COMP3419

Graphics & Multimedia

School of Information Technologies
The University of Sydney

Week02 • Semester 2 • 2018

Morphological Image Processing

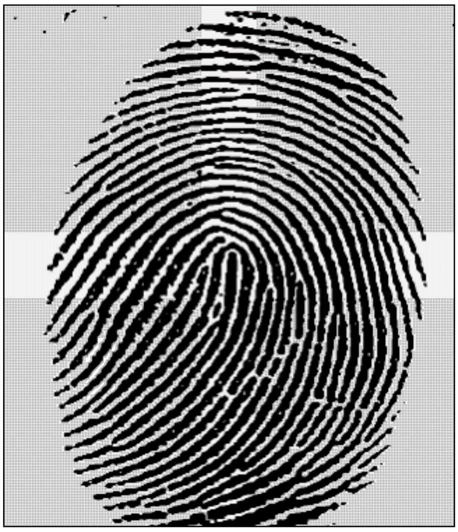
- Mathematical morphology
- Set theory and logic operations
- Primitive morphological operations: dilation and erosion
- Advanced morphological operations: opening and closing
- Appendix:
    - Morphological filters and algorithms
    - Morphological watersheds
    - Computer vision applications

Mathematical Morphology

- The word "*morphology*" refers to form and structure.
- We use the same word here in the context of *mathematical morphology* as a tool for extracting image components that are useful in the representation and description of region shape, such as boundaries and skeletons.
- Mathematical morphology is very often used in applications where **shape of objects** and **speed** is an issue.
- Applications: analysis of microscopic images (in biology, material science, geology, and criminology), industrial inspection, optical character recognition, and document analysis etc.

Mathematical Morphology and Set Theory

- The language of mathematical morphology is **set theory**.
- **Sets** in mathematical morphology represent **objects** in an image.



The set of all black pixels in a binary image is a complete morphological description of the image.

Some Basic Concepts from Set Theory

- Let Z be the set of integers. The sampling process used to generate digital images may be viewed as partitioning the *xy*-plane into a grid, with the coordinates of the center of each grid being a pair of elements from the Cartesian product, Z^2.
- In the terminology of set theory, a function $f(x, y)$ is said to be a **digital image** if (x, y) are integers from Z^2 and f is a mapping that assigns an intensity value (that is, a real number from the set of real numbers, R) to each distinct pair of coordinates (x, y).
- If the elements of R also are integers, a digital image then becomes a two-dimensional function whose coordinates and amplitude (i.e., intensity) values are integers.

Some Basic Concepts from Set Theory

- Let A be a set of Z^2, the elements of which are pixel coordinates (x, y). If $w = (x, y)$ is an element of A, then we write
$$w \in A$$
- Similarly, if w is not an element of A, we write
$$w \notin A$$
- The set with no elements is called the **null** or **empty set** and is denoted by the symbol $\varnothing$.
- A set is specified by the contents of two braces: $\{\cdot\}$. The elements of the sets with which we are concerned here are the **coordinates** of pixel representing objects or other features of interest in an image.
For example, when we write an expression of the form
$$C = \{w \mid w = -d, \text{for } d \in D\}$$
we mean that set C is the set of elements, w, such that w is formed by multiplying each of the two coordinates of all the elements set D by – 1

Some Basic Concepts from Set Theory

- If every element of a set A is also an element of another set B, then A is said to be a **subset** of B, denoted as
$$A \subseteq B$$
- The **union** of two sets A and B, denoted by
$$C = A \cup B$$
is the set of all elements belonging to either A, B, or both.
- Similarly, the **intersection** of two sets A and B, denoted by
$$D = A \cap B$$
is the set of all elements belonging to both A and B.
- Two sets A and B are said to be **disjoint** or **mutually exclusive** if they have no common elements. In this case,
$$A \cap B = \varnothing$$

Some Basic Concepts from Set Theory

- The **complement** of a set A is the set of elements not contained in A:

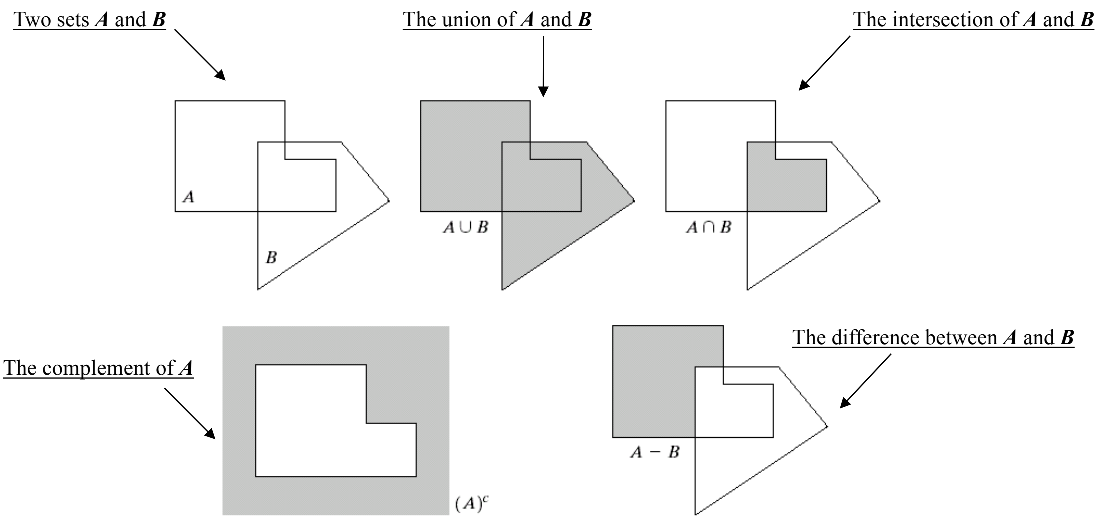
$$A^c = \{w \mid w \notin A\}$$

- The **difference** of two sets A and B, denoted $A - B$, is defined as

$$A - B = \{w \mid w \in A, w \notin B\} = A \cap B^c$$

We see that this is the set of elements that belong to A, but not to B.

Some Basic Concepts from Set Theory



EXAMPLE – The result of the set operation indicated in each figure is shown in gray.

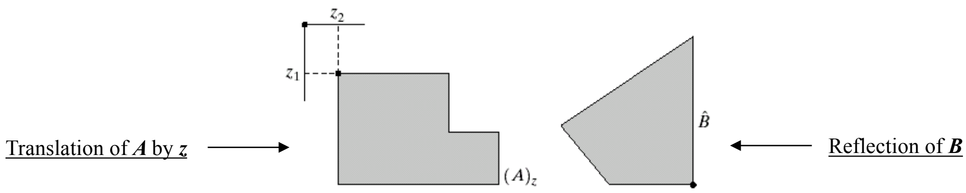
Translation and Reflection on Set Theory

- We need two additional definitions that are used extensively in morphology but generally are not found in basic texts on set theory.
- The **translation** of set A by point $z = (z_1, z_2)$, denoted $(A)_z$, is defined as

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

- The **reflection** of set B, denoted $\hat{B}$, is defined as

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



EXAMPLE – The sets A and B are from the previous slide, and the black dot identifies the origin of the sets shown in the figure.

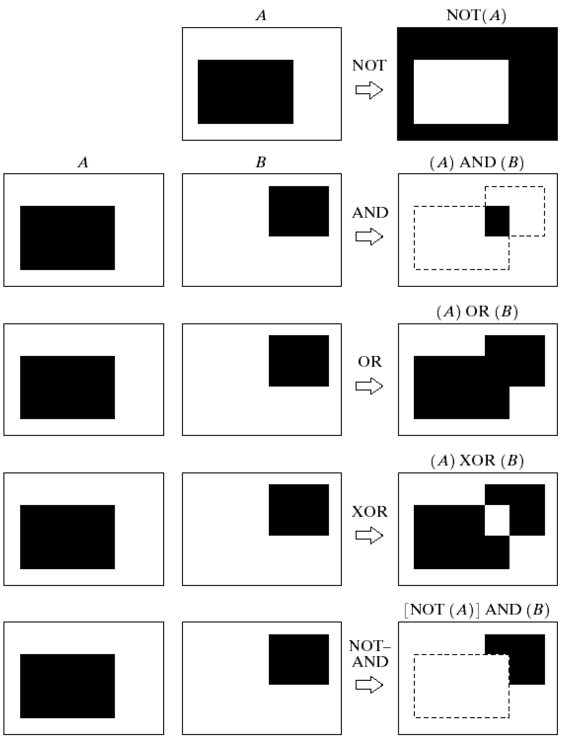
Binary Images, Sets, and Logical Operators

- The language and theory of mathematical morphology often present a dual view of binary images.
- A binary image can be viewed as a bi-valued **function** of x and y.
- Morphological theory views a binary image as the **set** of its foreground (1-valued, and shown in black) pixels, the elements of which are in Z^2.
- Set operations such as union and intersection can be applied directly to binary image sets.
  For example, **if *A* and *B* are binary images, then *C* = *A* U *B* is also a binary image**, where a pixel in *C* is a foreground pixel if either or both of the corresponding pixels in *A* and *B* are foreground pixels.

Logic Operations Involving Binary Images

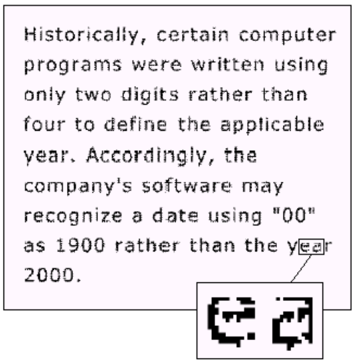
- Logic operations provide a powerful complement to implementation of image processing algorithms based on morphology.
- The principal logic operations used in image processing are **AND**, **OR**, and **NOT** (COMPLEMENT). These operations are functionally complete in the sense that they can be combined to form any other logic operation.
  For example, the **XOR** (exclusive OR) operation yields a 1 when one or the other pixel (but not both) is 1, and it yields a 0 otherwise. Similarly, the **NOT-AND** operation selects the black pixels that simultaneously are in *B*, and not in *A* (see next slide).
- Logic operations are performed on a pixel by pixel basis between corresponding pixels of two or more images (except NOT, which operates on the pixels of a single image).

Logic Operations Involving Binary Images

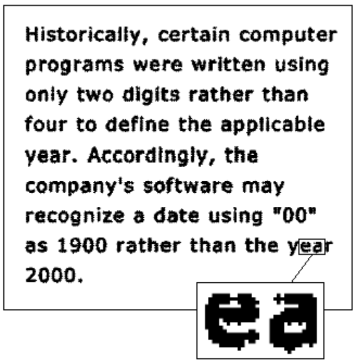


Some logic operations between binary images.

Black represents binary 1s and white binary 0s in this example



?



The original binary image shows sample text of poor resolution

Note that the broken characters in magnified view

The enhanced binary image with gap-repairing

Note that the broken segments have been joined.

Primitive morphological operations

Dilation and Erosion

- The operations of dilation and erosion are fundamental to morphological image processing.
- **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 referred to as a ***structuring element***.
- **Erosion** “shrinks” or “thins” objects in a binary image. The manner and extent of shrinking is also controlled by a ***structuring element***.
- Many of the algorithms presented later are based on these two primitive morphological operations.

Dilation

- With A and B as sets in Z^2, the **dilation** of A by B, denoted $A \oplus B$ is defined as

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

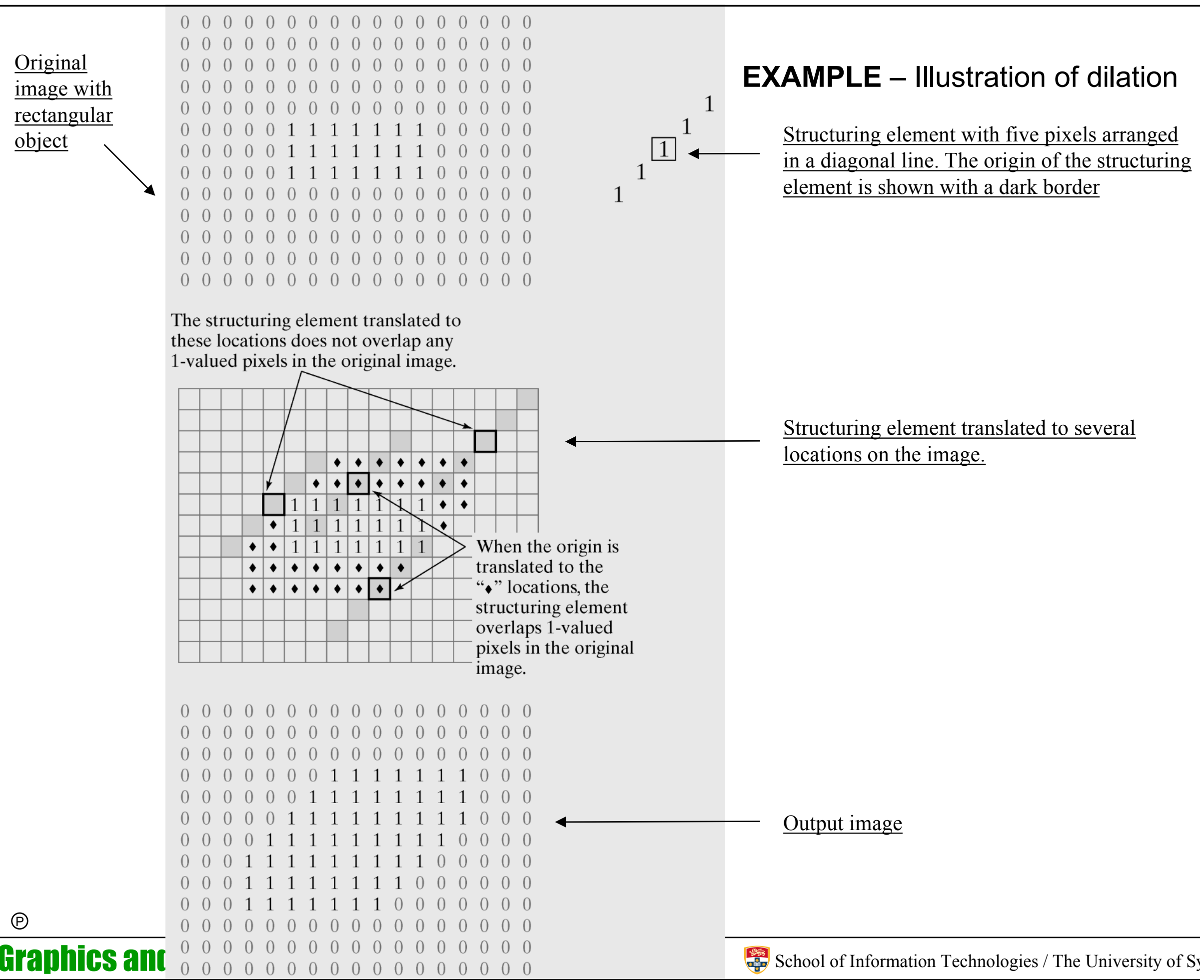
This equation is based on obtaining the reflection of B about its origin and shifting this reflection by z.

- The dilation of A by B then is the set of all displacements, z, such that $\hat{B}$ and A overlap by at least one element. Based on this interpretation, the above equation may be rewritten as

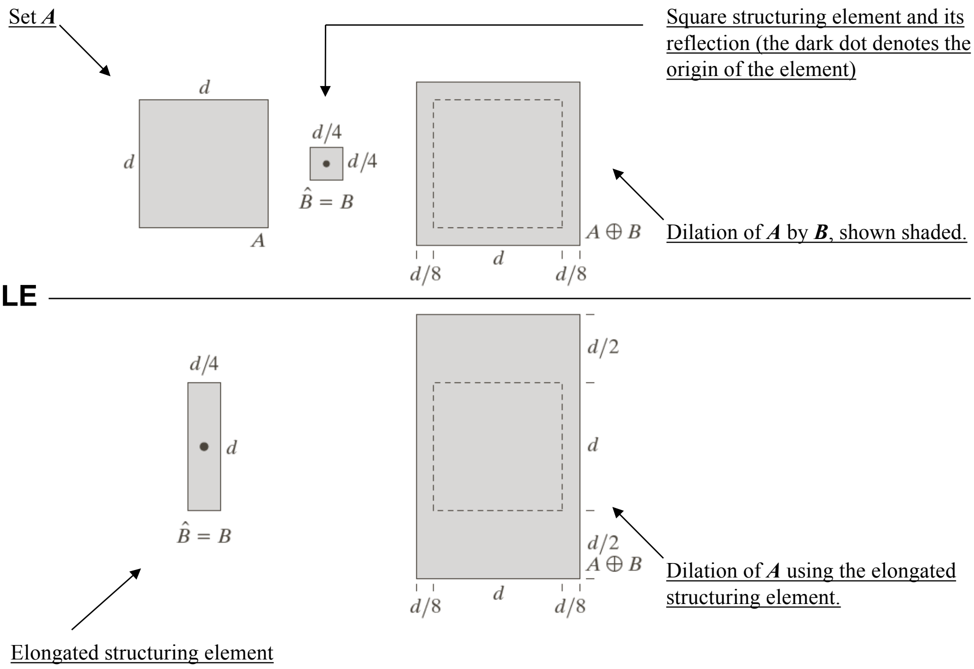
$$A \oplus B = \left\{ z \,\middle|\, \left[\left(\hat{B}\right)_z \cap A \right] \subseteq A \right\}$$

- Set B is commonly referred to as the **structuring element** in dilation, as well as in other morphological operations.

EXAMPLE – Illustration of dilation



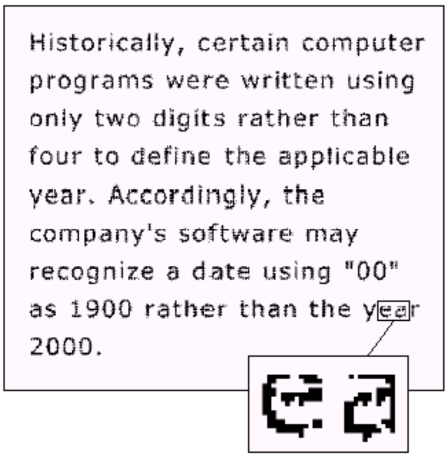
Dilation



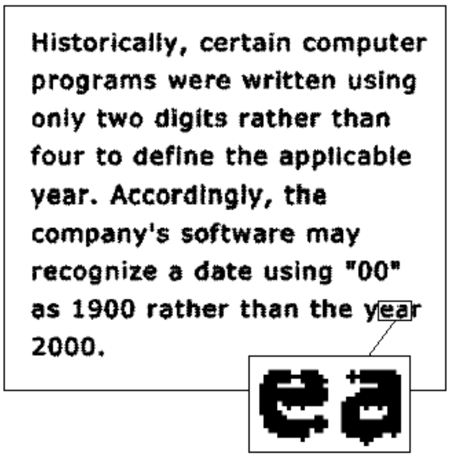
* The dotted border is the boundary of set A, shown only for reference.

Question: How to bridge gaps in binary image using dilation ?

The original binary image – sample text of poor resolution with broken characters (see magnified view); The maximum length of the breaks is known to be two pixels.


Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

Dilation of the original image by the structuring element. Broken segments have been joined.

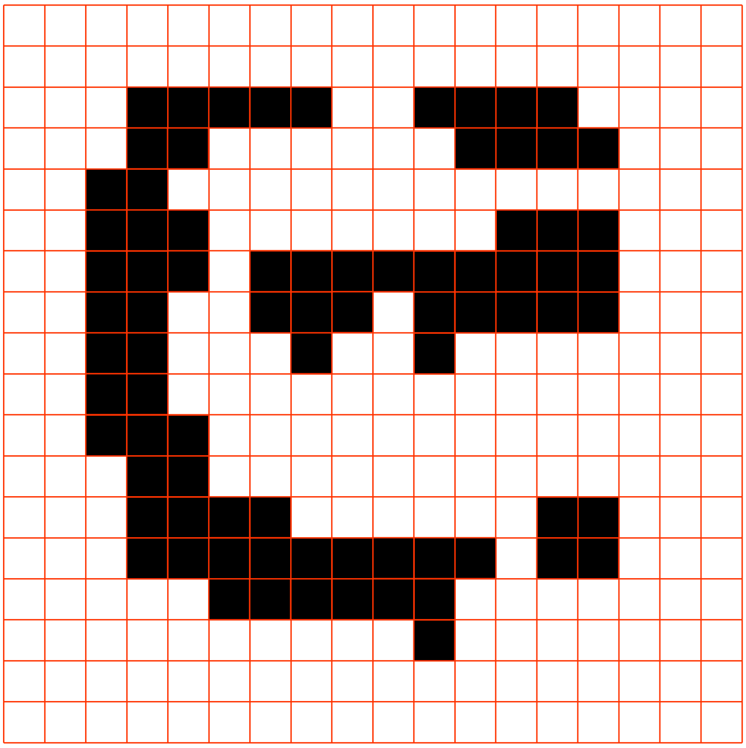

Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

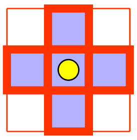
0	1	0
1	1	1
0	1	0

Structuring element in "cross" shape

Question: How to bridge gaps in binary image using dilation ?

Original Image:





Structuring element in "cross" shape

The central yellow dot denotes the origin of the element.

In this case, the structuring element and its reflection are equal because it is symmetric with respect to its origin.

Question: How to bridge gaps in binary image using dilation ?

Output Image:

Erosion

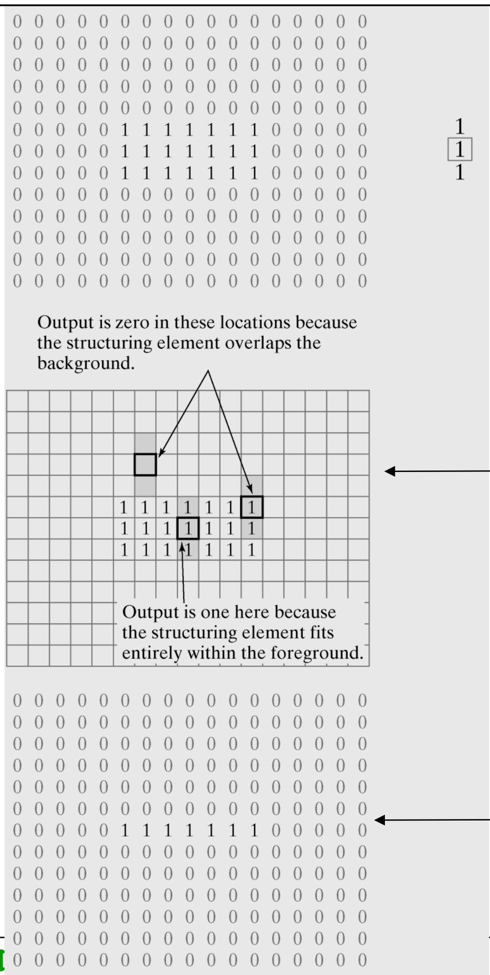
- For sets A and B in Z^2, the **erosion** of A by B, denoted $A \ominus B$ is defined as

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

This equation indicates that the erosion of A by B is the set of all points z such that B, translated by z, is contained in A.

EXAMPLE – Illustration of erosion

Original image with rectangular object

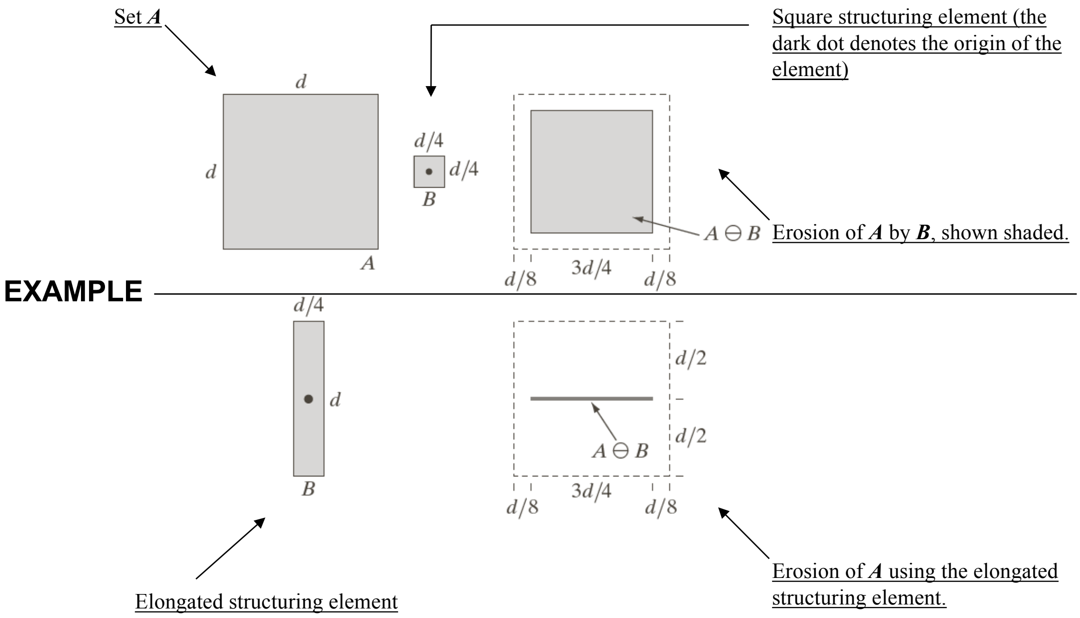


Structuring element with three pixels arranged in a vertical line. The origin of the structuring element is shown with a dark border

Structuring element translated to several locations on the image.

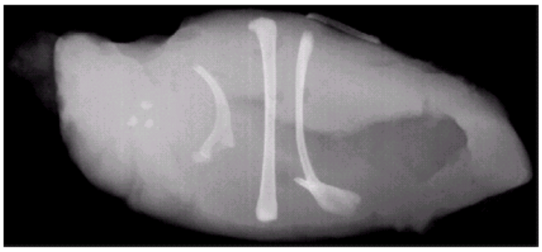
Output image

Erosion



* The dotted border is the boundary of set A, shown only for reference.

Example - *detect foreign objects in packaged food*



X-ray image of chicken filet with bone fragments

Thresholded image

Image **eroded** with a 5x5 structuring element of 1's – to determine if any significant foreign objects are contained in the original image.

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

Structuring Elements (SEs)

- The purpose of the structuring elements (SEs) is to act as probes of the binary image.
- One pixel of the structuring element is denoted as its origin; this is often the central pixel of a symmetric structuring element, but may in principle be any chosen pixel and problem dependent.
- The structuring element represents a shape. It can be of any size and have arbitrary structure that can be represented by a binary image.
- Some common structuring elements:
  - A rectangle of specified dimensions `[BOX(l,w)]`
  - A circular region of specified diameter `[DISK(d)]`

Common Structuring Elements

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

BOX(3,5)

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

DISK(5)

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

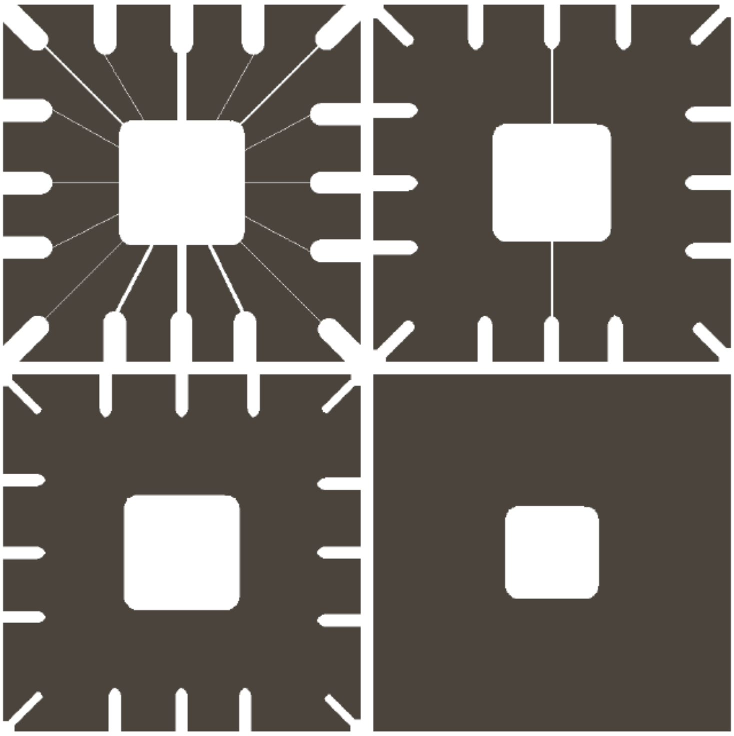
RING(5)

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

EXAMPLE – Using erosion to remove image components



(a) A 486 × 486 binary image of a wire-bond mask

(b) Image eroded using a square structuring element* of size 11 × 11

(c) Image eroded using a square structuring element* of size 15 × 15

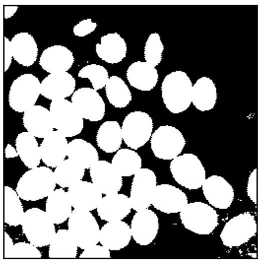
(d) Image eroded using a square structuring element* of size 45 × 45

* The elements of the SEs were all 1s.

EXAMPLE – Using erosion to remove image components

- Suppose that we wish to remove the lines connecting the center region to the border pads in Fig.(a) (shown in previous slide). Eroding the image with a square structuring element of size 11 × 11 whose components are all 1s removed most of the lines, as Fig.(b) shows. The reason the two vertical lines in the center were thinned but not removed completely is that their width is greater than 11 pixels.
- Changing the SE size to 15 × 15 and eroding the original image again did remove all the connecting lines, as Fig. (c) shows (an alternate approach would have been to erode the image in Fig. (b) again using the same 11 × 11 SE).
- Increasing the size of the structuring element even more would eliminate larger components. For example, the border pads can be removed with a structuring element of size 45 × 45, as Fig. (d) shows.
- We see from this example that erosion shrinks or thins objects in a binary image. In fact, we can view erosion as a morphological filtering operation in which image details smaller than the structuring element are filtered (removed) from the image. In this example, erosion performed the function of a “line filter”.

Erosion and Dilation – *are they inverse transformations ???*



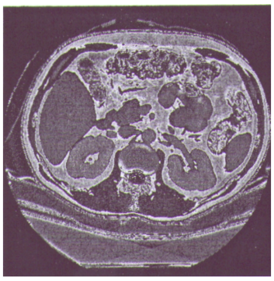
- Erosion and dilation are NOT inverse transformations – if an image is eroded and then dilated, the original image is not re-obtained. Instead, the result is a simplified and less detailed version of the original image.

- Erosion followed by dilation creates an important morphological transformation called **Opening**.
- Dilation followed by erosion is called **Closing**.

Opening and Closing

- ***Opening*** generally smoothes the contour of an object, breaks narrow isthmuses, and eliminates thin protrusions.
- ***Closing*** also tends to smooth sections of contours but, as opposed to opening, it generally fuses narrow breaks and long thin gulfs, eliminates small holes, and fills gaps in the contour.
- Opening and closing are useful in imaging applications where thresholding – or some other initial process – produces a binary image with tiny holes in the connected components or with a pair of components that should be separate joined by a thin region of foreground pixels.

Application – use of opening & closing in medical imaging



(a) Original medical image G

a 512 × 512 16-bit gray-scale chest image

(b) Preprocessing – thresholded image B

The image shown in (a) is thresholded by selecting pixels with gray tones above 1,070 to produce the binary image

(c) Result of morphological operations

The result of performing an opening operation with a DISK(13) structuring element to separate the organs and a closing with a DISK(2) to get rid of small holes.

Opening and Closing

- The **opening** of set A by structuring element B, denoted $A \circ B$, is defined as

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

Thus, the opening A by B is the erosion of A by B, followed by a dilation of the result by B.

- Similarly, the **closing** of set A by structuring element B, denoted $A \bullet B$, is defined as

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

which, in words, says that the closing of A by B is simply the dilation of A by B, followed by the erosion of the result by B.

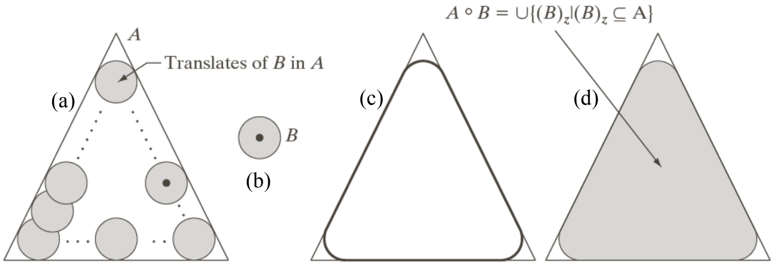
Opening

- An alternative mathematical formulation of **opening** is

$$A \circ B = \cup\{(B)_z \mid (B)_z \subseteq A\}$$

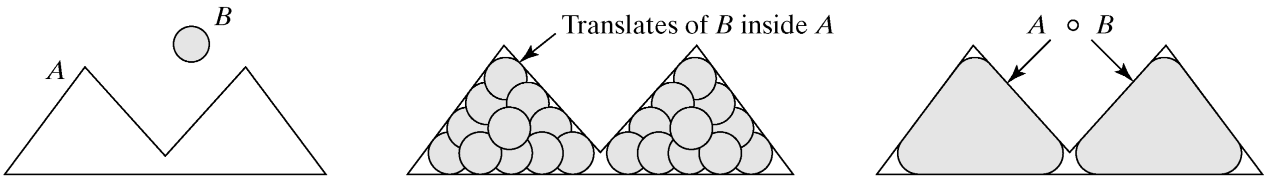
where $\cup\{\bullet\}$ denotes the union of all sets inside the braces.

- This formulation has a simple **geometric** interpretation: $A \circ B$ is the union of all translations of B that fit entirely within A.



(a) Structuring element B "rolling" along the inner boundary of A (the dot indicates the origin of B);
(b) Structuring element;
(c) The heavy line is the outer boundary of the opening;
(d) Complete opening (shaded). We did not shade A in (a) for clarity.

Opening



(a) Fig.(a) shows a set A and a disk-shaped structuring element B

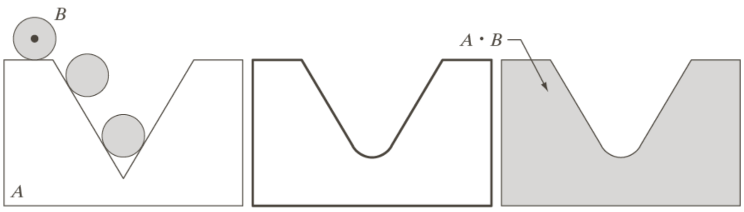
(b) Fig.(b) shows some of the translations of B that fit entirely within A. The union of all such translations is the shaded region in Fig.(c).

(c) This region is the complete opening. The white regions in Fig.(c) are areas where the structuring element could not fit completely within A, and, therefore, are not part of the opening.

- Morphological opening removes completely regions of an object that cannot contain the structuring element, smoothes object contours, breaks thin connections, and removes thin protrusions.

Closing

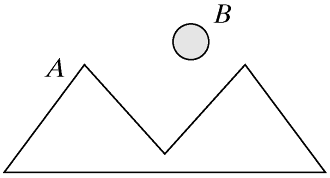
- **Geometrically**, $A \bullet B$ is the complement of the union of all translations of B that do not overlap A.



(a) (b) (c)

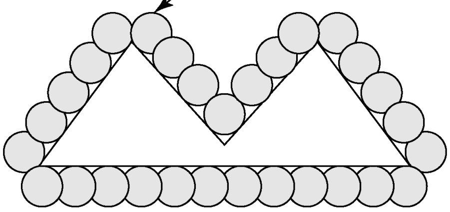
(a) Structuring element B "rolling" on the outer boundary of set A);
(b) The heavy line is the outer boundary of the closing;
(c) Complete closing (shaded). We did not shade A in (a) for clarity.

Closing



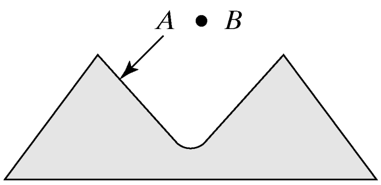
(a)

Fig.(a) shows a set A and a disk-shaped structuring element B



(b)

Fig.(b) illustrates several translations of B that do not overlap A. By taking the complement of the union of all such translations, we obtain the shaded region in Fig.(c).

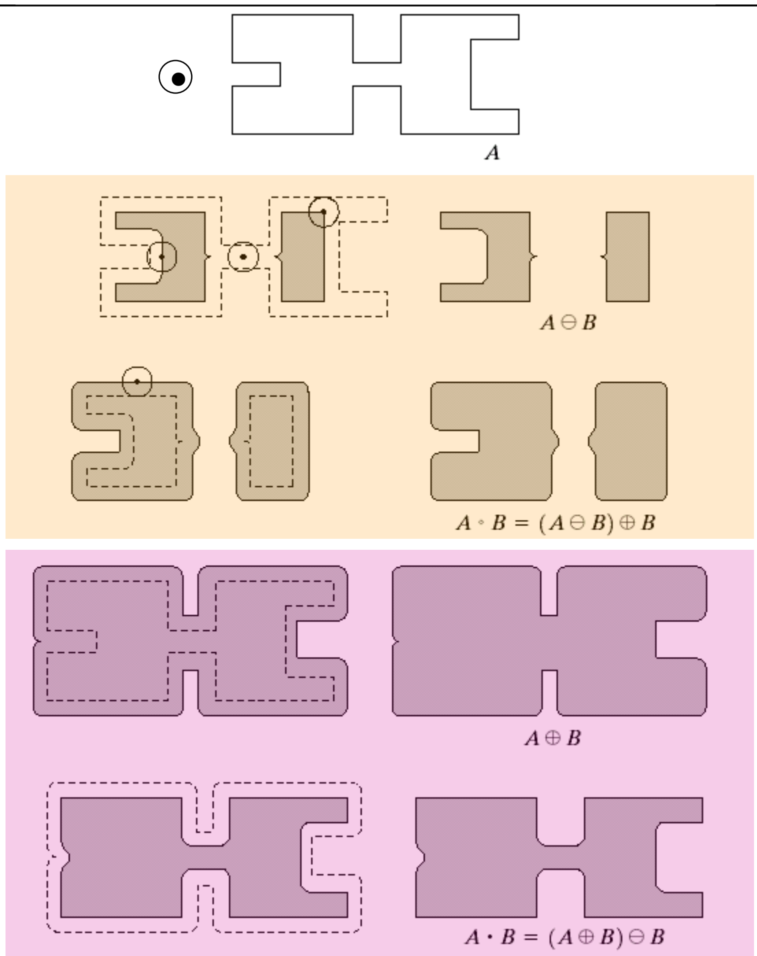


(c)

This region is the complete closing.

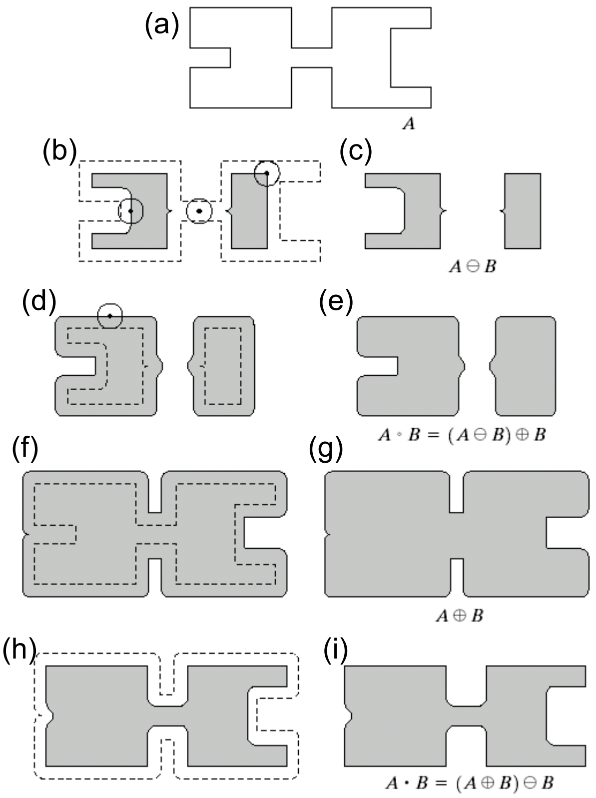
- Morphological closing tends to smooth the contours of objects. Unlike opening, it generally joins narrow breaks, fills long thin gulfs, and fills holes smaller than the structuring element.

EXAMPLE –



Opening

Closing



The structuring element is the small circle shown in various positions in Fig.(b). The dark dot is the center of the structuring element.

EXAMPLE – Opening and Closing

Fig.(a) shows a set *A*, and Fig.(b) shows various positions of a disk structuring element during the erosion process. When completed, this process resulted in the disjoint figure shown in Fig.(c). Note the elimination of the bridge between the two main sections. Its width was thin in relation to the diameter of the structuring element; that is, the structuring element could not be completely contained in this part of the set. The same also was true of the two rightmost members of the object. Protruding elements where the disk did not fit were eliminated. Fig.(d) shows the process of dilating the eroded set, and Fig.(e) shows the final result of opening. Note that outward pointing corners were rounded, whereas inward pointing corners were not affected.

Similarly, Fig.(f) through (i) show the results of closing *A* with the same structuring element. We note that the inward pointing corners were rounded, whereas the outward pointing corners remained unchanged. The leftmost intrusion on the boundary of *A* was reduced in size significantly, because the disk did not fit there. Note also the smoothing that resulted in parts of the object from both opening and closing the set *A* with a circular structuring element.

Appendix

- Morphological filters and algorithms
- Morphological watersheds
- Computer vision applications