Digital Image Processing

Shervin Halat

98131018

Homework 5

1.

a.

Resultant image ‘a’ is obtained by applying opening of SE ‘a’.

b.

Resultant image ‘b’ is obtained by applying closing of SE ‘b’.

c.

Resultant image ‘c’ is obtained by applying dilation of SE ‘c’.

d.

Resultant image ‘d’ is obtained by applying closing of SE ‘d’.

e.

0	0	0
1	1	1
0	0	0

f.

0	1	0
1	1	1
0	1	0

g.

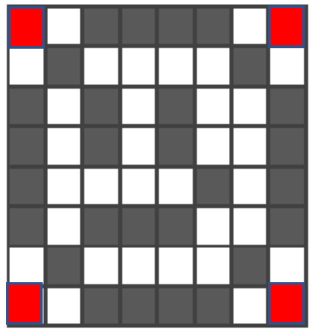
1	1	0
1	1	0
1	1	1

h.

0	1	0
1	1	1
0	1	0

i.

Applying hit and miss operation, with corresponding SEs for corner finding, results is the following:



Corresponding SEs are as follows:

x	1	x
0	1	1
0	0	x

x	0	0
1	1	0
x	1	x

x	1	x
1	1	0
x	0	0

0	0	x
0	1	1
x	1	x

j.

Based on defined SE, to detect boundaries, we are to compute following equation assuming B is the intended SE:

$$\boldsymbol{\delta X} = \boldsymbol{X} - (\boldsymbol{X} \ominus \boldsymbol{B})$$

k.

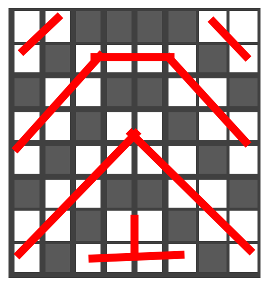
Based on specified initial point, by applying following equation iteratively:

$$X_k = (X_{k-1} \oplus B) \cap A \quad k = 1, 2, 3, \ldots$$

With the following SE:

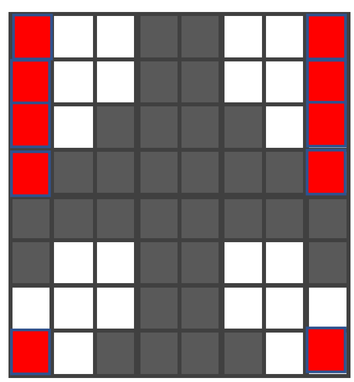
1	1	1
1	1	1
1	1	1

Disregarding which initial point is chosen, following regions will be found as connected components (all white pixels):



I.

Considering white pixels as one, and black pixels as zero, following skeleton will be obtained:



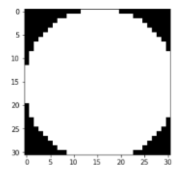
2.

???

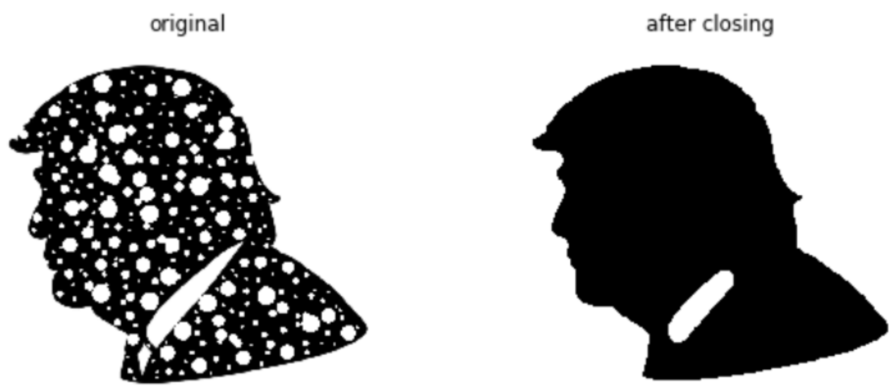
3.

a.

Applying closing (dilation then erosion) with following SE:



Results in:

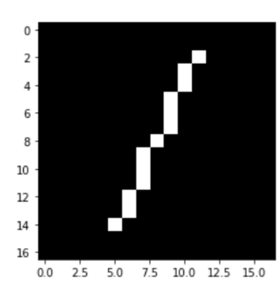


b.

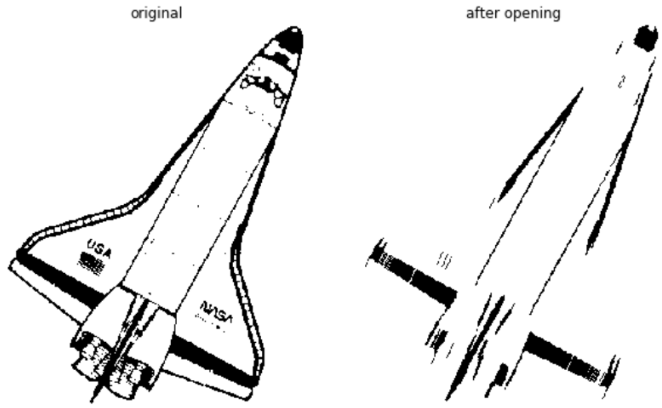
In order to extract bottom-left to top-right diagonal straight lines, following SE was applied in opening operation:

(Reversely, to eliminate diagonal lines it is enough to do closing operation with the same SE)

(following white line has an angle of 61 degree with horizontal line)



Which resulted in the following image:

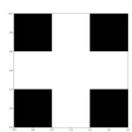


c.

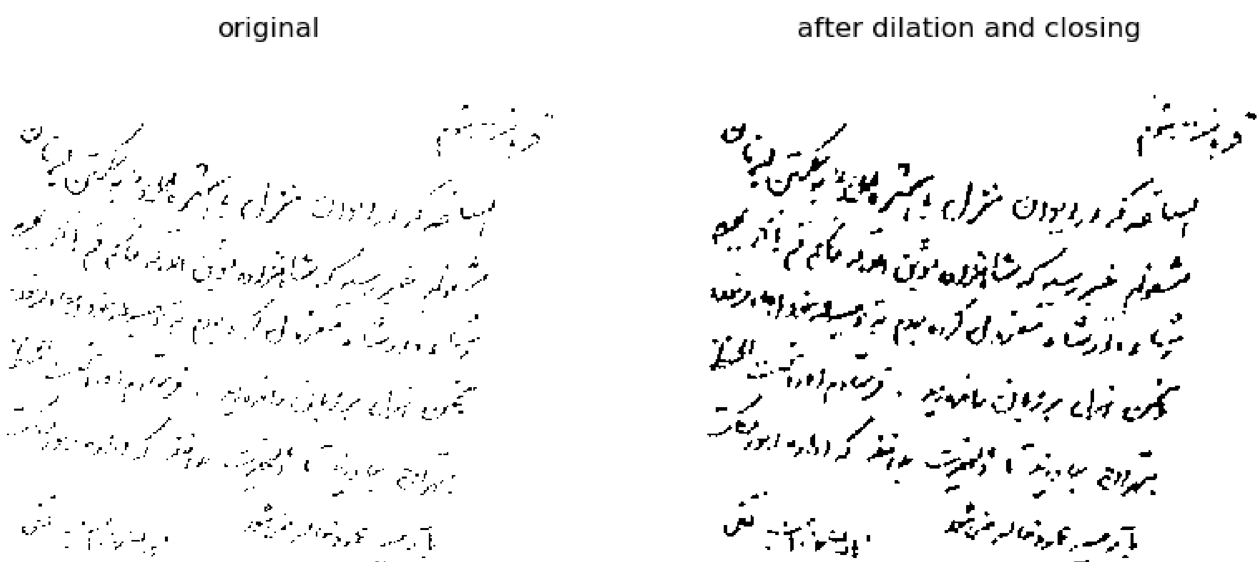
Unable to obtain reasonable results due to misfunction of implemented hit and miss function!!

d.

Applying dilation with following SE:



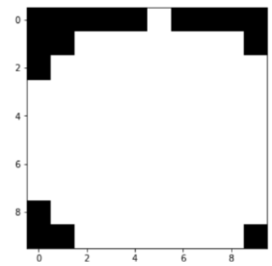
Then closing, the following result will be obtained:



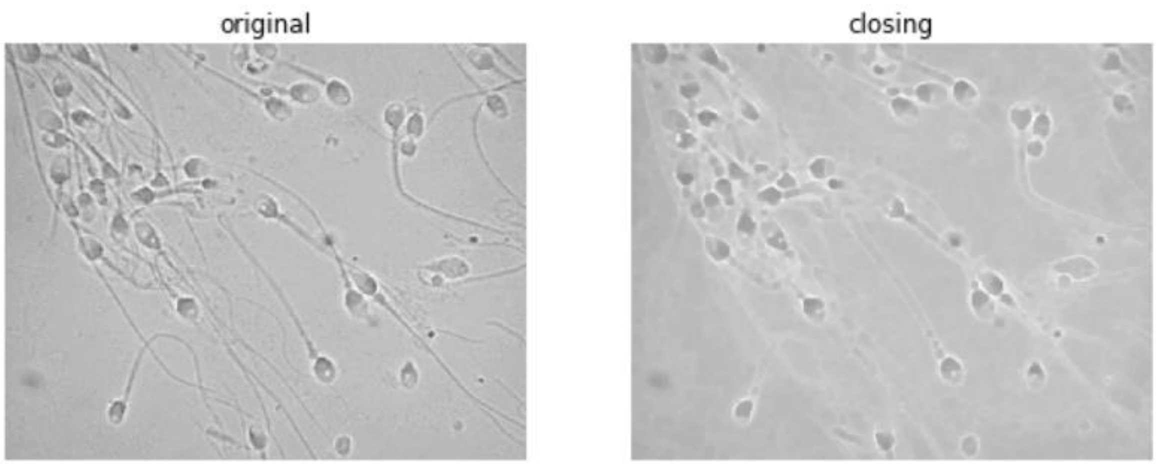
4.

a.

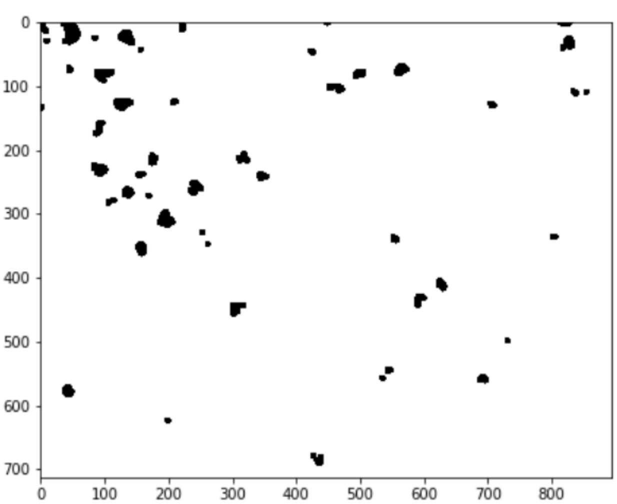
Firstly, by applying closing by the following SE:



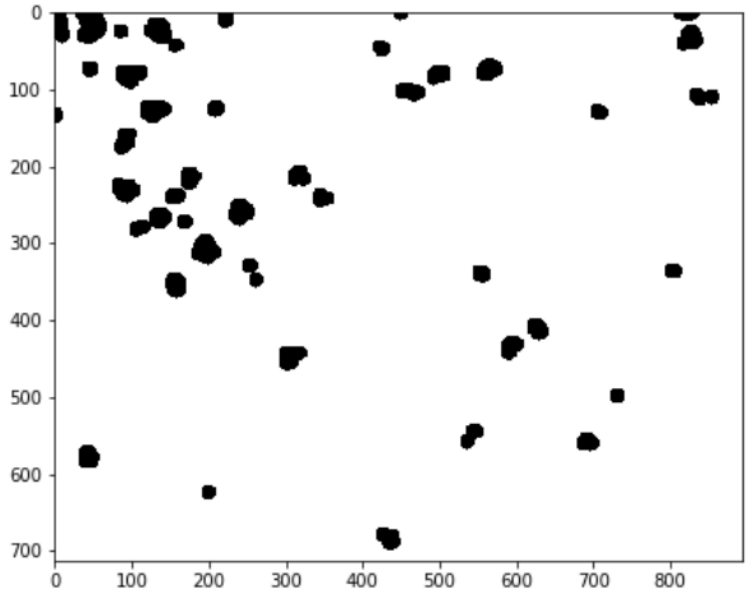
Following image will be obtained:



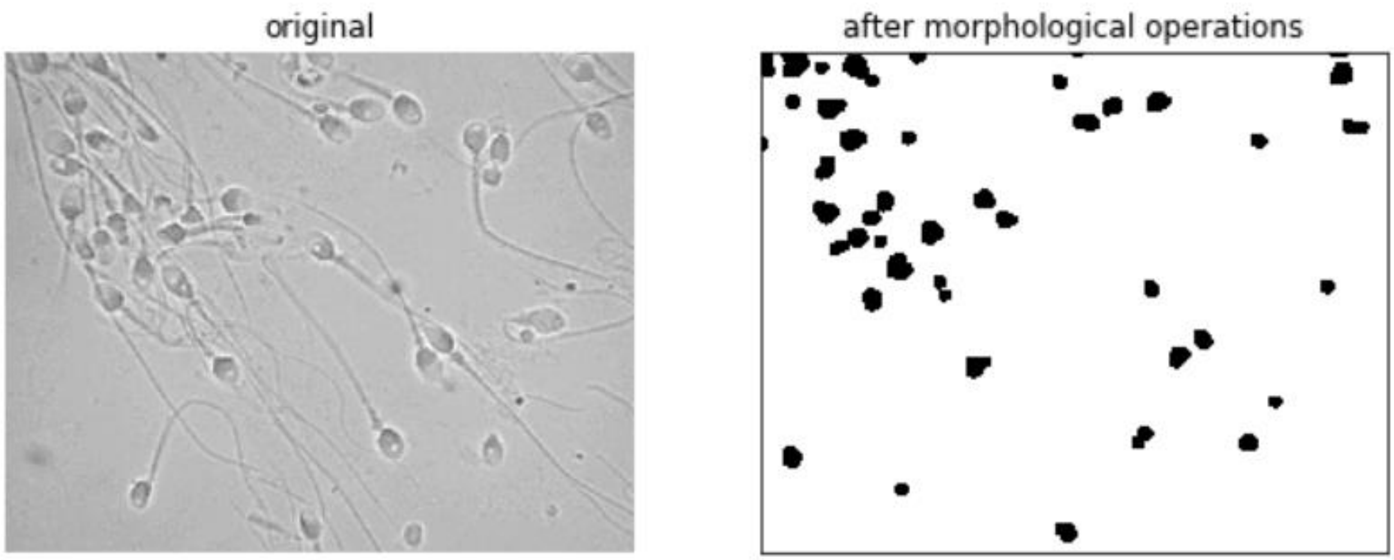
Then by thresholding (by value of 168) closed image will be transformed to the following:



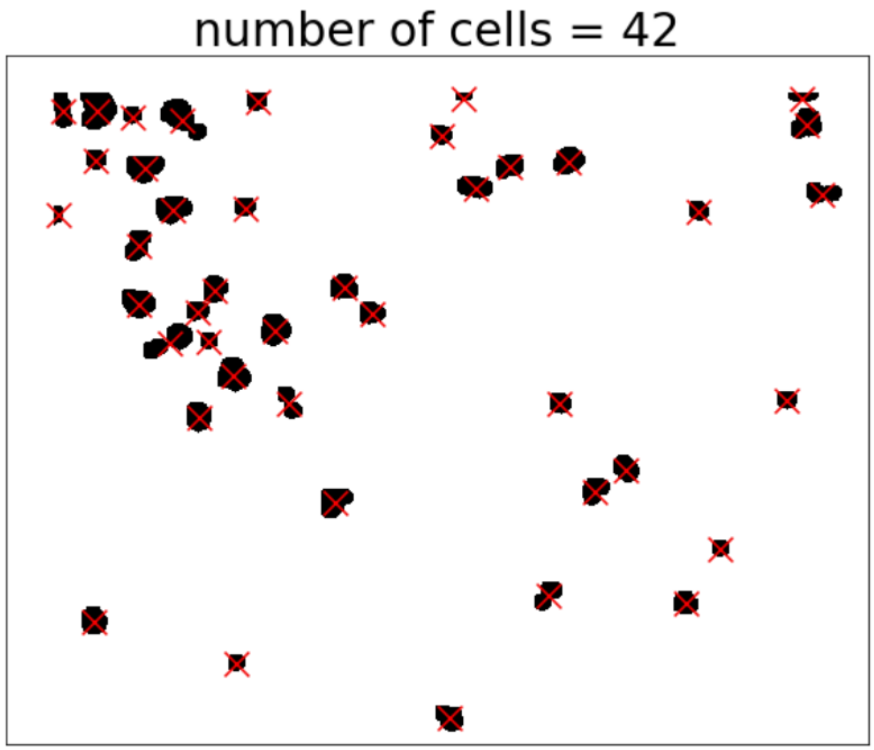
Next, by eroding the last image by the same SE as before to enlarge the black components we have:

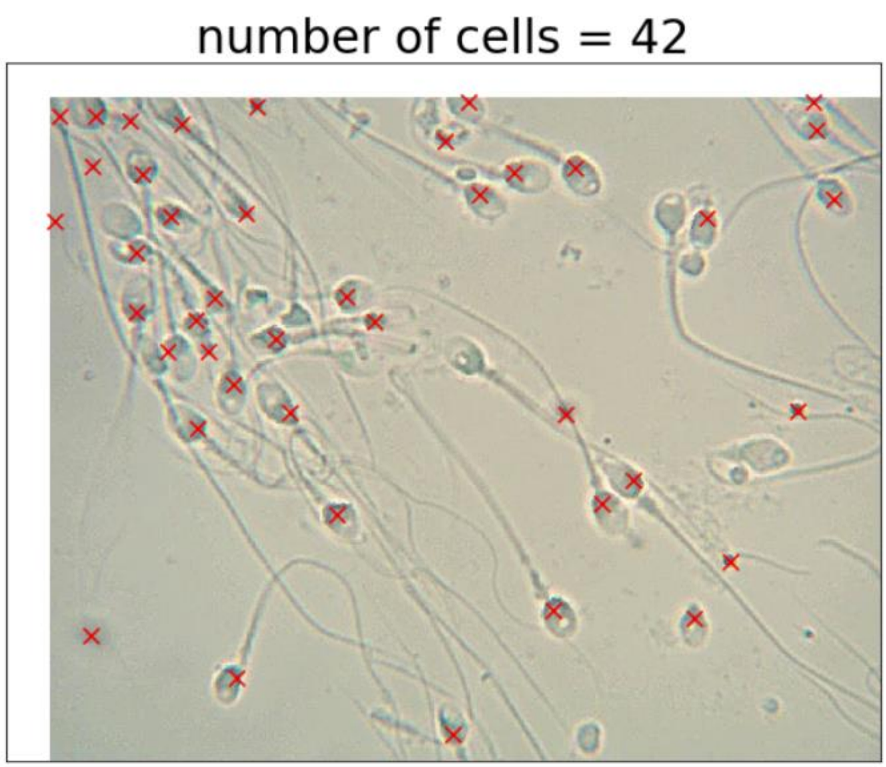


Comparing to the original image:



Lastly, in order to count number of black components, connected components will be counted which results to the following:

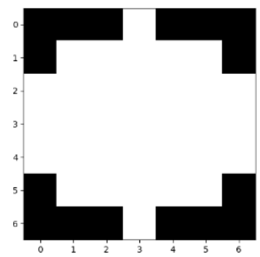


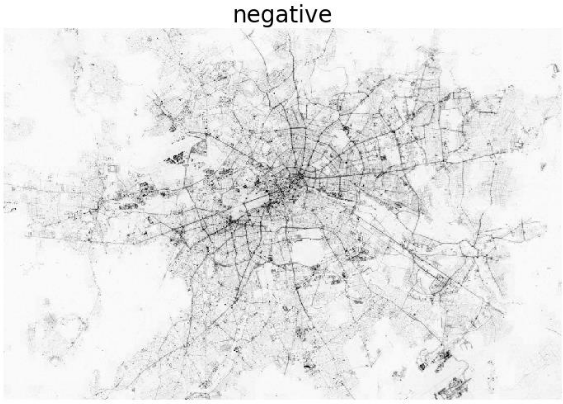


b.

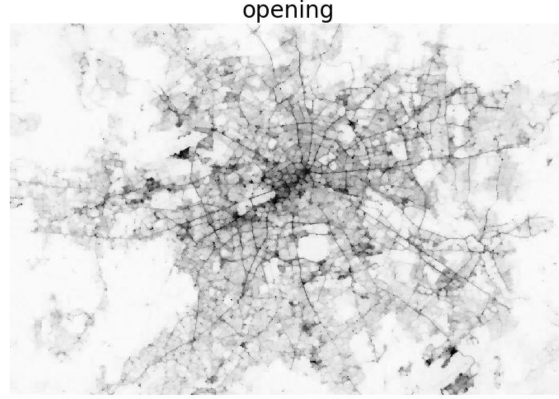
(**Following approach was applied on all three channels then results merged together)**

At first, open operation was applied on negative of each channel (kernel and obtained image):



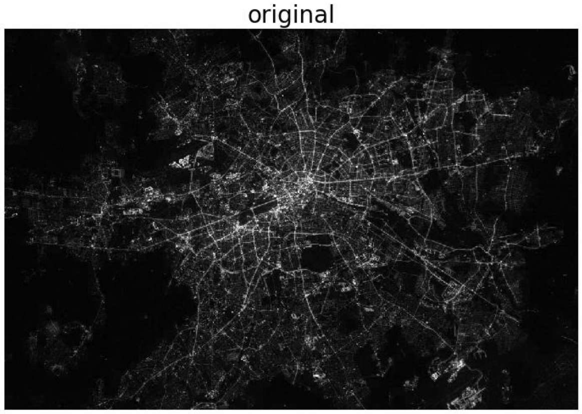


negative

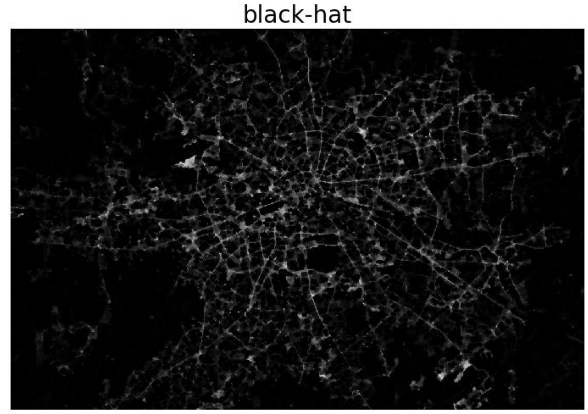


opening

Then, by applying black hat operation by similar kernel, we have:



original

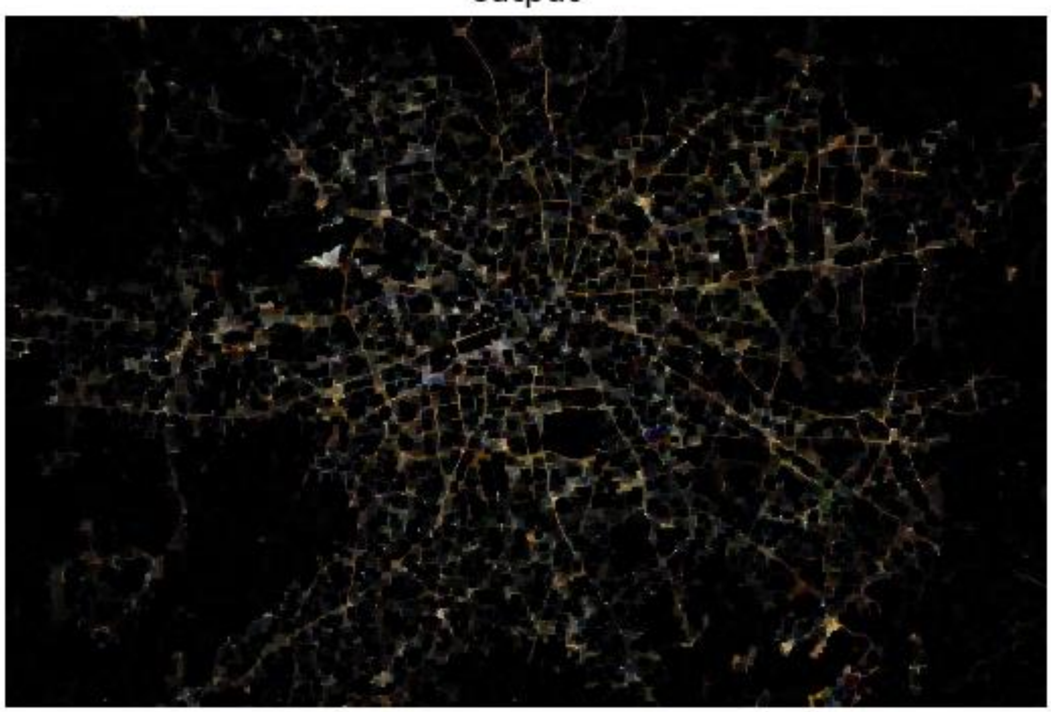


black-hat

Lastly, by merging three channels, following image will be obtained:

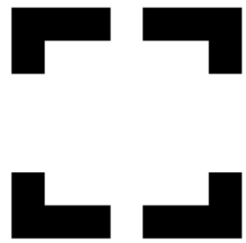
original

output

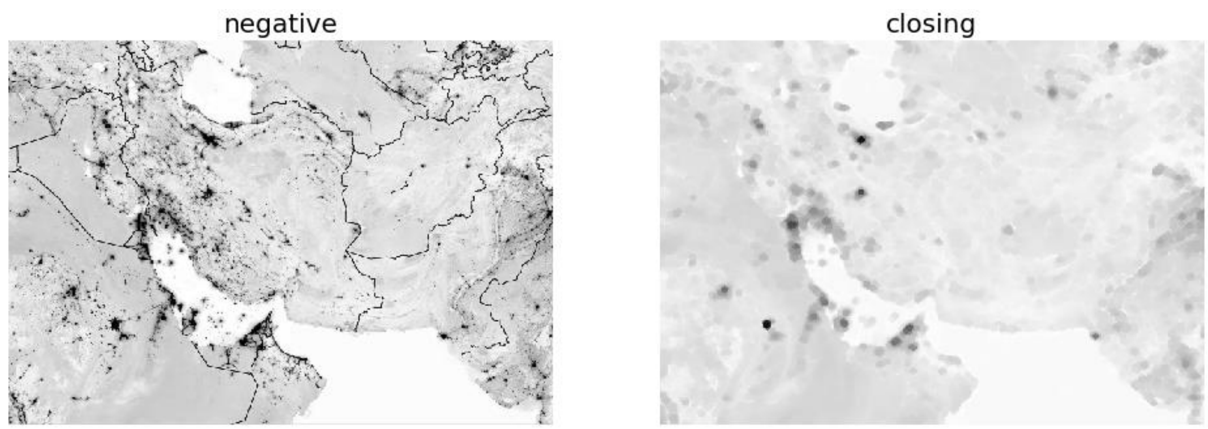


c.

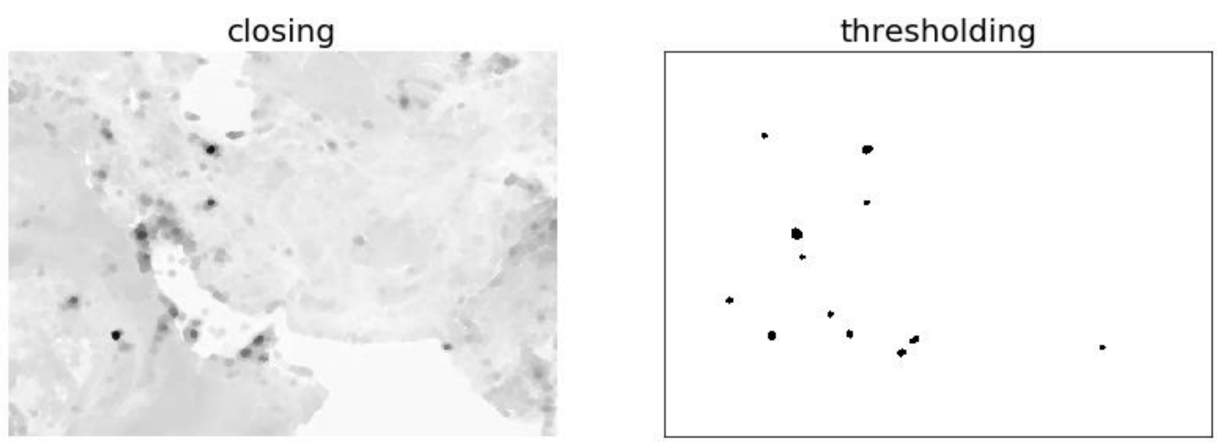
Applying following kernel:



on negative of grayscale image for closing operation on the negative of grayscale leads to the following:

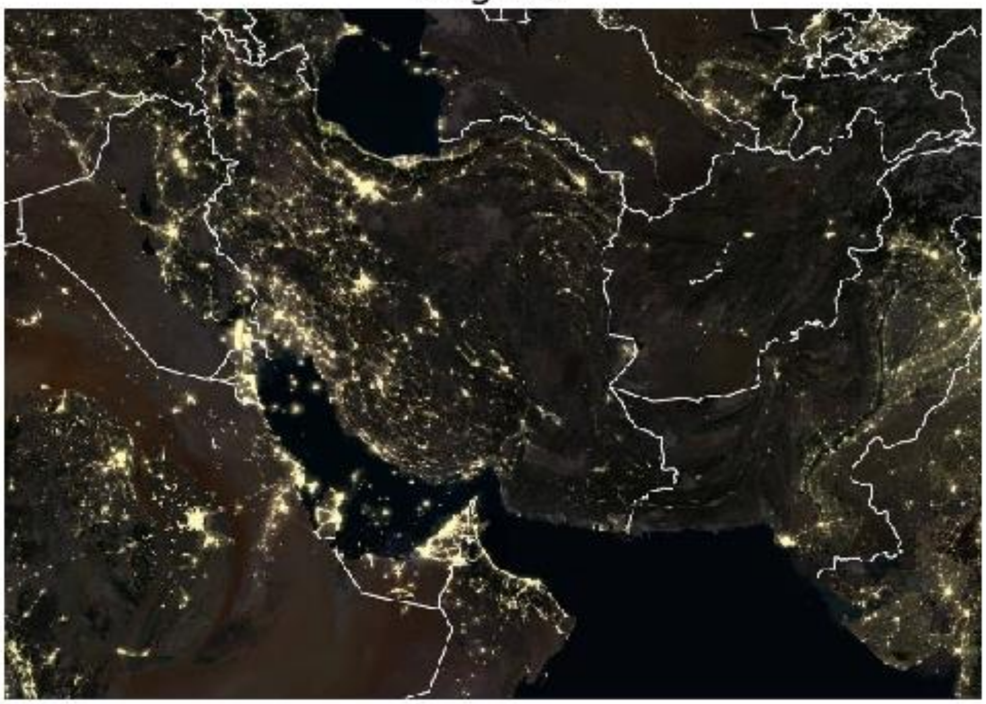


Then by thresholding (by value of 130) closed image will be transformed to the following:

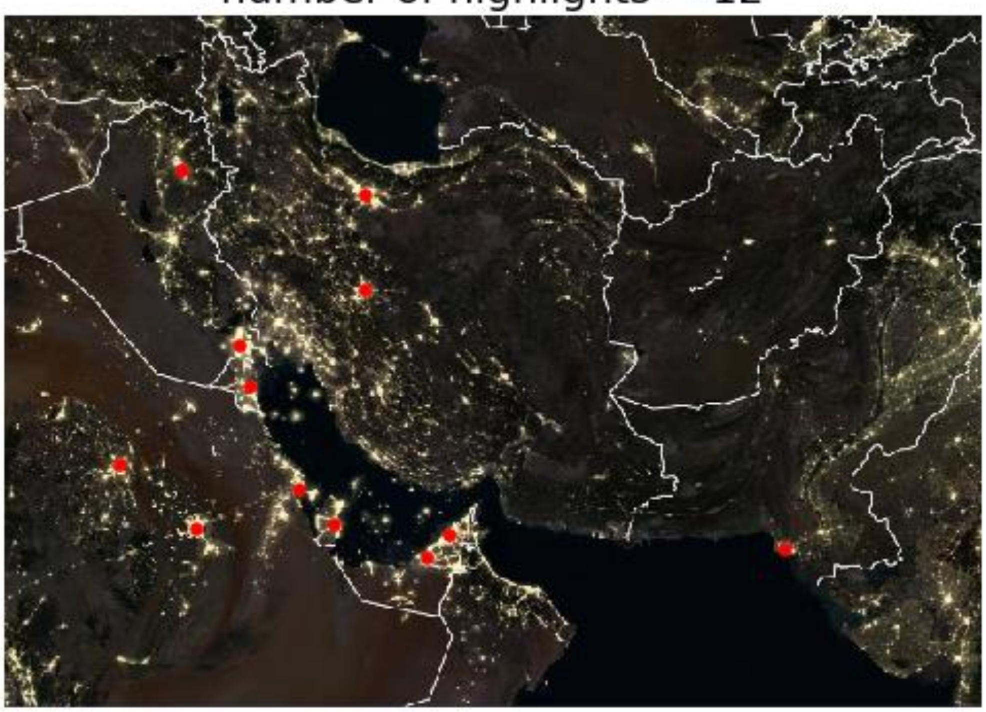


Lastly, by extracting connected components coordinates, following result will be obtained:

original



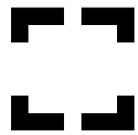
number of highlights = 12

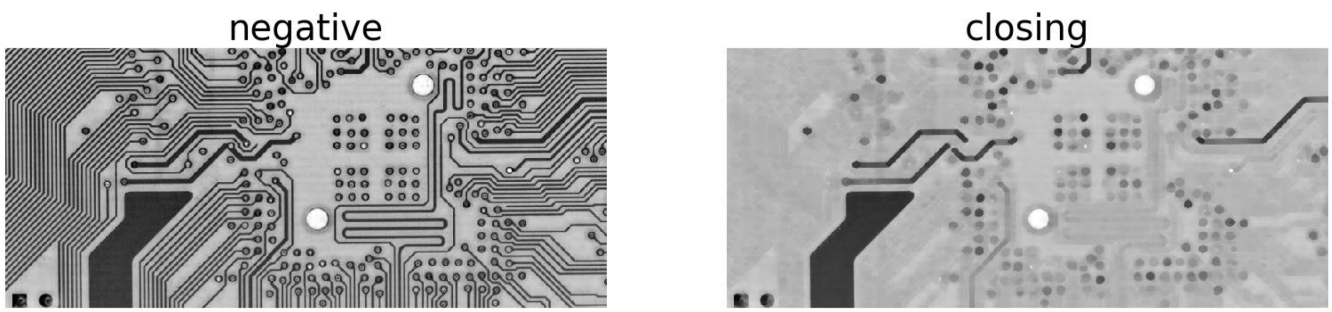


d.

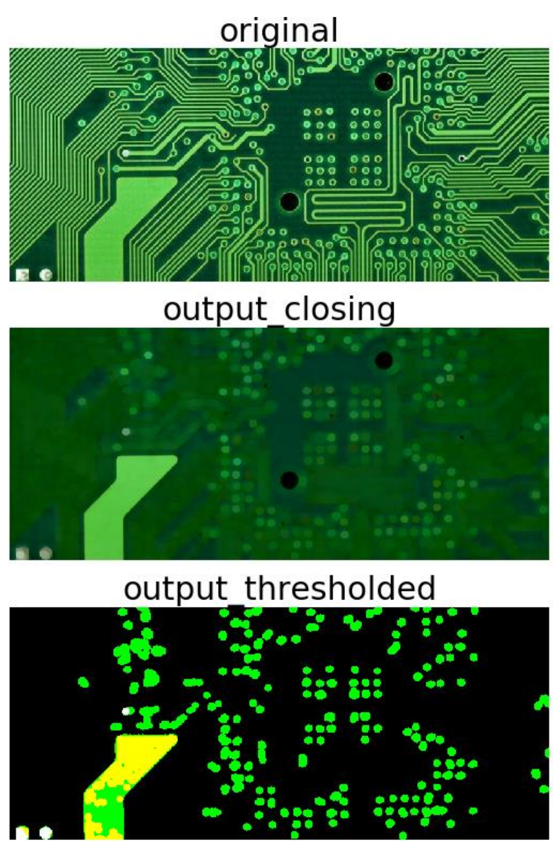
Removing lines (while keeping circles):

By applying closing operation on all channels (of the negative image!) with the following SE:



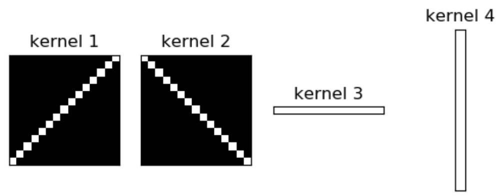


Then merging all channels, following image will be obtained:

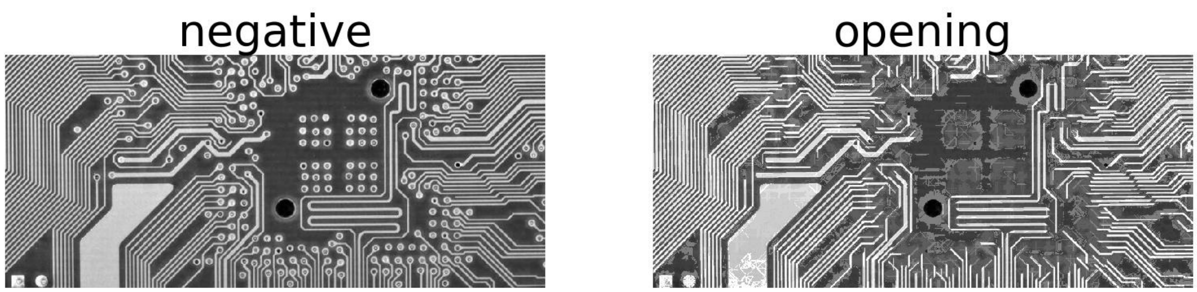


Removing circles (while keeping lines):

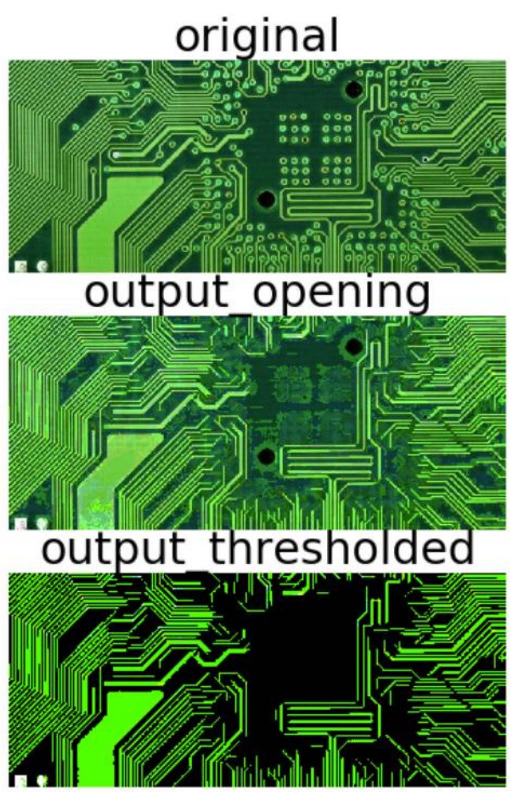
By applying opening operation on all channels (of the negative image!) with the following SEs:



Then merging all four results by ‘or’ logical operation, following result will be obtained:



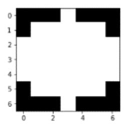
Then merging all channels, following image will be obtained:



5.

a.

Applying following SE in DYT morphological smoothing operation,

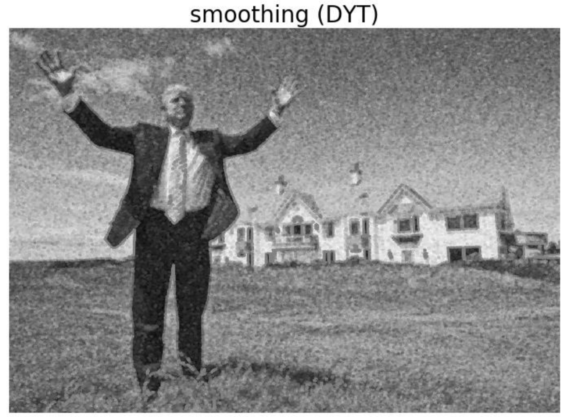


$$DYT(f) = \tfrac{1}{2}\left[\varepsilon_B(f) + \delta_B(f)\right]$$

Results in the following:

original

smoothing (DYT)



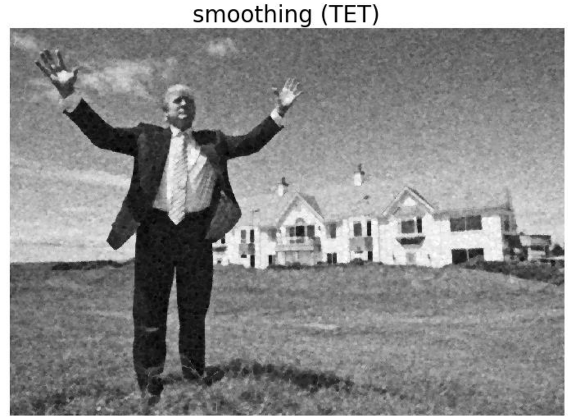
b.

Applying the same SE in TET morphological smoothing operation, results in the following:

$$TET(f) = \tfrac{1}{2}\left[\phi_B(f) + \gamma_B(f)\right]$$

original

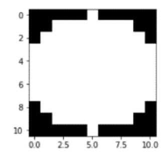
smoothing (TET)



Latter method (TET) has much better performance in noise reduction compared to first method (DYT).

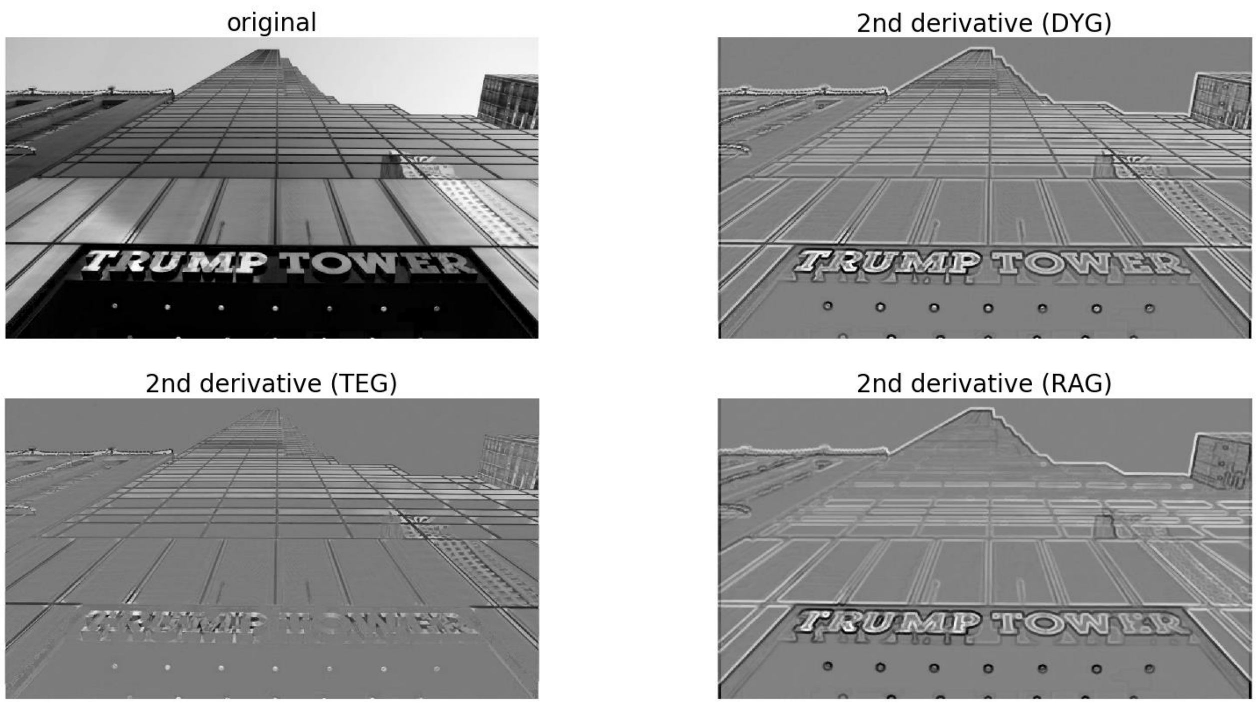
c & d & e.

Applying following SE in DYG, TEG and RAG morphological 2nd derivative operations,



$$dyg(f) = f - dyt(f)$$
$$teg(f) = f - tet(f)$$
$$rag(f) = dyg(f) - teg(f)$$

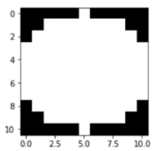
Following results will be obtained:



Each one of the mentioned operations are applied to distinguish two different kinds of edges of “ramp” and “texture” in an image. For example, DYG isolates all edges (both texture and ramp), RAG represents ramp edges and TEG represents non-ramp (texture) edges, whereby, each of them contributes in low, medium and high details of an image. Lastly, DYG has the most similar results to the linear Laplacian.

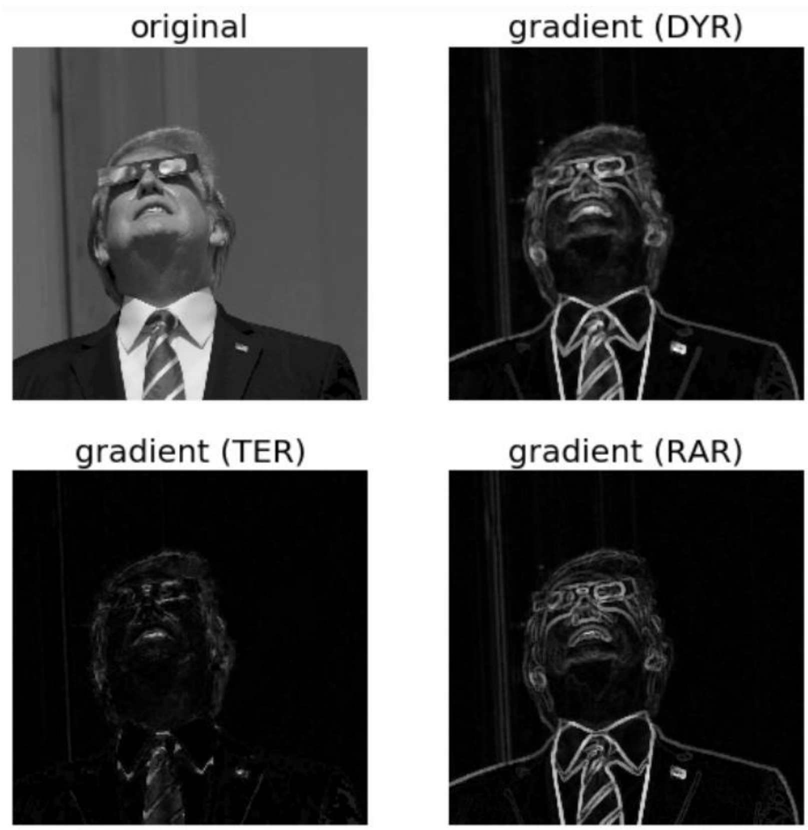
f & g & h.

Applying following SE in DYR, TER and RAR morphological gradient operations,



$$dyr(f) = dila(f) - eros(f)$$
$$ter(f) = close(f) - open(f)$$
$$rar(f) = dyr(f) - ter(f)$$

Following results will be obtained:



Very similar to the previous part, TER operation represents gradient of texture and noise edges, RAR represents only smooth edges and DYR represents both texture and smooth edges.

6.

???

7.

???

8.

a.

???

b.

The principle of this task is repeated conditional dilations until no more changes in result occurs. Initial dilation initiates from the given point of the connected component by, generally, applying a 3x3 square SE.

c.

Since, as opposed to closing, dilation changes objects' shapes and sizes and lastly degrades image significantly; specifically, some noisy holes may need more than one time of dilation which may lead to elimination of some details of the initial object. Therefore, closing (dilation followed by erosion) is suggested in such situations.

d.

It seems that there would be no difference in output between applying 3x3 twice and 5x5 once, but applying 3x3 SE twice may be faster because it needs 2x3x3 (18) comparisons for each pixel of initial image while 5x5 SE needs 5x5 (25) comparisons for each pixel of initial image.

e.

Generally, position of the origin does not affect the output of opening or closing operations but it affects the output of both erosion and dilation. This is mainly because of the fact that the change in origin position results in exactly the same displacement in both erosion and dilation outputs which will be eliminated after subtracting these two operations (when obtaining opening or closing).

f.

Generally, all basic morphological operations may be performed by use of hit and miss operation. For example, for erosion operation, “one” values of intended SE may be left unchanged while converting “zero” values to “don’t care” values. Then, result of hit and miss would be erosion.

g.

To perform dilation with NOT and hit-and-miss operations, we can first apply NOT operation on initial image then by applying hit-and-miss in such a way mentioned in part ‘f’ to erode the image with the intended SE and again applying NOT operation, it seems the desired dilation operation will be performed.

h.

Since determining convex hull using thickening requires 8 effective SEs (2 main SEs with all of their rotations by 90 degrees) to be used in each iteration until convergence which is a time-consuming approach.