Johns Hopkins Engineering

Computer Vision

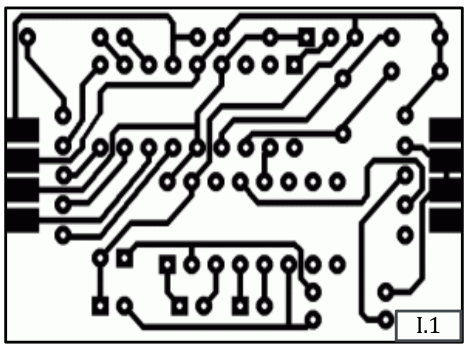
Binary Image Processing

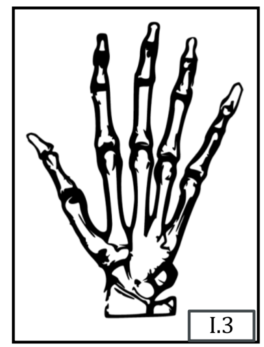
Johns Hopkins
Whiting School of Engineering

What are Binary Images?

Binary Image: Can have only two values (0 or 1).

Simple to process and analyze.





Binary Images: Properties and Methods

- Binary Image: Can have only two values (0 or 1).
- Simple to process and analyze.
- Topics
  1) Geometric Properties
  2) Multiple Objects (Connectivity)
  3) Binary Morphology

Representation

- A (grey) image I is a function $$I : \begin{cases} \Omega \subset \mathbb{R}^2 & \longrightarrow & \mathbb{R} \\ p = (x, y) & \mapsto & I(x, y) \end{cases}$$

- Represented, after sampling and quantization, by a matrix

(0,0)

x

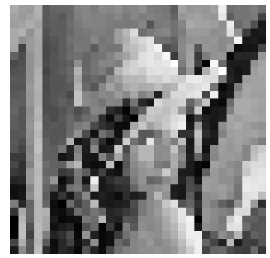
y

Representation

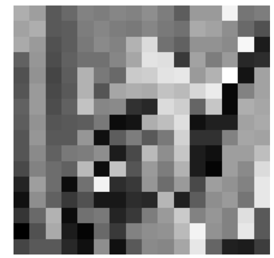
256x256

128x128

64x64



32x32



16x16

Number of pixels

32

16

8

4

2

Number of gray levels

Making Binary Images

Binary Image $b(x, y)$: Usually obtained from Gray-level (or other) image $g(x, y)$ by Thresholding.

Characteristic Function:

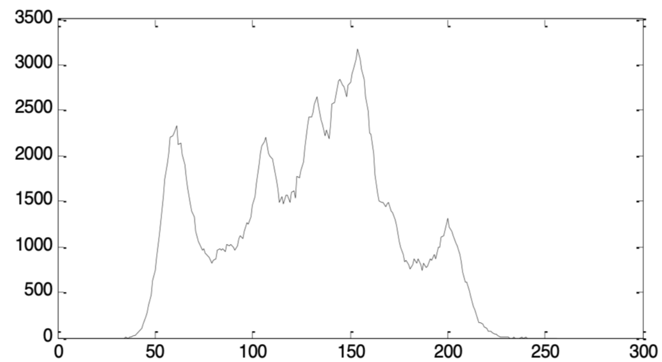
$$b(x,y) = \begin{cases} 0, & g(x,y) < T \\ 1, & g(x,y) \geq T \end{cases}$$

Histograms

I

512x512

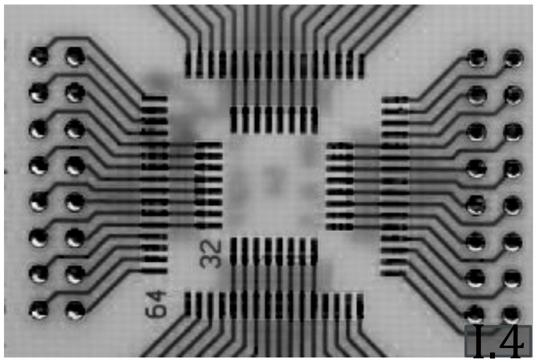
H



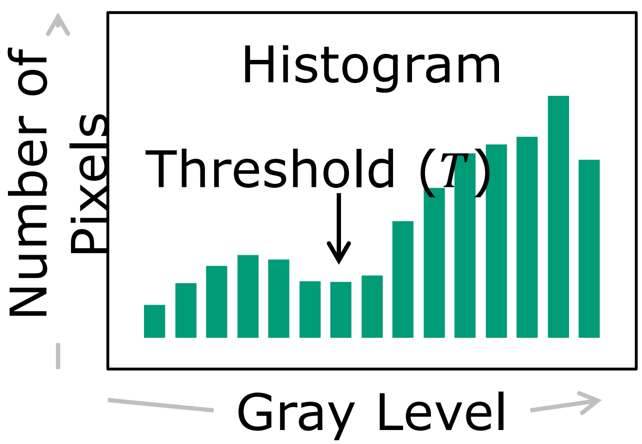
256 gray levels

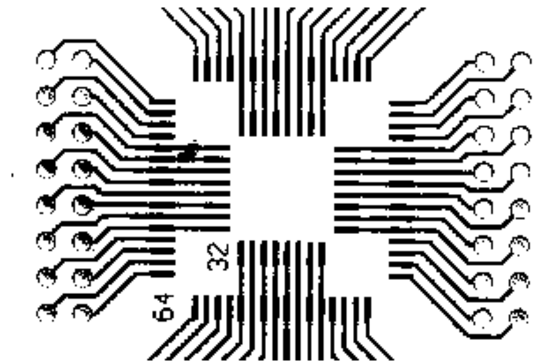
- H(x) is the number of pixels in image I with grey value x
- Probability of observing grey value x ? $$p(x) = \frac{H(x)}{s_x \times s_y}$$
- Invariant to pixel permutations

Selecting a Threshold (T)



Gray Image $g(x,y)$



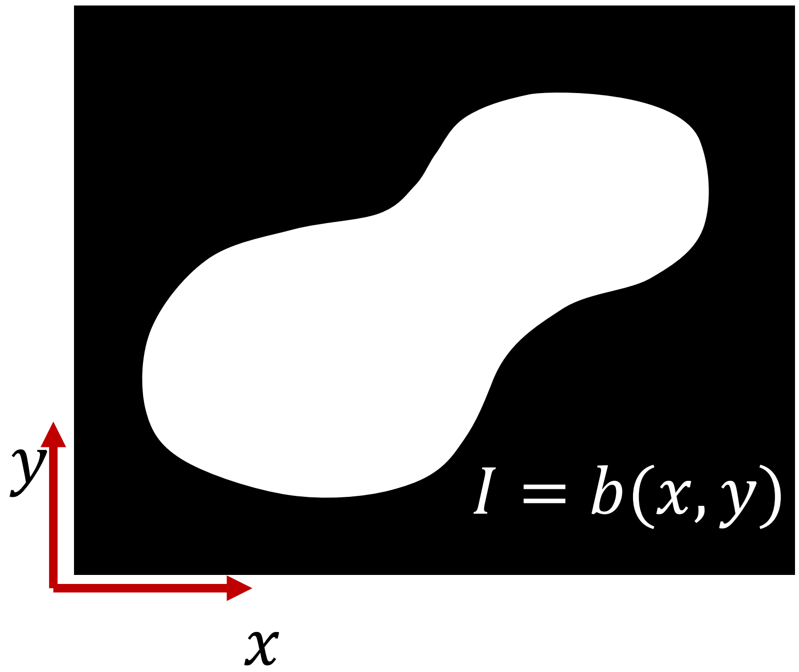


Binary Image $b(x,y)$

Geometric Properties of Binary Images

Assume:

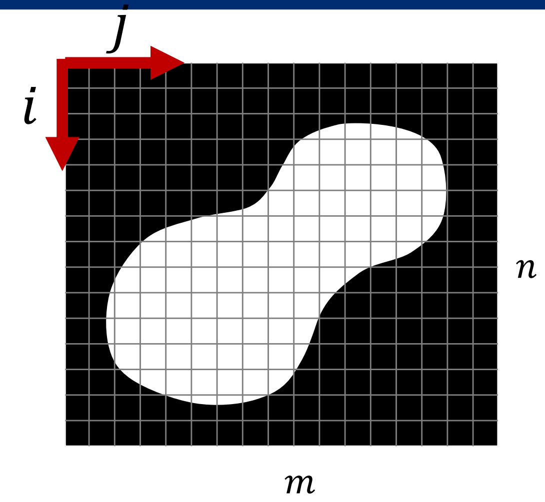
- $b(x, y)$ is continuous
- Only one object



Discrete Binary Images

$\boldsymbol{b_{ij}}$: Value at cell (pixel) in row i and column j.

Assume pixel area = 1.



Area: $$A = \sum_{i=1}^{n} \sum_{j=1}^{m} b_{ij}$$

Position: Center of Area (First Moment)

$$\bar{x} = \frac{1}{A} \sum_{i=1}^{n} \sum_{j=1}^{m} j b_{ij} \qquad \bar{y} = \frac{1}{A} \sum_{i=1}^{n} \sum_{j=1}^{m} i b_{ij}$$

Discrete Binary Images (cont.)

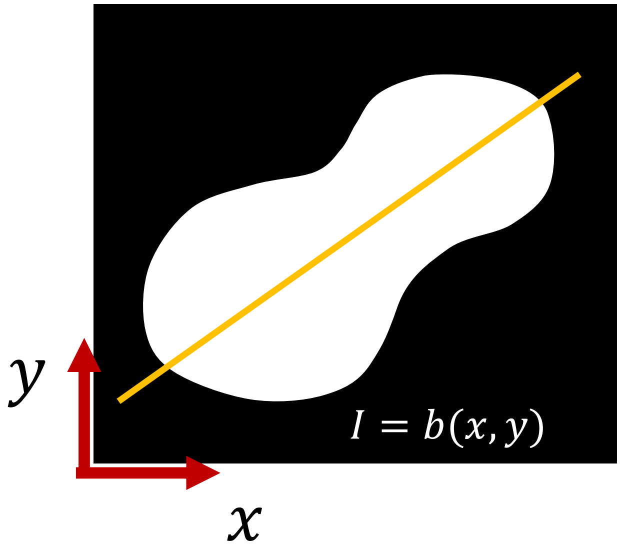
Second Moments:

$$a' = \sum_{i=1}^{n}\sum_{j=1}^{m} i^2 b_{ij} \qquad b' = 2\sum_{i=1}^{n}\sum_{j=1}^{m} ij b_{ij} \qquad c' = \sum_{i=1}^{n}\sum_{j=1}^{m} j^2 b_{ij}$$

- **Note**: a', b', c' are second moments w.r.t **origin**.
  a, b, c (w.r.t. **center**) can be found from a', b', c', $\bar{x}$, $\bar{y}$, A
- **Hint**: Expand $a = \sum_{i=1}^{n}\sum_{j=1}^{m}(i - \bar{y})^2 b_{ij}$ and represent in terms of a', $\bar{y}$, A.

Orientation

Difficult to define!



Use: Axis of Least Second Moment

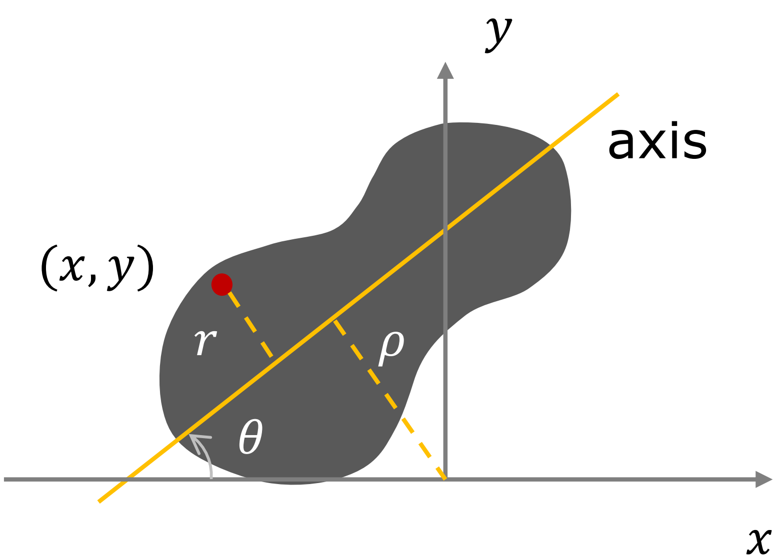
Orientation (cont.)

Axis of Least Second Moment minimizes:

$$E = \iint_I r^2 \, b(x, y) \, dx \, dy$$

Which equation to use for axis?

$$y = mx + b \ ? \qquad -\infty \leq m \leq \infty$$



Use: $\boxed{x \sin\theta - y \cos\theta + \rho = 0}$ ρ, θ are finite

Find ρ and θ that minimize E for given $b(x, y)$

Recall Polar Coordinates

x = r * cos(t)
y = r * sin(t)

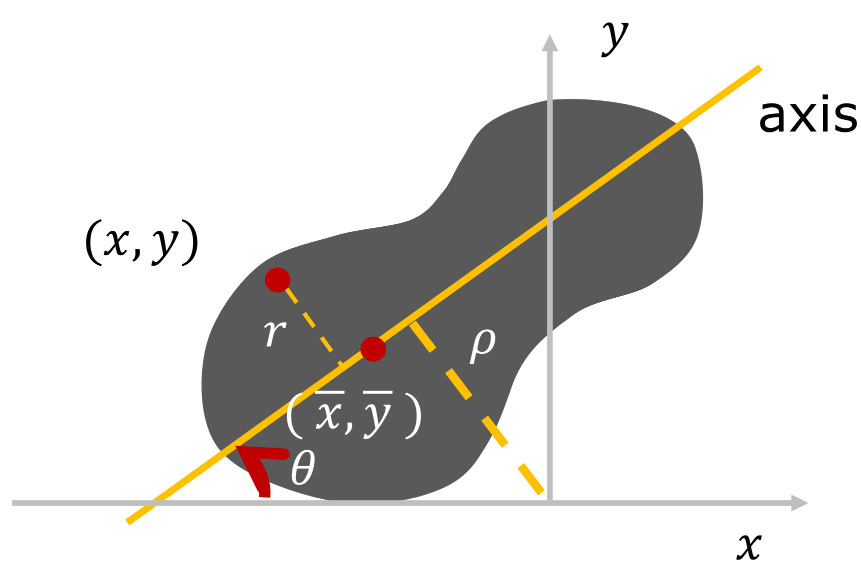
r = sqrt(x*x + y*y)
t = atan2(y,x)

Minimizing Second Moment

We can show that for any point (x, y):

$$r = x \sin\theta - y \cos\theta + \rho$$

[See Appendix A]



So, minimize:

$$E = \iint_I (x \sin\theta - y \cos\theta + \rho)^2 b(x, y)\, dx\, dy$$

Using $\dfrac{\partial E}{\partial \rho} = 0$ we get: $A(\underline{\overline{x} \sin\theta - \overline{y} \cos\theta + \rho}) = 0$

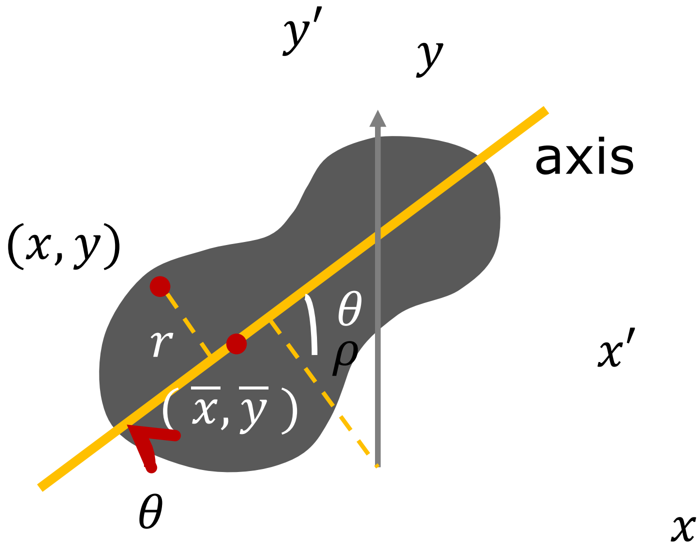
Axis passes through center $(\overline{x}, \overline{y})$!

Shift the Coordinate System

Change coordinates:

$x' = x - \overline{x},\ y' = y - \overline{y}$

$x \sin\theta - y\cos\theta + \rho$
$= x' \sin\theta - y' \cos\theta$



Therefore, we can rewrite E as:

$$\boxed{E = a\sin^2\theta - b\sin\theta\cos\theta + c\cos^2\theta}$$

where:

$$a = \iint_{I'} (x')^2\, b(x,y)\, dx'\, dy' \qquad b = 2\iint_{I'} (x'y')\, b(x,y)\, dx'\, dy'$$

$$c = \iint_{I'} (y')^2\, b(x,y)\, dx'\, dy'$$

(a, b, c are easy to compute)

Finally, Minimize E

Using $\frac{dE}{d\theta} = (a - c)\sin 2\theta - b\cos 2\theta = 0$ we get: $\boxed{\tan 2\theta = \frac{b}{a - c}}$

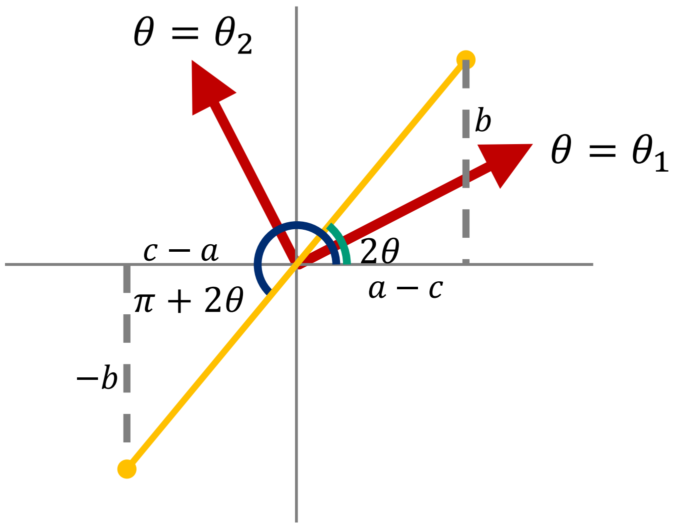
We know that: $\tan 2\theta = \tan(2\theta + \pi) = \frac{-b}{c - a}$

θ has two solutions.

1. $\theta = \theta_1$

2. $\theta = \theta_2 = \theta_1 + \frac{\pi}{2}$

One gives **Minimum of E**
and the other **Maximum of E**



Which One To Use?

Using second derivative test:

If $\dfrac{d^2E}{d\theta^2} = (a-c)\cos 2\theta + b\sin 2\theta$ > 0 then Minimum

< 0 then Maximum

Substituting $\cos 2\theta_1$, $\sin 2\theta_1$, $\cos 2\theta_2$ and $\sin 2\theta_2$:

$$\frac{d^2E}{d\theta^2}(\theta_1) > 0 \quad \text{and} \quad \frac{d^2E}{d\theta^2}(\theta_2) < 0$$

Therefore, Orientation: $\boxed{\theta = \theta_1 = \dfrac{atan2(b/a-c)}{2}}$

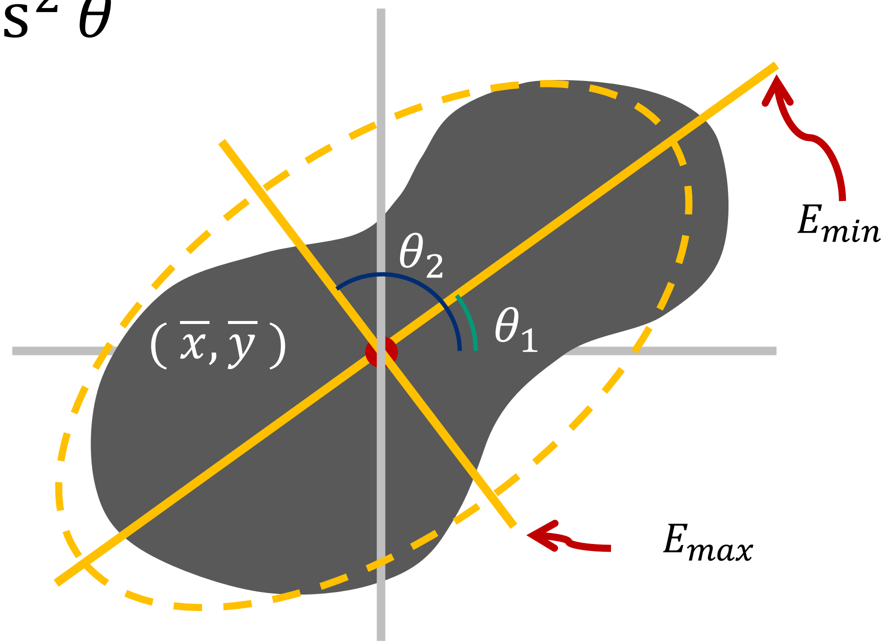
Roundedness

$$E = a\sin^2\theta - b\sin\theta\cos\theta + c\cos^2\theta$$

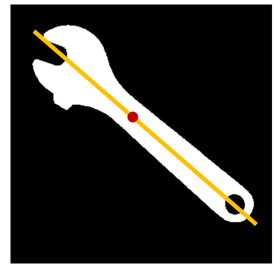
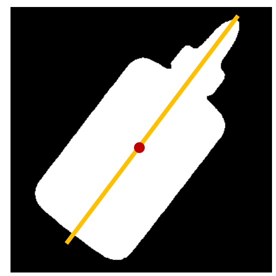
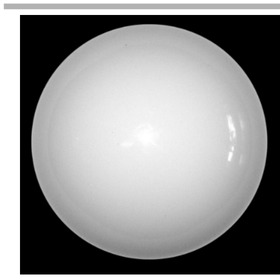
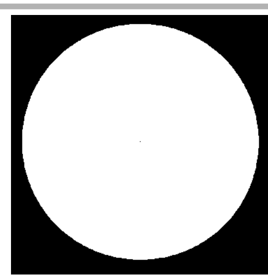
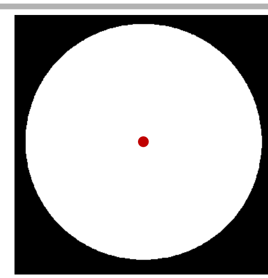
Roundedness = $\dfrac{E_{min}}{E_{max}}$

where:

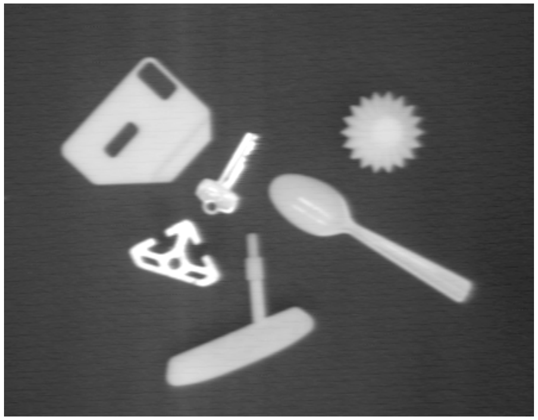
$E_{min} = E(\theta_1)$ and $E_{max} = E(\theta_2)$



Examples

Gray Image	Binary Image	Orientation	Roundedness
			0.19
			0.49
			1.0

Multiple Objects

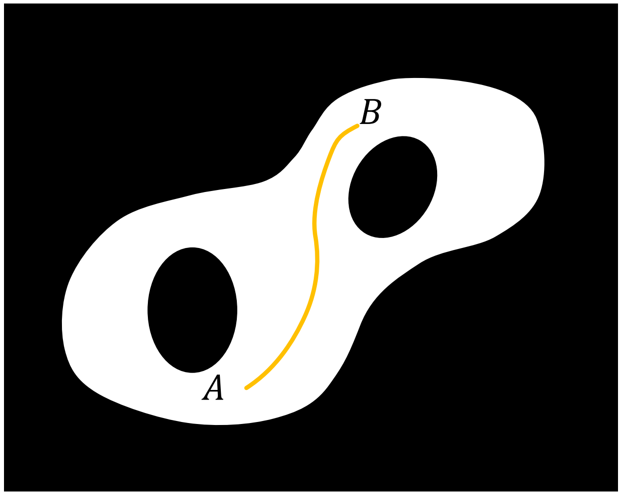


Need to **Segment** image into separate **Components**

Non-Trivial!

Connected Component

Maximal Set of Connected Points



A and B are connected if path exists between A and B along which $b(x, y)$ is constant.

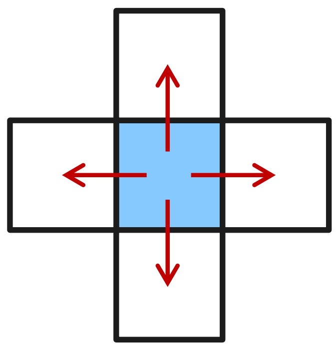
Connected Component Labeling

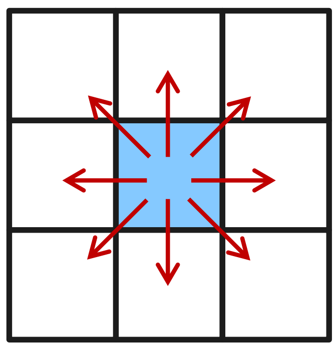
Region Growing Algorithm

a) Find **Unlabeled** “**Seed**” point with $b = 1$. If not found, Terminate.
b) Assign **New Label** to seed point
c) Assign **Same Label** to its Neighbors with $b = 1$
d) Assign **Same Label** to Neighbors of Neighbors with $b = 1$. Repeat until no more Unlabeled. Neighbors with b=1.
e) Go to (a)

What do we mean by Neighbors?

Connectedness



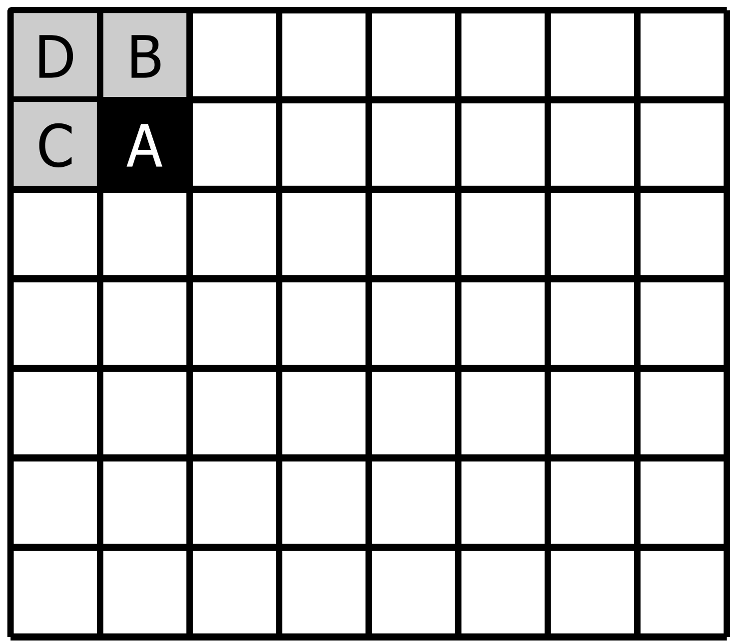


4-Connectedness
4-C

8-Connectedness
8-C

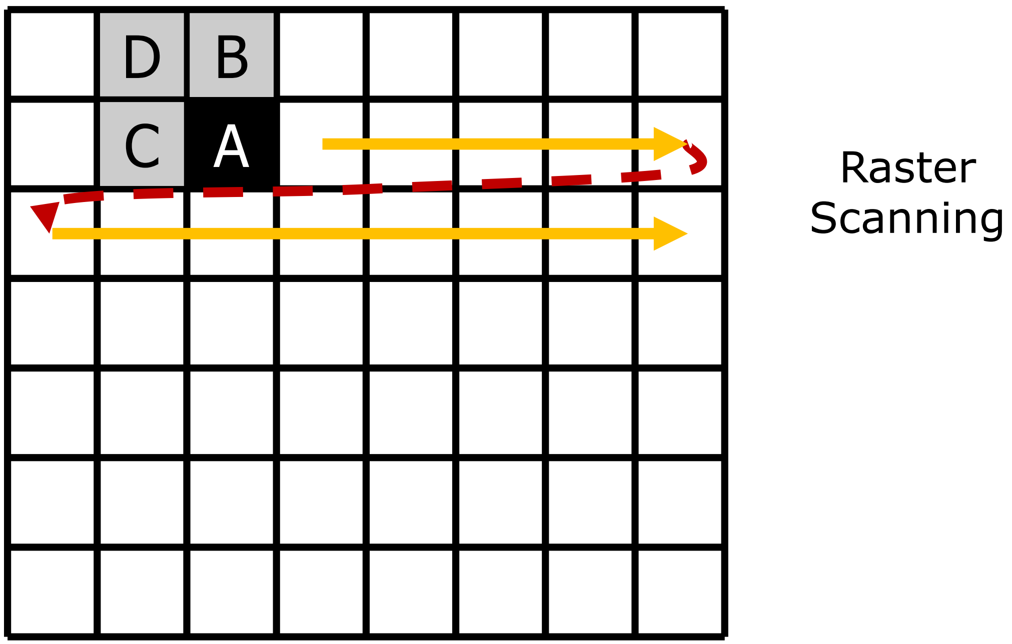
Neither is Perfect!

Sequential Labeling Algorithm



We want to label A.
B, C, D are already labeled.

Sequential Labeling Algorithm



We want to label A.
B, C, D are already labeled.

Sequential Labeling Algorithm (3)

X	X
X	0

→ label(A) = "background"

0	0
0	1

→ label(A) = new label

D	X
X	1

→ label(A) = label(D)

0	0
C	1

→ label(A) = label(C)

0	B
0	1

→ label(A) = label(B)

0	B
C	1

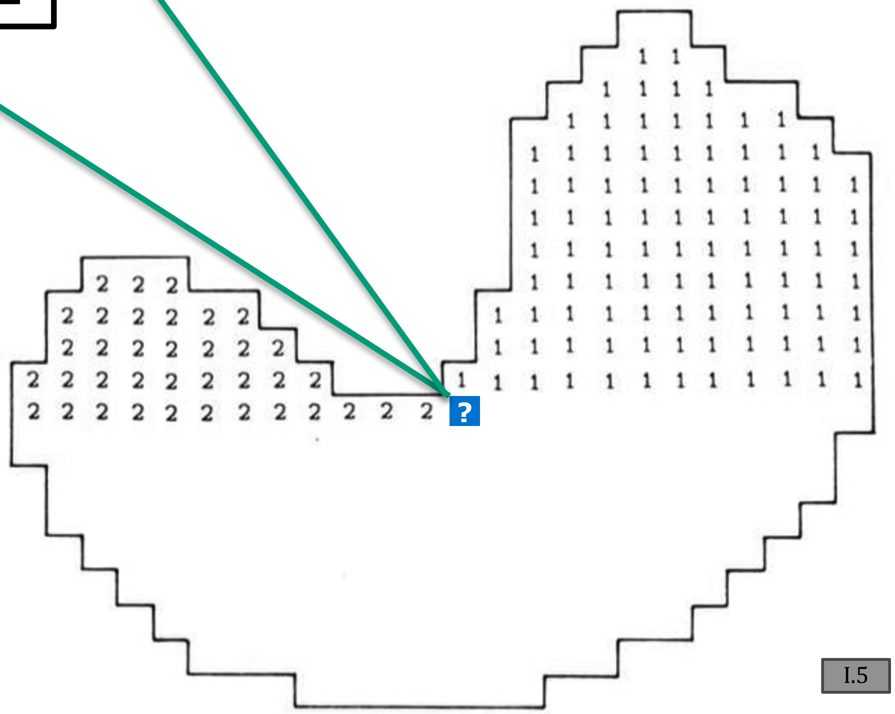
→ If
label(B) = label(C)
then,
label(A) = label(B)

X: Value does not matter (Can be 0 or 1)

Sequential Labeling Algorithm (4)

0	B
C	1

→ What if label(B) not equal to label(C)?



Sequential Labeling Algorithm

0	B
C	1

→ What if label(B) not equal to label(C)?

Solution: Create Equivalence Table

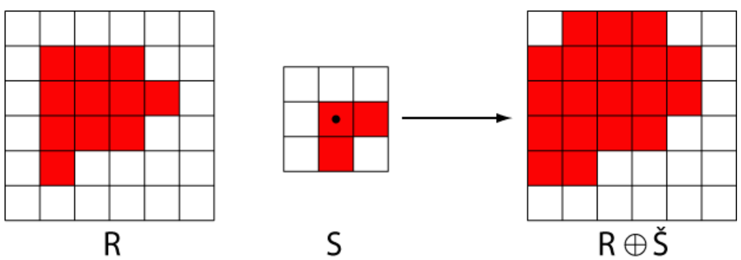
- Note down that label(B) ≡ label(C)
- Assign label(A) = label(B)

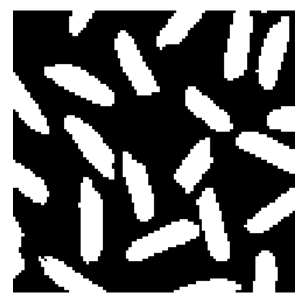
Resolve Equivalence in Second Pass

Morphological operators

Binary Dilation

- Defined by a Morphological structuring element ***S*** (a binary template)
- Images are represented by the sets $(\subset Z^2)$ containing the positions of their non-zero elements
- Binary dilatation $D(R,S) = R \oplus S = \{u - v | u \in R, v \in S\}$
- (Intuitively: set of all possible positions of the center of ***S*** such that the two patterns overlap by at least one element)



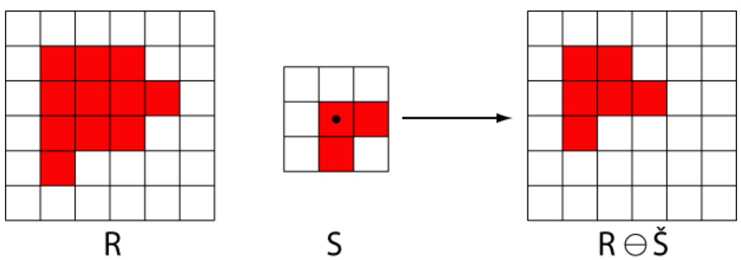

Original binary image

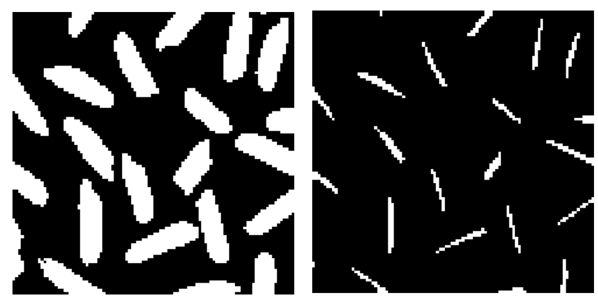

Dilated image

$$S = \{(0,0),(1,0),(0,1)\}$$
$$R = \{(1,1),(1,2),(1,3),(2,1),(2,2),(2,3),(2,4),(3,1),(3,2),(3,3),(4,1)\}$$

Binary Erosion

- Defined by a Morphological structuring element ***S***
- Binary erosion $E(R,S) = R \ominus S = \{u \mid \forall v \in S, u+v \in R\}$
- (Intuitively: all positions of the center of S such that pattern ***S*** is contained in pattern R)





Original binary image   Eroded image

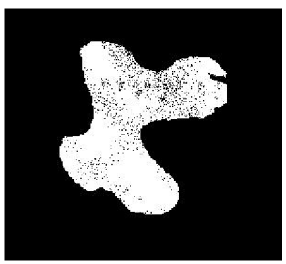
$$S = \{(0,0),(1,0),(0,1)\}$$
$$R = \{(1,1),(1,2),(1,3),(2,1),(2,2),(2,3),(2,4),(3,1),(3,2),(3,3),(4,1)\}$$

Binary Closing

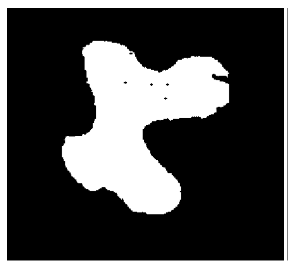
- Defined by a Morphological structuring element S
- Binary closing
  - Properties: $C(R,S) = E(D(R,S),S)$

    Fill the **holes smaller** than the structuring elements

    Smooth the contours by filling the cavities



Original binary image



Radius of the structuring element R = 1

R = 3

R = 10

Binary Opening

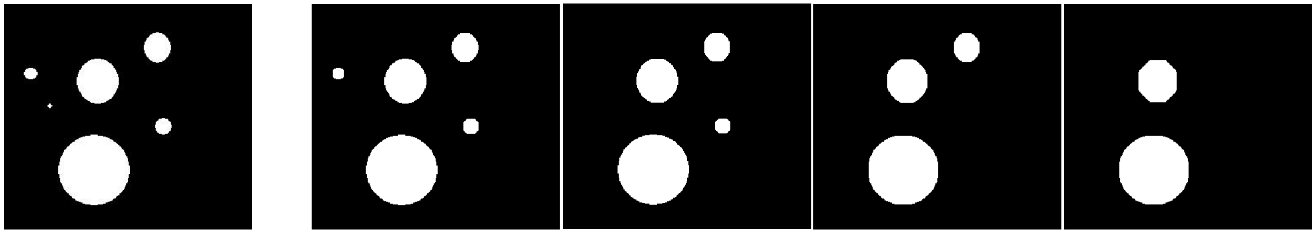
- Defined by a Morphological structuring element ***B***
- Binary opening
  - Properties: $O(R,S) = D(E(R,S),S)$

    Suppress the **structures smaller** than the structuring elements

    Delete the link between weak connected components

    Smooth the contours by deleting the outgrowths

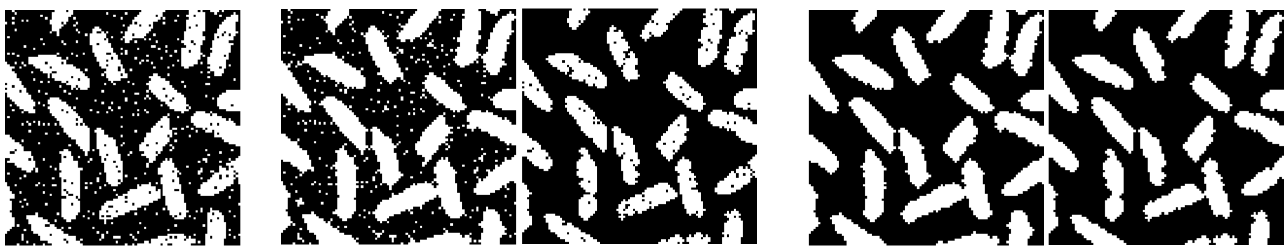
    Applications: **Granulomet**



Original binary image R = 5 R = 7 R = 10 R = 20

Binary Examples

- Removing the noise perturbation
  - Close-open operation: $O\left(C(R,S),S\right)$
  - Open-close operation: $C\left(O(R,S),S\right)$



Original binary image | Close | Open | Close-open | Open-Close

- Note: morphological operations can be generalized to grey value images

References: Textbooks

- Computer Vision: Algorithms and Appications (Chapter 3.3-3.4) Recommended Reading
- Szelinski, 2011 (available online)
- Digital Image Processing (Chapter 3 and 4)
- González, R and Woods, R., Prentice Hall
- Computer Vision: A Modern Approach (Chapter 7)
- Forsyth, D and Ponce, J., Prentice Hall
- Robot Vision (Chapter 3, 4)
- Horn, B. K. P., MIT Press
- Robot Vision (Chapter 6 and 7)
- Horn, B. K. P., MIT Press

Image Credits

I.1 http://en.wikipedia.org/wiki/File:Fourier2.jpg

I.2 http://www.instructables.com/image/FY1T8VKG79F1MO7/Rubiks-cubepranks.jpg

I.3 Matlab Demo Image

I.4 Matlab Demo Image

I.5 http://en.wikipedia.org/wiki/File:Moire_pattern_of_bricks.jpg

I.6 http://www.todayandtomorrow.net/wp-content/uploads/2010/06/shirt_video.jpg

I.7 http://www.svi.nl/wikiimg/StFargeaux_kasteel_buiten1_aliased.jpg

I.8 http://learn.hamamatsu.com/articles/images/lenslet.jpg

Image Credits

I.9 http://www.astrosurf.com/luxorion/Physique/nikon-d200-low-pass-ir.jpg

I.10 http://en.wikipedia.org/wiki/File:Fingerprint_picture.svg

I.11 http://www.thinkmore.in/software-tools-for-pcb-design/

I.12 http://www.clker.com/clipart-hand-x-ray.html

I.13 http://www.greenbang.com/wp-content/uploads/2009/02/green-pcb.jpg

I.14 B. K. P. Horn, Robot Vision, Figure 4-4

I.15 B. K. P. Horn, Robot Vision, Figure 4-7

JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING