

Comp3302 Unit1 Morphological Operations -1-

C.B.Price September 2021

Purpose	To explore “erosion” and “dilation”
Files Required	Octave scripts in the zipped folder “Octave_ImProc_Release” on the module site.
ILO Contribution	Position Paper 1 (part 1) Learning Outcome 2
Send to Me	
Homework	

Activities

1 Investigating Structuring Elements

(a) Generate the square structuring element by typing this directly in the Octave Command Window

```
se = strel('square',3)
```

Now change the size of the SE.

(b) Try out some of the following.

	Parameters
<code>se = strel("diamond",RADIUS)</code>	Max vertex distance from centre
<code>se = strel("line",LEN,DEG)</code>	Length of line, angle in degrees from horiz.
<code>se = strel("rectangle", [HEIGHT WIDTH])</code>	Height and width
<code>se = strel("disk",RAD,0)</code>	radius. The “0” needs to be there

2 Exploring Erosion and Dilation using Various SEs

The script “Morph2” contains a small synthetic binary image with a white square. Here you can test out the effects of erosion and dilation operations with various SEs.

(a) Open up the script “Morph2” and look for the line **SE1 = strel('diamond',1);** Also look for the line **BW1 = imdilate(BW,SE1);**

(b) Run the script as it stands and see the effect of the dilation. Note the red asterisks mark the original pixels.

(c) Now investigate the effects of other sizes and other SEs.

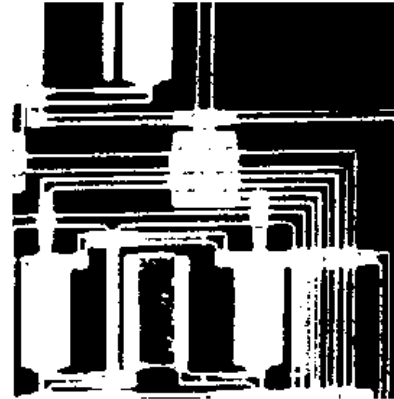
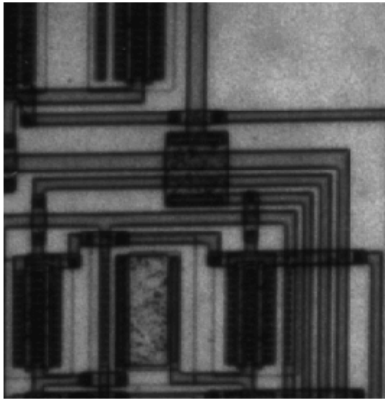
(d) Now replace the dilation line with erosion **imerode(...);**

(e) Now try erosion followed by dilation, and of course dilation followed by erosion.

3 Segmenting an Image based on Object Shape

Here we start with an image of a printed circuit in which there are two types of object: the darker rectangles (chips) and lines (connecting wires). First the image is thresholded, and then inverted so the object are white. This is done for you.

Orig gray level image



- (a) Run the script “Morph3” and you will be told the size of the image.
- (b) The SE we shall use to segment the chips is “rectangle”. Estimate the width and height of the SE and see what happens.
- (c) Repeat until you obtain a good segmentation, showing all four chips and zero wires.
- (d) You could always try out different SEs

4 Cleaning up the segmented image from task “Thresh4”

If you look back at your Thresh4 investigations, you should have found the following:

- (i) For a smoothing kernel of size 3, there were more than the 10 objects identified
- (ii) For a smoothing kernel of size 5, 10 objects were identified
- (iii) For a kernel size 7, 10 objects were identified, but their areas reported were smaller

So there is a trade-off between kernel size and accuracy of reported areas. Here we shall use morphological operations to address this problem.

Here we shall take this approach. Let’s say we want to fill the holes, so first we dilate using a SE, and then we erode to remove extra pixels added to the objects boundaries.

If you look at the script, we create a structuring element `se`, then we use it first to erode

- (a) Run the script “Morph1” and look at the images. Can you see the effect of dilation? Has this done what we wanted. Can you see the effects of erosion? Has this done what we wanted?
 - (b) Explain why this order of processing is doomed to fail.
-

(c) Now run Morph1a where the order of dilation and erosion is reversed, so we remove those bright pixels in the background first. Does this work? Is there a side-effect

(d) So how can we fix the problem. Return to Morph1 where we have successfully filled in the holes but have those bright pixels in the background remaining. Can you think of a combination of morphological operations to remove these while not affecting the real objects? Try to add code in the script to do this.

5 Exploring the “Hit-Or-Miss” Transformation

The script “HOM_Expts” asks you to experiment with the HOM Transformation.

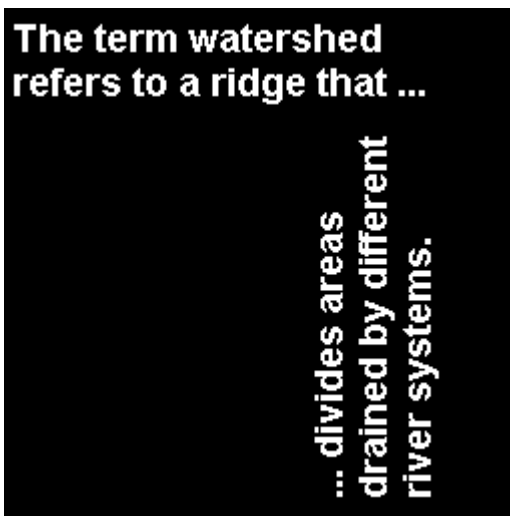
(a) Run the script and you’ll see that the figure “3” is identified.

(b) Type “se1” in the Command Window and check that the HIT SE is indeed a “3”. Now type “se2” which will show you the MISS SE.

(c) Change the image in any way you wish, adding or deleting pixels to the “3”. Do this using the array “BW” which has format **BW(row,col) = 1;** to set a pixel.

6 Applying the “Hit-or-Miss” Transformation

(a) Run the script “HitOrMiss” and you will be presented with this binary image



Choose a character with is repeated from the horizontal text. Do this by imaging a box around the character. Click first at the top-left of the box, then at the bottom right of the box. Your character should appear in a separate window and its location(s) in the original image labelled. Repeat!
