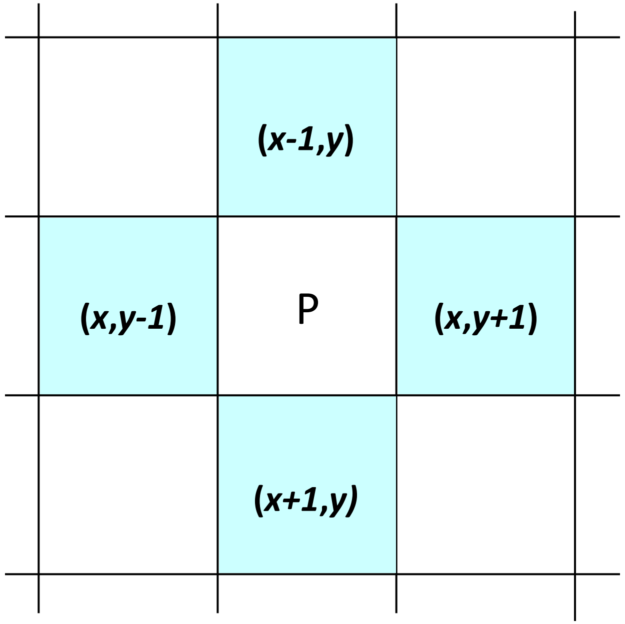
Basic Relationship between Pixels

Connectivity & Adjacency

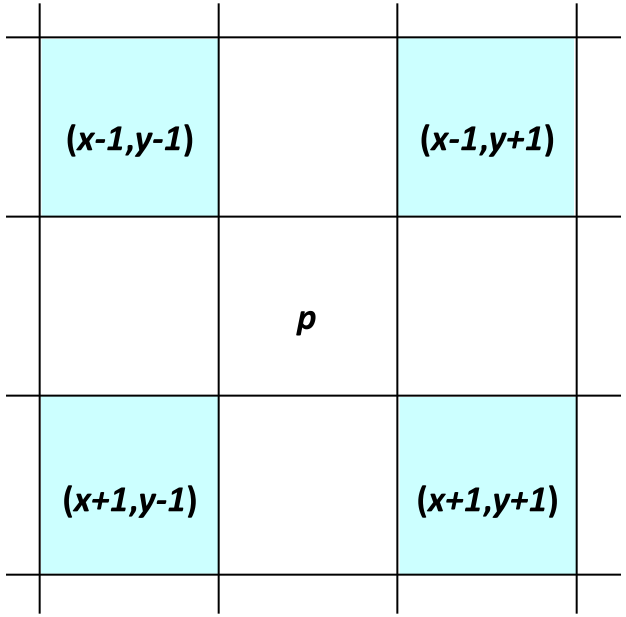
Neighbourhood

- Neighbors of pixels



4-neighbors of *p*:

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



Diagonal neighbors of *p*:

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

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

8-neighbors of *p*:

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

Connectivity

- 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.

For p and q from the same class

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

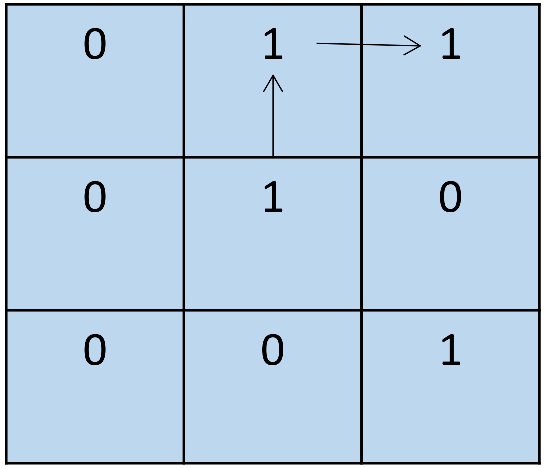
8-connectivity: p and q are 8-connected if $I(p,q) \in v$ & $q \in N_8(p)$

mixed-connectivity (m-connectivity):

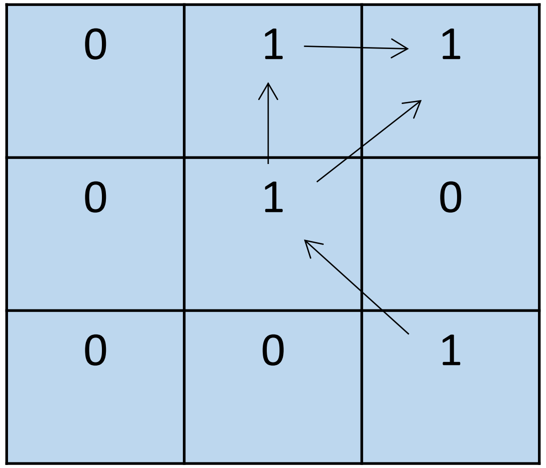
p and q are m-connected if $I(p,q) \in v$ & $q \in N_4(p)$ or

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

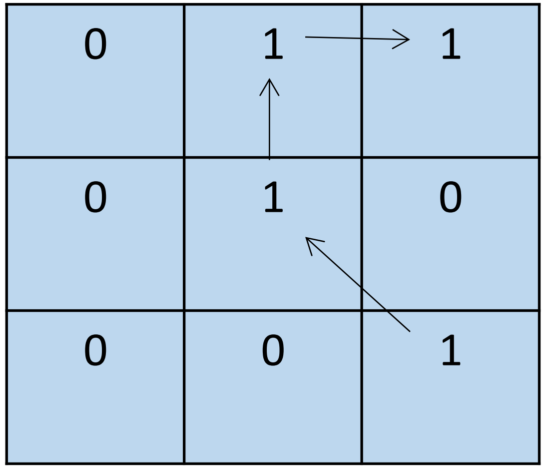
Connectivity



4-connectivity



8-connectivity



m-connectivity

Connectivity

- Establishing Object boundary
- Defining Object component/region


Basic Thresholding Algorithm

If $f(x,y) > Th$

$\Rightarrow$(x,y) is an element of foreground else background

Adjacency

- A pixel p is adjacent to pixel q if they are connected to each other. Two image subsets S_1 and S_2 are adjacent if some pixel in S_1 is adjacent to some pixel in S_2

4-adjacency: p and q are 4-adjacent if $q \in N_4(p)$

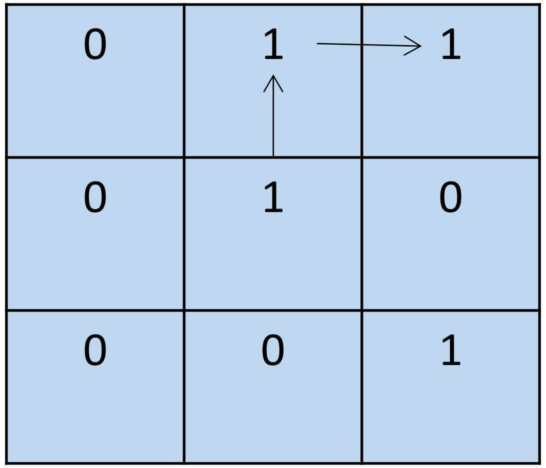
8-adjacency: p and q are 8-adjacent if $q \in N_8(p)$

mixed-adjacency (m-adjacency):

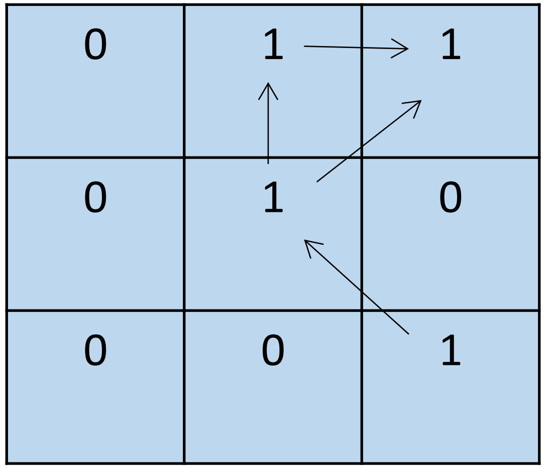
p and q are m-adjacent if $q \in N_4(p)$ or

$q \in N_D(p)$ and $N_4(p) \cap N_4(q) = \varnothing$

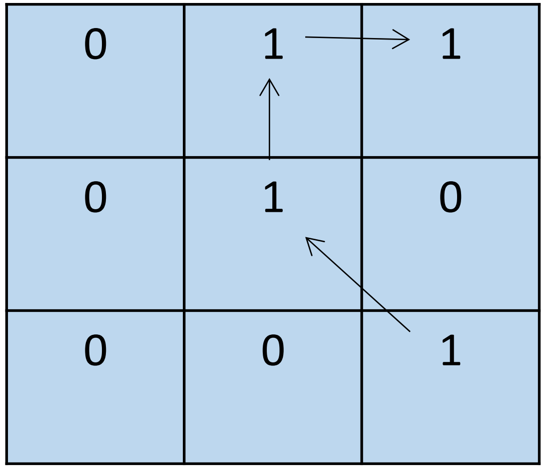
Adjacency



4-adjacency



8-adjacency



m-adjacency

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

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

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

Connected Component

S is subset of image I and p,q $\in$ S

p is connected to q if path exist

All pixels in S connected to p forms a connected component

- Region
  - R is a region of the image if R is a connected set
- Boundary
  - The boundary of a region R is the set of pixels in the region that have one or more neighbors that are not in R
- Edge
  - Pixels with derivative values that exceed a preset threshold

Food for thought!

1. When can we say that two pixels in an image belong to the same object?
2. Why does it matter whether we consider only side-by-side neighbors or also diagonal neighbors while analyzing an image?
3. How can a computer identify all pixels that form one complete object in a binary image?
4. What do we mean by the boundary of an object in an image, and why is it important?
5. How can we decide whether there is a path between two pixels inside an object?

Programming assignments

1. Display 4-neighbours and 8-neighbours of a pixel.

Concepts: Adjacency, neighborhood

Task:

Given a binary or grayscale image as a 2D NumPy array and a pixel location (x, y), write a program to:

Print its 4-neighbours

Print its d-neighbours

Print its 8-neighbours

2. Implement simple thresholding to create a binary image.

Concepts: Foreground, background, preprocessing for connectivity

Task:

Given a grayscale image and a threshold value:

Convert it into a binary image using thresholding.

AI supported self-learning on **Connectivity** (Prompts compatible with ChatGPT)

Active Learners (Learning by Doing)

1. Give me a small grid-based problem where I identify 4-connectivity, 8-connectivity, and m-connectivity between pixels. Let me try first, then explain the correct answer.
2. Create a short exercise where I compare 4- and 8-connectivity and explain which pixels are connected and why.

Reflective Learners (Learning by Thinking)

1. Explain 4-connectivity, 8-connectivity, and m-connectivity step by step, and summarize the differences clearly at the end.
2. Explain why different types of connectivity are defined in digital image processing and what problems they are intended to solve.

Sensing Learners (Concrete & Practical)

1. Explain pixel connectivity using a small binary image matrix with actual pixel coordinates and values.
2. Show practical examples where 4-connectivity and 8-connectivity lead to different connected components.

Intuitive Learners (Concepts & Patterns)

1. Explain the conceptual idea behind pixel connectivity and how different definitions change the interpretation of image regions.
2. Explain m-connectivity conceptually and why it is introduced instead of relying only on 4- or 8-connectivity.

Visual Learners (Diagrams & Structure)

1. Explain 4-connectivity, 8-connectivity, and m-connectivity using diagrams or grid-based illustrations.
2. Show a visual comparison of how the same pixel pattern forms different connected regions under 4-, 8-, and m-connectivity.

Verbal Learners (Words & Explanation)

1. Explain pixel connectivity in simple language using clear descriptions and everyday analogies.
2. Explain the difference between 4-, 8-, and m-connectivity as if teaching it to someone new to image processing.

Sequential Learners (Step-by-Step Logic)

1. Explain how to determine pixel connectivity step by step starting from identifying neighbors to defining connected regions.
2. List the rules for 4-connectivity, 8-connectivity, and m-connectivity and explain each rule in order.

Global Learners (Big Picture First)

1. First explain the overall idea of connectivity in digital images, then explain how 4-, 8-, and m-connectivity fit into this concept.
2. Explain why connectivity definitions are important in image analysis before explaining the details of each type.