Home Assignment 1

Shahin Mammadov
Abumansur Sabyrrakhim

February 20, 2021

1 DILATION AND EROSION

Dilation and erosion are two fundamental morphological operations. Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels on object boundaries. The number of pixels added or removed from the objects in an image depends on the size and shape of the structuring element used to process the image.By examining the expressions for morphological dilation and erosion we see that we need to get minimum and maximum of a pixel within structuring element. The functions called minVal and maxVal are responsible for setting the value of output pixel of th epixels that are in the neighbourhood of the size of SE within input image. Algorithm maxVal gets the maximum value which is used in dilation and minVal gets the minimum val which is used for erosion.

2 EXPERIMENTS

2.1 erosion of f1 by square of size 3

Input: `py final.py e SE1.txt f1.txt ef1_e1`

2.2 erosion of f2 horizontal line of length 3

Input: `py final.py e SE2.txt f2.txt ef2_e2`

2.3 erosion of f3 by horizontal line (–) of length 3

Input: `py final.py e SE3.txt f3.txt ef3_e2`

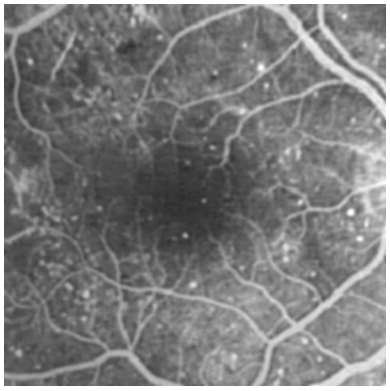

Figure 2.1: Original image

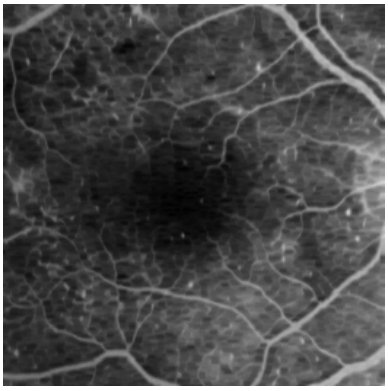

Figure 2.2: Erosion

2.4 erosion of f3 by square of size 5

Input: `py final.py e SE4.txt f3.txt ef3_e3`

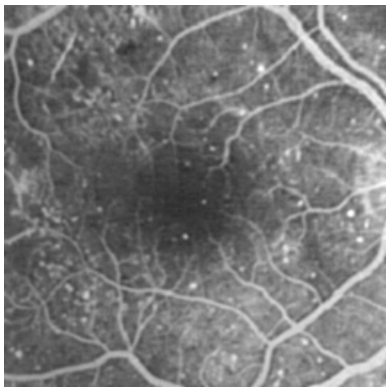

Figure 2.3: Original image

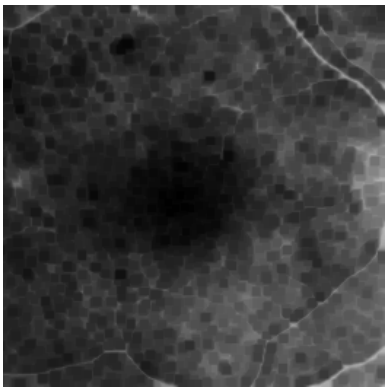

Figure 2.4: Erosion

2.5 erosion of f3 by backward diagonal of length 9

Input: `py final.py e SE5.txt f3.txt ef3_e4`

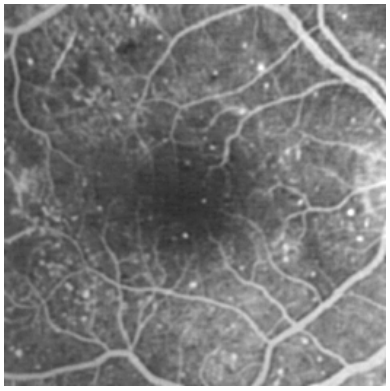

Figure 2.5: Original image

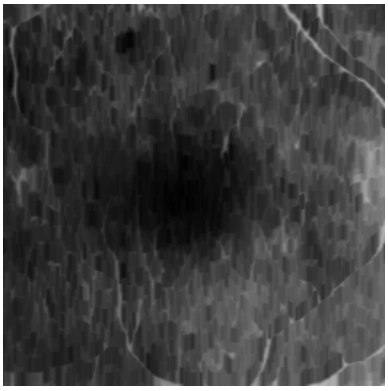

Figure 2.6: Erosion

2.6 erosion of f3 by forward diagonal of length 9

Input: `py final.py e SE6.txt f3.txt ef3_e5`

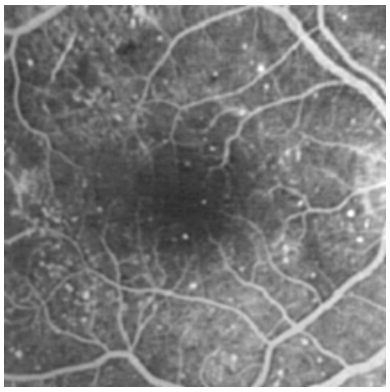

Figure 2.7: Original image

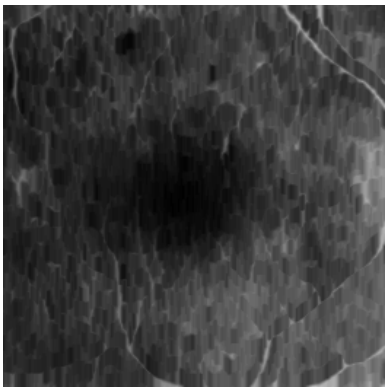

Figure 2.8: Erosion

2.7 dilation of f3 by square of size 5

Input: `py final.py d SE4.txt f3.txt df3_d3`

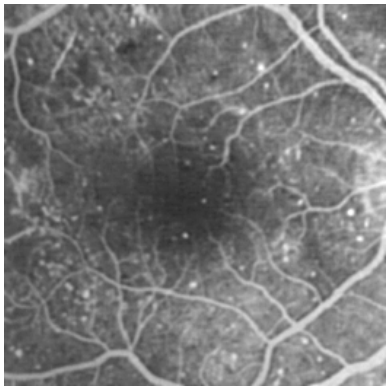

Figure 2.9: Original image

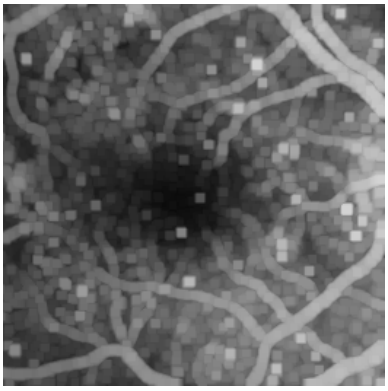

Figure 2.10: Dilation

2.8 dilation of f3 by backward diagonal of length 9

Input: `py final.py d SE5.txt f3.txt df3_d4`

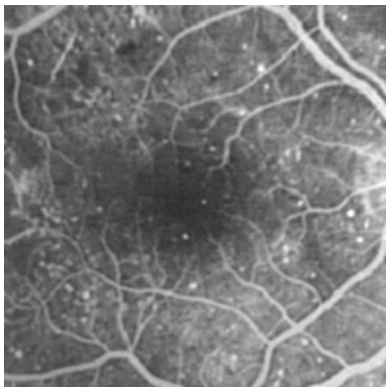

Figure 2.11: Original image

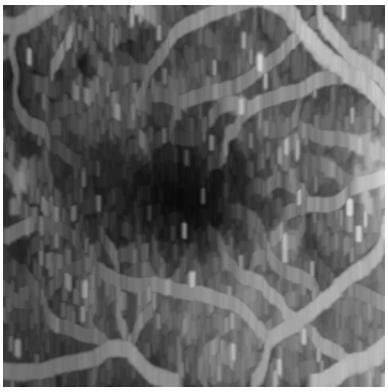

Figure 2.12: Dilation

2.9 dilation of f3 by forward diagonal of length 9

Input: `py final.py d SE6.txt f3.txt df3_d5`

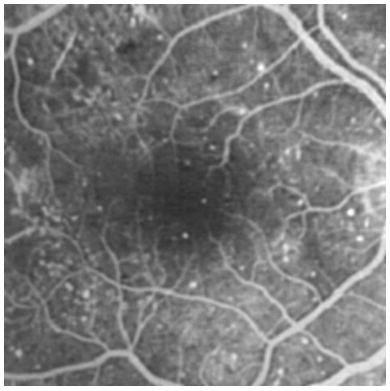

Figure 2.13: Original image

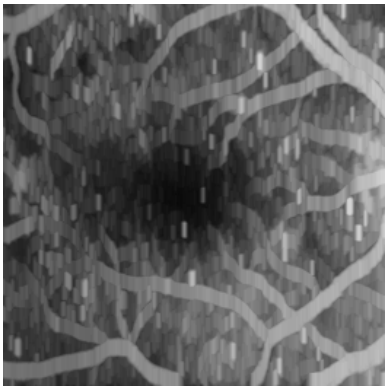

Figure 2.14: Dilation

2.10 dilation of f3 by forward diagonal of length 9

Input: `py final.py d SE4.txt f3.txt of3_o3`

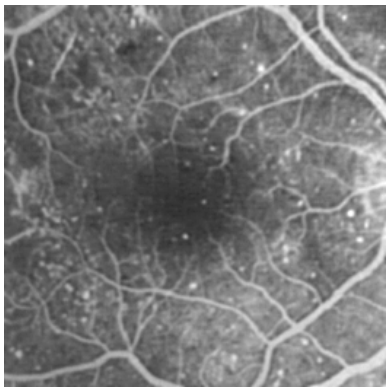

Figure 2.15: Original image

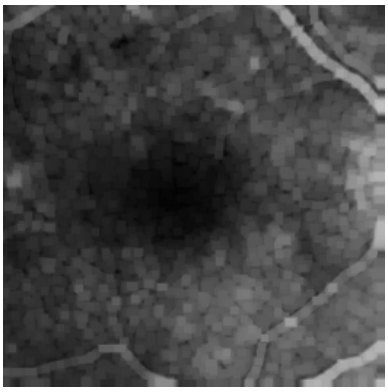

Figure 2.16: Opening

2.11 dilation of f3 by forward diagonal of length 9

Input: `py final.py d SE5.txt f3.txt of3_o4`

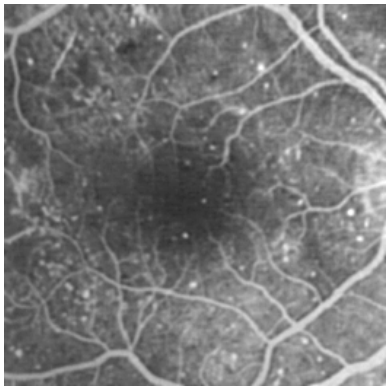

Figure 2.17: Original image

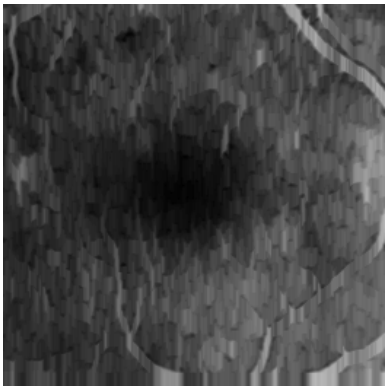

Figure 2.18: Dilation

2.12 asymmetric SE for erosion, dilation, closing, opening

1. Input for erosion:
    - `py final.py e SE_AS.txt f3.txt as_e`
2. Input for dilation:
    - `py final.py d SE_AS.txt f3.txt as_d`
3. Input for opening:
    - `py final.py o SE_AS.txt f3.txt as_o`
4. Input for closing:
    - `py final.py c SE_AS.txt f3.txt as_c`

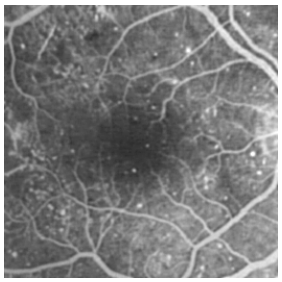

Figure 2.19: Original image

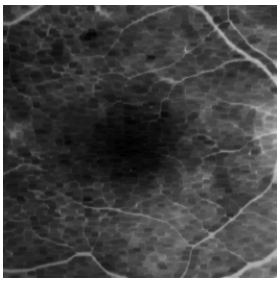

Figure 2.20: Erosion

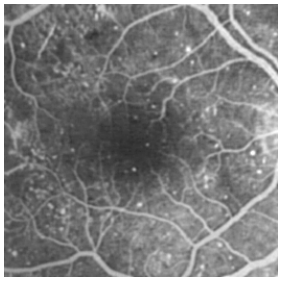

Figure 2.21: Original image

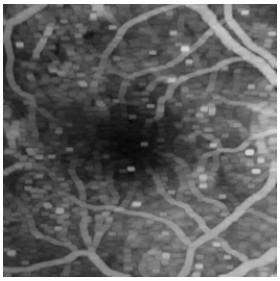

Figure 2.22: Dilation

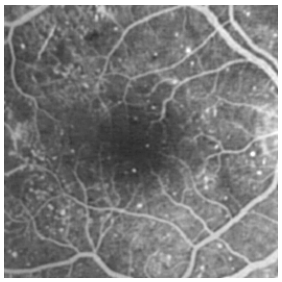

Figure 2.23: Original image

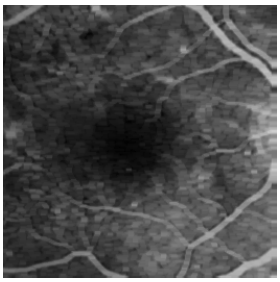

Figure 2.24: Opening

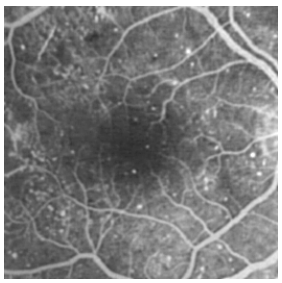

Figure 2.25: Original image

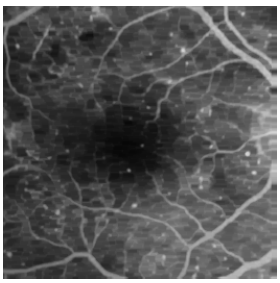

Figure 2.26: Closing

3 SELF-CHOSEN IMAGES AND EXPERIMENT

3.1 Image 1

3.1.1 Erosion

AFTER EXPERIMENTING WITH IMAGE 1, IT IS CLEAR TO OBSERVE THAT THE IMAGE BECOMES MORE BLURRY, AND HARDER TO DETECT DETAILS, ESPECIALLY WOMEN'S EYES AND SMILE Input: `py final.py e SE7.txt exp.txt exp1_ero`

Figure 3.1: Original image

Figure 3.2: Erosion

3.1.2 Dilation

Input: `py final.py e SE7.txt exp.txt exp1_dil`

Figure 3.3: Original image

Figure 3.4: Dilation

3.1.3 Opening

Input: `py final.py e SE7.txt exp.txt exp1_open`

Figure 3.5: Original image

Figure 3.6: Opening

3.1.4 Closing

Input: `py final.py e SE7.txt exp.txt exp1_close`

Figure 3.7: Original image

Figure 3.8: Closing

3.2 Image 2

AFTER EXPERIMENTING WITH IMAGE 2, IT IS CLEAR TO OBSERVE THAT THE BUTTERFLY ON THE IMAGE BECOMES EITHER TOO DARK OR TOO BRIGHT TO DETECT

3.2.1 Erosion

Input: `py final.py e SE7.txt exp2.txt exp2_ero`

Figure 3.9: Original image

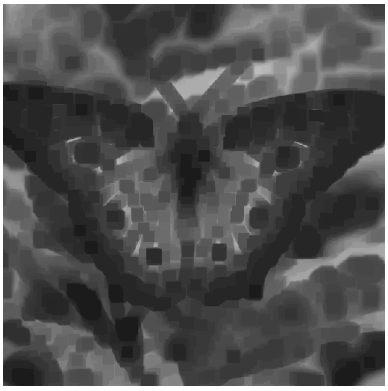

Figure 3.10: Erosion

3.2.2 Dilation

Input: `py final.py e SE7.txt exp2.txt exp2_dil`

Figure 3.11: Original image

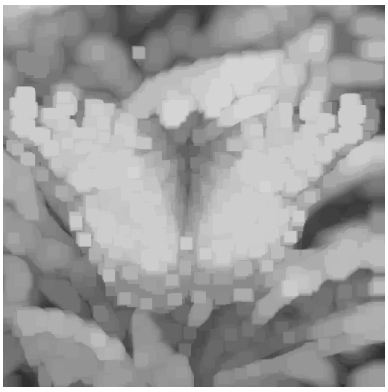

Figure 3.12: Dilation

3.2.3 Opening

Input: `py final.py e SE7.txt exp2.txt exp2_open`

Figure 3.13: Original image

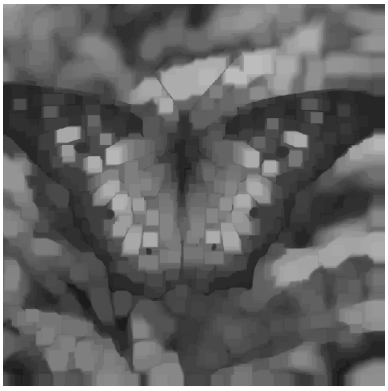

Figure 3.14: Opeining

3.2.4 CLosing

Input: `py final.py e SE7.txt exp2.txt exp2_close`

Figure 3.15: Original image

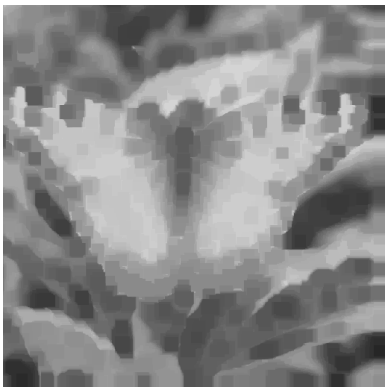

Figure 3.16: Closing

3.3 Image 3

AFTER EXPERIMENTING WITH IMAGE 3, IT IS CLEAR TO OBSERVE THAT THE TREES ON THE IMAGE BECOME TOO BLURRY TO DETECT

3.3.1 Erosion

Input: `py final.py e SE7.txt exp3.txt exp3_ero`

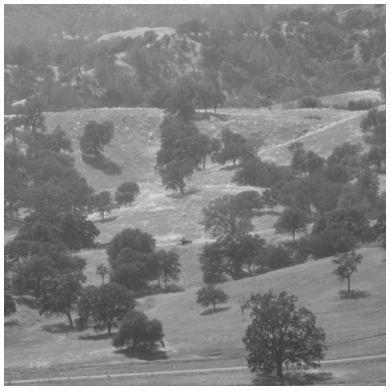

Figure 3.17: Original image

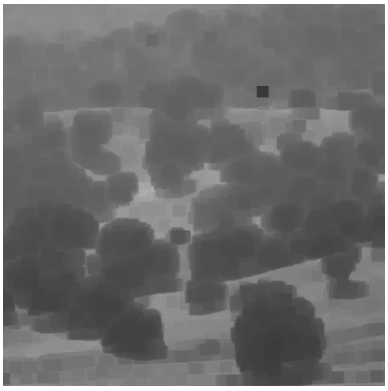

Figure 3.18: Erosion

3.3.2 Dilation

Input: `py final.py e SE7.txt exp3.txt exp3_dil`

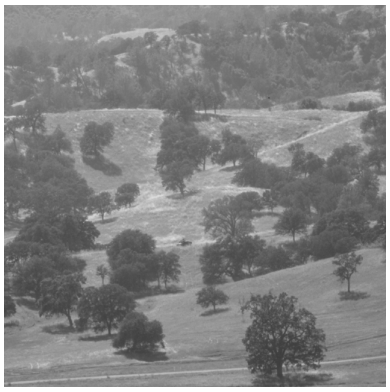

Figure 3.19: Original image

Figure 3.20: Dilation

3.3.3 Opening

Input: `py final.py e SE7.txt exp3.txt exp3_open`

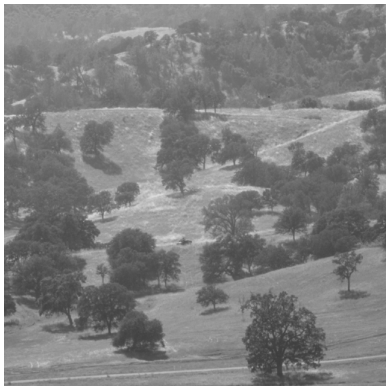

Figure 3.21: Original image

Figure 3.22: Opening

3.3.4 Closing

Input: `py final.py e SE7.txt exp3.txt exp3_close`

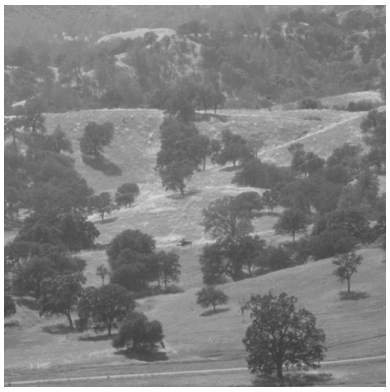

Figure 3.23: Original image

Figure 3.24: Closing

4 Work Distribution

Since it is a team work, the distribution of work was following:

Shahin Mammadov was responsible for implementing Erosion and finding minimum value.

Abumansur Sabyrrakhim was responsible for implementing Dilation and finding maximum value.

The experiments were done together.