

University of Nottingham UK | CHINA | MALAYSIA

Introduction to Image Processing

Lecture 5 Morphology



COMP2005: Introduction to Image Processing Week 23 – 10:00am Friday – 01 March 2024







Learning Outcomes



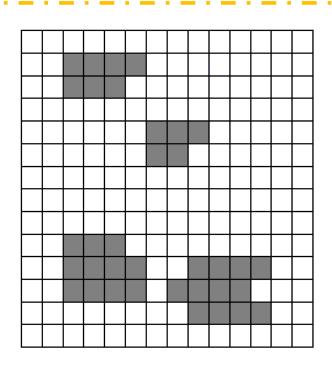
- 1. Connected Components
- 2. What is Mathematical Morphology
- 3. Erosion and Dilation
- 4. Opening and Closing
- 5. ROIs and Masks

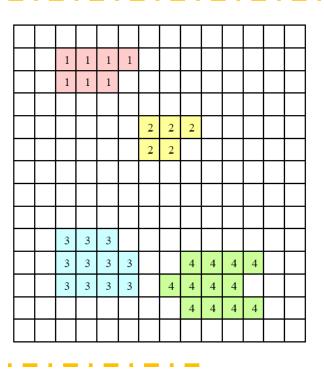






If black pixels are objects & white are background, give each connected set of black pixels a different label





Connected can mean 4- or 8- neighbours REMEMBER



- Connected component algorithms are slow and often a bottleneck

- Keep a separate output array for labels
- Two passes over the image

A sequential algorithm using 4-neighbours

- Scan image top left to bottom right
- Look at top and left neighbours (already given labels)
- Can we assign either of their labels to the current pixel?

First pass



- If the current pixel is **foreground**, there are 3 cases:

- 1. left = top = background → assign current pixel a new label
- 2. one of left and top is background, the other foreground → assign current pixel the foreground pixel's label
- 3. Left = top = foreground → assign current pixel one of their labels and note that their labels are equal in an *equivalence table*

Step 3 detects mergers between two previously separate components

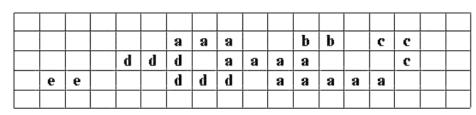


 Consider each equivalent pair of labels, set all instances of the higher label to the lower

Second pass

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	0	0	1	1	0	1	1	0	0
0	0	0	0	1	1	1	0	1	1	1	1	0	0	0	1	0	0
0	1	1	0	0	0	1	1	1	0	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

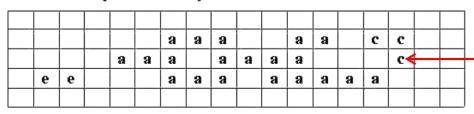
1st Scan



d=a, b=a d=a

:equivalence table

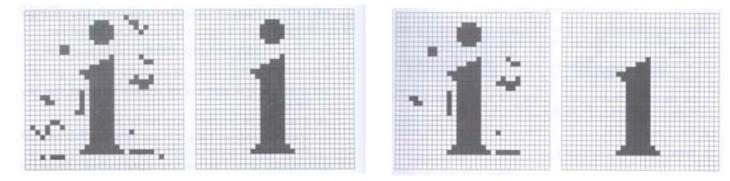
2nd Scan: Replace b & d by a



Using 8 neighbours makes the algorithm (a little) more complex and gives (slightly) different results



- Can compute features of and apply tests to components to process the underlying image
 - e.g., apply a (size) threshold T to the number of pixels in the component



T=10 T=25

- Shape is captured by measures like area/boundary length
- Many application-specific features and tests exist



What is Mathematical Morphology



Mathematical Morphology

- A branch of image processing which treats images as sets of pixels and uses set theoretic operations to process them
- Developed for binary images, extended to grey level images
- Elements of sets are (x,y) coordinates of black (or white) pixels
- Perform operations by combining two sets:
 - A patch of the binary image to be processed
 - A Structuring Element, similar to the mask in a convolution process

Underlying mathematics is beyond the scope of COMP2032.

Instead, we will focus on how they work in practice and what they do

DISCLAIMER



Structuring Elements

- Binary masks, c.f. filter masks but identifying rather than weighting pixels
- Larger structuring elements produce more extreme effects
- Very similar effects can be achieved by repeated operations using a smaller but similarly shaped structuring element
- With larger structuring elements, it is quite common to use an approximately disk-shaped structuring element
- Need not be square, origin need not be in the centre

1	1	1
1	\bigcirc	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		

ACK: Prof. Tony Pridmore, UNUK

COMP 2005 | Morphology 1



Morphological Operations

Expands a foreground (or background) object A using structuring element B

Dilation

13

Erosion

Shrinks a foreground (or background) object A using structuring element B

- The boundaries between foreground and background are often smoothed in the process
- The amount and the way objects grow and shrink depend upon the choice of the structuring element
- Dilating or eroding without specifying the structural element makes no sense than trying to filter an image without specifying the filter



University of Nottingham

UK | CHINA | MALAYSIA

Erosion & Dilation



Dilation

2 inputs

- Binary image to dilate
- Set of points to be considered; a structuring element
- Origin is the central element

1	1	1
1	1	1
1	1	1

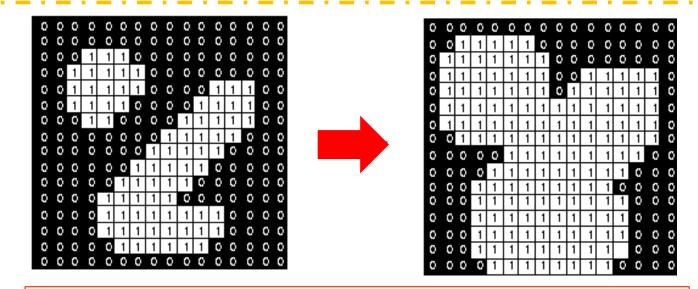
- The structuring element is superimposed on each of the background pixels such that origin of the structuring element coincides with the input pixel position
- If any of the '1' pixels in the structuring element overlap (intersect) the foreground then the background pixel is also set to foreground

Algorithm (dilate foreground)



Dilation

- Gradually enlarges the boundaries of regions of foreground pixels (i.e., white pixels, typically)
- Areas of foreground pixels grow in size while holes within those regions become smaller



Result of dilation with a square 3 x 3 structuring element

ACK: Prof. Tony Pridmore, UNUK

16



A More interesting Example

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.

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.



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.

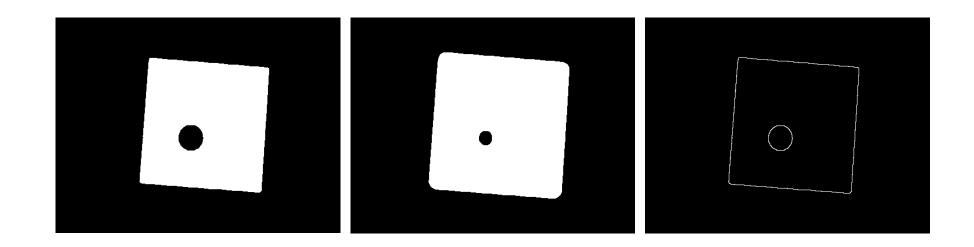
17





Edge Detection by Dilation

- Dilation input image (note rounding of corners)
- Subtract from original
- Edges remain (pixels on the outside of the boundary)



ACK: Prof. Tony Pridmore, UNUK

18



Erosion



Binary image to erode

- Set of points to be considered; a structuring element

Origin is the central element

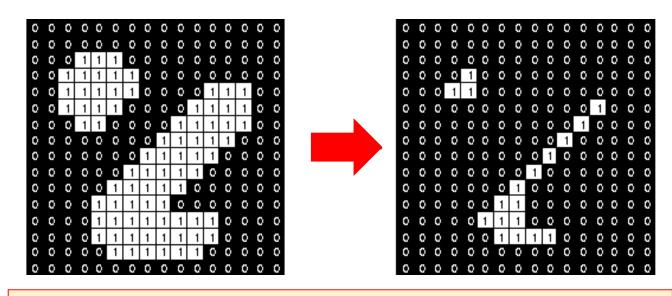
1	1	~
1	1	1
1	1	1

- The structuring element is superimposed on each of the foreground pixels such that origin of the structuring element coincides with the input pixel position
- If any of the '1' pixels in the structuring element overlap (intersect) the background then the foreground pixel is also set to background

Algorithm (erode foreground)



Erosion



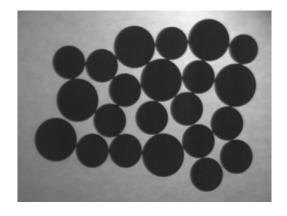
Result of erosion with a square 3 x 3 structuring element

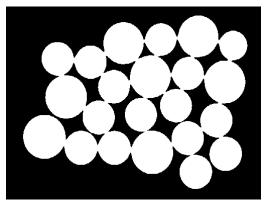
- Erosion is the *dual* of dilation, *i.e.*, eroding foreground pixels is equivalent to dilating the background pixels with the same structuring element

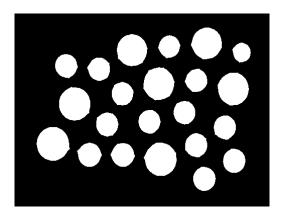


Erosion

- Counting objects (cells, coins) can be difficult if they touch
- Erosion can separate them







Erosion can be used for edge detection too – giving pixels on the inside of the boundary

REMEMBER

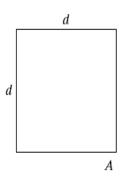


Asymmetric Structuring Elements



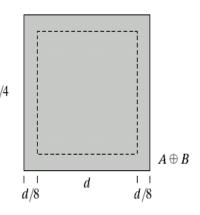
FIGURE 9.4

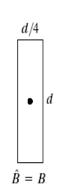
- (a) Set *A*.
- (b) Square structuring element (dot is the center).
- (c) Dilation of *A* by *B*, shown shaded.
- (d) Elongated structuring element.
- (e) Dilation of *A* using this element.

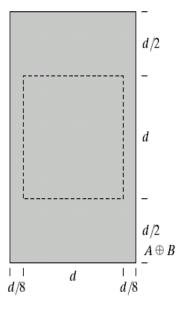


d/4

 $\hat{B} = B$







Dilation



Asymmetric Structuring Elements

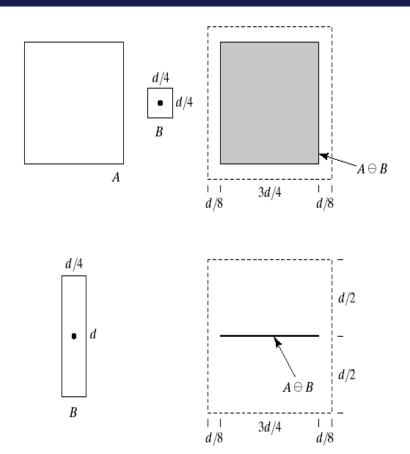




FIGURE 9.6 (a) Set A. (b) Square structuring element. (c) Erosion of *A* by *B*, shown shaded. (d) Elongated structuring element. (e) Erosion of *A* using this element.

Erosion



Break





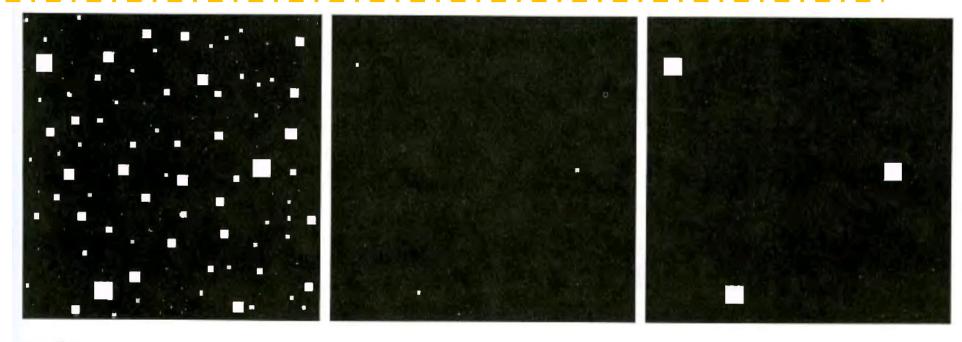
University of Nottingham UK | CHINA | MALAYSIA

Opening & Closing



Combining Dilation and Erosion

Its rare to need only erosion and dilation, and they are much more useful when combined



abc

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 and Closing

- First erode A with B, then dilate A with B
- Smoothes contours, eliminates protrusions

Opening

Closing

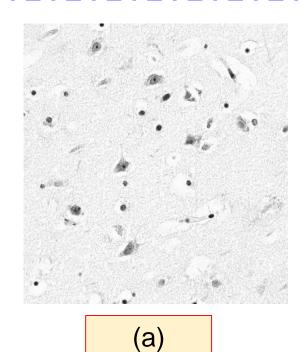
- First dilate A with B, then erode A with B
- Smoothes sections of contours, fuses narrow breaks and long thin gulfs, eliminates small holes and fill gaps in contours

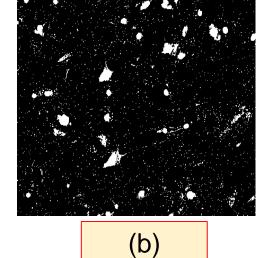
- These operations are dual to each other
- These operations can be applied multiple times, but have an effect only once (the first time)

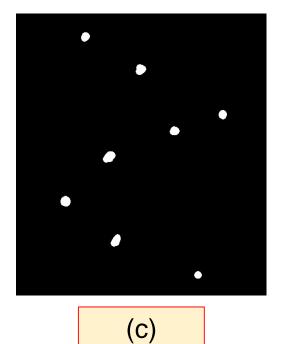


Opening

(a) Contains two kinds of cells: small, black ones and larger, grey ones. Thresholding with a value of 210 yields (b). Opening (b) using an 11 pixel circular structuring elements yields (c)





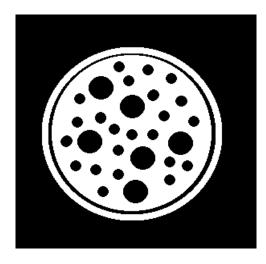


ACK: Prof. Tony Pridmore, UNUK

COMP 2005 Morphology 28



Closing



To retain only the large holes, we can simply perform a closing with a disk-shaped structuring element with a diameter larger than the smaller holes, but smaller than the large holes





The result of a closing with a 22 pixel diameter disk



University of Nottingham UK | CHINA | MALAYSIA

ROIs & Masks



The Whole Image

- All our examples so far have applied processes to the whole image or section of image – input
- The tools we have become more powerful if we can apply them to selected pixels, and more powerful still if the selection can be automatic

OpenCV supports Region of Interest Processing through NumPy array slicing

- ROI Object Creation and Modification
- Mask creation
- ROI filtering



Python OpenCV

Supports filtering and inpainting of:

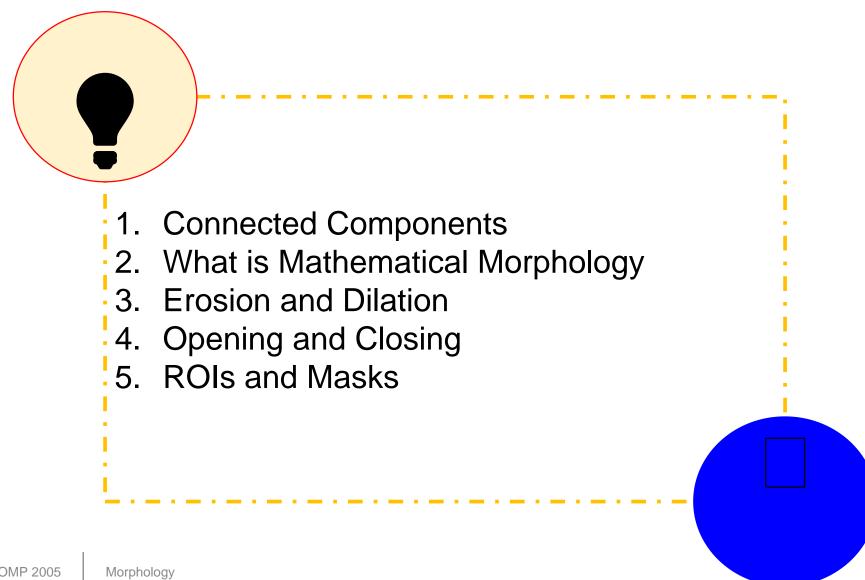
- Manually defined rectangular ROIs
- Interactively drawn bounding box ROIs
- ROIs defined by any appropriately sized binary image
- ROIs defined by colour

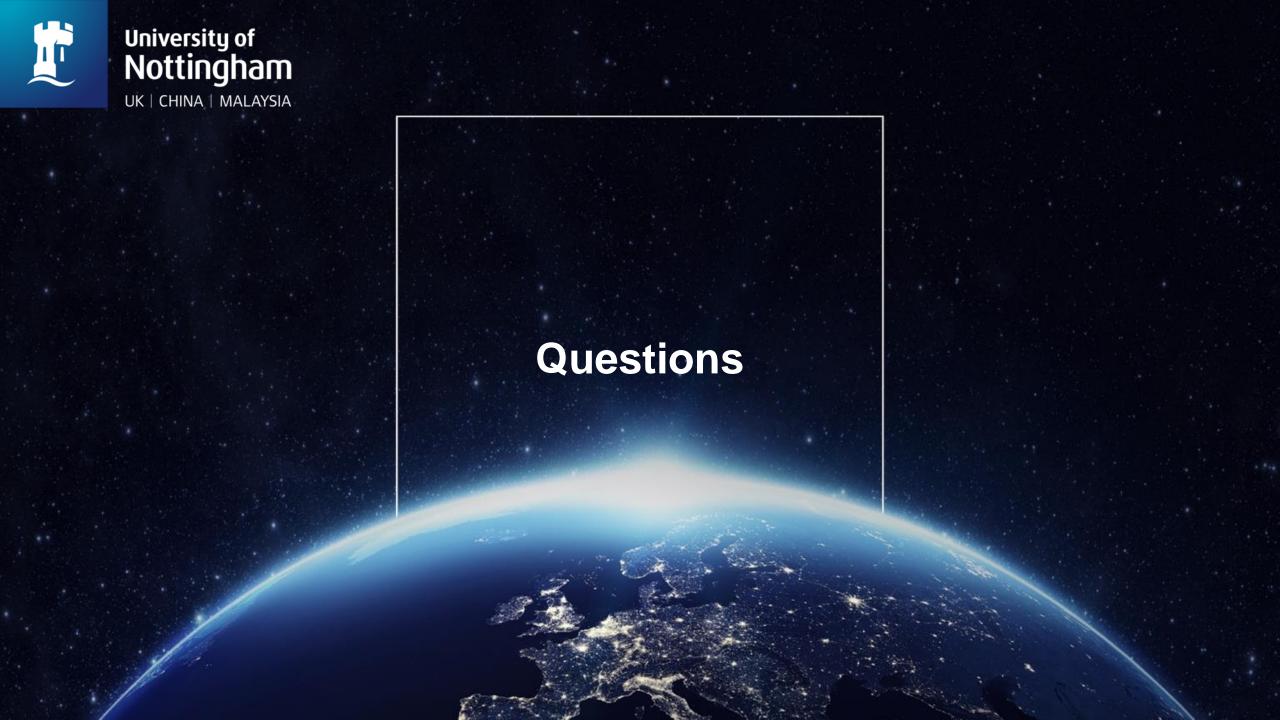
The concept of ROIs is very powerful; code can be written to allow the output of one process to determine where another is applied

REMEMBER



Summary







NEXT:

Derivates and Edges