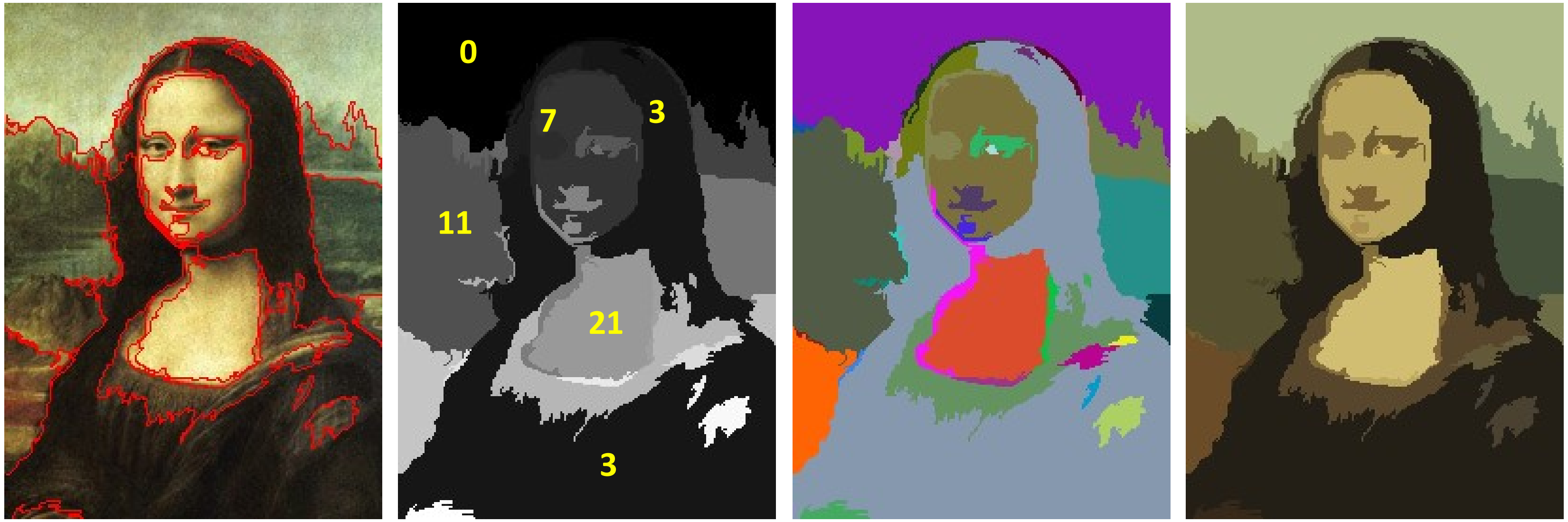
Basic Image Manipulation

What is segmentation?

- Segmentation divides an image into groups of pixels
- Pixels are grouped because they share some local property (gray level, color, texture, motion, etc.)



boundaries **labels** **pseudocolors** **mean colors**

(different ways of displaying the output)

algorithm used: Pedro F. Felzenszwalb and Daniel P. Huttenlocher, Efficient Graph-Based Image Segmentation, IJCV, 59(2), 2004

S. Birchfield, Clemson Univ., ECE 847, http://www.ces.clemson.edu/~stb/ece847

Segmentation

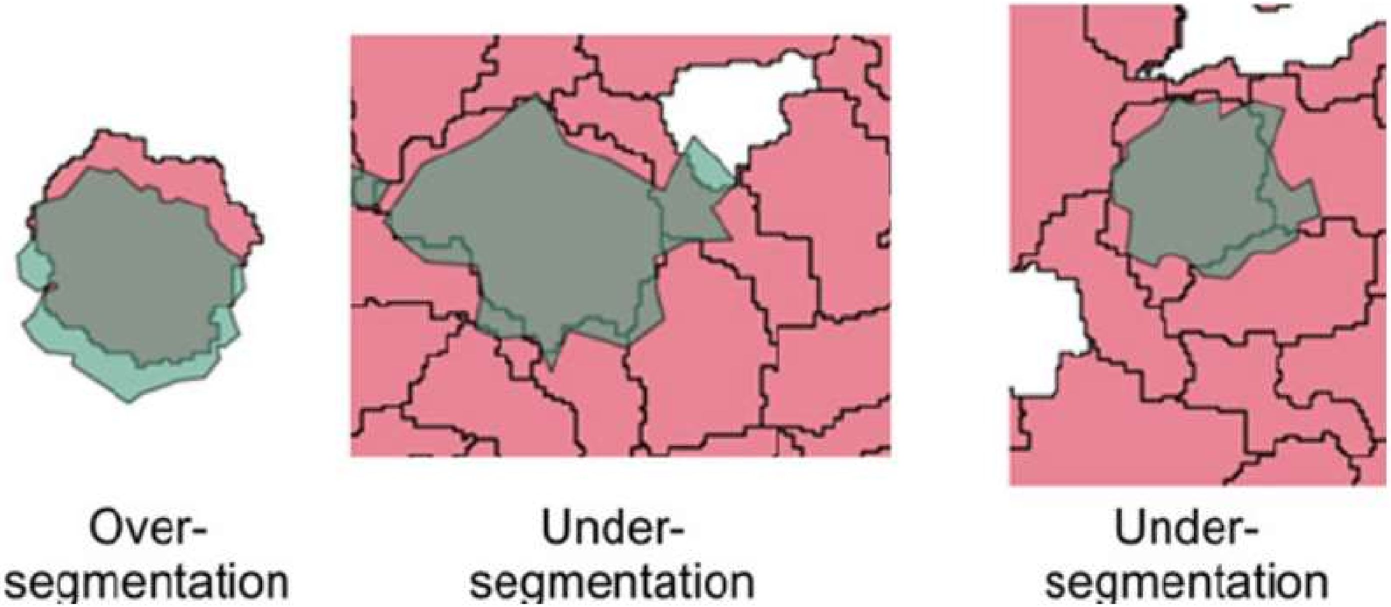
- *Segmentation = partitioning*
  Divide image based on pixel similarity
  - Divide spatiotemporal volume based on image similarity (shot detection)
  - Figure / ground separation (background subtraction)
  - Regions can be overlapping (layers)

Foreground / background separation

Background subtraction provides figure-ground separation,
which is a type of segmentation

C. Wren et al., Pfinder: Real-Time Tracking of the Human Body
http://vismod.media.mit.edu/vismod/demos/pfinder/

Oversegmentation vs undersegmentation



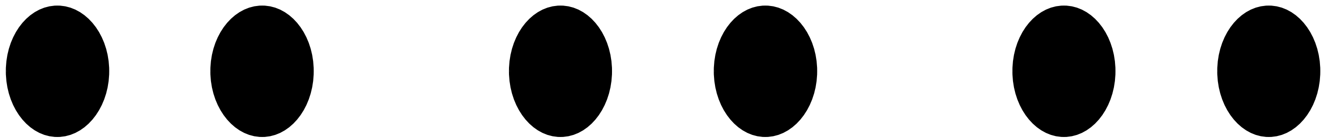
Oversegmentation versus Undersegmentation. The green polygons are the manually digitized polygons that were overlaid on multiresolution segmentation polygons (shown in pink).

An experiment:

What do you see?

Just six dots

Now what do you see?



Three groups of dot pairs

Why?

Dots that are close together ("proximity")
are grouped together by the human visual system

S. Birchfield, Clemson Univ., ECE 847, http://www.ces.clemson.edu/~stb/ece847

And now?

Again, three groups of dot pairs

Why?

Dots are similar in appearance ("similarity")

How about now?

Again, three groups of dot pairs

Why?

Dots move similarly ("common fate")

S. Birchfield, Clemson Univ., ECE 847, http://www.ces.clemson.edu/~stb/ece847

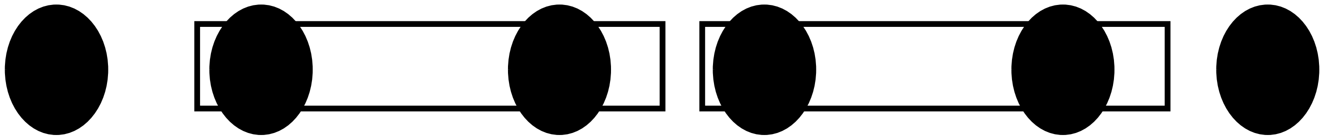
Last one

Again, three groups of dots

Why?

Dots are enclosed together ("common region")

But wait!



Note that the “common region” can overwhelm the “proximity” tendency

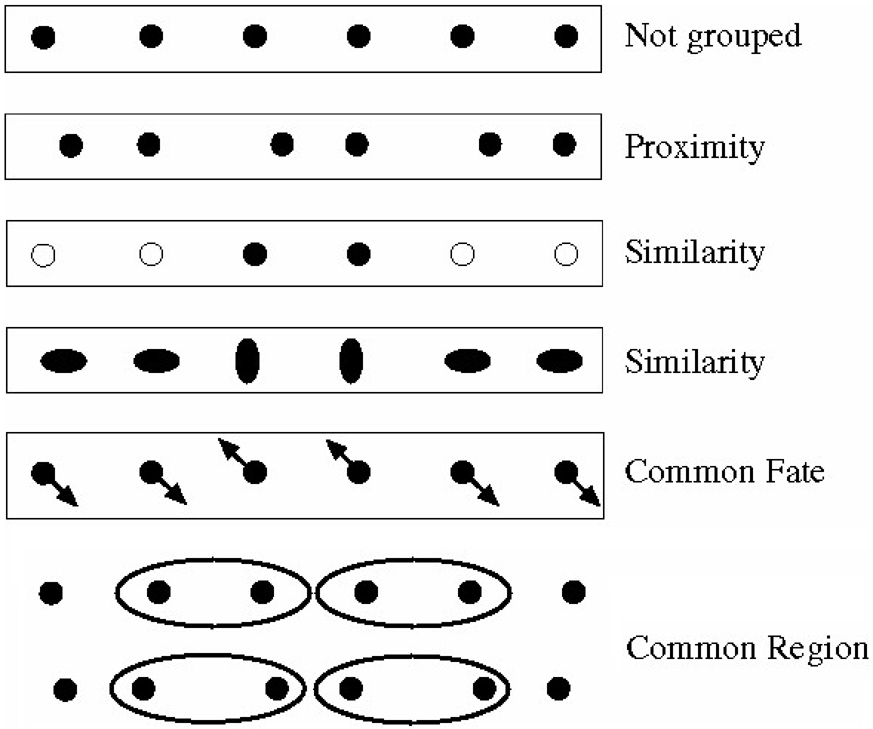
Gestalt psychology

Gestalt school of psychologists emphasized grouping as the key to understanding visual perception.

Recall: Context affects how things are perceived

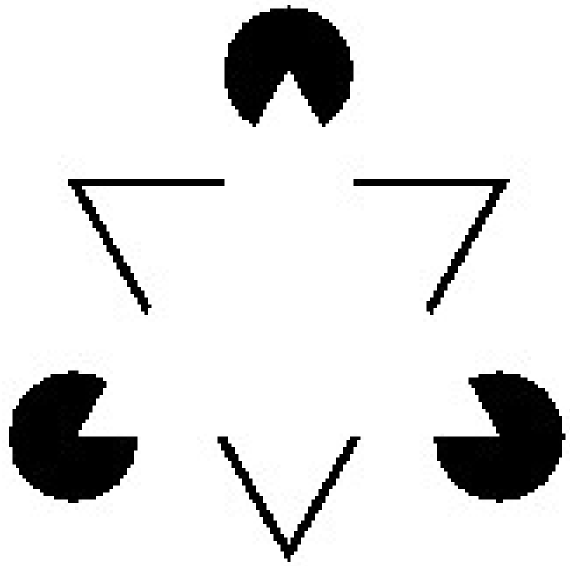
***gestalt* – whole or group**

***gestalt qualitat* – set of internal relationships that makes it a whole**



Max Wertheimer, Laws of Organization in Perceptual Forms, 1923
http://psy.ed.asu.edu/~classics/Wertheimer/Forms/forms.htm

Can you see anything invisible?



These are illusory contours, formed by grouping the circles

This is the well-known Kanizsa triangle

S. Birchfield, Clemson Univ., ECE 847, http://www.ces.clemson.edu/~stb/ece847

Segmentation as partitioning

- A *partition* of image is collection of sets $S_1, .., S_N$ such that

  $I = S_1 \cup S_2 \dots \cup S_N$ (sets cover entire image)

  $S_i \cap S_j = 0$ for all $i \neq j$ (sets do not overlap)

- A *predicate* $H(S_i)$ measures region *homogeneity*

$$H(R) = \begin{cases} \textit{true} & \text{if pixels in region } R \text{ are similar} \\ \textit{false} & \text{otherwise} \end{cases}$$

- We want
  1. Regions to be homogeneous

     $H(S_i) = \textit{true}$ for all i

  2. Adjacent regions to be different from each other

     $H(S_i \cup S_j) = \textit{false}$ for all adjacent S_i, S_j

Region growing

- Start with (random) seed pixel as cluster
- Repeat:
  - Aggregate neighboring pixels that are similar to cluster model
  - Update cluster model with newly incorporated pixels
- This is a generalized floodfill
- When cluster stops growing, begin with new seed pixel and continue
- An easy cluster model:
  - Store mean and covariance of pixels in cluster
  - Use Mahalanobis distance to cluster
    This leads to a natural threshold, e.g., ± 2.5 σ
  - Update mean and covariance efficiently by keeping track of sum(x) and sum(x^2)
- One danger: Since multiple regions are not grown simultaneously, threshold must be appropriate, or else early regions will dominate

GrowSingleRegion($I, O, p, label$)

```
model.INITIALIZE( I(p) )
frontier.push(p)
O(p) ← label
while NOT frontier.isEmpty() do
        p ← frontier.pop()
        for q ∈ N(p) do
                if model.ISSIMILAR( I(q) )
                then frontier.push(q)
                        O(q) ← label
                        model.UPDATE( I(q) )
```

S. Birchfield, Clemson Univ., ECE 847, http://www.ces.clemson.edu/~stb/ece847

Region growing results

S. Birchfield, Clemson Univ., ECE 847, http://www.ces.clemson.edu/~stb/ece847

Balloons

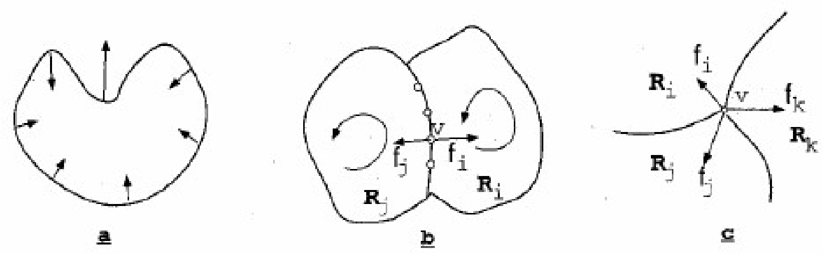
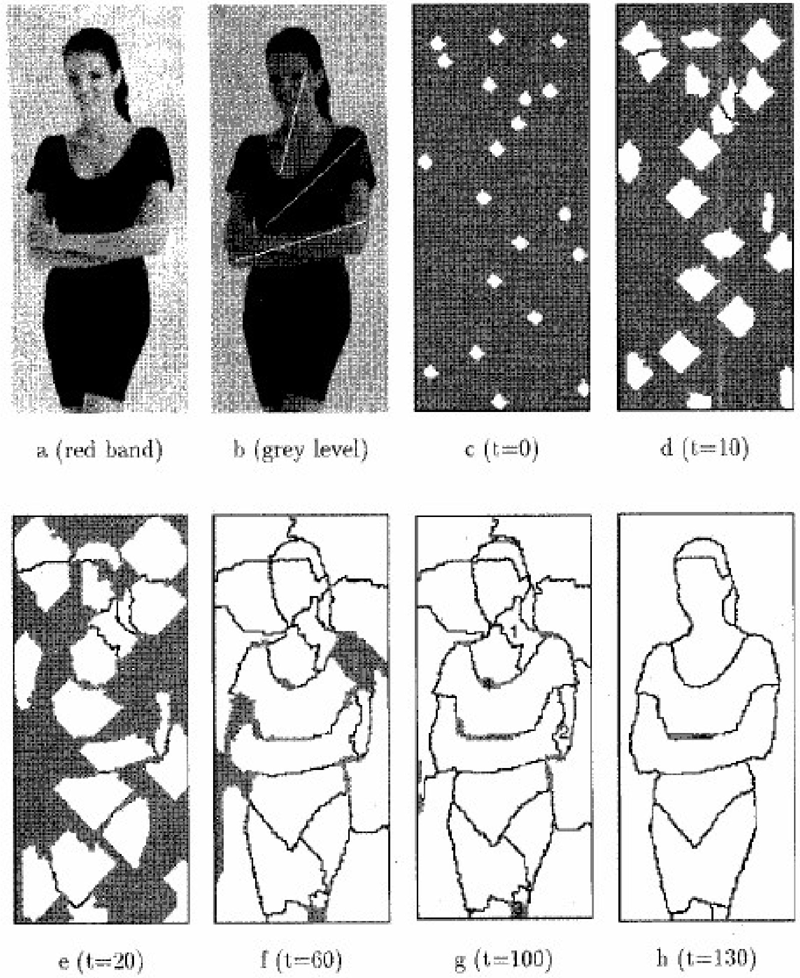


Fig. 2. The forces acting on the contour: (a) the smoothing force, (b the statistics force at a boundary point, (c) the statistics force at a junc tion point.



S. C. Zhu and A.L Yuille**, Region Competition**: Unifying Snake/balloon, Region Growing and Bayes/MDL/Energy for Multi-band Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.18, no.9, pp.884-900, Sept. 1996.

Watershed

- Smooth the image
- Run an edge detector (get gradient magnitude = edge strength)
- Create a ‘seed’ at each local min i.e. where there is no edge.
- Essentially, the idea is to ‘flood’ the image, marking each pixel as it goes ‘underwater’

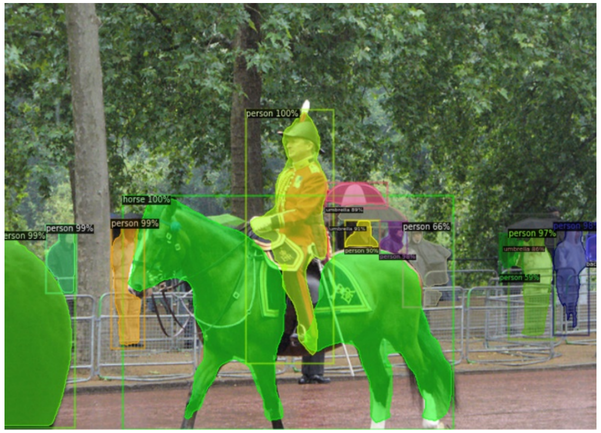
Watershed Example

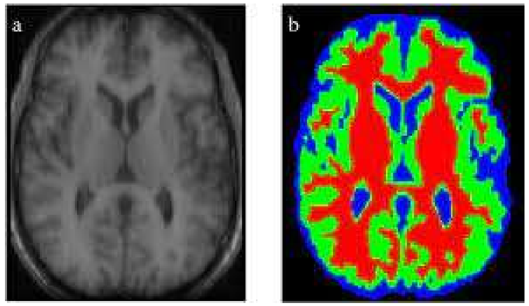
5	5	6	1	1
1	3	5	2	1
1	2	4	5	3
0	1	1	2	2
1	1	1	1	1

```
val = 0
While (unmarked pixels exist){
    for(pixel in image)
          if (pixel.val = val)
             if 2 neighbors are different
                            pixel is border
             else if no neighbors are marked
                             pixel is seed
             else
                            mark pixel = neighbor
                 val++
}
```

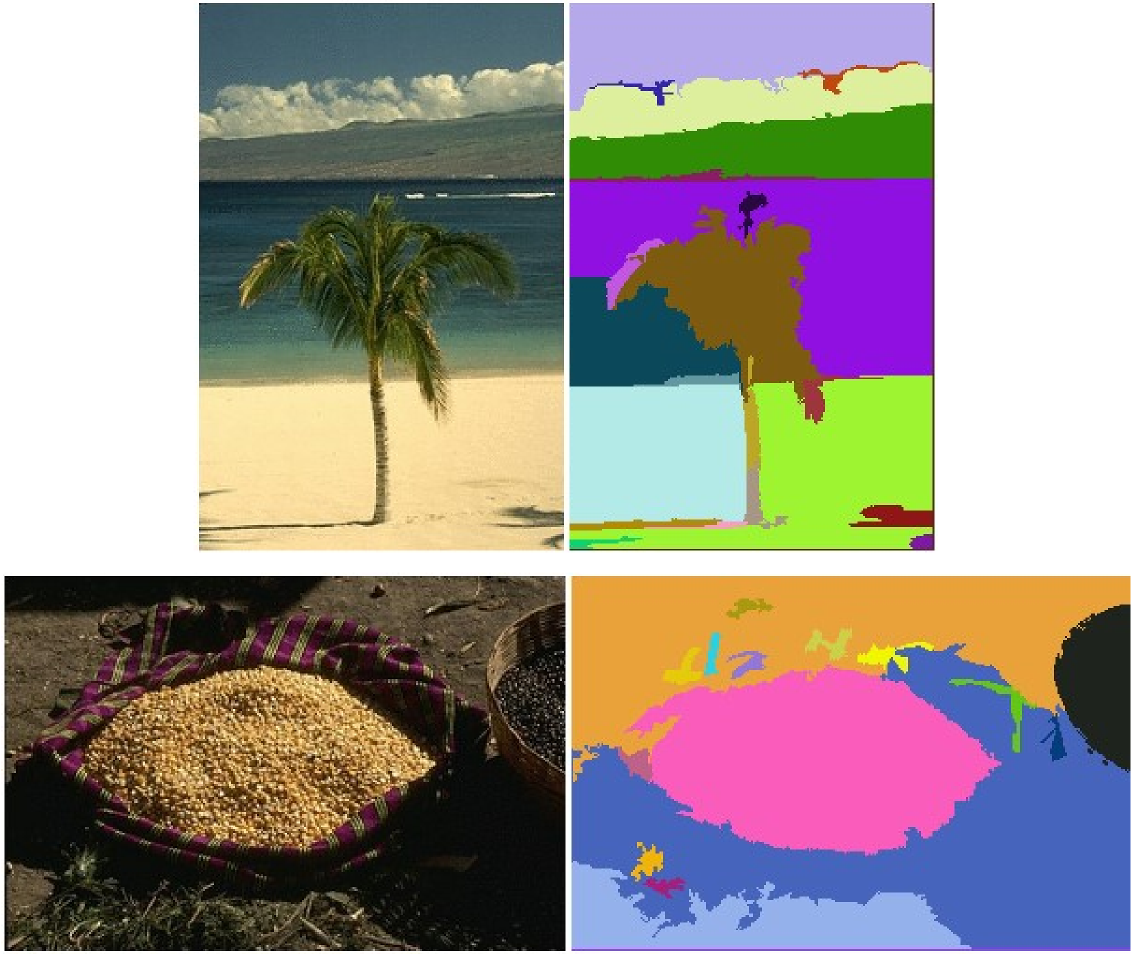
Displaying & Representing Regions

- Overlays (for display)
  - Use bright colors to show regions over greytone image
  - Color border pixels to contrast (e.g. white, red)
- Labeled image
  - Each region has a unique identifier (e.g. integer)
  - In a copy of the image, set each pixel value to its region label
  - For display, use well-separated values (grey or color)





Segmentation examples



from Pedro F. Felzenszwalb and Daniel P. Huttenlocher, Efficient Graph-Based Image Segmentation, IJCV, 59(2), 2004
http://people.cs.uchicago.edu/~pff/segment/

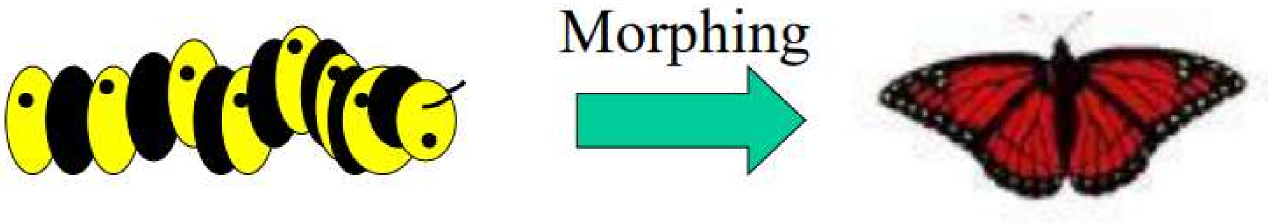
More examples

*

* Pictures from Mean Shift: A Robust Approach toward Feature Space Analysis, by D. Comaniciu and P. Meer http://www.caip.rutgers.edu/~comanici/MSPAMI/msPamiResults.html

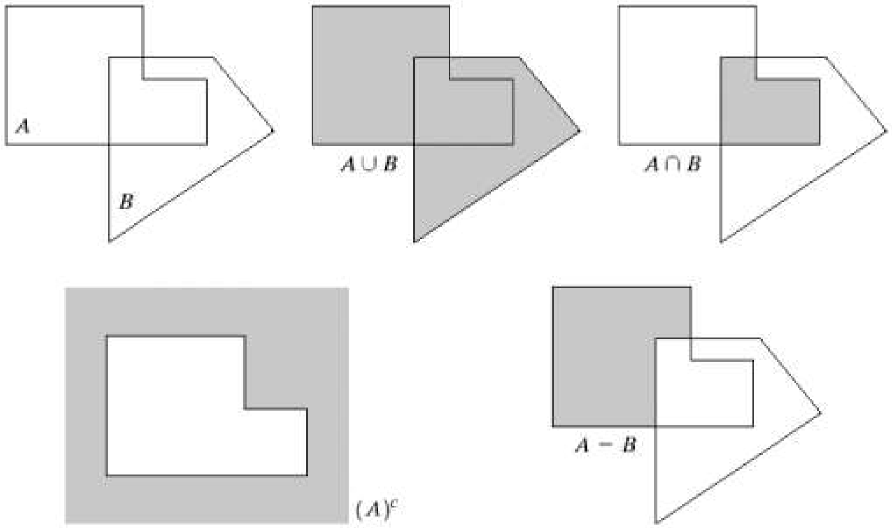
What are Morphological Operations?

- Morphological operations come from the word “morphing” in Biology which means “changing a shape”.
- Image morphological operations are used to manipulate object shapes such as thinning, thickening, and filling
- Binary morphological operations are derived from set operations.



Basic Set Operations

- Concept of a set in binary image morphology: Each set may represent one object. Each pixel (x,y) has its status: belong to a set or not belong to a set.



a b c
d e

FIGURE 9.1 (a) Two sets A and B. (b) The union of A and B. (c) The intersection of A and B. (d) The complement of A. (e) The difference between A and B.

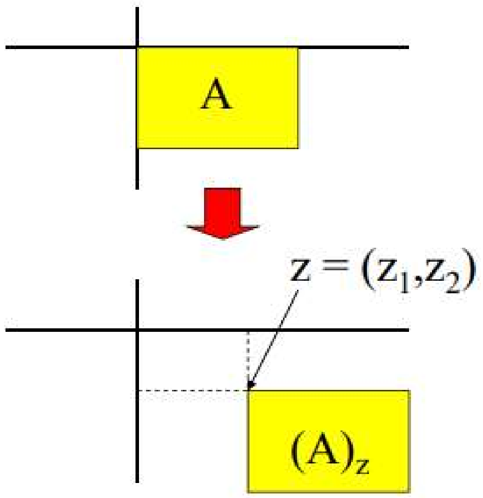
Translation and Reflection Operations

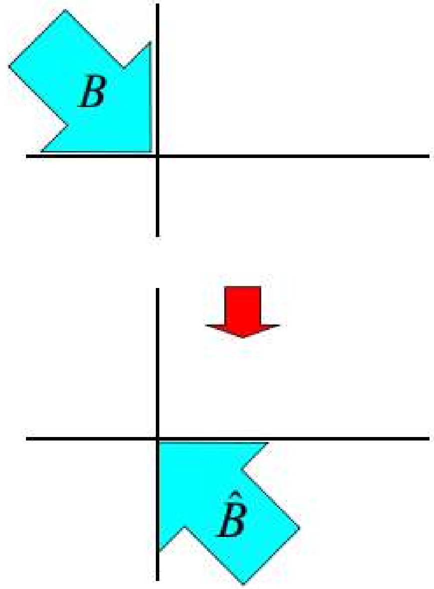
Translation

$$(A)_z = \{c \mid c = a + z, \text{ for } a \in A\}$$

Reflection

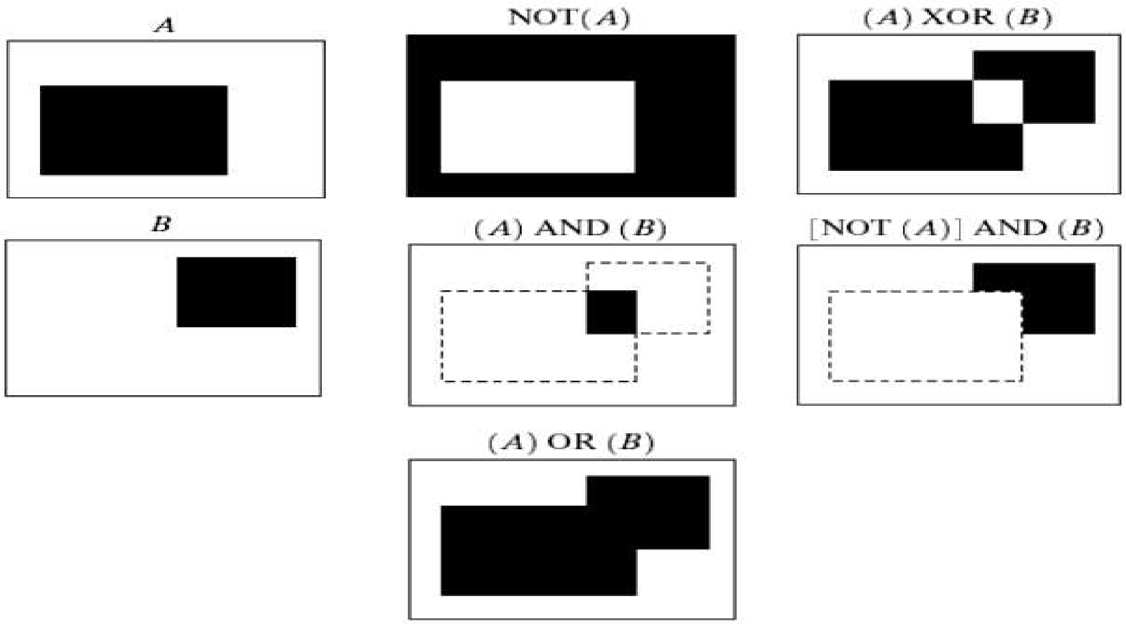
$$\hat{B} = \{w \mid w = -b, \text{ for } b \in B\}$$





Logical Operations

p	q	p AND q (also $p \cdot q$)	p OR q (also $p + q$)	NOT (p) (also $\bar{p}$)
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0



Dilation Operations

- This equation is based on obtaining the reflection of B about its original and shifting this reflection by z. The dilation of A by B the is the set of all displacement, z, such $\widehat{B}$ and A overlap by at least one element.

$$A \oplus B = \left\{ z \middle| \left(\hat{B}\right)_z \cap A \neq \phi \right\}$$

ϕ = Empty set

Dilate means “extend”

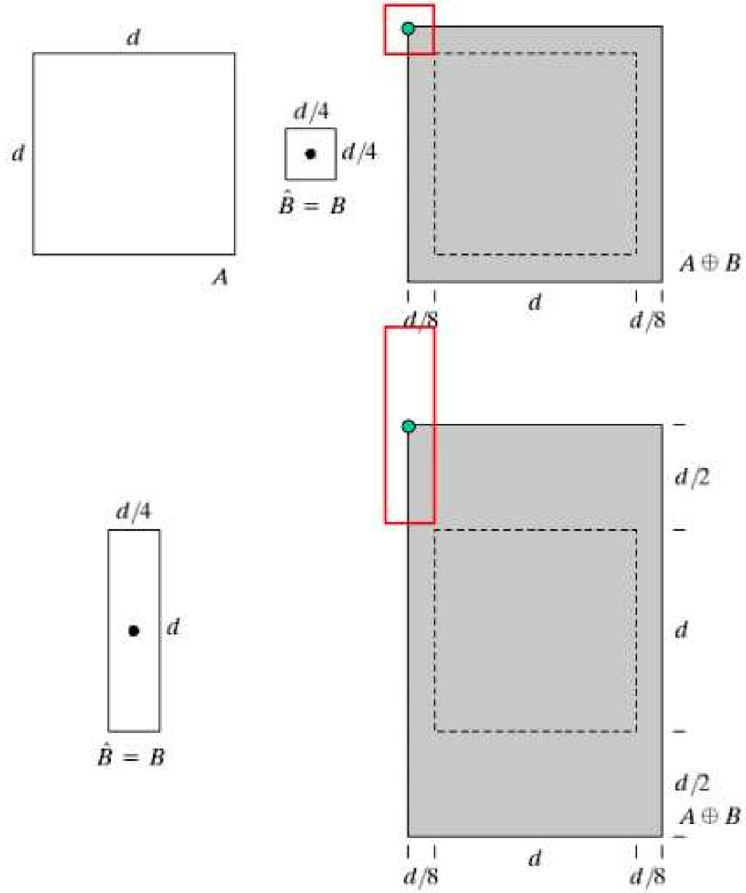
A = Object to be dilated

B = Structuring element

- Set B is commonly referred to as the structuring element

$$A \oplus B = \left\{ z \middle| \left[\left(\hat{B}\right)_z \cap A\right] \subseteq A \right\}$$

Dilations



ϕ = Empty set

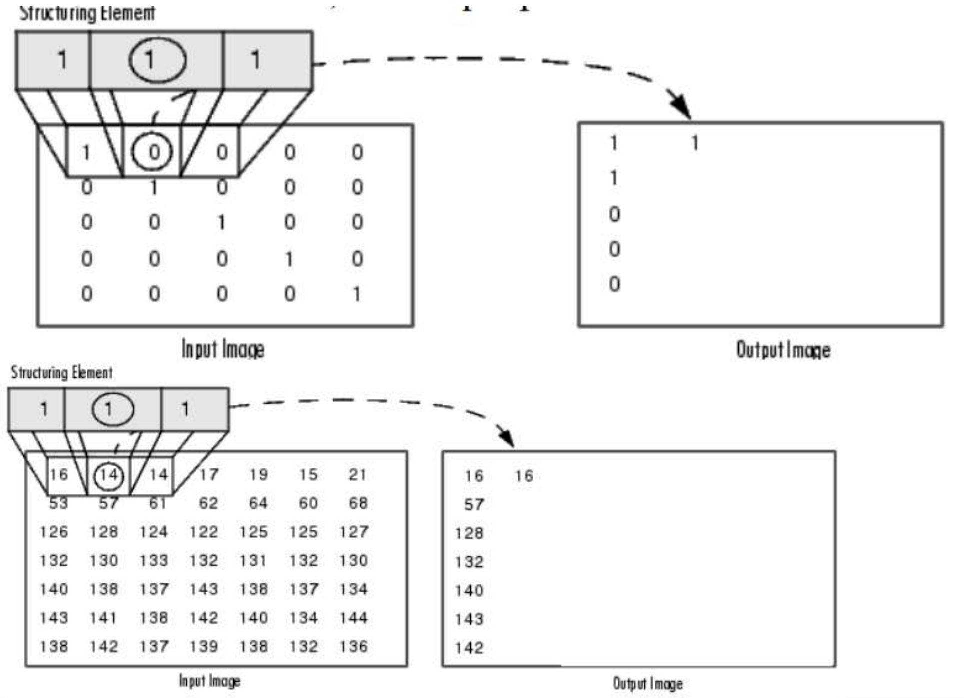
Dilate means “extend”

A = Object to be dilated

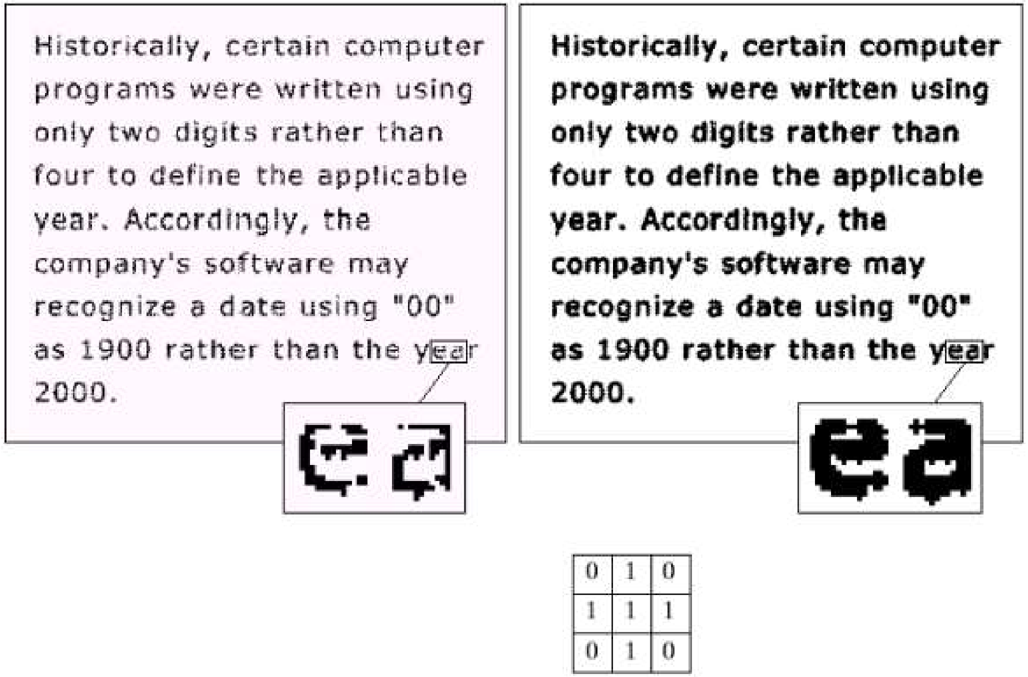
B = Structuring element

Dilations

- Dilation: The value of the output pixel is the maximum value of all the pixels in the input pixel's neighborhood. In a binary image, if any of the pixels is set to the value 1, the output pixel is set to 1



Applications



a c
b

FIGURE 9.5
(a) Sample text of poor resolution with broken characters (magnified view).
(b) Structuring element.
(c) Dilation of (a) by (b). Broken segments were joined.

"Repair" broken characters

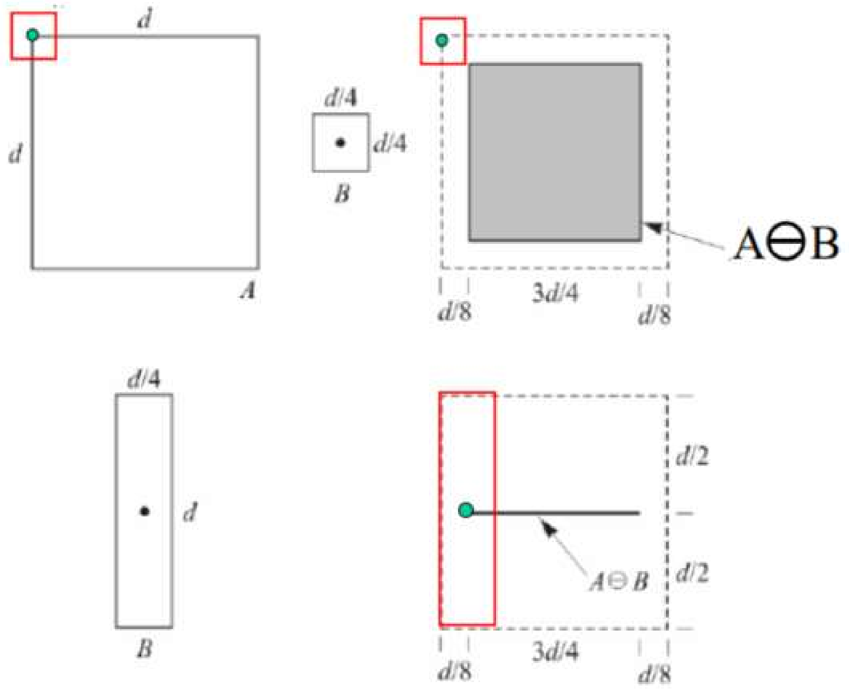
Applications

Dilation is an operation that “grows” or “thickens” objects in a binary image. The specific manner and extent of this thickening is controlled by a shape refer to as a structuring element

Erosion Operation

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

- The Erosion of A by B is the set of all points z such that B, translated by z is contained in A.



Erosion means “trim”

A = Object to be eroded
B = Structuring element

https://www.cpe.ku.ac.th/~jan/204584/09-morph.pdf

Erosion Operation

- Erosion: The value of the output pixel is the minimum value of all the pixels in the input pixel's neighborhood. In a binary image, if any of the pixels is set to 0, the output pixel is set to 0

Example: Application of Dilation and Erosion

- Remove small objects such as noise
  - Remember! Erosion “shrinks” or “thins”
  - objects in binary image.

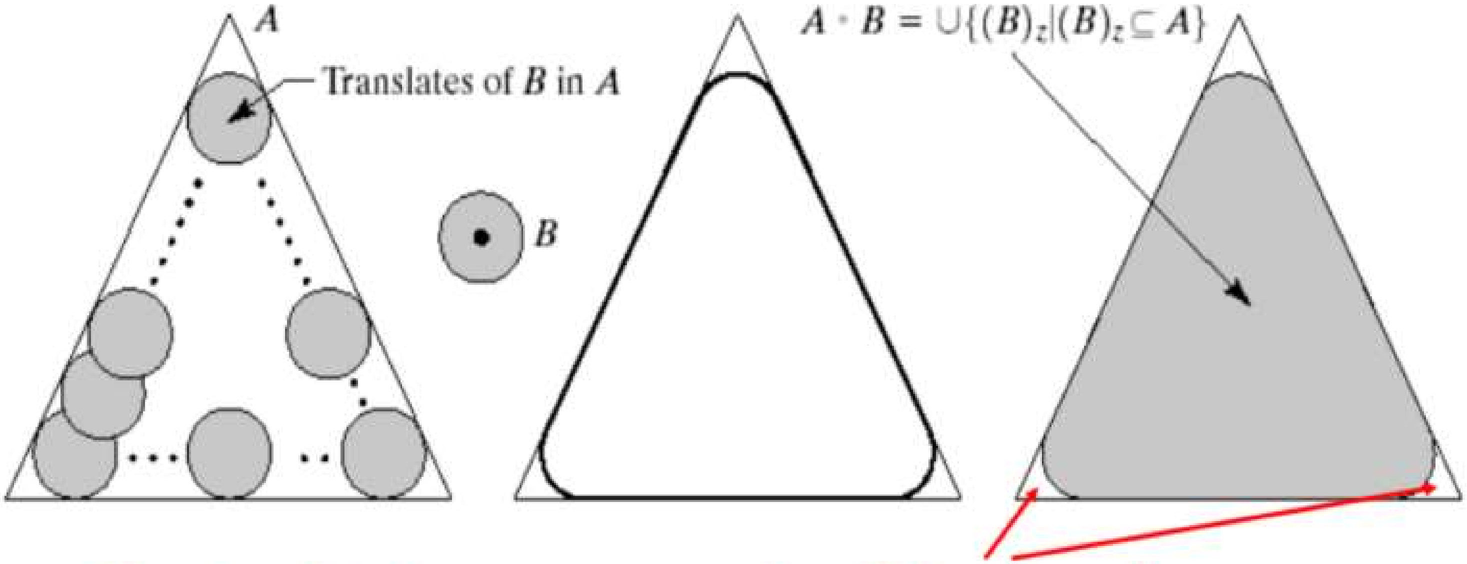
a b c

FIGURE 9.7 (a) Image of squares of size 1, 3, 5, 7, 9, and 15 pixels on the side. (b) Erosion of (a) with a square structuring element of 1’s, 13 pixels on the side. (c) Dilation of (b) with the same structuring element.

Opening Operation

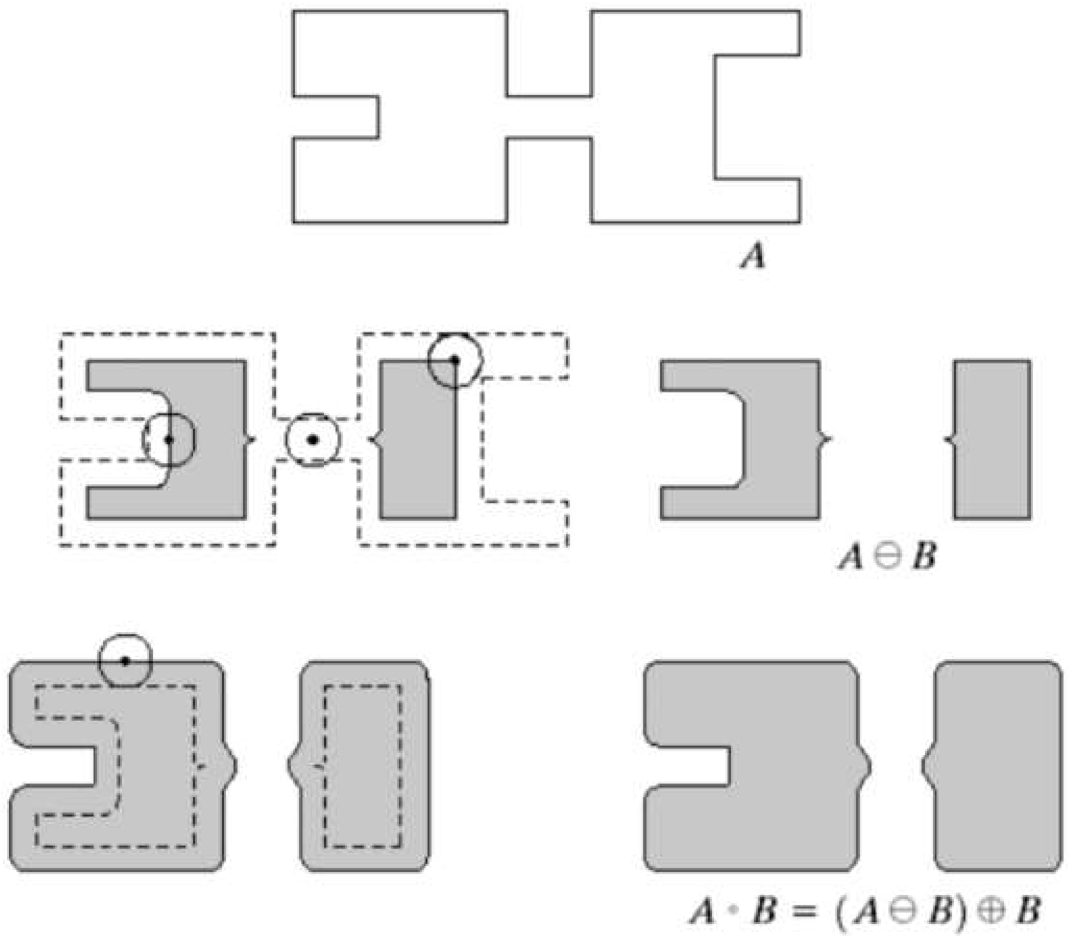
$$A \circ B = (A \ominus B) \oplus B$$

= Combination of all parts of A that can completely contain B



Opening eliminates narrow and small details and corners.

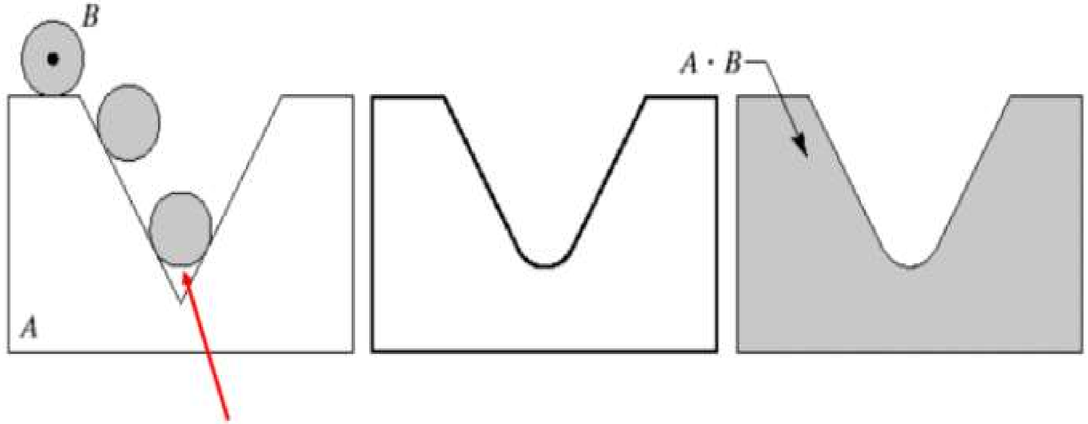
Example of Opening



(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition

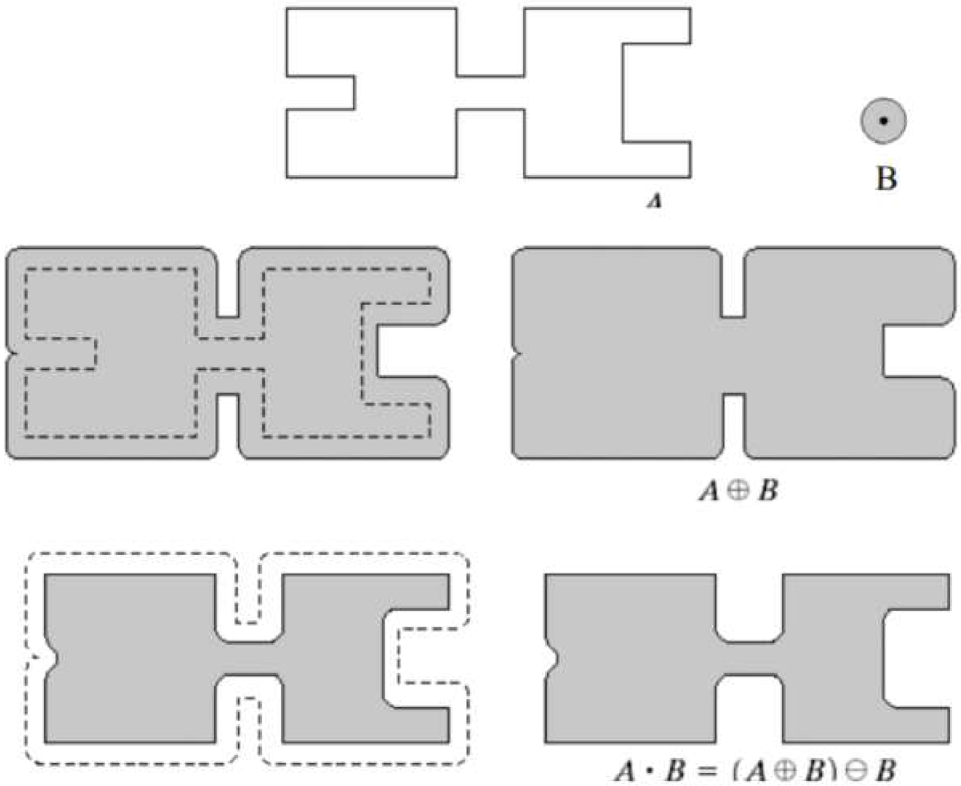
Closing Operation

$$A \bullet B = (A \oplus B) \ominus B$$



Closing fills narrow gaps and notches

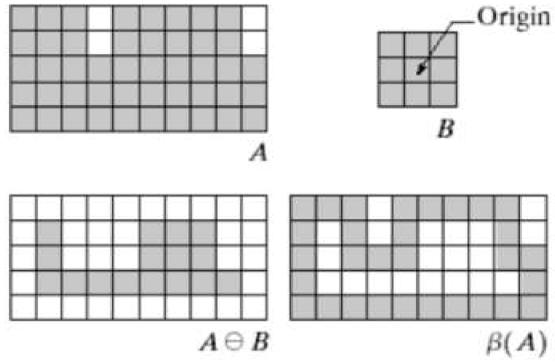
Example of Closing



Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition

Boundary Extraction

$$\beta(A) = A - (A \ominus B)$$



- The boundary of set A, denoted by , can be obtained by first eroding A by B (suitable structuring element) and then performing the set difference between A and its erosion.

Original image

Boundary