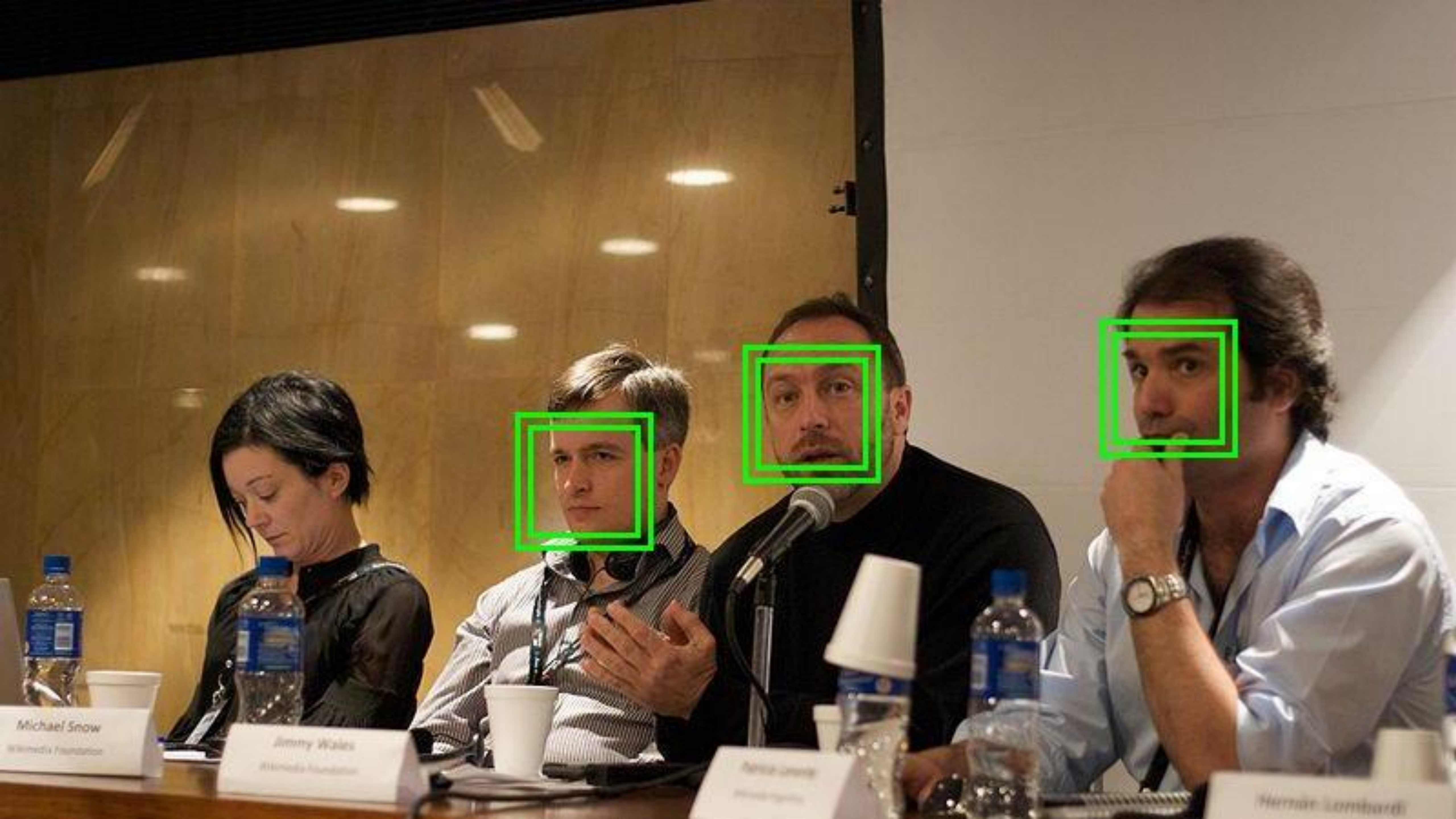
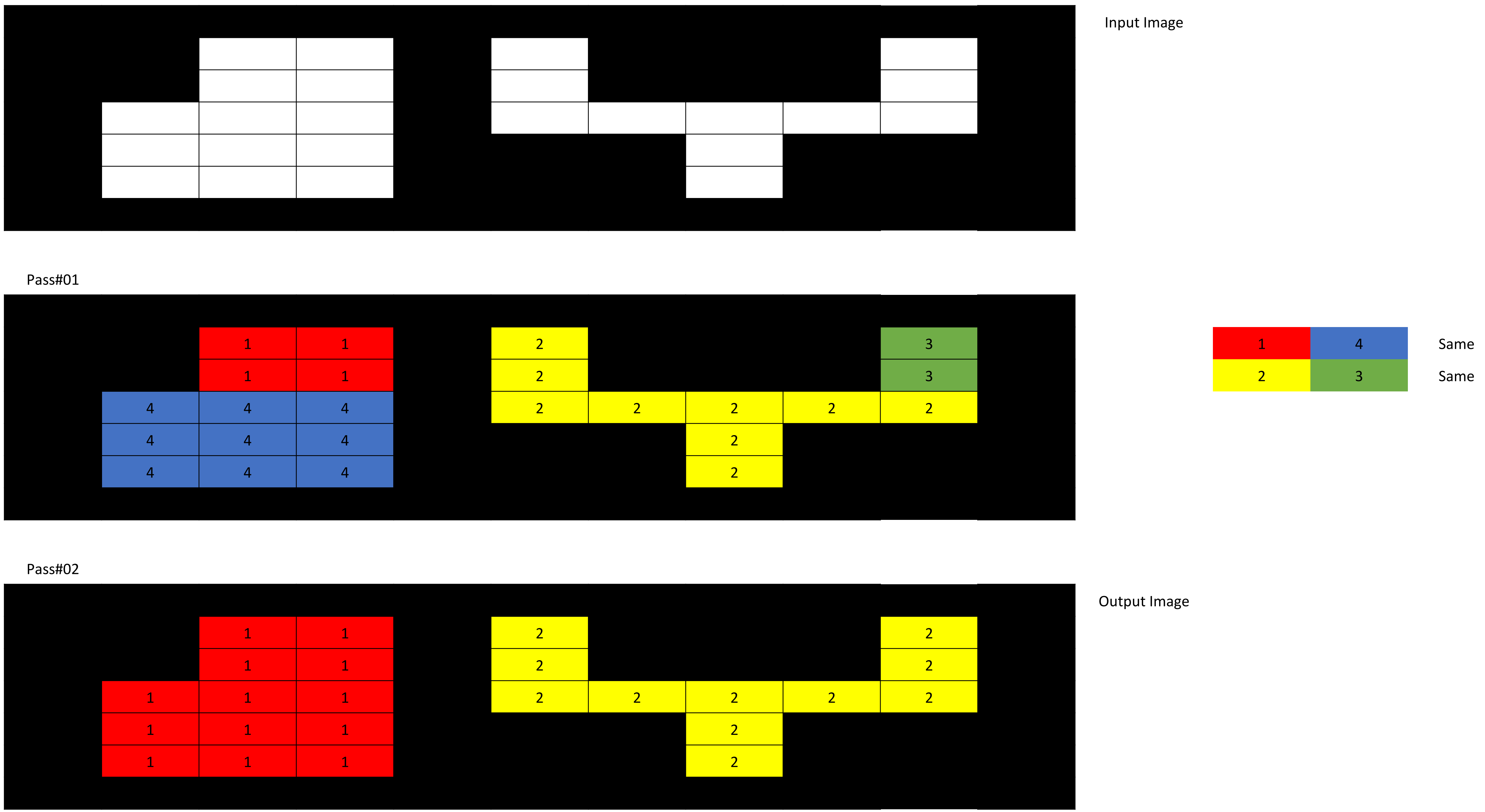
Component Labelling

Count how many?


Michael Snow
Jimmy Wales
Hernán Lombardi


Input Image
Pass#01
1 4 Same
2 3 Same
Pass#02
Output Image

if $f(x,y) = '1'$

then

⋮

Case 1

$L(T) = \phi$ & $L(L) = \phi$

$L(P) =$ 'New Label'

Case 2

$L(T) \neq \phi$; $L(L) = \phi$

$L(P) = L(T)$

Case 3

$L(T) = \phi$; $L(L) \neq \phi$

$L(P) = L(L)$

Case 4

$L(T) \neq \phi$; $L(L) \neq \phi$

Sub case a

$L(T) = L(L)$

$L(P) = L(T)$

Sub case b

$L(T) \neq L(L)$

$L(P) = L(L)$

$L(L) = L(T)$

Pass 2

Only if:

Case 4 Subcase b is true.

$$L(T) = L(L) = L(T)$$

$$1 = 4 = 1 \checkmark$$

$$2 = 3 = 2 \checkmark$$

$\vdots$

Count Labels

Food for thought!

1. What is component labeling in digital image processing?
2. Why is component labeling used in image analysis?
3. What is meant by a connected component?
4. How does 4-connectivity differ from 8-connectivity in component labeling?
5. What is the basic output of a component labeling algorithm?

Programming assignment

- Implement a connected component labeling algorithm to identify and count distinct objects in a binary image.
- **Concepts Used**
  - Binary image representation
  - Connected components
  - 4-connectivity and 8-connectivity
  - Component labeling algorithm (two-pass or flood-fill)
- **Tasks**
  - Read or create a binary image.
  - Implement connected component labeling using 4-connectivity.
  - Modify the algorithm to perform labeling using 8-connectivity.
  - Assign a unique label to each connected component.
  - Count the total number of connected components.
  - Display the original binary image and the labeled output image.
  - Briefly compare the results obtained using 4-connectivity and 8-connectivity.

AI supported self-learning (Prompts compatible with ChatGPT)

Active Learners (Learning by Doing)

1. Give me a small binary image matrix and ask me to manually label connected components using 4-connectivity. Let me try first, then explain the correct labeling.
2. Create a simple exercise where I count the number of connected components in a binary image using 8-connectivity and then explain the answer.

Reflective Learners (Learning by Thinking)

1. Explain component labeling step by step and summarize the purpose of each step in the algorithm.
2. Explain why component labeling is important in image analysis and what information we gain from it.

Sensing Learners (Concrete & Practical)

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

Intuitive Learners (Concepts & Patterns)

1. Explain the concept of a connected component and how connectivity definitions change image interpretation.
2. Explain why component labeling algorithms are designed as multi-pass or region-growing methods.

Visual Learners (Diagrams & Structure)

1. Explain component labeling using grid-based diagrams showing labeled regions for 4-connectivity and 8-connectivity.
2. Visually compare original binary images and their labeled outputs using different connectivity rules.

Verbal Learners (Words & Explanation)

1. Explain component labeling in simple language using everyday examples like grouping nearby objects.
2. Explain the difference between 4-connectivity and 8-connectivity as if teaching it to a beginner.

Sequential Learners (Step-by-Step Logic)

1. Explain the two-pass connected component labeling algorithm step by step from scanning to label equivalence resolution.
2. Explain step by step how a flood-fill algorithm identifies and labels a connected component.

Global Learners (Big Picture First)

1. First explain the overall goal of component labeling in image analysis, then explain how the algorithm works.
2. Explain how component labeling fits into a complete image processing pipeline (segmentation, analysis, counting).