Morphological Image Processing

Logic Operations involving binary images,

Dilation and Erosion,

Opening and Closing,

The Hit or-Miss Transformation

Introduction

Morphology: Shape, Form, Structure / Extracting and describing image component regions /Applied on binary image / Based on set theory

- Binary images may contain numerous imperfections.
- In particular, the binary regions produced by simple thresholding are distorted by noise and texture.
- Morphological image processing pursues the goals of removing these imperfections by accounting for the form and structure of the image. These techniques can be extended to greyscale images.

Primitives:

Question

What is Mathematical Morphology ?

An (imprecise) Mathematical Answer

A mathematical tool for investigating geometric structure in **binary** and **grayscale** images.

Shape Processing and Analysis

Visual perception requires transformation of images so as to make explicit particular **shape information**.

Goal: Distinguish meaningful shape information from irrelevant one.

The vast majority of shape processing and analysis techniques are based on designing a **shape operator** which satisfies desirable properties.

Morphological Operators

Erosions and **dilations** are the most elementary operators of mathematical morphology.

More complicated **morphological operators** can be designed by means of combining erosions and dilations.

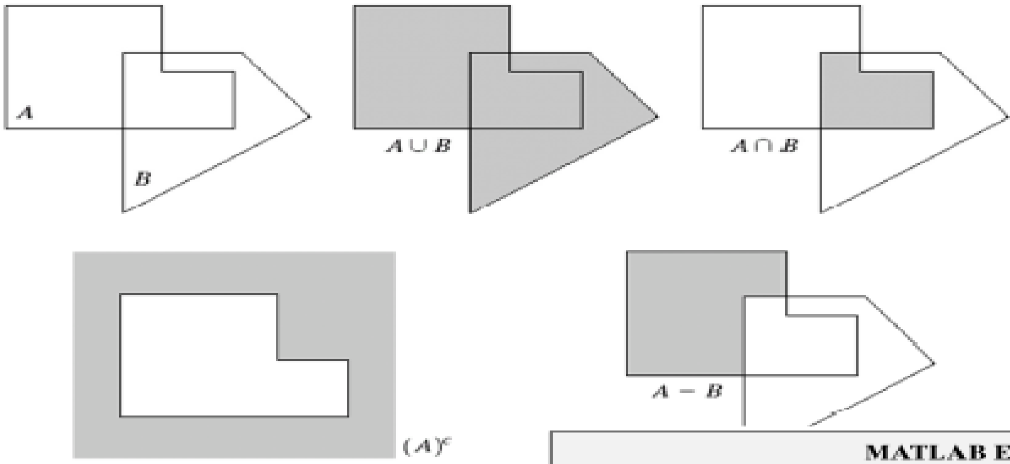
Some History

George Matheron (1975) *Random Sets and Integral Geometry*, John Wiley.

Jean Serra (1982) *Image Analysis and Mathematical Morphology*, Academic Press.

Petros Maragos (1985) *A Unified Theory of Translations-Invariant Systems with Applications to Morphological Analysis and Coding of Images*, Doctoral Thesis, Georgia Tech.

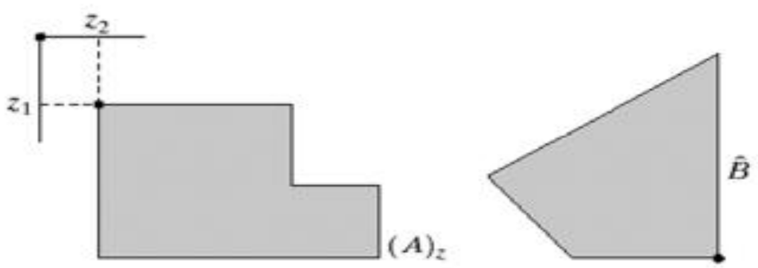
Logic Operations involving binary images



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.

Set Operation	MATLAB Expression for Binary Images	Name
$A \cap B$	A & B	AND
$A \cup B$	A | B	OR
A^c	~A	NOT
$A - B$	A & ~B	DIFFERENCE



a b

FIGURE 9.2
(a) Translation of A by z.
(b) Reflection of B. The sets A and B are from Fig. 9.1.

Translation	$(A)_z = \{w \mid w = a + z, \quad \text{for } a \in A\}$	Translates the origin of A to point z.
Reflection	$\hat{B} = \{w \mid w = -b, \quad \text{for } b \in B\}$	Reflects all elements of B about the origin of this set.

Logic operations are fundamental image processing techniques that apply logical operations to binary images. These operations consider the binary images as sets and perform set operations on corresponding pixels. The primary logical operations are AND, OR, NOT, and XOR

AND Operation

- The AND operation is used to find the intersection of two binary images.
- Each pixel in the output image is 1 if both corresponding pixels in the input images are 1, otherwise, it is 0.
- **Application**: This operation is useful for masking, where you want to retain certain parts of an image.

OR Operation

- The OR operation is used to find the union of two binary images.
- Each pixel in the output image is 1 if at least one of the corresponding pixels in the input images is 1, otherwise, it is 0.
- **Application**: This operation is useful for combining multiple binary masks.

NOT Operation

- The NOT operation inverts the binary image. Each 0 becomes 1 and each 1 becomes 0.
- **Application**: This operation is used to find the complement of an image.

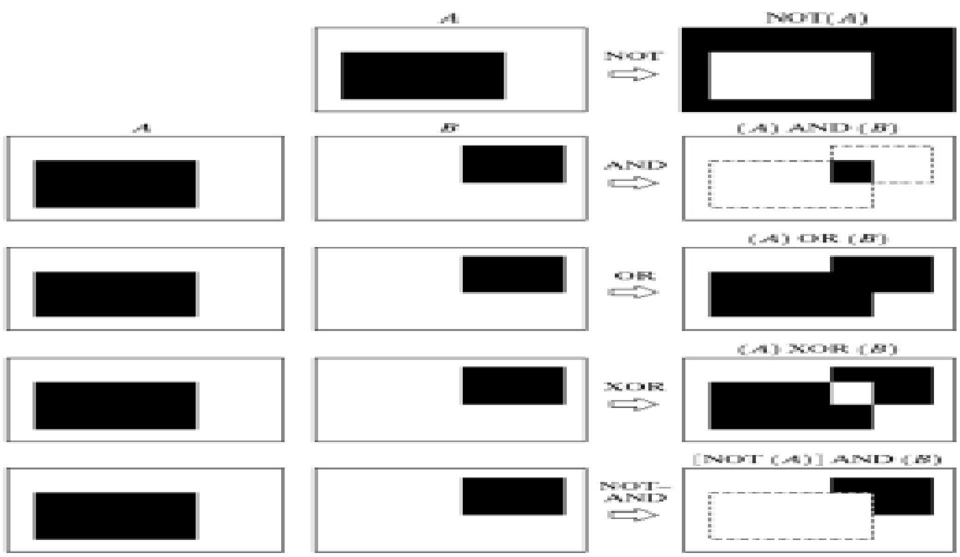


FIGURE 9.3 Some logic operations between binary images. Black represents binary 1s and white binary 0s in this example.

XOR Operation

- The XOR operation is used to find the difference between two binary images.
- Each pixel in the output image is 1 if the corresponding pixels in the input images are different, otherwise, it is 0.
- **Application**: This operation is useful for detecting changes between two images.

Morphological image processing

- **It** is a collection of non-linear operations related to the shape or morphology of features in an image.

- Morphological operations rely only on the ***relative ordering of pixel values, not on their numerical values***, and therefore are especially suited to the processing of binary images.

- Morphological operations can also be applied to greyscale images such that their light transfer functions are unknown and therefore their absolute pixel values are of no or minor interest.

The Hit or (Fit)Miss Transformation:

Morphological techniques probe an image with a small shape or template called a **structuring element**.

The structuring element is positioned at all possible locations in the image and it is compared with the corresponding neighbourhood of pixels.

Some operations test whether the element "**fits**" within the neighbourhood, while others test whether it "**hits**" or intersects the neighbourhood:

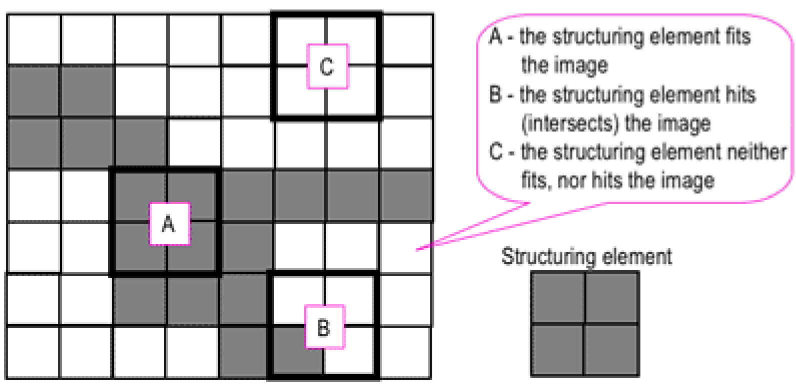
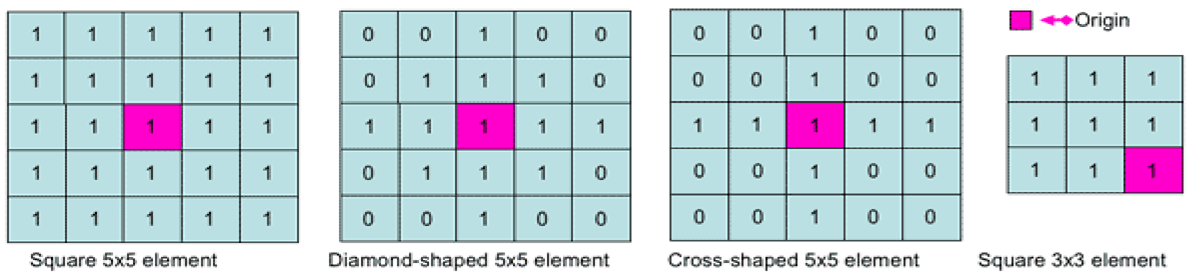


fig. Probing of an image with a structuring element
(white and grey pixels have zero and non-zero values, respectively).

A morphological operation on a binary image creates a new binary image in which the pixel has a non-zero value only if the test is successful at that location in the input image.

The **structuring element** is a small binary image, i.e. a small matrix of pixels, each with a value of zero or one:

- The matrix dimensions specify the *size* of the structuring element.
- The pattern of ones and zeros specifies the *shape* of the structuring element.
- An *origin* of the structuring element is usually one of its pixels, although generally the origin can be outside the structuring element.



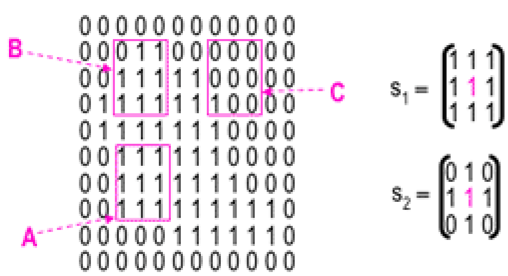
Examples of simple structuring elements.

A common practice is to have odd dimensions of the structuring matrix and the origin defined as the centre of the matrix.

Stucturing elements play in moprphological image processing the same role as convolution kernels in linear image filtering.

When a structuring element is placed in a binary image, each of its pixels is associated with the corresponding pixel of the neighborhood under the structuring element.

The structuring element is said to **fit** the image if, for each of its pixels set to 1, the corresponding image pixel is also 1. Similarly, a structuring element is said to **hit**, or intersect, an image if, at least for one of its pixels set to 1 the corresponding image pixel is also 1.



		A	B	C
fit	s_1	yes	no	no
	s_2	yes	yes	no
hit	s_1	yes	yes	yes
	s_2	yes	yes	no

Fitting and hitting of a binary image with structuring elements s_1 and s_2.

Zero-valued pixels of the structuring element are ignored, i.e. indicate points where the corresponding image value is irrelevant.

Erosion and dilation

Dilation and erosion are basic morphological processing operations. They are defined in terms of more elementary set operations, but are employed as the basic elements of many algorithms.

Both dilation and erosion are produced by the interaction of a set called a structuring element with a set of pixels of interest in the image.

The structuring element has both a shape and an origin.

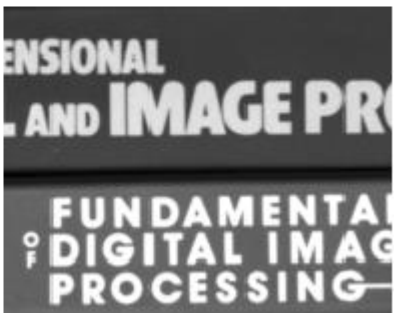
Let A be a set of pixels and let B be a structuring element. Let (ˆB)s be the reflection of B about its origin and followed by a shift by s. Dilation,

written $A \oplus B$, is the set of all shifts that satisfy the following:

$$A \oplus B = \{s | (\hat{B})_s \cap A \neq \emptyset\}$$

Equivalently,

$$A \oplus B = \{s | \left((\hat{B})_s \cap A\right) \subseteq A\}$$



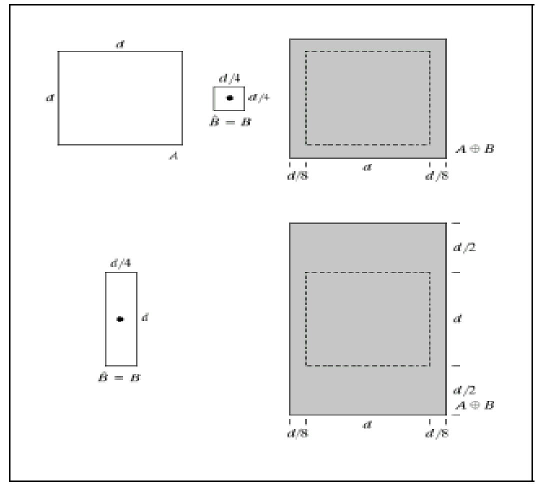
Greyscale image

Binary image by thresholding

Erosion: a 2×2 square structuring element

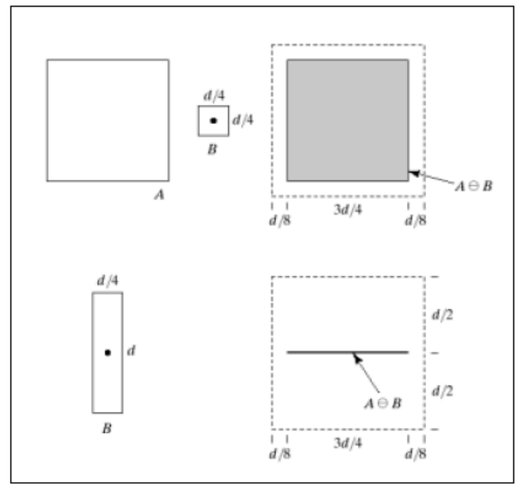
Erosion with small (e.g. 2×2 - 5×5) square structuring elements shrinks an image by stripping away a layer of pixels from both the inner and outer boundaries of regions. The holes and gaps between different regions become larger, and small details are eliminated:

Morphological Dilation



Any pixel in the output image touched by the · in the structuring element is set to ON when any point of the structuring element touches a ON pixel in the original image.
This tends to close up holes in an image by expanding the ON regions. It also makes objects larger.
Note that the result depends upon both the shape of the structuring element and the location of its origin.

Morphological Erosion:



Any pixel in the output image touched by the · in the structuring element is set to ON when every point of the structuring element touches a ON pixel in the original image.
This tends to makes objects smaller by removing pixels.

Figure: with Example

Morphological Erosion + Dilation

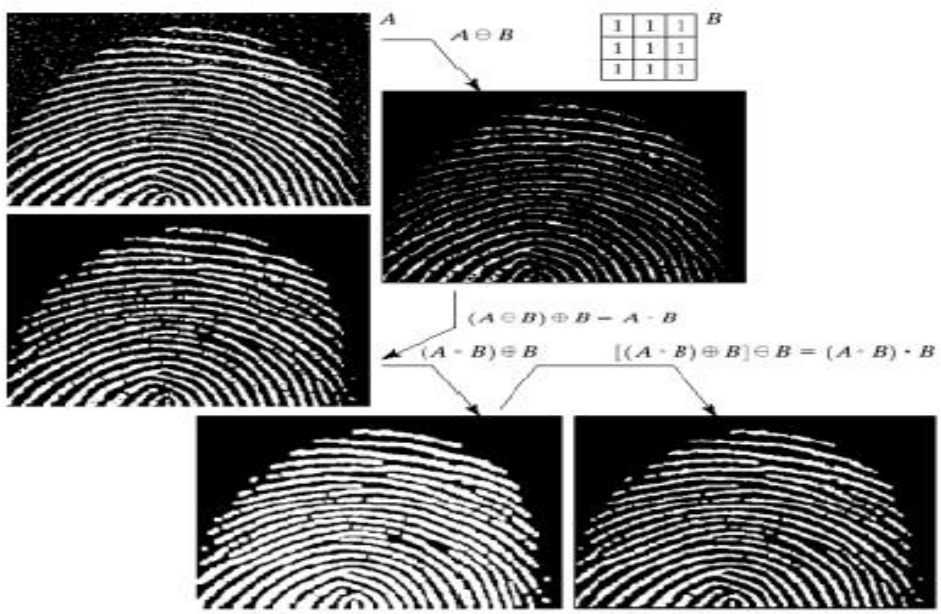
The effect of erosion followed by dilation is illustrated below.

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.

Fingerprint Image Cleanup

The use of ERODE + DILATE is illustrated by this example



a b
d c
e f

FIGURE 9.11
(a) Noisy image.
(c) Eroded image.
(d) Opening of *A*.
(d) Dilation of the opening.
(e) Closing of the opening. (Original image for this example courtesy of the National Institute of Standards and Technology.)

Opening and Closing

In mathematical morphology, **opening** is the dilation of the erosion of a set A by a structuring element B:

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

where $\ominus$ and $\oplus$ denote erosion and dilation, respectively.

Together with closing, the opening serves in computer vision and image processing as a basic workhorse of morphological noise removal. Opening removes small objects from the foreground (usually taken as the bright pixels) of an image, placing them in the background, while closing removes small holes in the foreground, changing small islands of background into foreground. These techniques can also be used to find specific shapes in an image. Opening can be used to find things into which a specific structuring element can fit (edges, corners, ...).

One can think of *B* sweeping around the inside of the boundary of *A*, so that it does not extend beyond the boundary, and shaping the *A* boundary around the boundary of the element.

Opening and closing

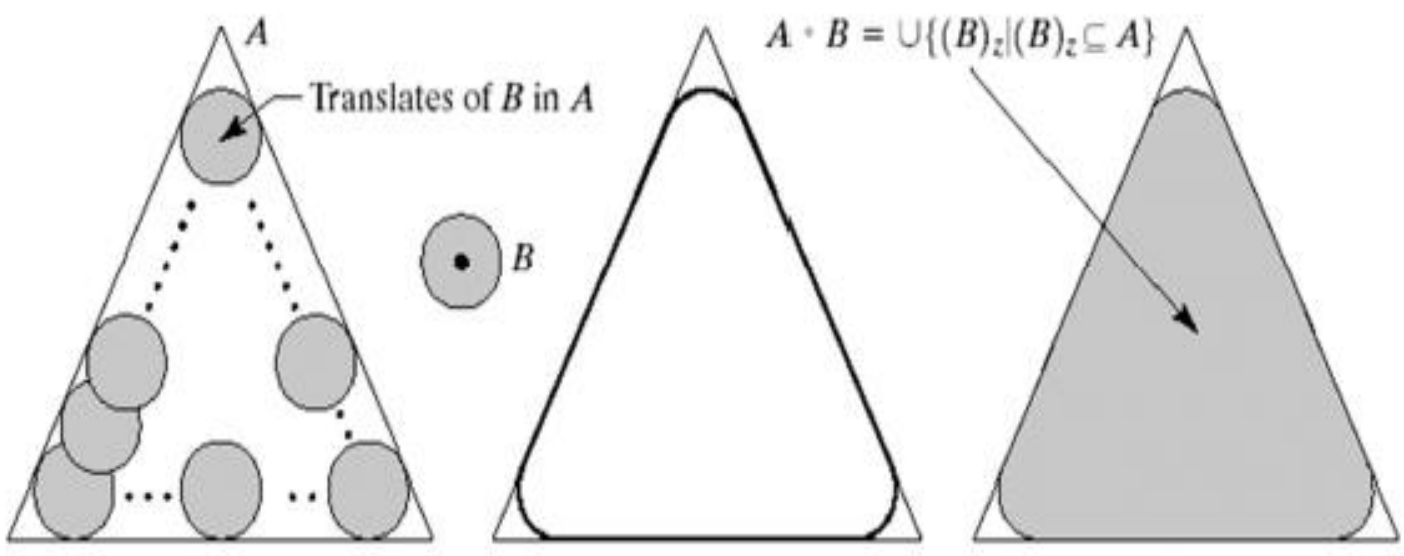
OPENING is erosion followed by dilation

CLOSING is dilation followed by erosion

Opening	$A \circ B = (A \ominus B) \oplus B$	Smoothes contours, breaks narrow isthmuses, and eliminates small islands and sharp peaks. (I)
Closing	$A \bullet B = (A \oplus B) \ominus B$	Smoothes contours, fuses narrow breaks and long thin gulfs, and eliminates small holes. (I)

Opening

A different formulation:



a b c d

FIGURE 9.8 (a) Structuring element B "rolling" along the inner boundary of A (the dot indicates the origin of B). (c) The heavy line is the outer boundary of the opening. (d) Complete opening (shaded).

Closing

A different formulation:

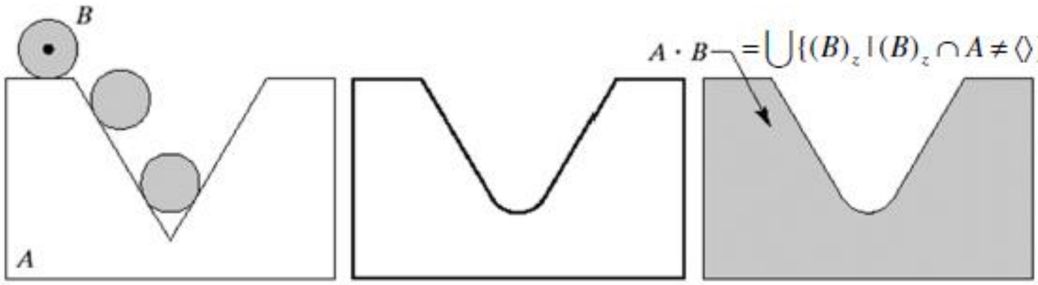


FIGURE 9.9 (a) Structuring element B "rolling" on the outer boundary of set A. (b) Heavy line is the outer boundary of the closing. (c) Complete closing (shaded).