CAS CS 585
Image and Video Computing

Lecture by Margrit Betke

Projections, Multi-object Labeling, and Face Detection,

January 25, 2024

Last time: Geometric Properties of Object

- Representations of lines
  - Binary Image $B(x,y) = 1$ if $I(x,y) > \tau$,
    $= 0$ otherwise
  - Size, area: $A = \sum_{x=0}^{x=xdim-1} \sum_{y=0}^{y=ydim-1} B(x,y)$

    $A =$ “0th moment”
  - Position: use 1st moment
  - Orientation: use 2nd moments

Class Poll:
Which line equation is wrong?

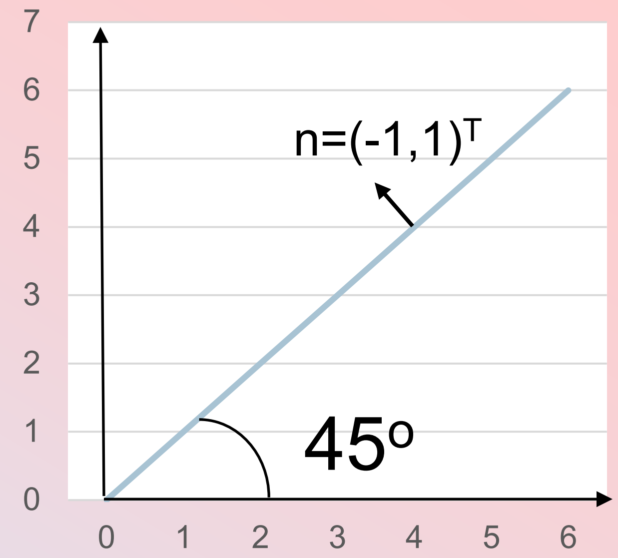
Computer Science

A. $y=mx+n$, where $m=1$, $n=0$

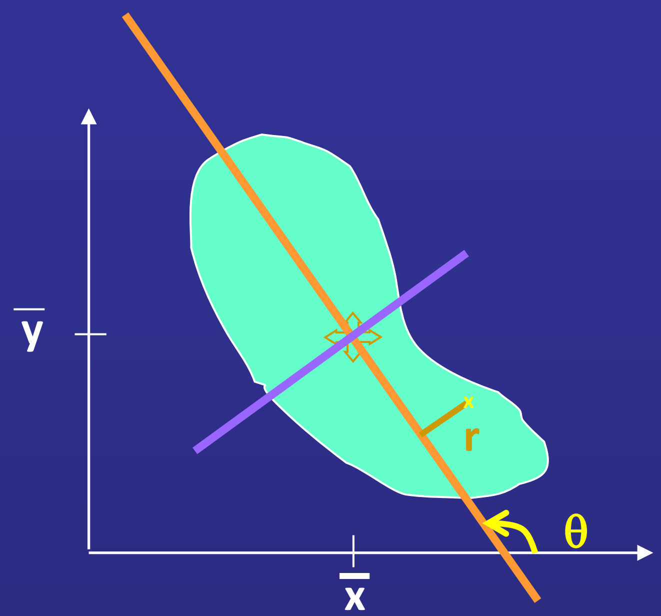
B. $(-1,1)^T \mathbf{x} - g = 0$, where $g=\sqrt{2}$

C. $-x \sin \alpha + y \cos \alpha = 0$, where $\alpha = 45$ degrees

D. $(-\sin \pi/4, \cos \pi/4)^T \mathbf{x} = 0$



Object Shape: Round or Elongated?



Axis of least inertia =
Direction of elongation

$\min E = \min \sum\sum r^2$

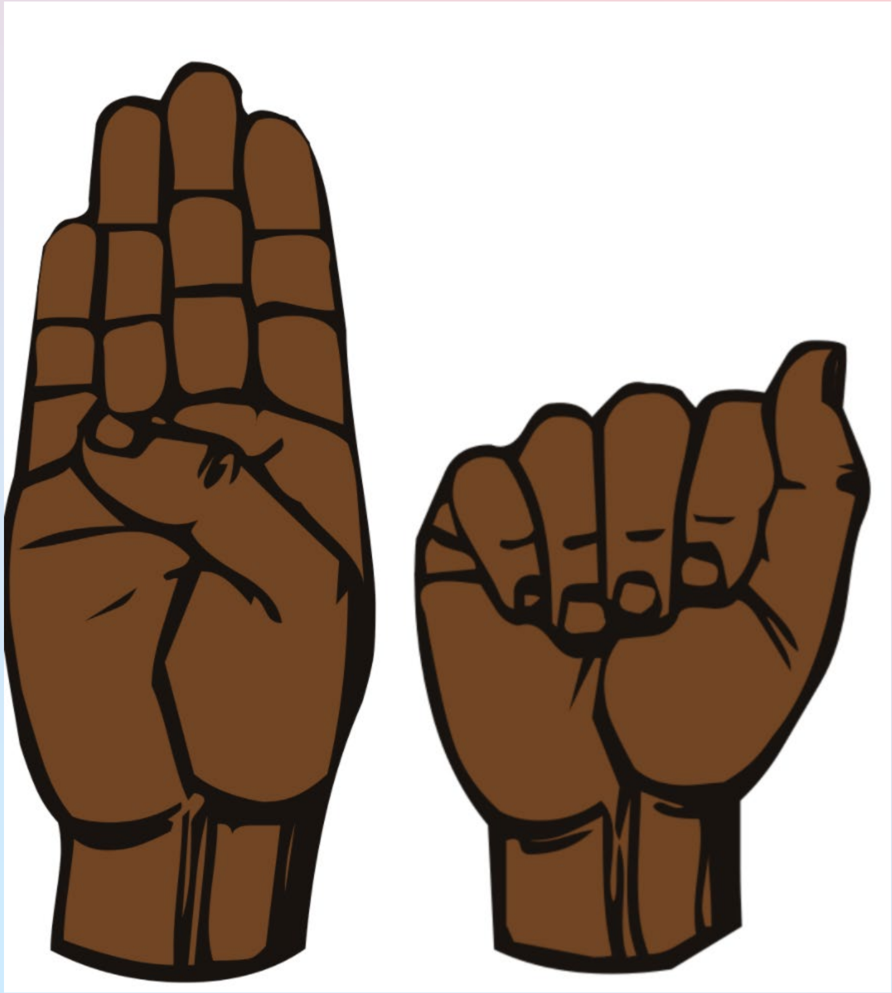
a, b, c =2nd central moments

Circularity: $0 <= E_{min}/E_{max} <= 1$

$$E = \frac{1}{2}(a+c) - \frac{1}{2}(a-c)\cos 2\theta - \frac{1}{2} b \sin 2\theta$$

$$\sin 2\theta = \pm \frac{b}{\sqrt{b^2 + (a-c)^2}} \qquad \cos 2\theta = \pm \frac{a-c}{\sqrt{b^2 + (a-c)^2}}$$

Example Application: American Sign Language Recognition



Sign for A   Sign for B

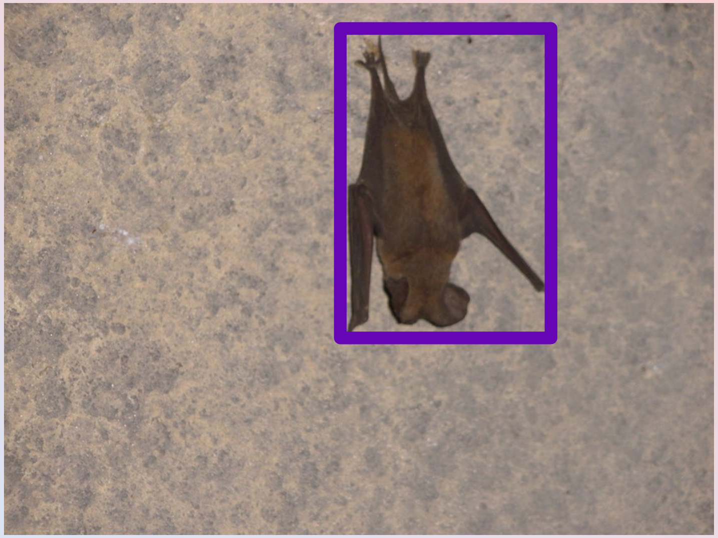
Idea:

Distinguish the two signs by their size and circularity/elongation of hand.

Task: Object Localization

Computer Science

Goal: Find an algorithm that draws a bounding box around the bat



Object Localization Algorithm: Conversion to Binary Image

Computer Science

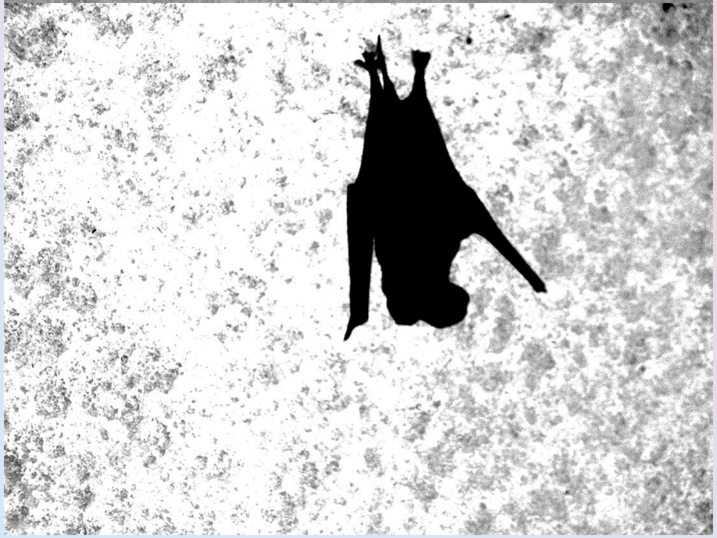
Color image

→

Greyscale image

→

Back-and-white image
= binary image:



Object Localization

Computer Science

1. Find bounding box around black object in grayscale background

Background may contain black pixels

Algorithm ??

Using Black Color & Projections for Object Detection

Computer Science

1. Find bounding box around black object in grayscale background

   Background may contain black pixels

Algorithm:

- Count number of black pixels in each row and column
- Analyze these histograms or projections of black pixels onto x- and y-axes.

Using Black Color & Projections for Object Detection

Computer Science

1. Find bounding box around black object in grayscale background

Background may contain black pixels

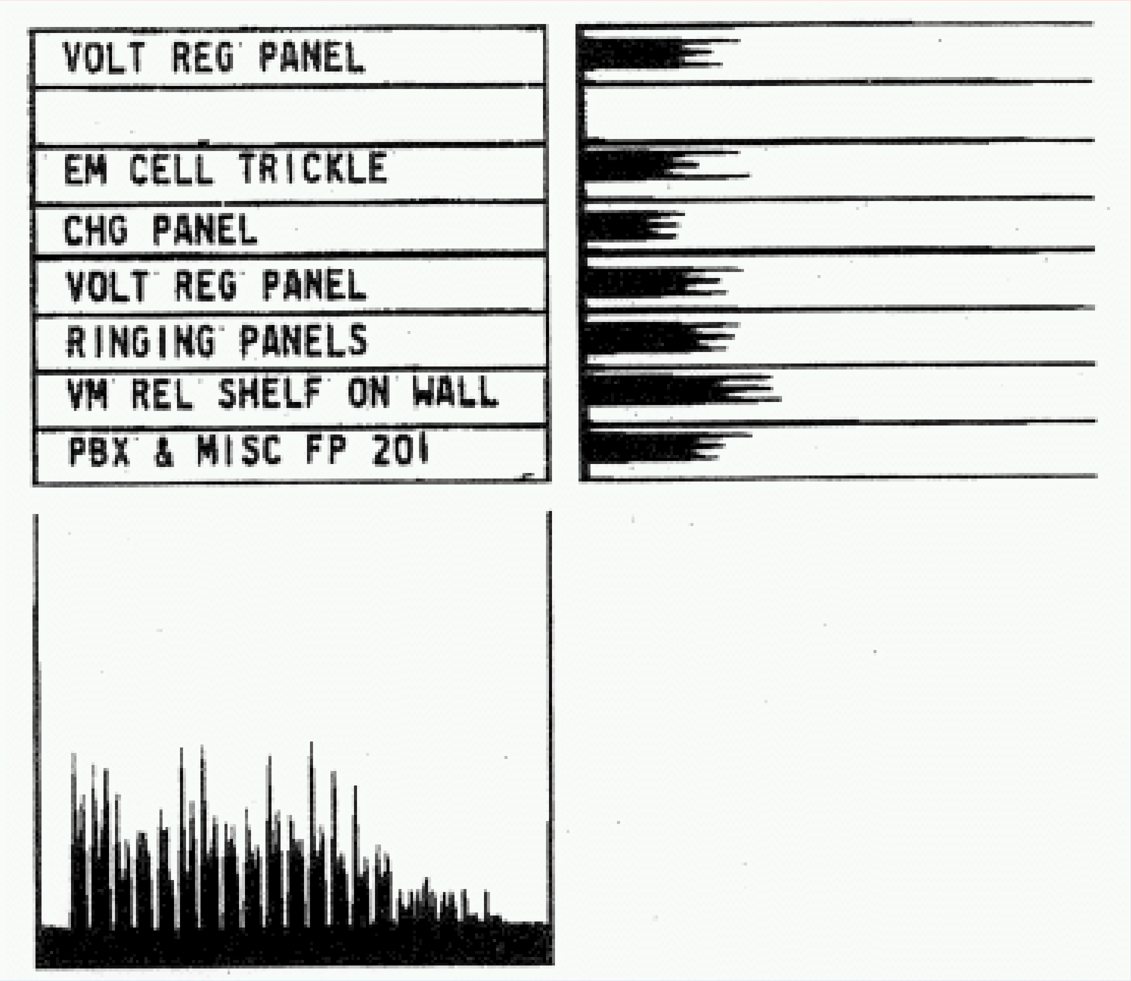
Algorithm:

- Count number of black pixels in each row and column
- Analyze these histograms or projections of black pixels onto x- and y-axes.

$$P1(y) = \sum_x B(x,y), \quad P2(x) = \sum_y B(x,y)$$

Examples of Projections

Computer Science

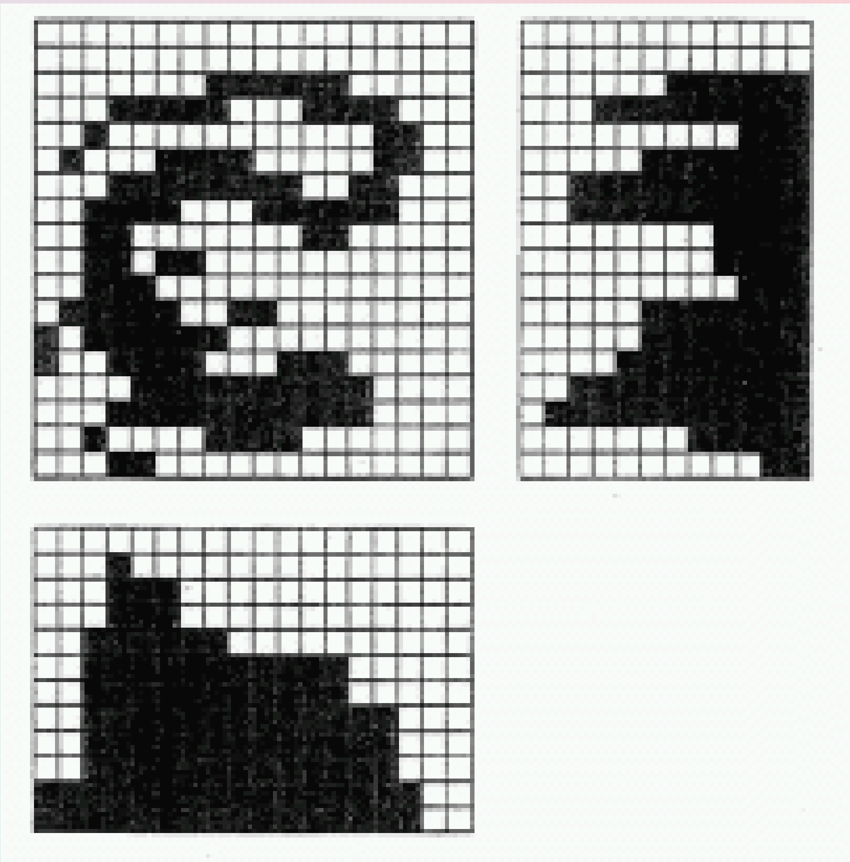


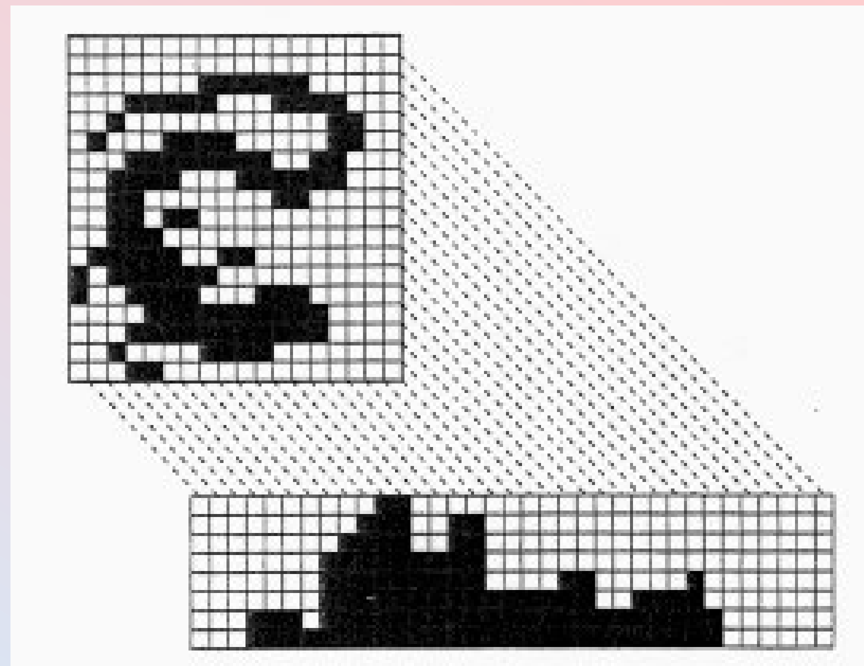
Examples of Projections

Computer Science

Examples of Projections

Computer Science





Images from Machine Vision by
Jain et al.

Using Skin Color for Face Detection

Computer Science

Task: Find bounding box around face in color image

Algorithm ??

Using Skin Color for Face Detection

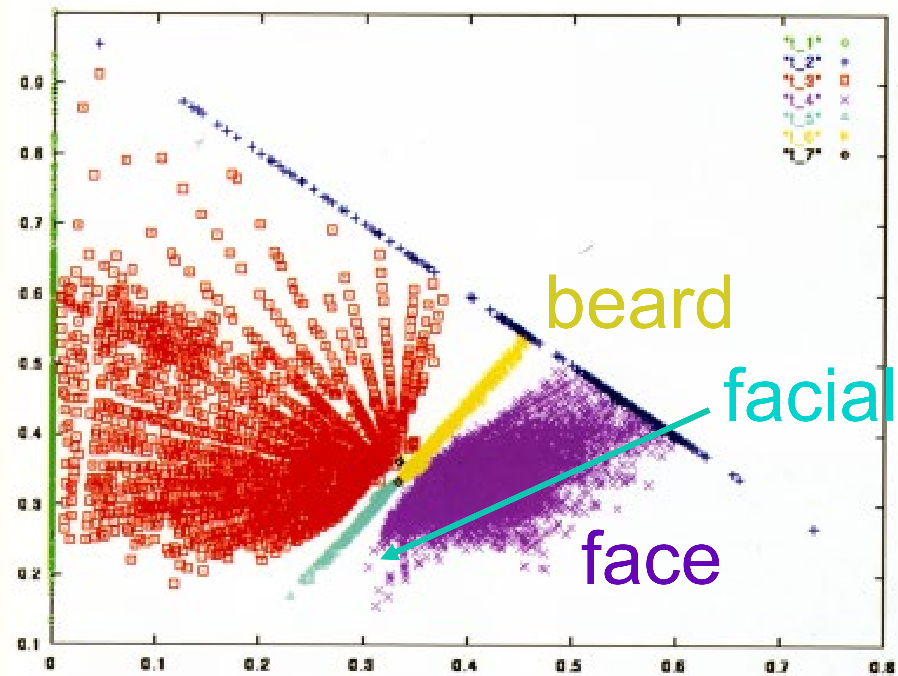
Find bounding box around face in color image

Algorithm: Same idea as for bat example

- Find all pixels with skin color
- Count number of skin color pixels in each row and column
- Analyze these histograms or projections of skin-color pixels onto x- and y-axes.

$P1(y) = \Sigma_x B(x,y), \quad P2(x) = \Sigma_y B(x,y)$

Using skin color for face detection



▲ **Figure 6.12**
Skin color clusters obtained from training: the horizontal axis is Rnorm and the vertical axis is G_{norm}. The cluster labeled as t_4 is the primary face color, cluster t_5 and t_6 are secondary face clusters associated with shadowed or bearded areas of a face. (Figure from V. Bakic.)

▼ **Figure 6.13**
Face extraction examples: (left) input image; (middle) labeled image; and (right) boundaries of the extracted face region. (Images from V.Bakic.)

From Computer Vision by Shapiro & Stockman

Object Localization: Alternative to Projection Algorithm: "Flood Fill"

Computer Science

1. Convert to binary image:

 → 

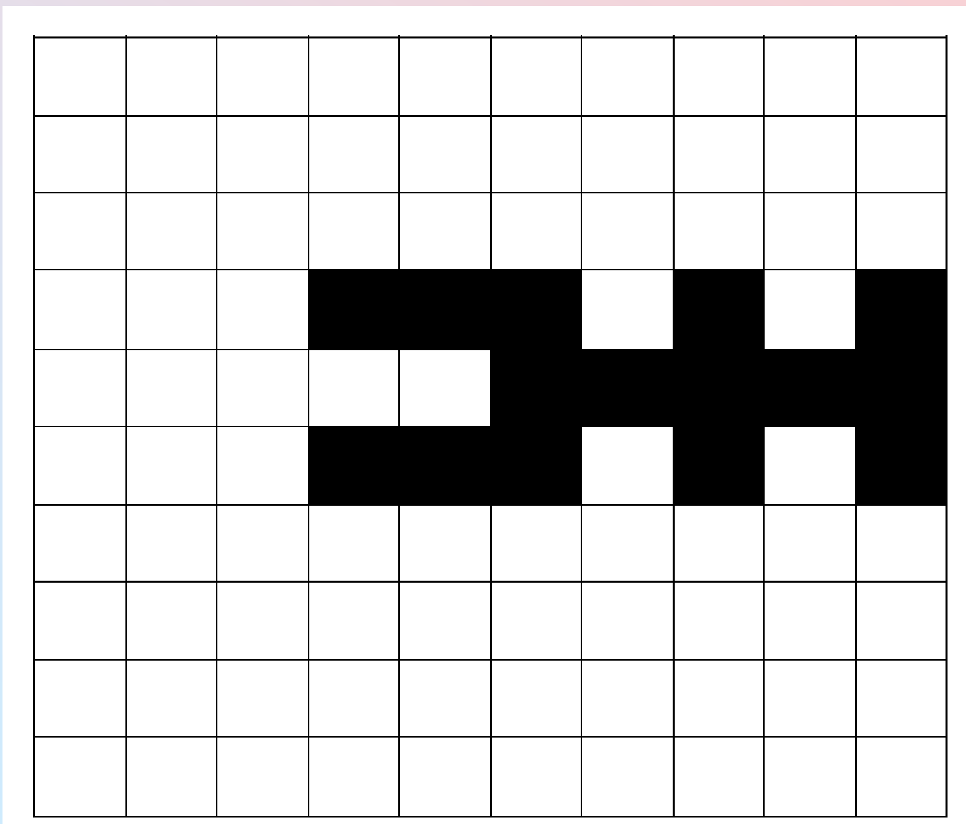
Black = "-1" pixels

White = "1" pixels

2. Scan row by row until first "-1" pixel is reached and label it "object 1."
3. Find all neighbors of the current pixel that are "-1" and assign the object label of the current object in a recursive, depth first search manner.
4. When there are no more "-1" neighbors, continue scanning the image until the next "-1" pixel is reached and label it with the next object label and go to step 3.

Object Localization: Alternative Algorithm: "Flood Fill"

Example



“Flood Fill” first step:

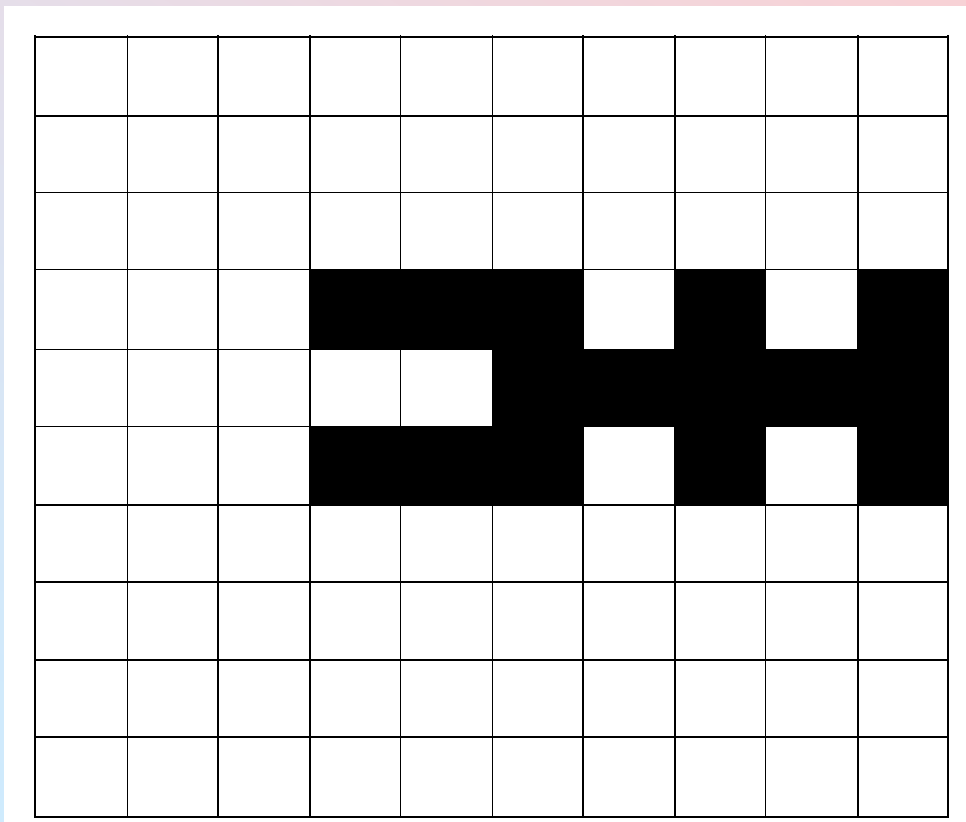
Example

1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	-1	-1	-1	1	-1	1	-1
1	1	1	1	1	-1	-1	-1	-1	-1
1	1	1	-1	-1	-1	1	-1	1	-1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

“Flood Fill” demo

Computer Science

Example



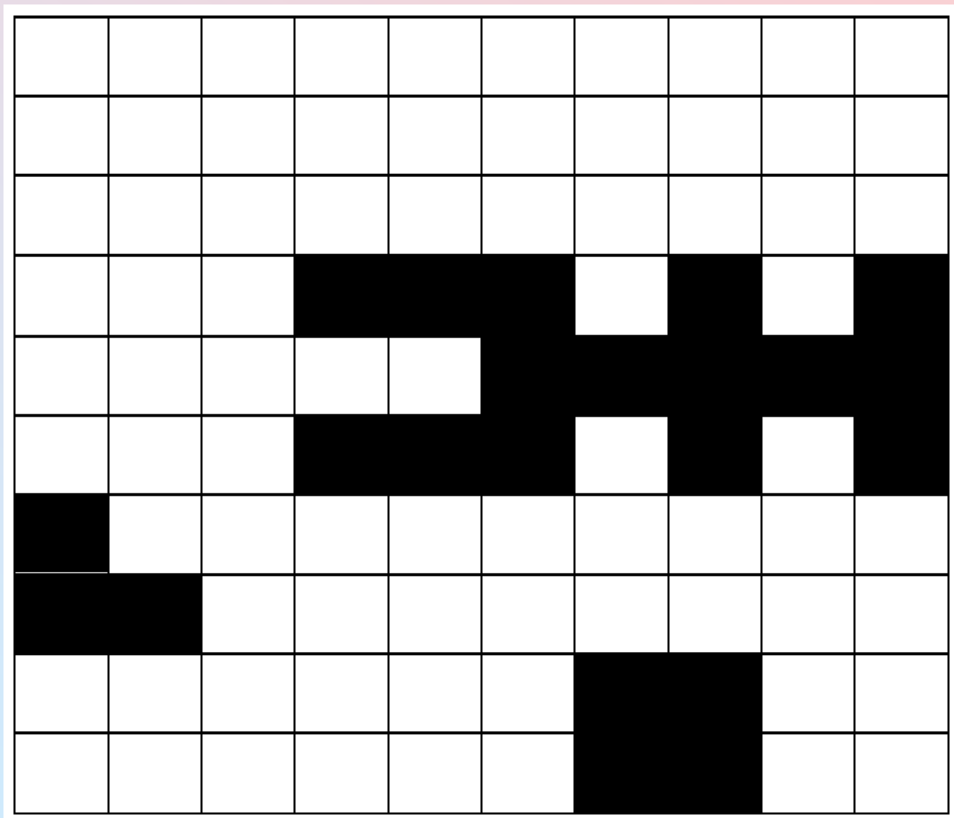
“Flood Fill” Result

Result: Red labeled pixels indicate Object 1

1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

“Flood Fill” as a Multiple Object Labeling Algorithm

Example



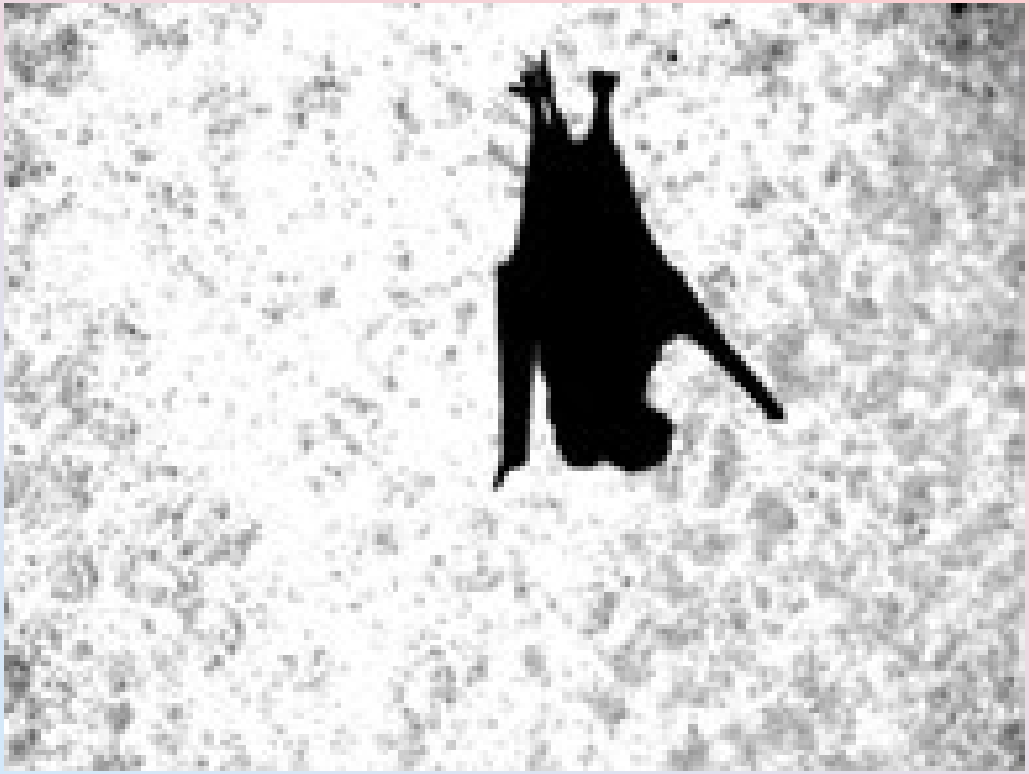
“Flood Fill” as a Multiple Object Labeling Algorithm: Result

Example: Red labeled pixels indicate Objects 1, 2, 3

1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1
2	2	1	1	1	1	1	1	1	1
1	1	1	1	1	1	3	3	1	1
1	1	1	1	1	1	3	3	1	1

Critique of Flood Fill Algorithm

Computer Science



Critique of Flood Fill Algorithm

Computer Science

- Recursive: Can lead to huge stacks

- Alternative: Sequential multi-object labeling algorithm
  - 1st pass through image: Label objects based on local neighborhood
  - 2nd pass solve any ambiguities

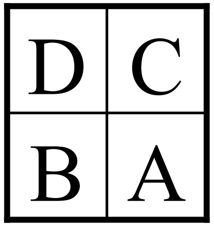
Main Ideas of Sequential Labeling Algorithm:

Scan through the image only twice.
1st scan: Determine object labels.
2nd scan: Fix cases where more than 1 label appears in an object.

During 1st scan: Find the label of pixel 'A' based on the labels of pixels 'B', 'C', and 'D':

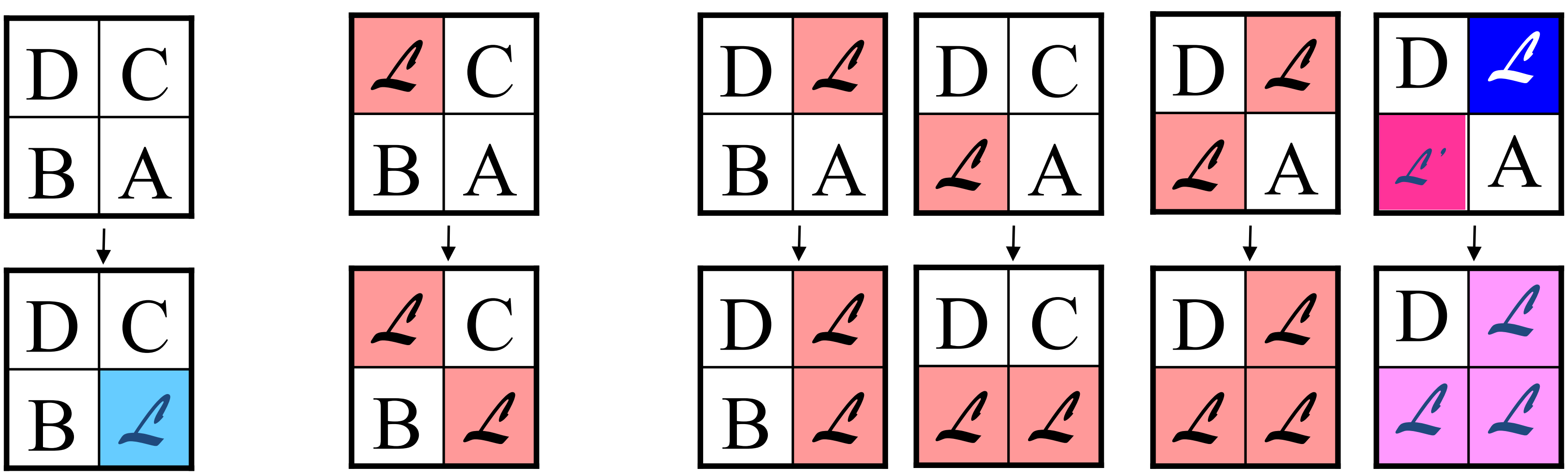
D	C
B	A

Organize the comparison by using this 4 pixel "mask."



Sequential Labeling Algorithm

Case 1:	Case 2:	Case 3:	Case 4:
Neither B,C, or D labeled	D labeled B, C not	Either B or C labeled	B, C labeled same different



Sequential Multiple Object Labeling Algorithm: 1st Phase

1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	2	1	3
1	1	1	1	1	1	1	2’	2’	3’
1	1	1	4	4	1’	1	2’	1	3’
5	1	1	1	1	1	1	1	1	1
5	5	1	1	1	1	1	1	1	1
1	1	1	1	1	1	6	6	1	1
1	1	1	1	1	1	6	6	1	1

Label Equivalence Class:
1, 2, 2’, 3, 3’, 4, 1’

Sequential Multiple Object Labeling Algorithm: 2nd Phase

1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1
2	2	1	1	1	1	1	1	1	1
1	1	1	1	1	1	3	3	1	1
1	1	1	1	1	1	3	3	1	1

Label Equivalence Class: 1, 2, 2', 3, 3', 4, 1' => new Label "1"

5=> 2
6=> 3

Learning Objectives

Computer Science

Can explain and apply to image analysis problems:

- Circularity/elongation measure
- Projections
- Multi-object Labeling: Flood fill & sequential algorithm