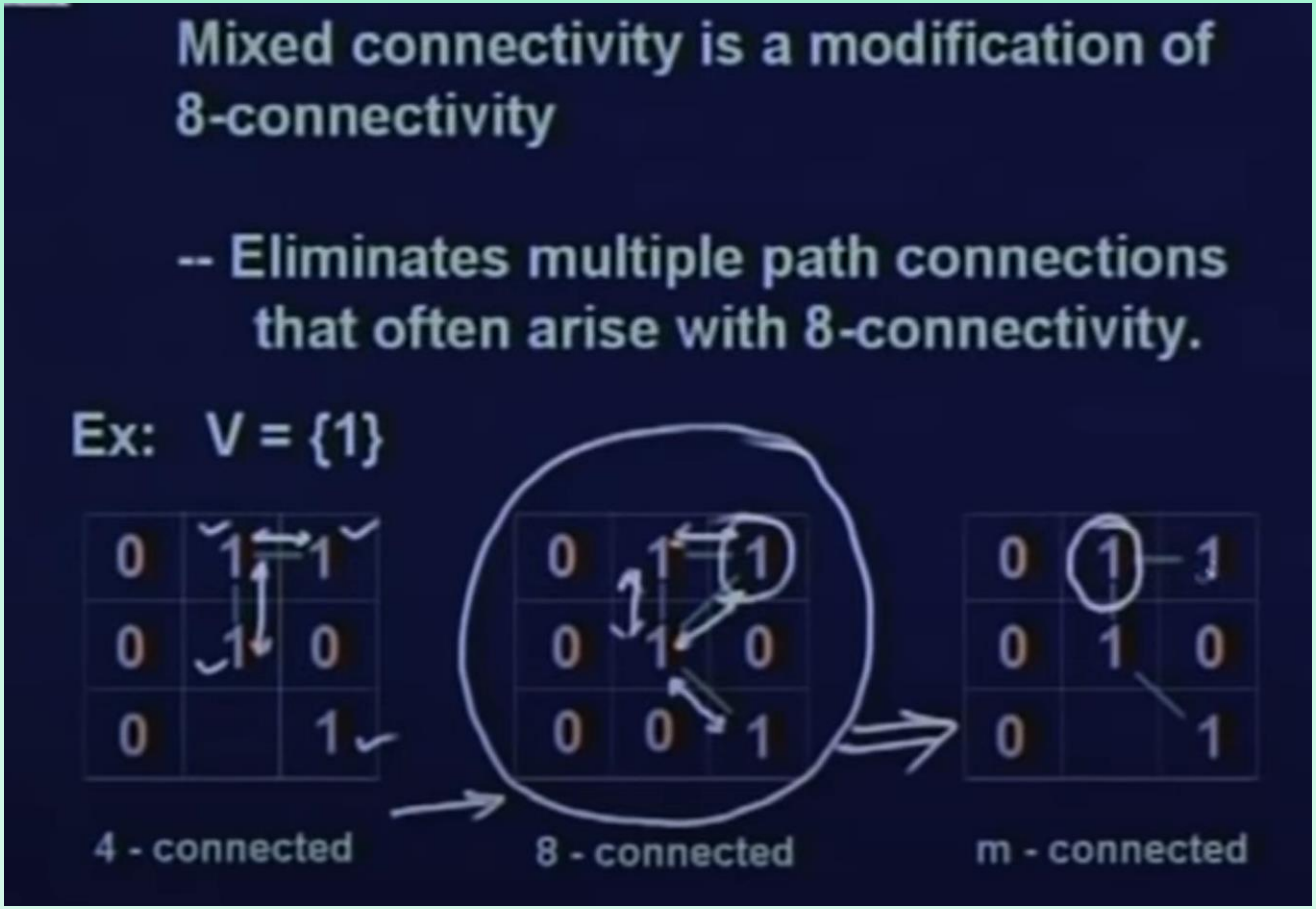
a b c

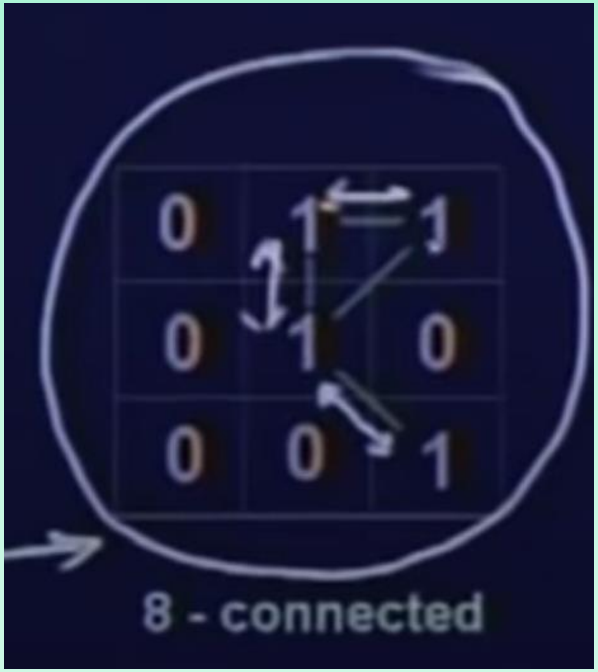
FIGURE 2.26 (a) Arrangement of pixels; (b) pixels that are 8-adjacent (shown dashed) to the center pixel; (c) *m*-adjacency.

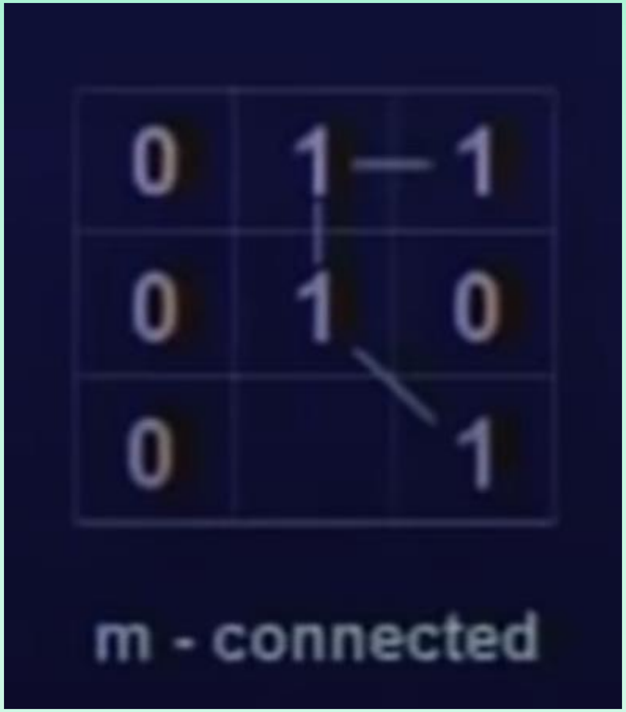
Adjacency of pixels



The role of *m*-adjacency is to define a single path between pixels. It is used in many image analysis and processing algorithms.


Mixed connectivity is a modification of 8-connectivity
-- Eliminates multiple path connections that often arise with 8-connectivity.
Ex: V = {1}
4 - connected
8 - connected
m - connected


0 1 1
0 1 0
0 0 1
8 - connected


0 1 1
0 1 0
0 1
m - connected

- Let V: a set of intensity values used to define adjacency and connectivity.
- In a binary image, $V = \{1\}$, if we are referring to adjacency of pixels with value 1.
- In a gray-scale image, the idea is the same, but V typically contains more elements, for example, $V = \{180, 181, 182, \ldots, 200\}$
- If the possible intensity values 0 – 255, V set can be any subset of these 256 values.

Adjacency, Connectivity, Region and Boundary

- Two pixels are *connected* if they are neighbors and if their gray levels satisfy a specified criterion of similarity.

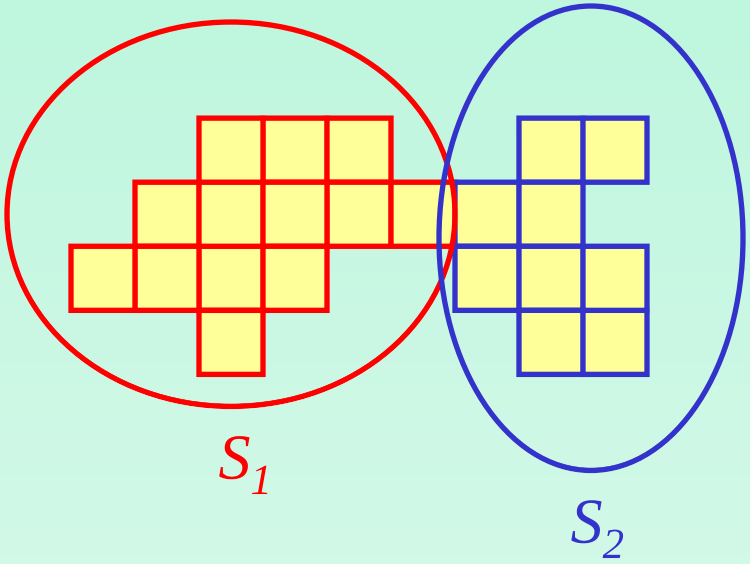
$145 < V \leq 170$

165	104	101
110	150	165
102	155	170

- Two pixels *p* and *q* are *adjacent* if they are connected.

Adjacency

Two image subsets S_1 and S_2 are adjacent if some pixel in S_1 is adjacent to some pixel in S_2



We can define type of adjacency: 4-adjacency, 8-adjacency or m-adjacency depending on type of connectivity.

Region Adjacency

- Two image subsets S_1 and S_2 are *adjacent* if some pixel in S_1 is adjacent to some pixel in S_2.

$$V = \{1\}$$

$$\begin{array}{cccccccccccc} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{array} \qquad \begin{array}{cccccccccccc} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{array}$$

$$S_1 \qquad\qquad S_2 \qquad\qquad\qquad S_1 \qquad\qquad S_2$$

Path

A ***path*** from pixel p at (x,y) to pixel q at (s,t) is a sequence of distinct pixels:

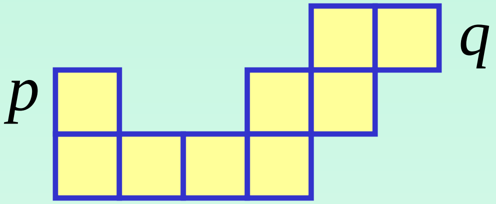
$$(x_0,y_0), (x_1,y_1), (x_2,y_2),\ldots, (x_n,y_n)$$

such that

$$(x_0,y_0) = (x,y) \text{ and } (x_n,y_n) = (s,t)$$

and

$$(x_i,y_i) \text{ is adjacent to } (x_{i-1},y_{i-1}), \qquad i = 1,\ldots,n$$



We can define type of path: 4-path, 8-path or m-path depending on type of adjacency.

A Digital Path

- A digital path (or curve) from pixel p with coordinate (x,y) to pixel q with coordinate (s,t) is a sequence of distinct pixels with coordinates (x_0,y_0), (x_1,y_1), …, (x_n, y_n) where $(x_0,y_0) = (x,y)$ and $(x_n, y_n) = (s,t)$ and pixels (x_i, y_i) and (x_{i-1}, y_{i-1}) are adjacent for $1 \leq i \leq n$
- n is the length of the path
- If $(x_0,y_0) = (x_n, y_n)$, the path is closed.
- We can specify 4-, 8- or m-paths depending on the type of adjacency specified.

Digital Path (or Curve)

q

p

0 0 0 0 0 0 1 1 1 0 0

0 0 1 1 0 1 0 0 0 1 0

0 0 1 0 0 0 1 1 1 0 0

0 0 1 0 0 0 1 0 0 0 0

0 0 0 1 1 1 0 0 0 0 0

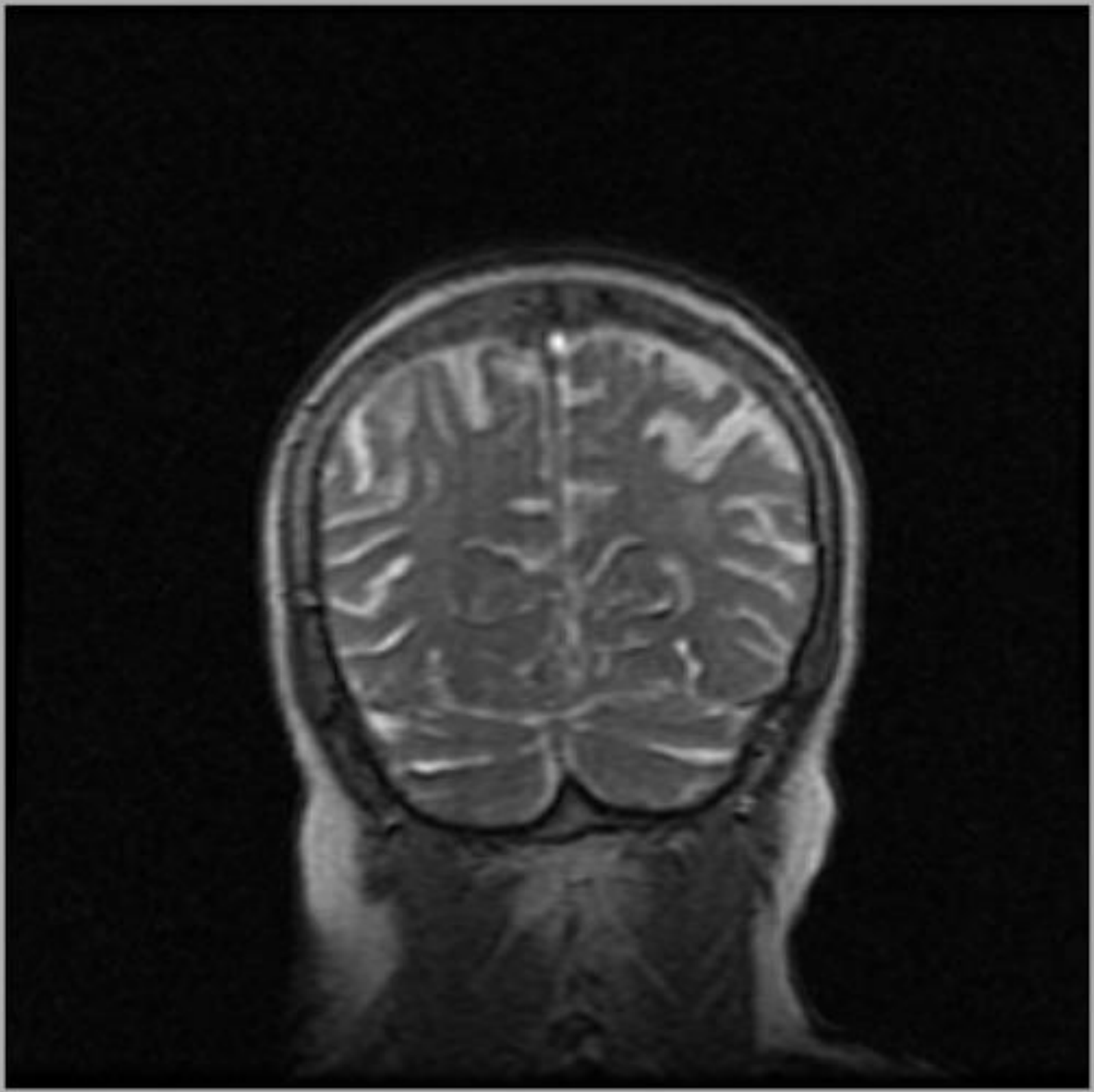
(2,4) , (2,3), (3,3), (4,3), (5,4), (5,5), (5,6), (4,7), (3,7), (2,6)

Region and Boundary

Let *R* be a subset of pixels in an image.
R is a *region* of the image if *R* is a *connected set*.
The *boundary* (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*.

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

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



Distance

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

- $D(p,q) \geq 0 \quad (D(p,q) = 0$ if and only if $p = q)$
- $D(p,q) = D(q,p)$
- $D(p,z) \leq D(p,q) + D(q,z)$

Example: Euclidean distance

$$D_e(p,q) = \sqrt{(x-s)^2 + (y-t)^2}$$

Distance (cont.)

D_4-distance (*city-block distance*) is defined as

$$D_4(p,q) = |x-s| + |y-t|$$

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

Pixels with $D_4(p) = 1$ is 4-neighbors of p.

Distance (cont.)

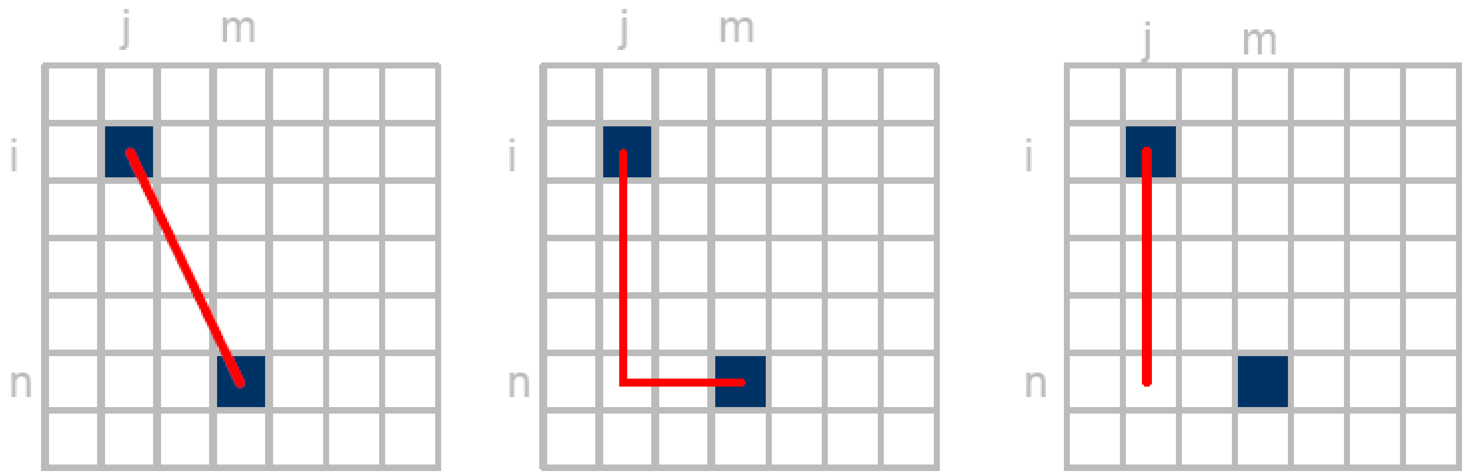
D_8-distance (*chessboard distance*) is defined as

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

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

Pixels with $D_8(p) = 1$ is 8-neighbors of p.

Distance measures



Euclidean Distance

$$= \sqrt{(i-n)^2 + (j-m)^2}$$

City Block Distance

$$= |i-n| + |j-m|$$

Chessboard Distance

$$= \max\left[\, |i-n|, |j-m| \,\right]$$

Example

Compute the distance between the two pixels using the three distances :

q:(1,1)

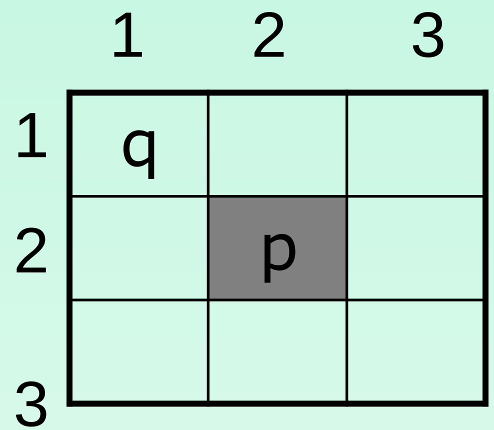
P: (2,2)

Euclidian distance : $((1-2)^2+(1-2)^2)^{1/2}$ = sqrt(2).

D4(City Block distance): |1-2| +|1-2| =2

D8(chessboard distance) : max(|1-2|,|1-2|)= 1

(because it is one of the 8-neighbors)



Distance measures

Example :

Use the city block distance to prove 4-neighbors ?

Pixel A : |2-2| + |1-2| = 1

Pixel B: |3-2| + |2-2| = 1

Pixel C: |2-2| + |2-3| =1

Pixel D: |1-2| + |2-2| = 1

	1	2	3
1		d	
2	a	p	c
3		b	

Now as a homework try the chessboard distance to proof the 8- neighbors!!!!