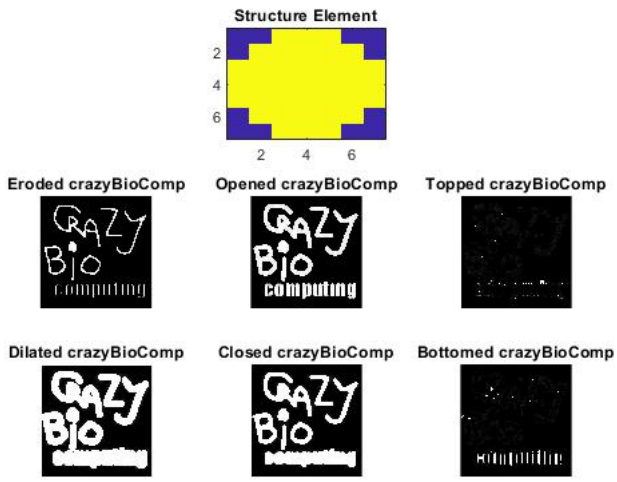
ש.ב רטוב 2 ענ"ת

29 במאי 2019

סהר סלע כרמל 305554453
אלכס שנדר 328626114

1. סינון מרחבי ופעולות מורפולוגיות

א.



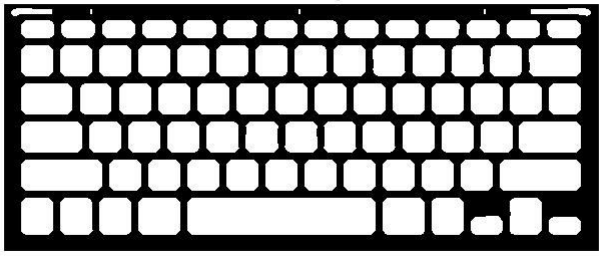
ב.

ג+ד.

ניתן לראות כי המבנים שנשארים בתמונה הם המבנים שהם קווים אופקיים ואנכיים. בהתאם לפילטרים שהעברנו על התמונה. כמו שניתן לראות בתמונת הסכום אנחנו מקבלים את המבנים האנכיים והאופקיים כסכום של שני הפילטרים הקודמים.

ה.

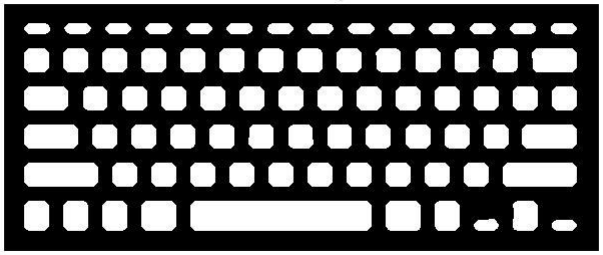
"NOT" filtered keyboard



בחרנו במסנן חציון בכדי להוריד את נקודות האי רציפות בתמונה, בעוד שמסנן מיצוע היה נותן לנו רמות אפור לא רצויות.

ו.

Eroded filtered keyboard



ז.

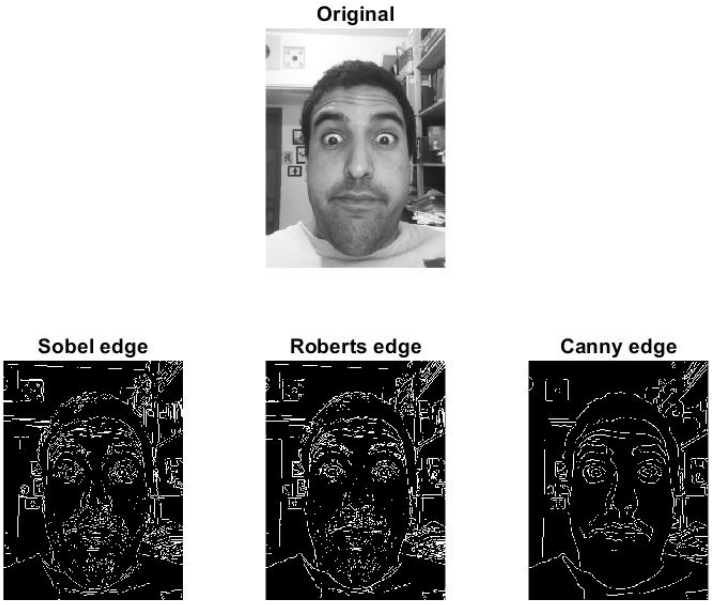
Sharpened intersectioned keyboard

ח.

תחילה בודדנו את המקשים על ידי שימוש בפילטרים של קווים הוריזונטליים ואנכיים וסכמנו את שתי התמונות. לאחר מכן ניקינו רעשים על ידי מסנן חציון, אז בודדנו את מסגרות המקשים לאחר מכן השתמשנו בבידוד זה בכדי להתמקד על המקשים ולהתנתק מסביבתם, לבסוך בודדנו את המקשים מהמקלדת.

2. גילוי וחידוד שפות בתמונה רועשת

ב.



האופרטור העדיף מבחינתנו הוא אופרטור $Canny$ שכן השפות שהוא תופס הם העיקריות והם גם הכי ברורות.

ג.

הערך שנותן את התוצאה האופטמילית ביותר מבחינתנו הוא $a = 2$ שכן עבור ערך זה השפות אכן מתחדדות ועבור ערכים גדולים יותר מתקבלת תופעה לא רצויה של "הילה" סביב העצמים. הפרמטר a הוא סקלר שמכפיל את חוזר אופרטור הנגזרת. מסנן זה מדגיש שפות מכיוון שהוא מבצע הפרשים ממוצעים סביב כל פיקסל כמו אופרטור הנגזרת שלמדנו.

ד.

a = 0.3

a = 0.5

a = 0.7

רעש מלח פלפל הוא רעש בעל תדירות גבוהה, ומכיוון שפילטר זה מחדד שפות הוא גם מגביר את סוג הרעש הזה.

ה.

ניתן לעביר תחילה את התמונה במסנן חציון בכדי להפחית את כמות הרעש, ולאחר מכן ולהפעיל את אופרטור הגזירה.

a = 0.3

a = 0.5

a = 0.7

3. לכידת תמונה

א.

Up to down

ב.

Left to right

ג.

Right to left

מכיוון שבסעיף ב. סרקנו משמאל לימין, בעצם סרקנו בכיוון תנועת הסוס ולכן התמונה נמרחה כאשר היא ניסתה "לתפוס" את הסוס. בעוד שעבור סעיף ג כיוון תפיסת התמונה היה בניגוד לתנועת הסוס ולכן הסוס התכווץ בחצי שכן קו תפיסת התמונה חלף עליו במהירות.

I. Question 4.

A.

included in the code

B.

included in the code.

C.

The algorithm did not succeed to successfully identify and match the template. The algorithm failed in the cases where the occlusion was present, or the object was rotated by a certain value.
Example for the successful detection:

Not successful detections:

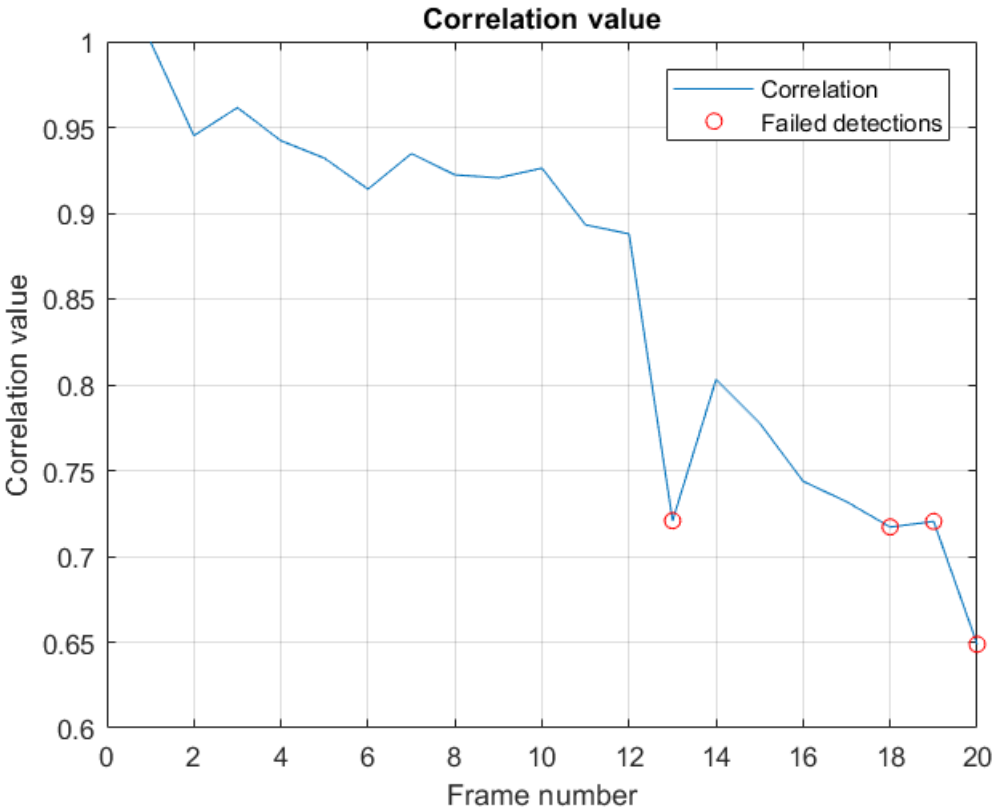

5000


610
5000


610
5000


2

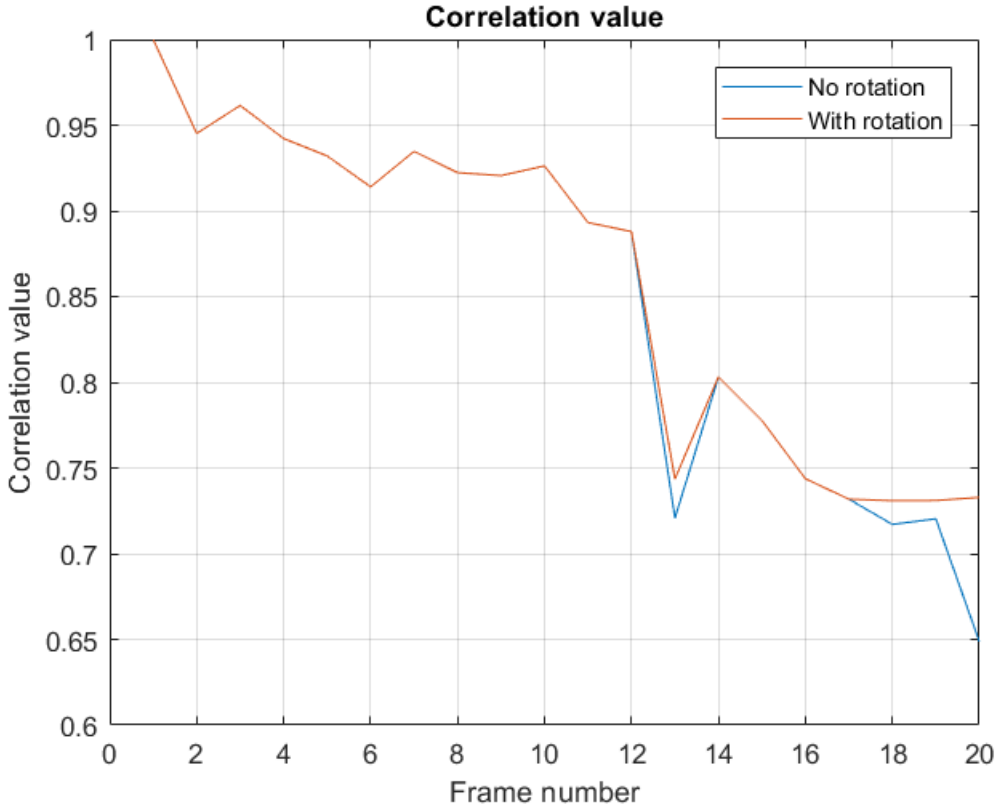
The following figure demonstrates the Maximum correlation value that was found for each of the correlations. Not surprisingly, the cases where the correlation is low, the prediction was not correct



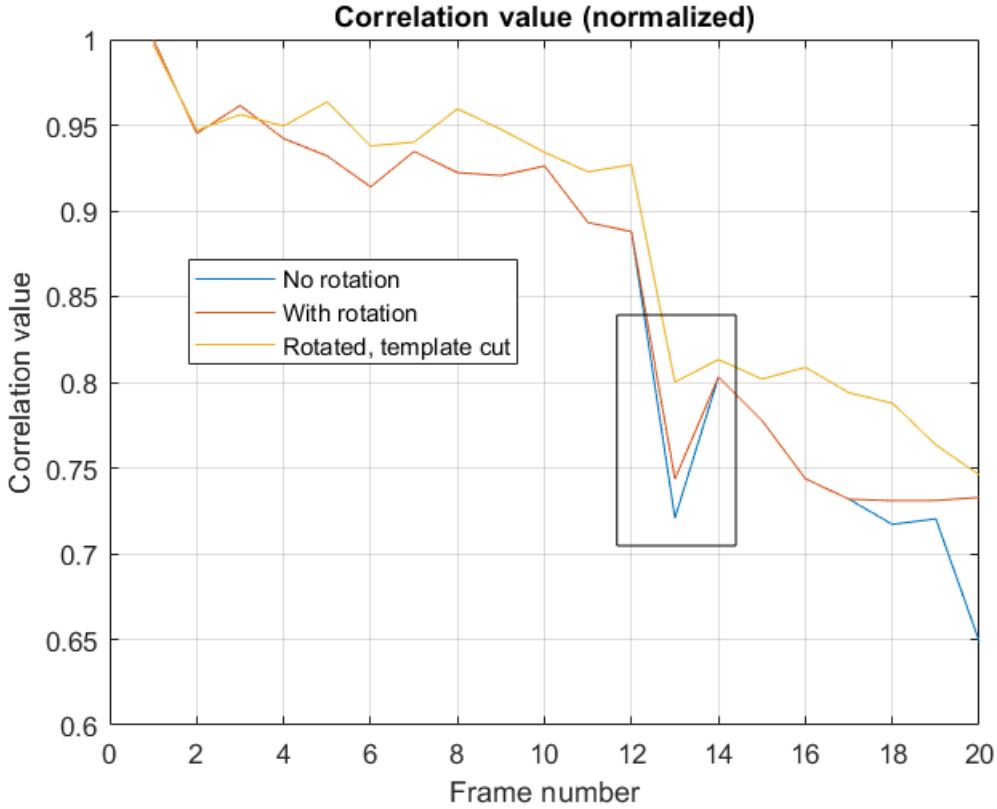
D.

There were 2 fixes to the algorithm, to deal with both of the reasons for failure.

1. First, the loop was created which goes over different angles from $-\pi/2$ to $\pi/2$. This has solved the problem for the object at an angle. But still the problem of occlusion exists The following graph shows the improvement in correlation for the last 3 frames where the detection was wrong.



2. To solve the problem of the occlusion, we have divided the template into 2 parts. The left part and the right part. The initial plan was to calculate the correlation for both of them, and if the found coordinates were not close enough - we take the coordinates of the template with the highest correlation value. (This case means than the second template must have been occluded and gave a low correlation value with some other object). But the reality did not go as planned, and this 'fusion' model did not succeed. What did succeed is just taking 1 part of the template, the right part (which is not occluded on the frame 13), and then all the cases detected the bird successfully. Below is the image of the Correlation values for 3 algorithms. We can see than on Frame 13, where the bird has been occluded, the correlation value was raised. (But the correlation values on the graph are normalized, so we cannot compare different algorithm correlations values).



The following images show the frames, which have previously failed, being successful again.


610
5000

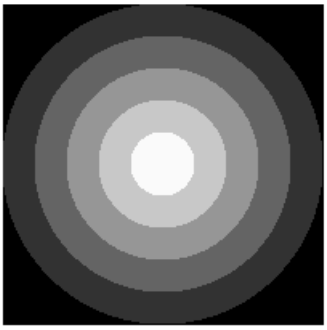

610
5000


6
5000

II. Question 5.

A.

The circle has been built. The fuction which creates a noisy image was created as well. The following images show the circle built.

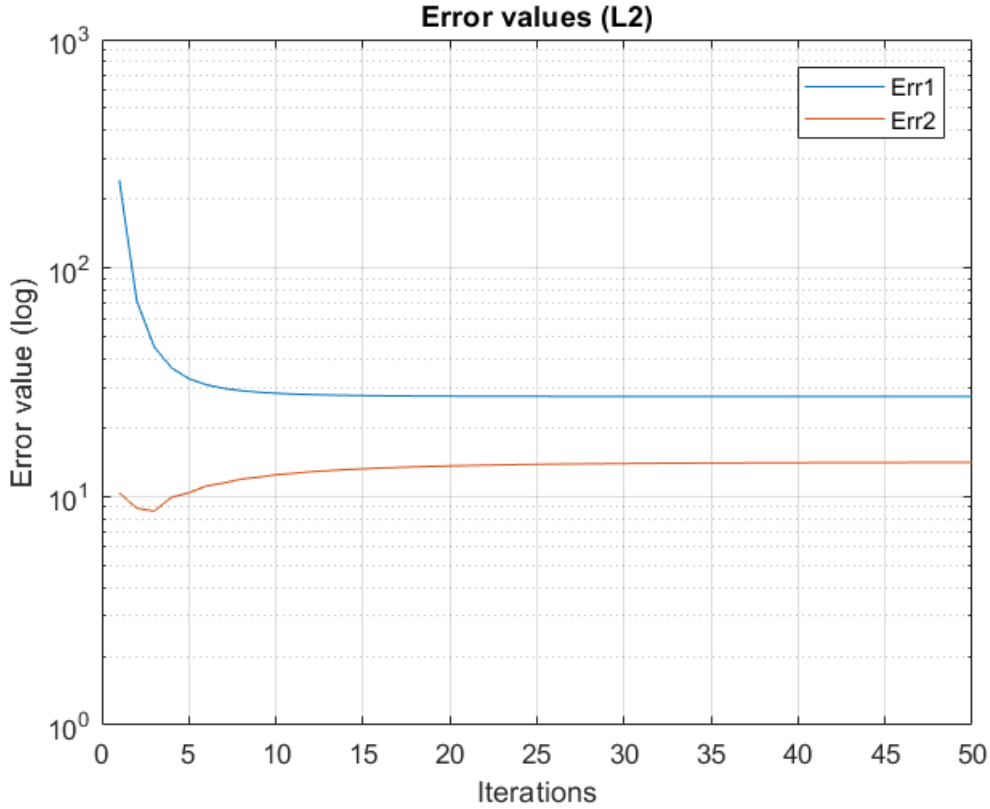


B.

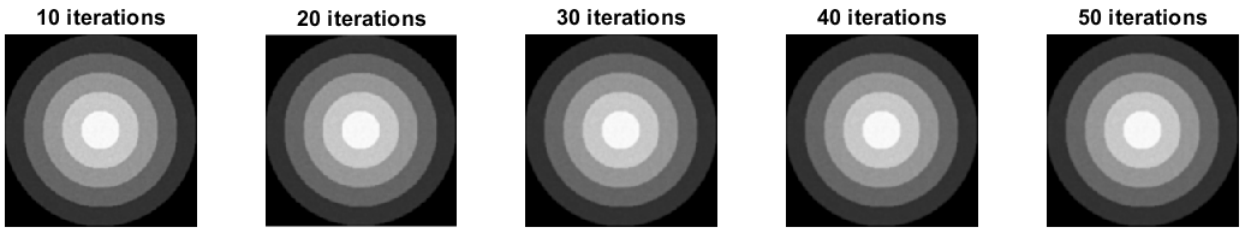
The function exists in the MATLAB code.

C.

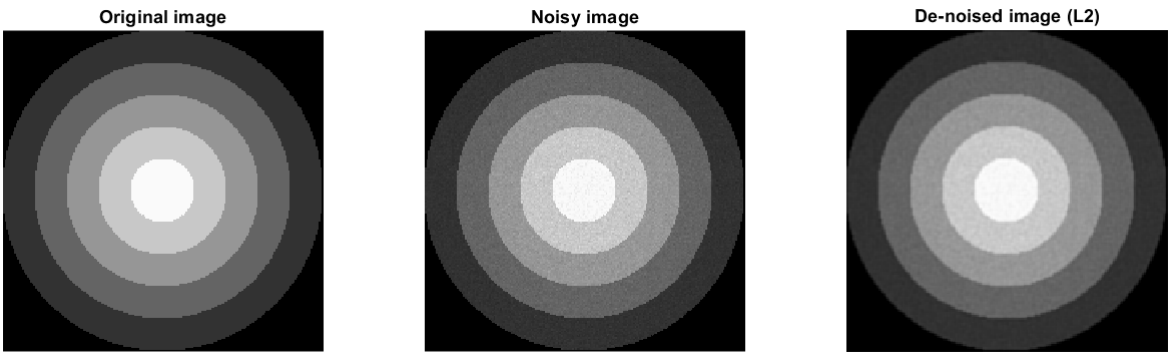
By operating the L2 denoise algorithm with the prescribed parameters we obtain the following Err_1 and Err_2 results:



By printing the image every 10 iterations we obtain:



It is definitely hard to distinguish the image with the least noise from those images. So we take the minimum value from the Err_2 graph, which shows the difference of the de-noised image from the original image. We obtain this value at the iteration=3. Displaying the best image from the best iteration:

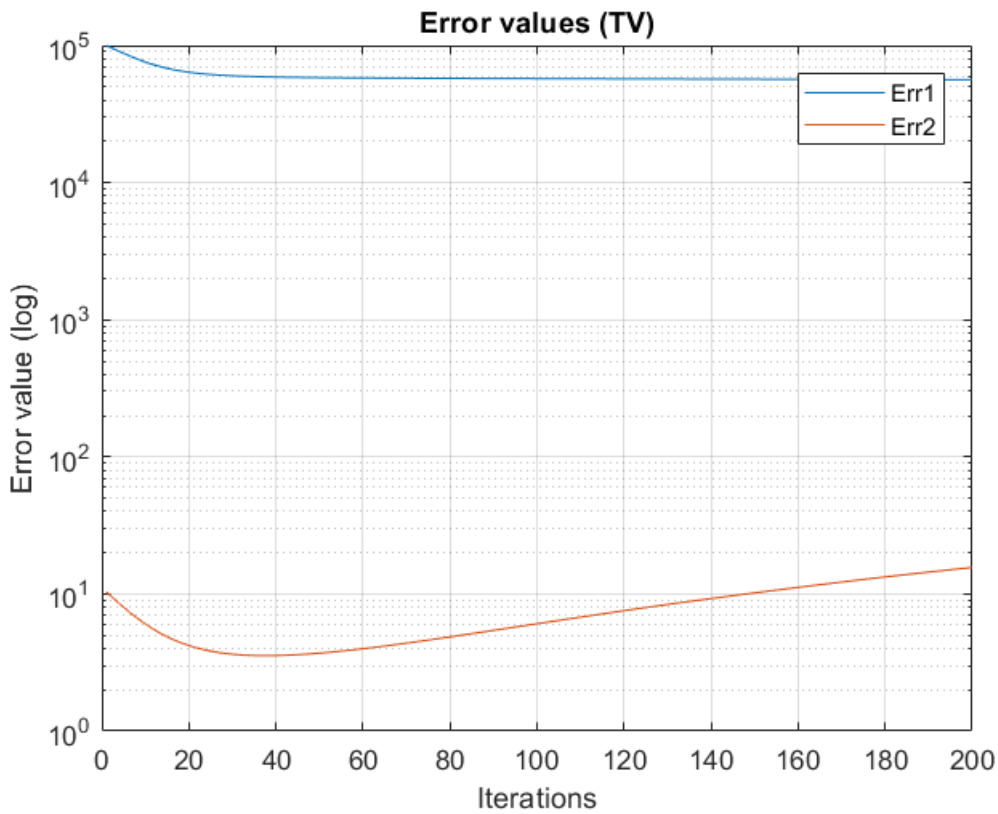


We can observe than some denoising did take place indeed, the image became smoother. Also the edges got smoother, which should be fixed with the TV prior, which we see in

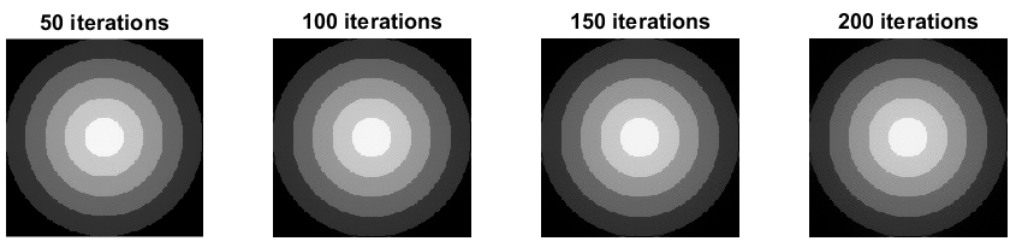
the next subsection.

D.

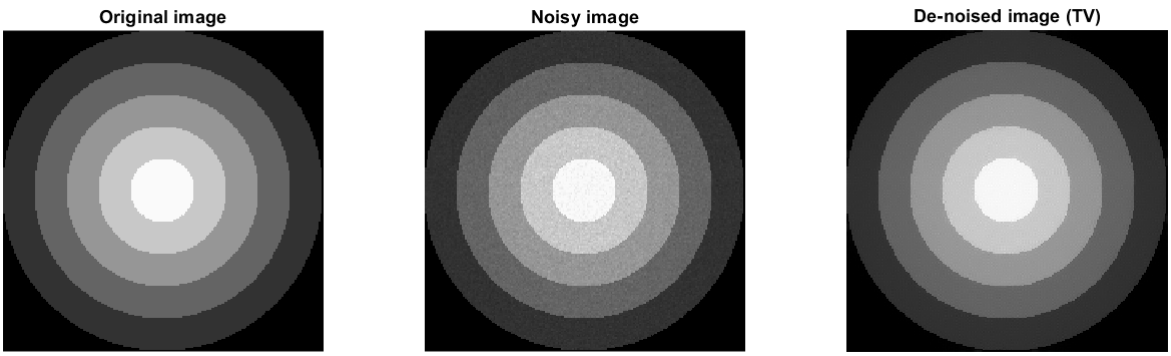
The parameters were set as prescribed, the results obtained are the following. For the error values:



By printing the image every 50 iterations we obtain:



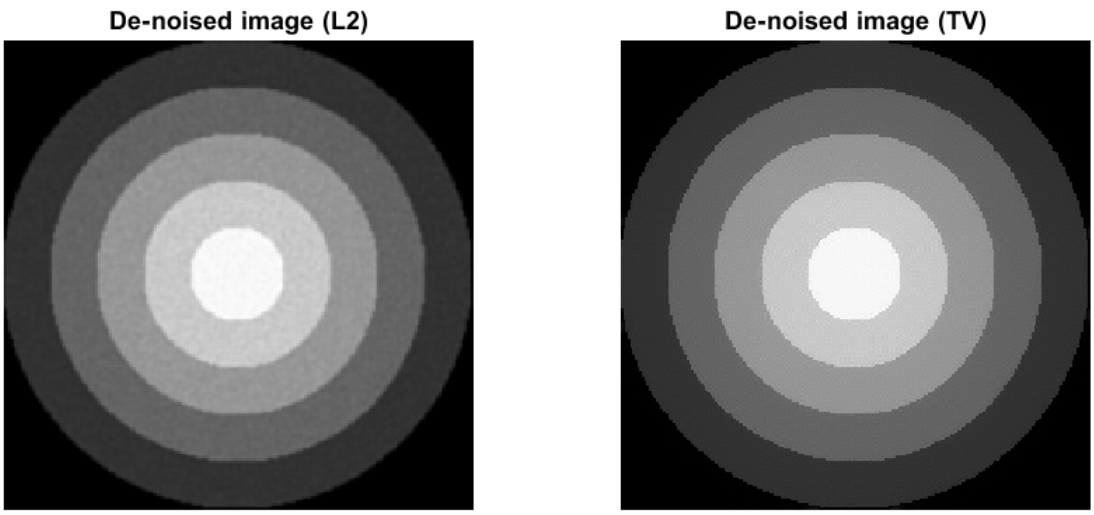
Similarly, it is hard to choose the best image by looking at them. So we choose at the iteration with the smallest Err_2 value. This happens approximately at iteration 30. Displaying this best result:



We can see the improvement, also by looking at the minimal Err_2 value for both methods. More on this in next subsection.

E.

Let's first compare both denoises:



Several things can be said:

1. The denoise process with the TV prior performs better than with the L2 prior
2. TV tends to preserve the edges!
3. L2 tends to blur both noise, and edges, without distinguising between the two

F.

The image taken is presented below. The same noise was added to it. Below are presented 3 images:

1. The original image

2. The de-noise by L2 prior

3. The de-noise by TV prior

We can indeed see that the de-noise using the TV prior is much more successful and gives better results