Image Processing - Assignment 2

Group 3
Martin Eek Gerhardsen

November 22

Department of Engineering Cybernetics

Contents

1 Task 1: Theory

a)

Opening can be defined as:

$$f \circ b = (f \ominus b) \oplus b \tag{1}$$

Or, first applying an erosion, then applying a dilation using the same structuring element on the image.

Closing can be defined as:

$$f \bullet b = (f \oplus b) \ominus b \tag{2}$$

Or, first applying a dilation, then applying a erosion using the same structuring element on the image.

After the first opening or closing, applying more openings or closings will not have any effect on the same image as you will inevitably erode and dilate the same shape each time.

b)

This is because the edge detection (and derivatives in general) enhance noise, making it virtually impossible to detect edges on a noisy image. Therefore it would be necessary to apply smoothing to the image before we can detect any edges. See fig. 1, which illustrates how noise is amplified when derivated.

c)

Rather than using a single threshold, we use two thresholds, one lower and one higher. Anything larger than the higher is marked as an edge, and anything below the lower is marked as not an edge. Those between these threshold can be seen as weak edges, and is marked as an edge if there is a strong edge next to it.

d)

We use hysteresis thresholding instead of a single threshold because it is difficult to get a single useful value for thresholding. If the value is too high, the weak edges (as mentioned in c)) would be ignored, and if the value is too low, then unwanted values (noise) would be counted as real edges. Hysteresis thresholding gives us a decent way to include likely edges (weak edges close to strong edges).

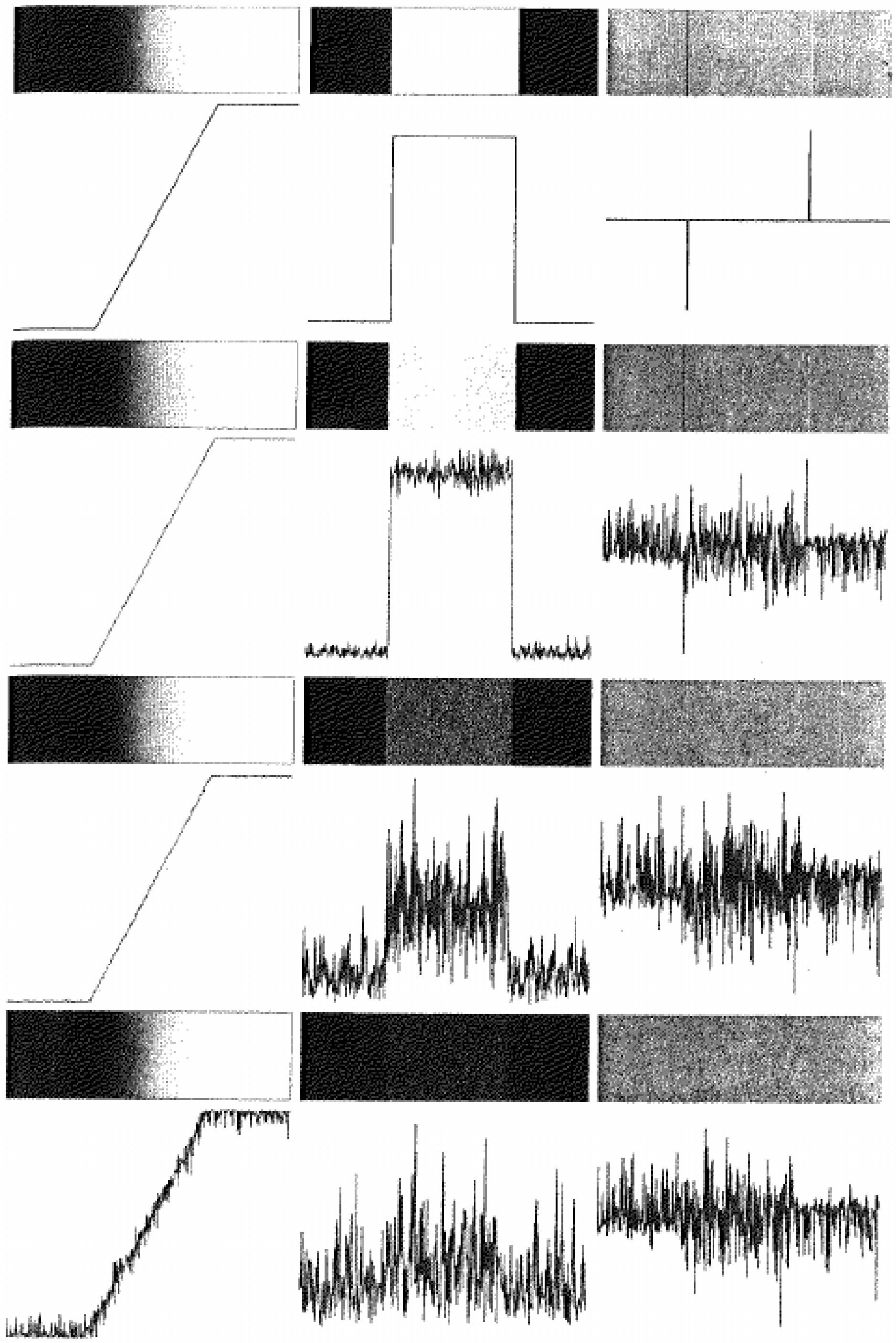

Figure 1: Ramp edge with different levels of noise. From figure 10.7 Digital Image Processing.

e)

Reflecting B has no effect. Centre has been highlighted with bold. Values outside A was handled as zeros.

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & \mathbf{1} & 1 \end{bmatrix}$$

$$A \oplus B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \oplus \begin{bmatrix} 1 & \mathbf{1} & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 2: Thumbprint segmented using Otsu's algorithm for thresholding.

2 Programming

2.1 Task 2: Segmentation

a)

For the result of the thumbprint, see fig. 2.

For the result of the polymer cell, see fig. 3.

2.2 Task 3: Morphology

a)

Opening is particularly well suited for removing the noise / particles outside the triangle shape, as it firstly erodes information with the structuring element, then dilates the shape of the triangle back. Similarly, closing is particularly well suited for removing the noise / holes inside the triangle,

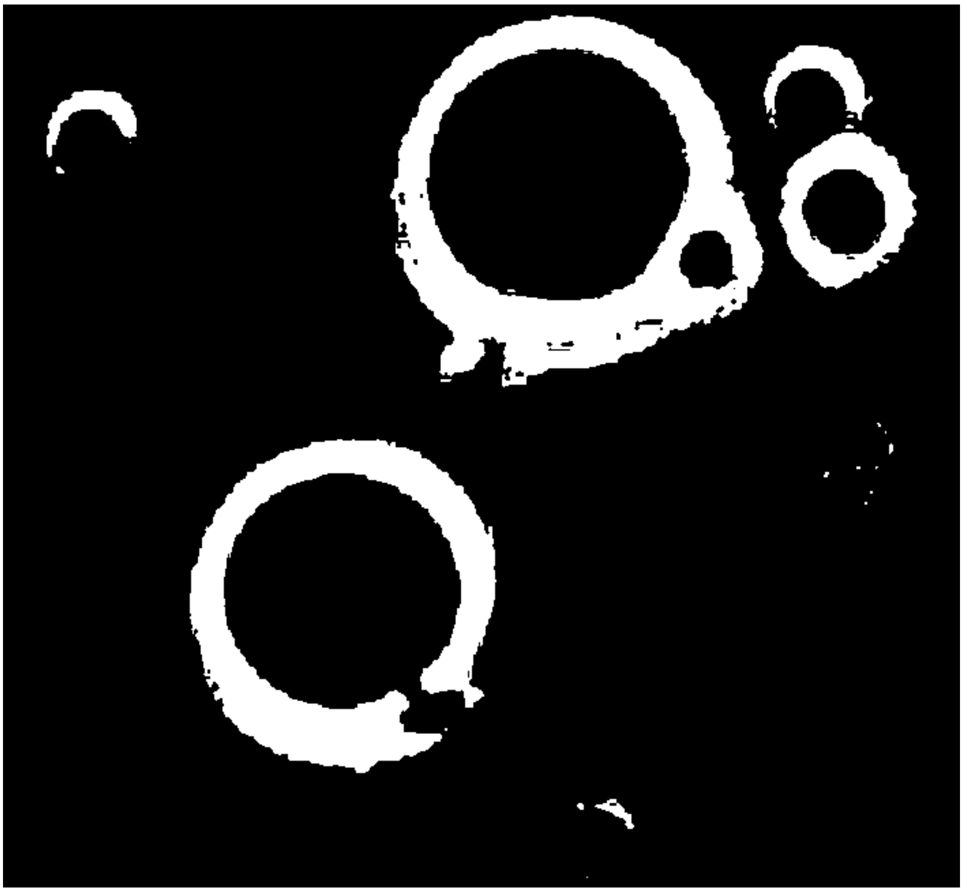

Figure 3: Polymer cell segmented using Otsu's algorithm for thresholding.

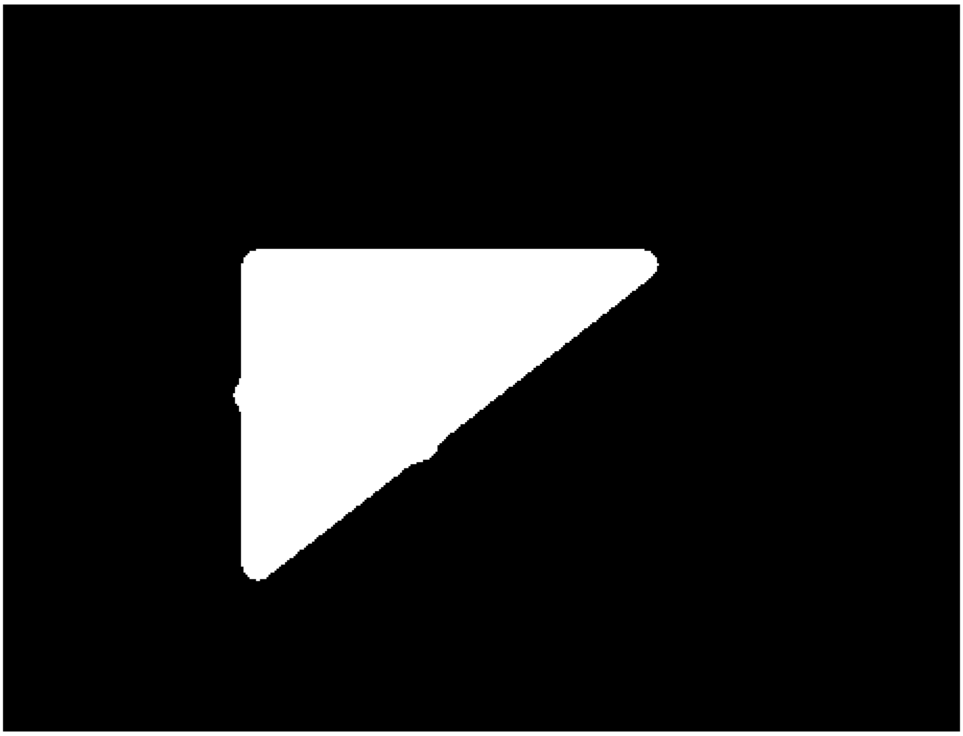

Figure 4: Filtered version of the noisy image.

as firstly the holse are dilated away, then the extra *padding* added by the dilation is removed by erosion.

See fig. 4 for the result. The structuring element used was a disk generated from the *skimage.morphology*-library, with a radius of 7. This was the best radius I found for removing all the noise, but keeping the structure of the triangle shape.

b)

See fig. 5 for the result of the distance transform.

c)

For the resulting boundary of the image, see

$(A \ominus B)$ removes the inner boundary, which we want to extract from A, then we find it by taking the difference $A - (A \ominus B)$. If we instead find the dilation $(A \oplus B)$, we will increase A by the outer boundary, and we may find this outer boundary as $(A \oplus B) - A$.

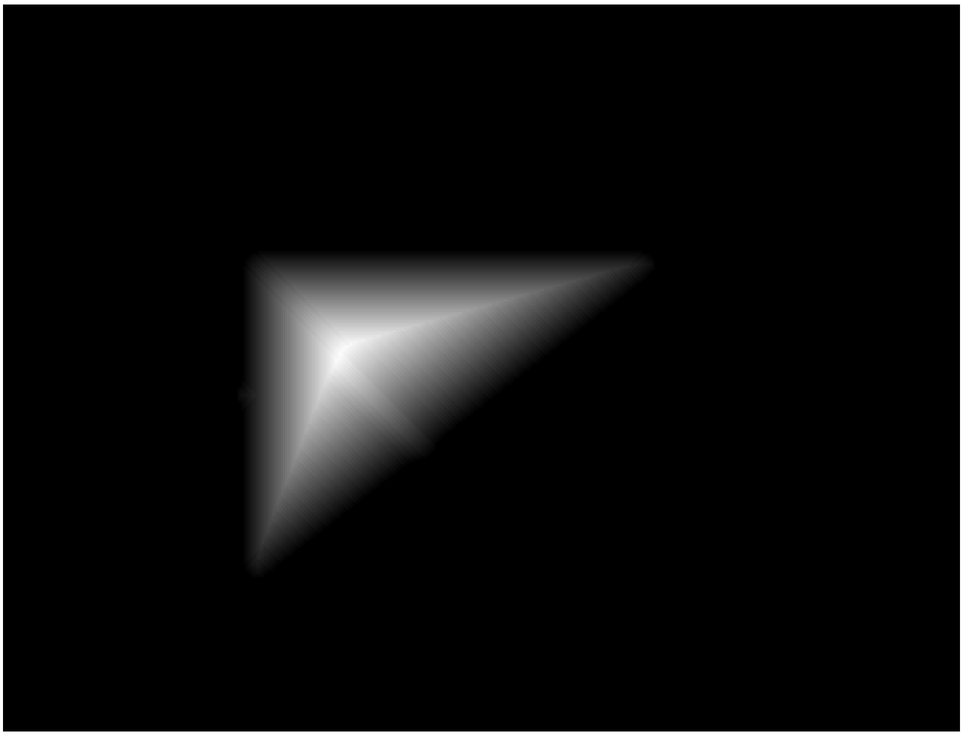

Figure 5: Distance transform of the filtered noisy triangle shape.

d)

See fig. 7.

Figure 6: Boundary of the **lincoln.png** image.

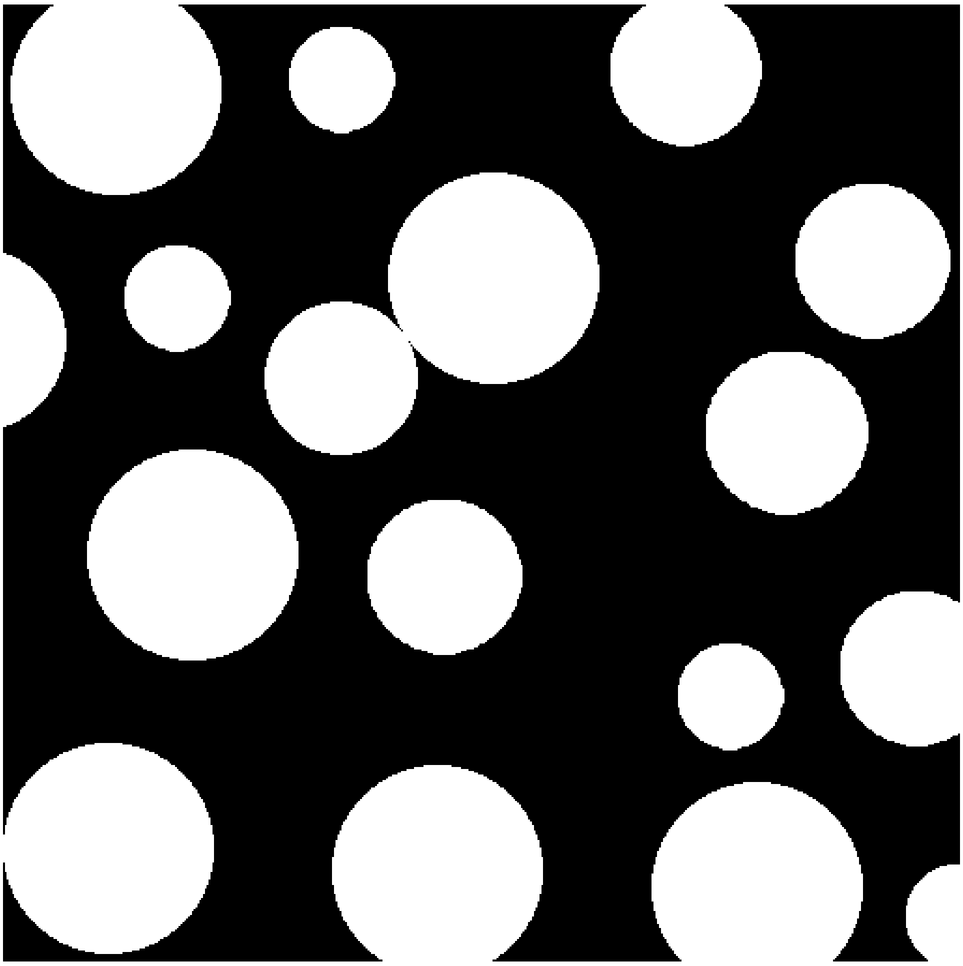

Figure 7: Balls filled by the algorithm.