Autofocus Methods in spacial and DCT domains*

Authors:
Issam Manai
Viktor Truderung

Hochschule Kaiserslautern

Fraunhofer, Institut für Techno- und Wirtschaftsmathematik, Kaiserslautern

Abstract:

This work is a part of the InStent project, where the surface of medical devices is captured with an optical inspection system. Due to the very complex surface, the movement during the acquisition and the extremely shallow depth of field, focusing is a challenge. In addition, due to the cylindrical shape of the stent, the neighbor regions are less focused, although a selected region is well focused. Therefore, it cannot be solved by a global approach and must be automatically adjusted depending on the selected region of interest.

This paper is a summary of all the steps followed, the methods applied, the challenges faced and the results developed. The conclusion that we came out with, is specific to the nature and the demands of this project and the hardware we worked with (Cameras, lights ..).

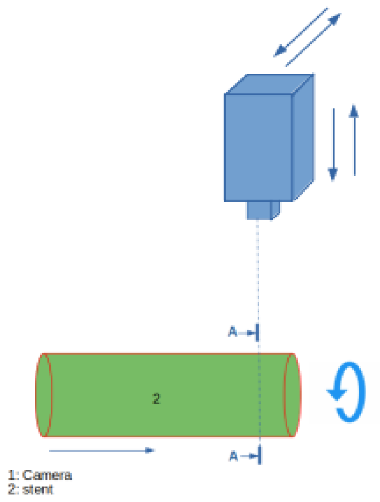
Introduction:

The goal is to analyze a photo of the stent and determine the region with an acceptable focusing rate (The acceptance will be studied and defined). Then based on the region of interest (The surface on the Stent defined by the user), decide whether to move the camera up, down, back or forward in order to centralize the focus region with the region of interest.

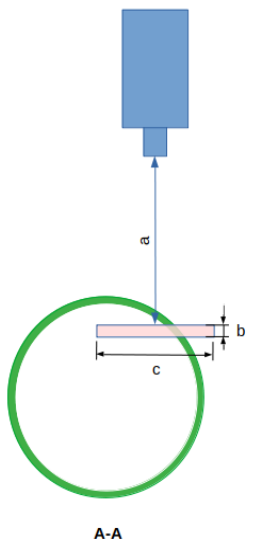
A simplified model of the image acquisition unit is presented in (1). The camera is able to move up-down and back-forward, and the stent is rotating- while moving sideways- with a predefined speed which depends on the diameter of the stent and the speed at which the camera takes photos.

In the section A-A in (2), the camera is represented with the blue rectangle and section of the stent with the green circle despite the stent has not a perfect cylindrical shape . The section shows the camera parameters a, b and c, which are the most important factors in this part of the project. a represents the focus distance, b the depth of field and c the comp size.

By moving the camera up-down and back-forward, we can adjust the position of depth of field to match the pre-selected surface of the stent that will be later analyzed.

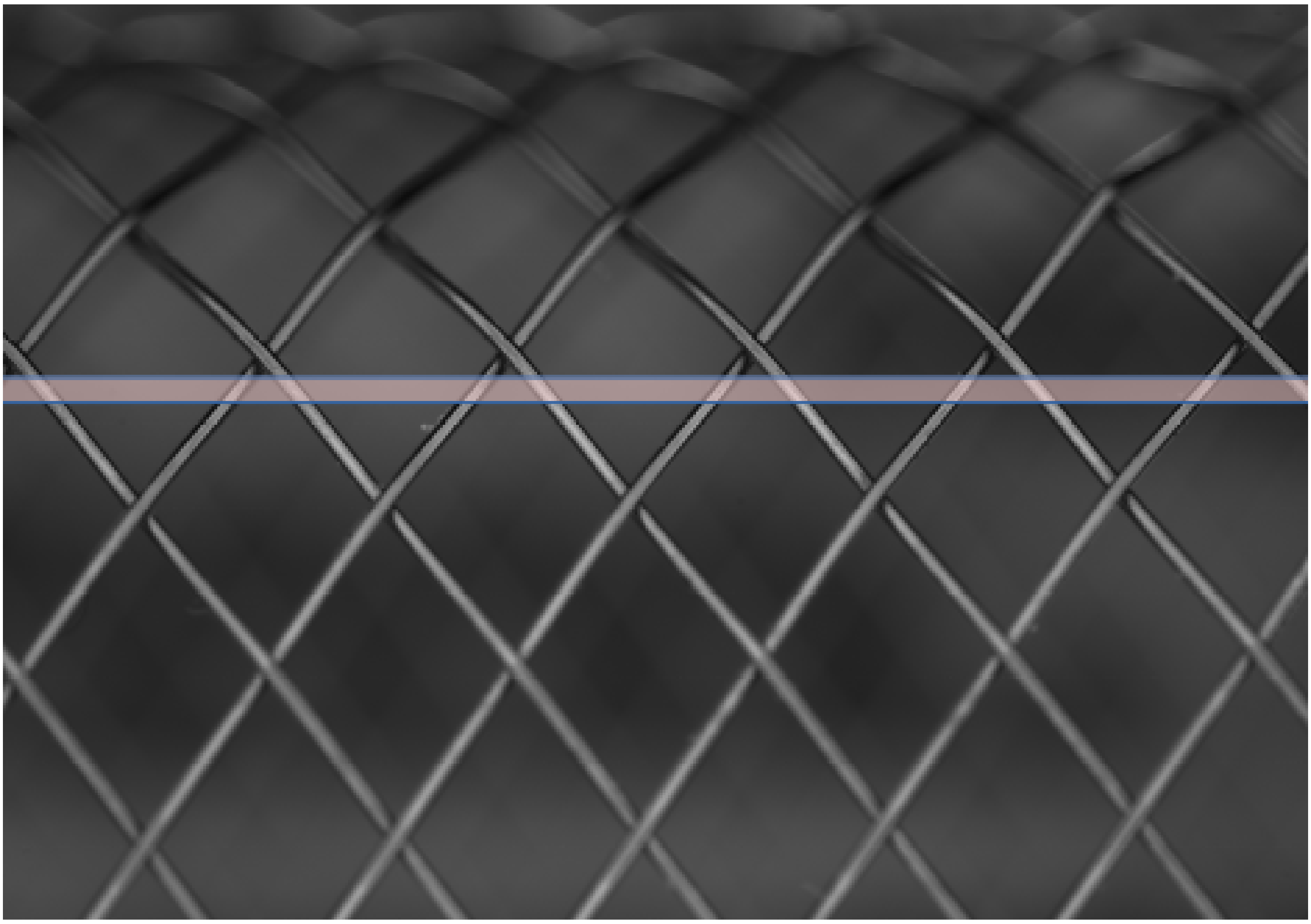


(1)



(2)

An example photo of the stent (3) shows that the focus loss takes place just in vertical direction. Moreover, we expect horizontally the same focus quality. Variations happen due to non-perfect diameter and the uncertainty during the movement. Later we will divide the photo into segments- like the pink box in (3)- with a length equal to photo length and a width that will be later defined.

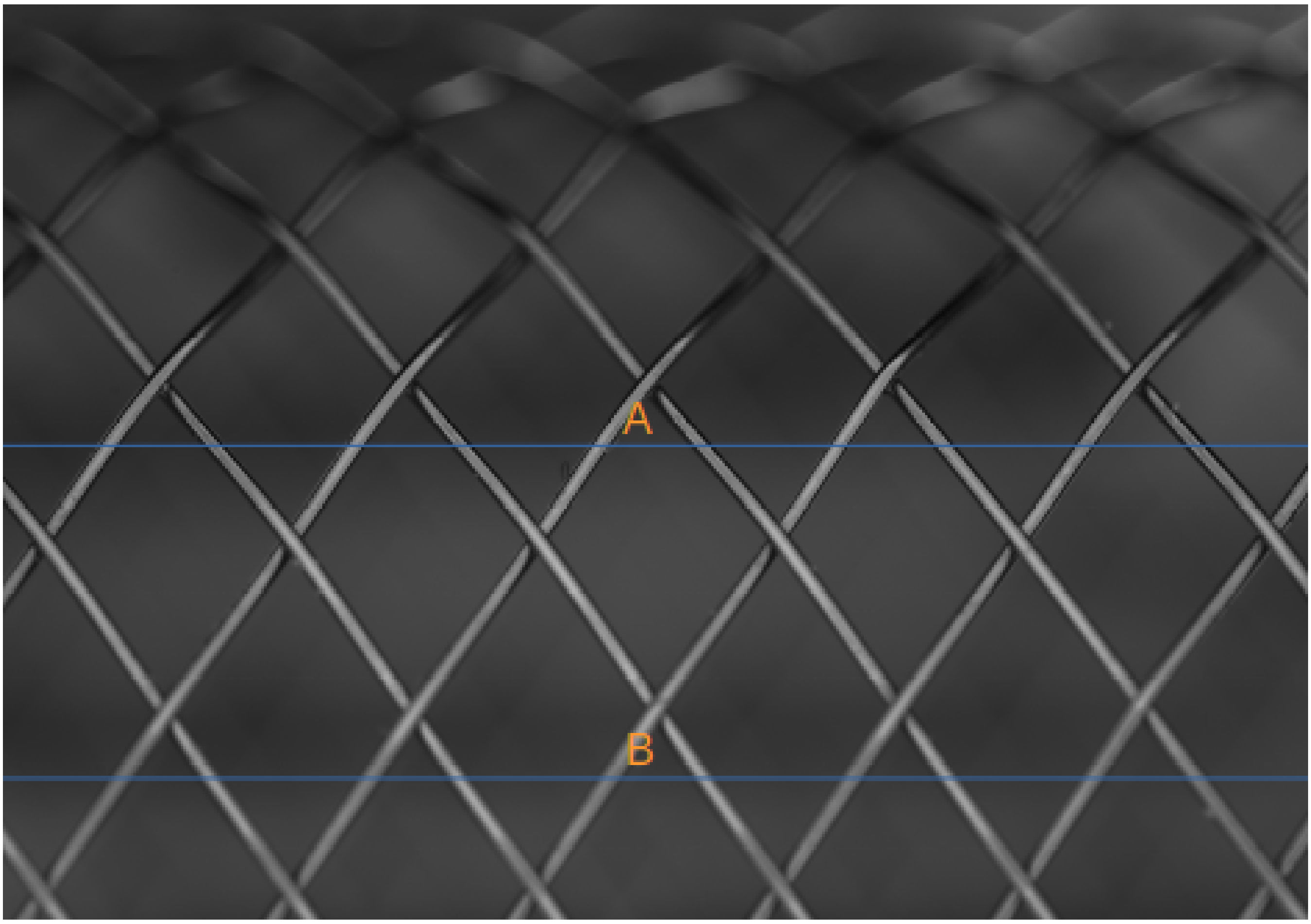


(3)

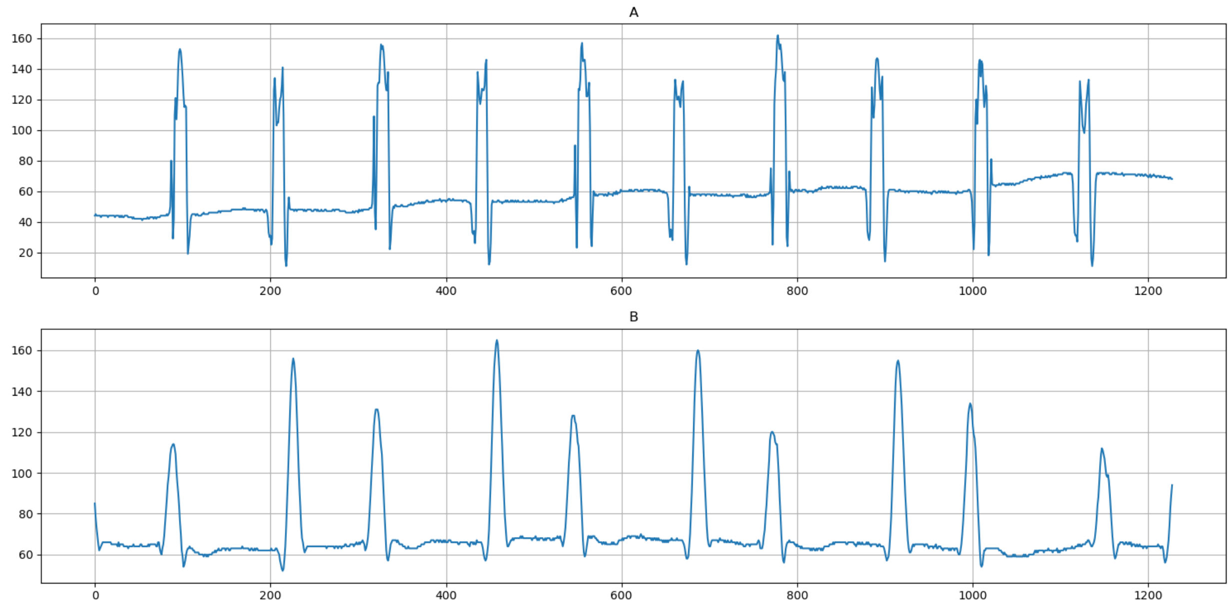
1. First approach:

We started by working on the idea that a region with a good focus is a region with sharp edges. So we studied the pixel intensity variation in horizontal lines of the image as an example: A and B in (4). (The pixel intensity is a single value in the gray-level, or three values in a color images. We study the photos in gray-level, so the intensity varies between black=0 and white = 255)

The graphs A and B shows the pixel intensity variation of all the pixels under the line A, respectively B.

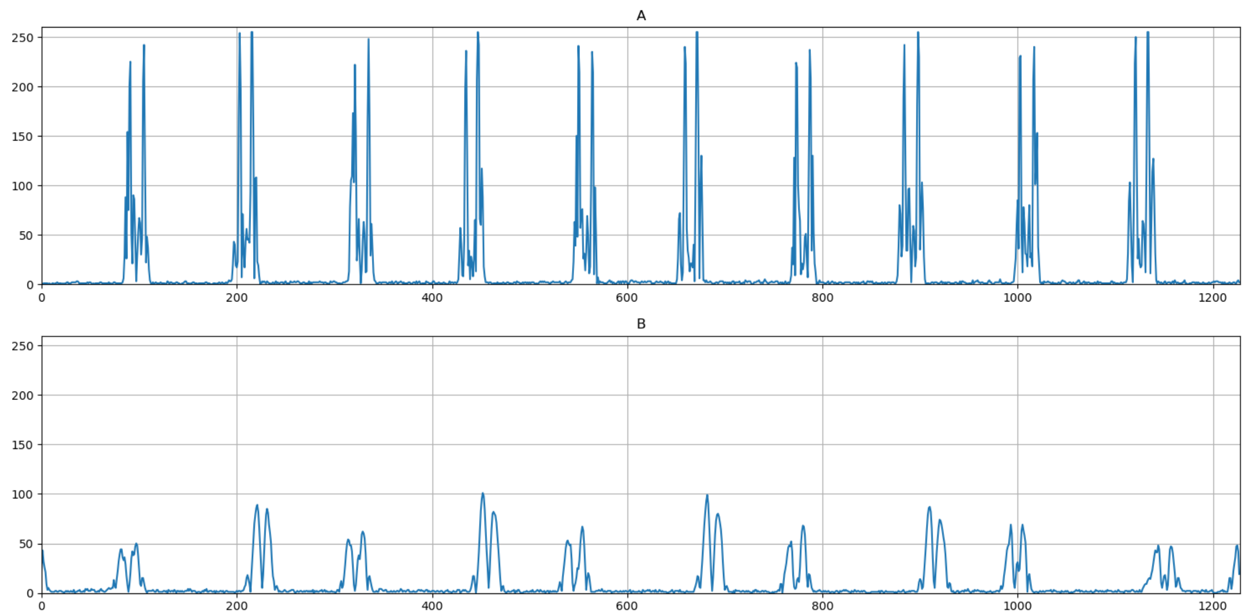


(4)



(5)

As shown in the two previous graphs, the pixel intensity variation under the line B is smoother than the one under the line A due to the unsharpness- respectively the sharpness- of the edges. (6) is the result of applying the mathematical operator Sobel to determine the increasing or decreasing rate of a pixel intensity compared to its two neighbor pixels.



(6)

The Sobel operator can be mathematically described as followed:

It uses two 3x3 kernels: **Sx** for horizontal- and **Sy** for vertical changes:

$$\mathbf{Sx} = \begin{matrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{matrix} \quad \mathbf{Sy} = \begin{matrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{matrix}$$

If we define **A** as the source image, and $\mathbf{G}_x$ and $\mathbf{G}_y$ are two images which at each point contain the horizontal and vertical derivative approximations respectively, the computations are as follows:

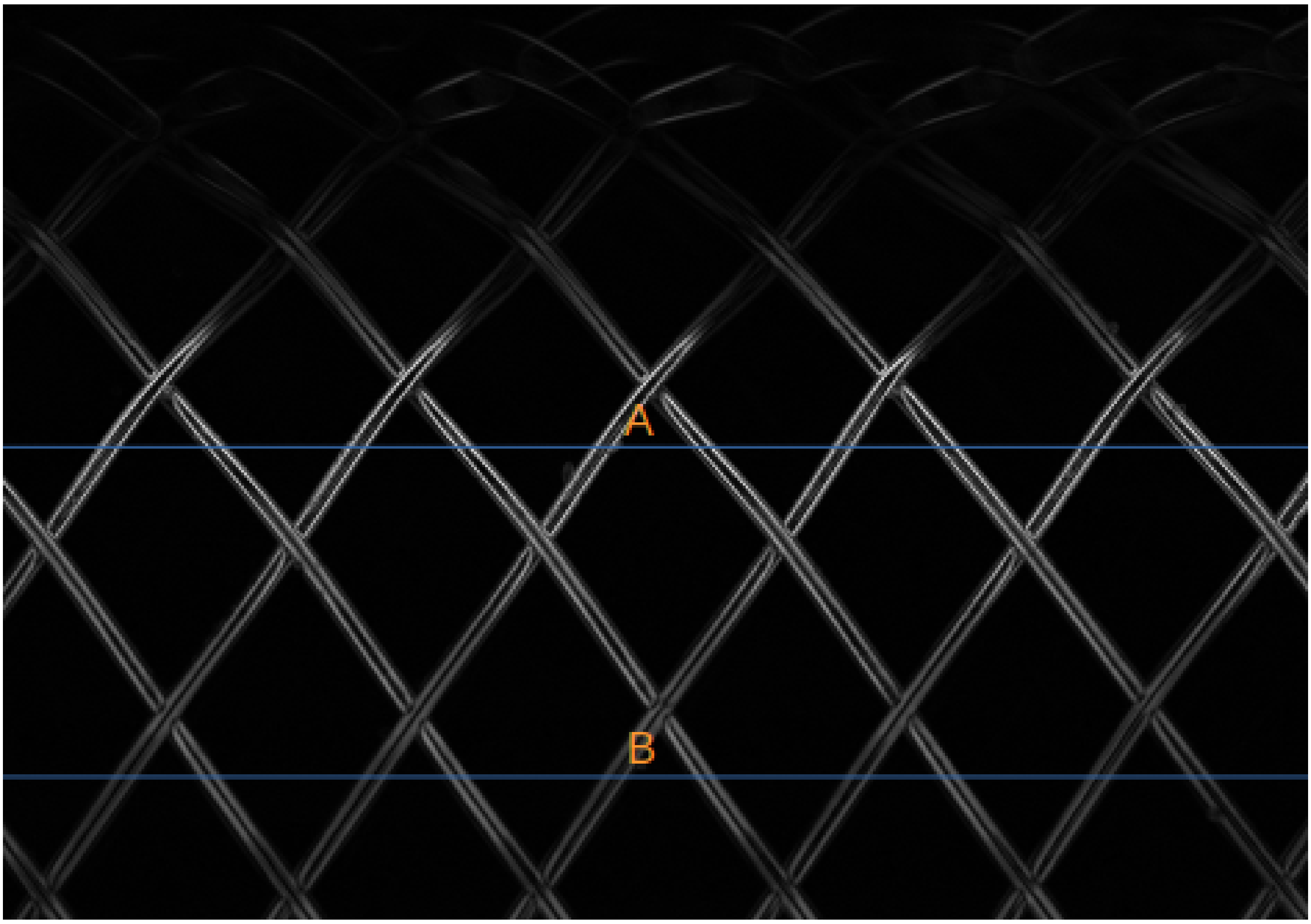
$$\mathbf{G}_x = \begin{matrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{matrix} * A \quad \text{and} \quad \mathbf{G}_y = \begin{matrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{matrix} *A$$

where here denotes the 2-dimensional signal processing convolution operation.

At each point (x,y) in the image, the resulting gradient approximations can be combined to give the gradient magnitude, using:

$$G = \sqrt{G_x^2 + G_y^2}$$

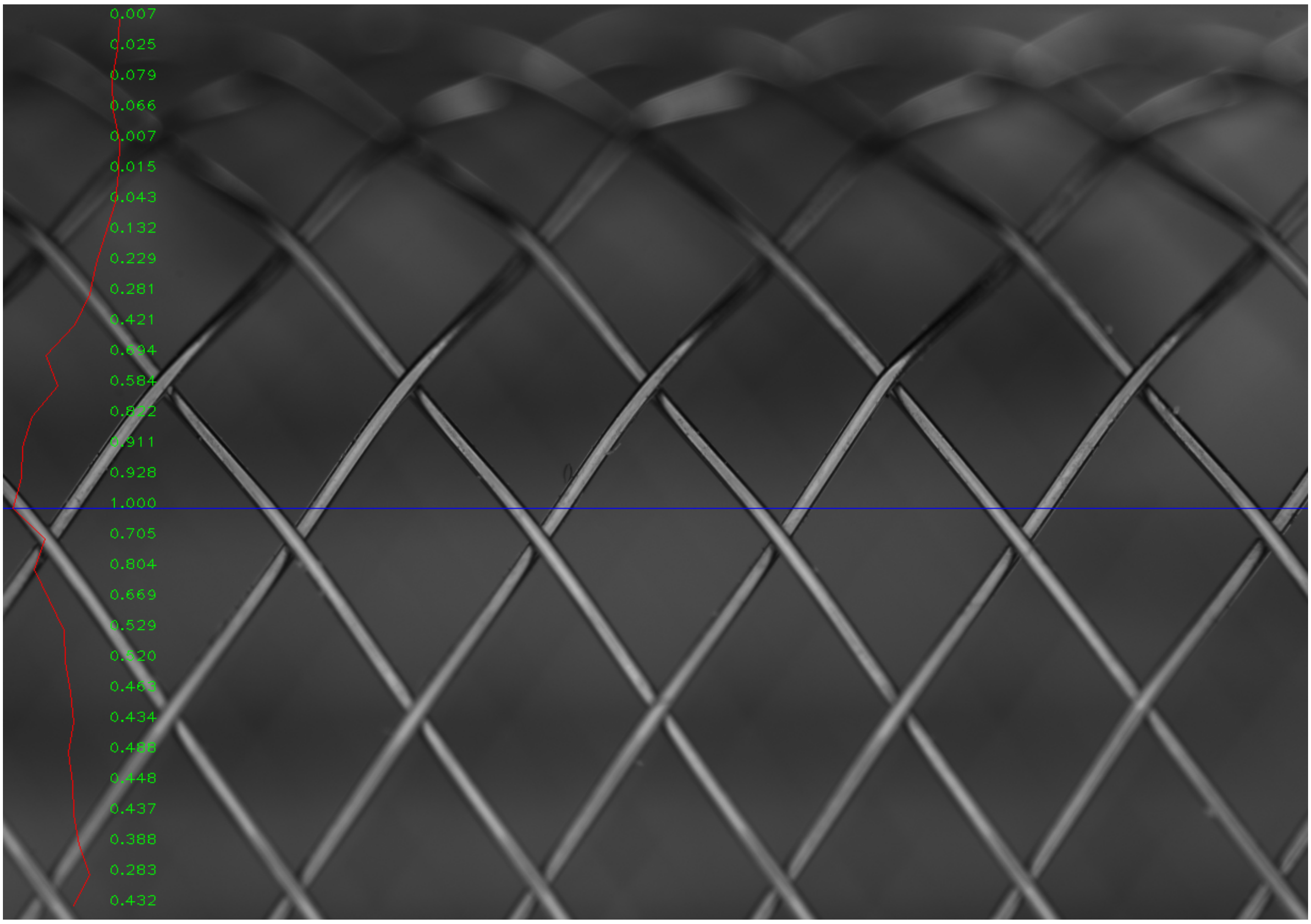
In (7), each pixel has the value of its gradient magnitude **G**. It varies between black=0 and white=255, where 0 represent no difference between the pixel intensity and the intensities of the neighbor pixels (no edge) and 255 a big difference (sharp edge).



(7)

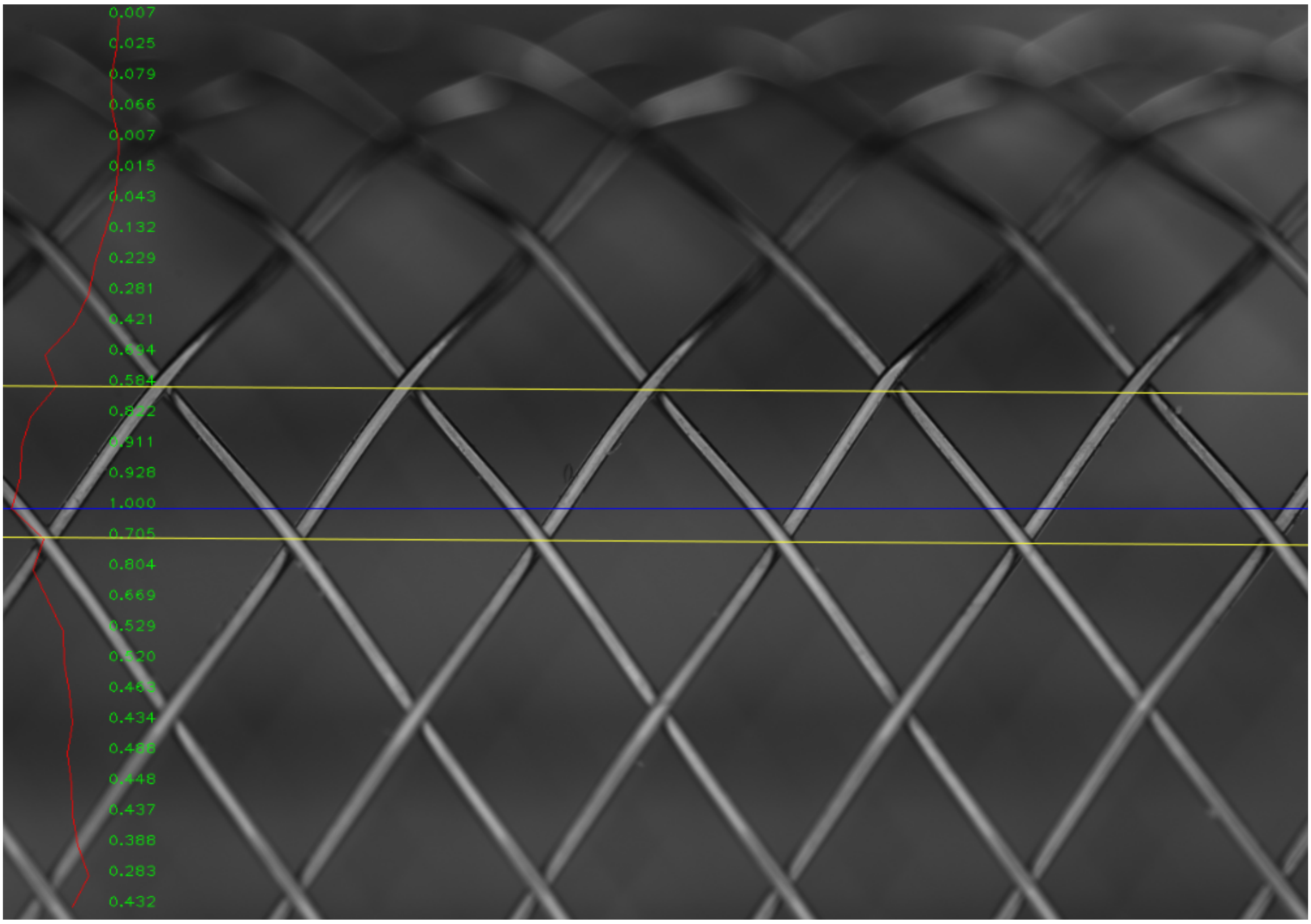
A first approach to calculate the focus rate is to apply the Sobel operator on the whole photo, divide the photo into horizontal segments and then sum up the pixel gradient magnitudes in each segment. The segment with the highest sum is the one with the highest focus rate.

In (8), we divided the photo into 30 segment. In order to normalize the results between 0 and 1, we divided each sum with the highest occurring sum. The blue line is the center of the segment with the highest focus rate. The red graph indicate the variation of the focus rate.



(8)

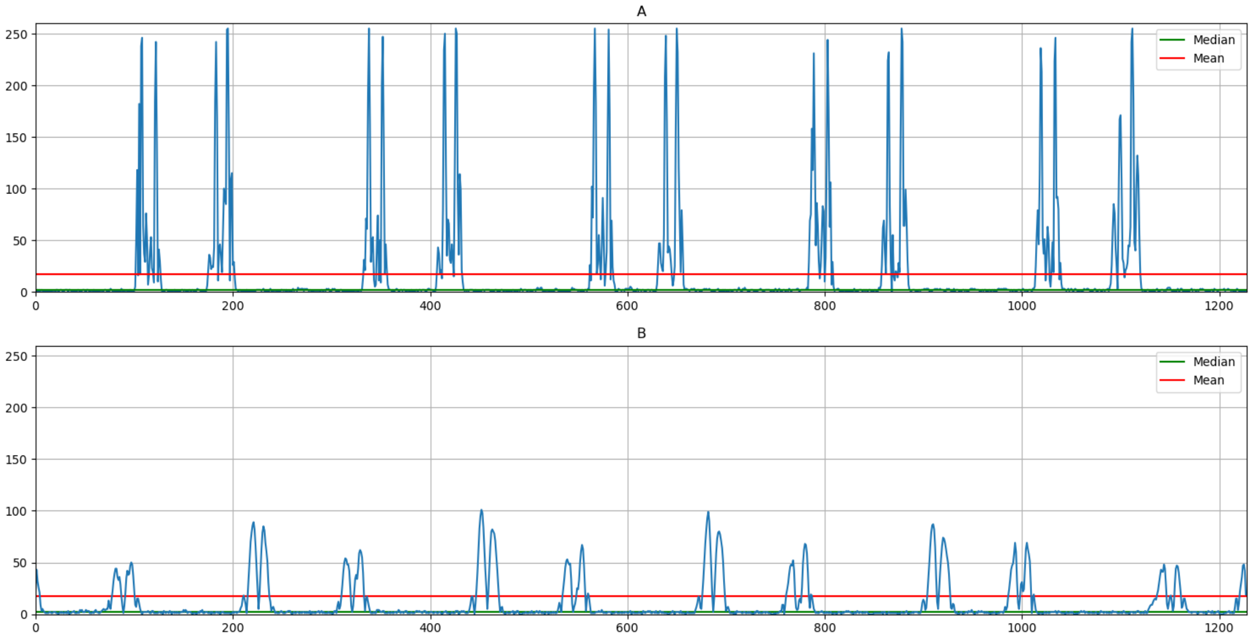
Due to the texture of the stent, some segments have less number of edges compared to others because of the intersections of the wires. So if a segment which has few edges and all of them are perfectly sharp, may be considered less focused compared to a segment with many less sharp edges. In (9) the two yellow lines shows that the segments with the wires intersections have a smaller focus rate compared to their neighbor segments.



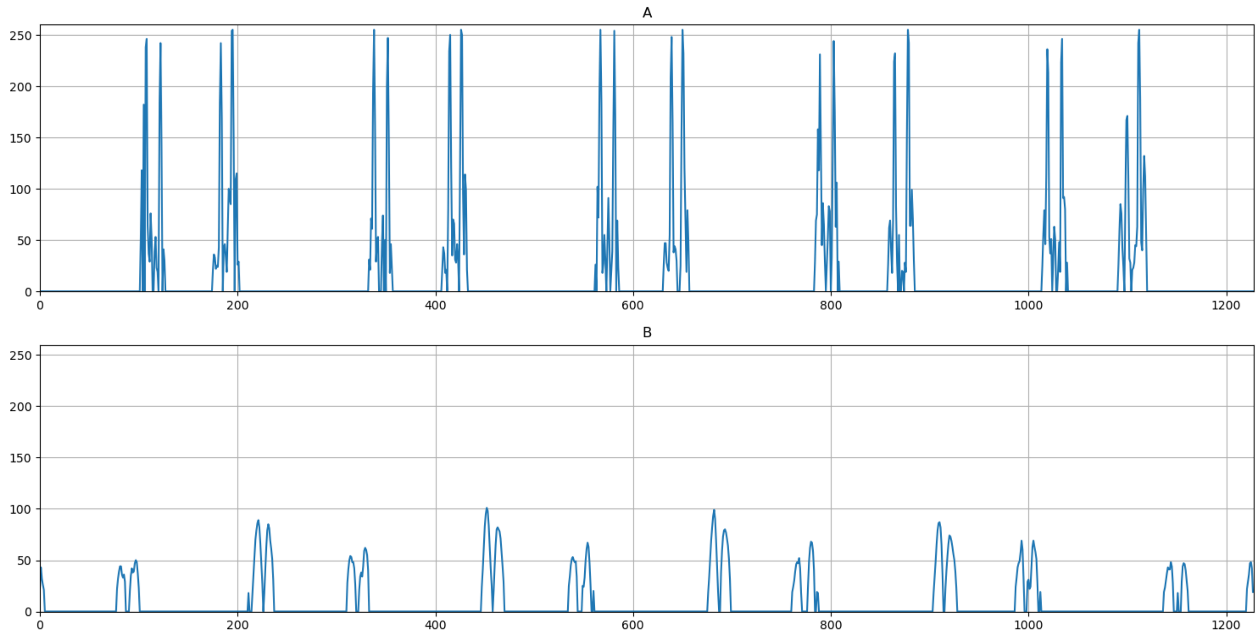
(9)

A solution to solve this problem is to divide the sum of the gradient magnitudes by the number of edge pixels in that segment. An edge pixel is the one that has a gradient magnitude bigger that a certain threshold. (The threshold will be later studied and defined)

In (10), the blue graphs are for the gradient magnitudes under the lines A and B in (7). The red line is the mean value and the green line is the median. By taking the highest mean value of all pixel lines in the photo as a threshold, we eliminate the noise caused by the background by setting its gradient magnitude to zero, and extract the pixels that are more likely to be edge pixels, (11).



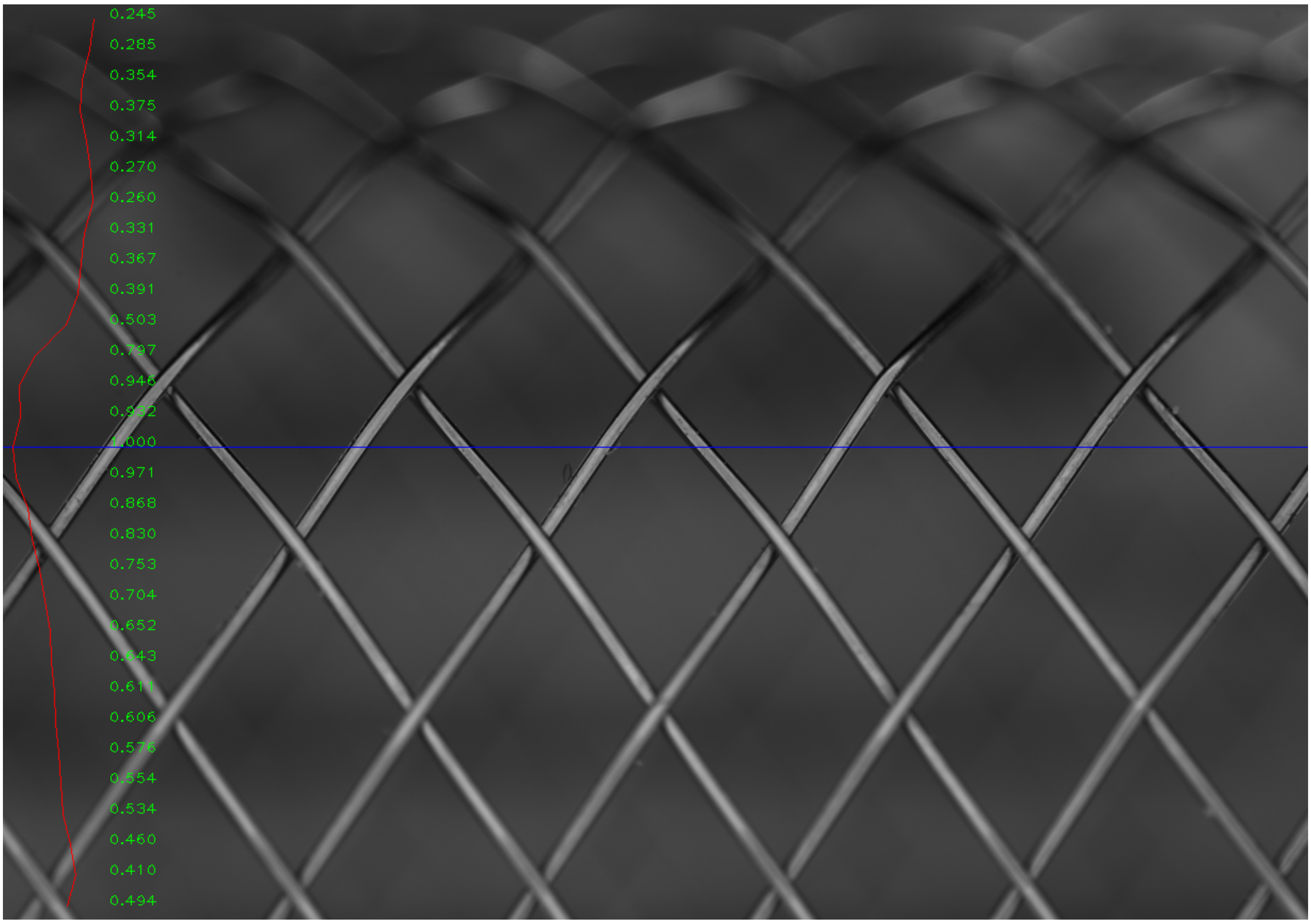
(10)



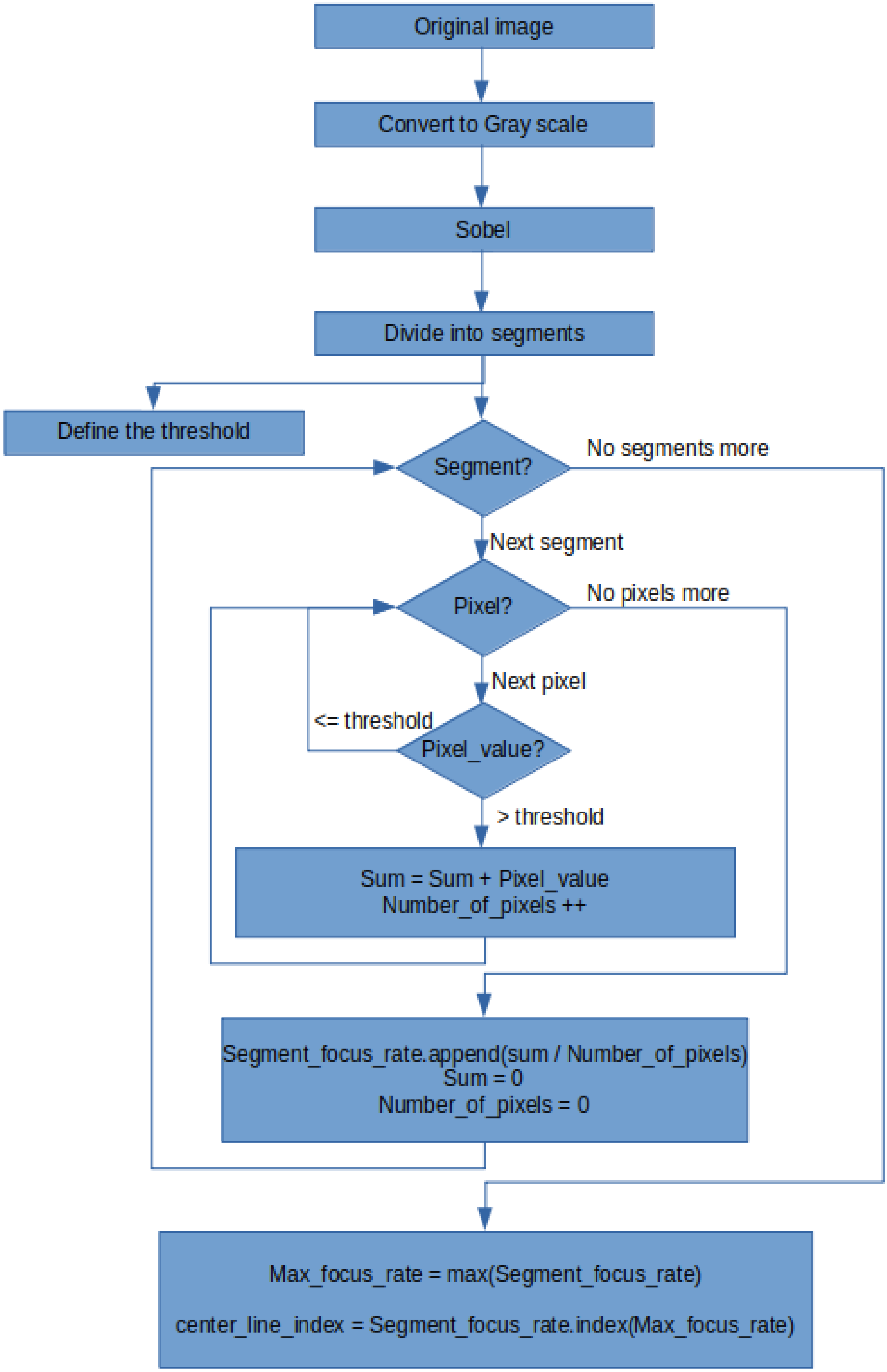
(11)

Final result:

S_Sobel_E_Mean: Segment, Sobel, Edges, Mean



(12)


Original image
Convert to Gray scale
Sobel
Divide into segments
Define the threshold
Segment?
No segments more
Next segment
Pixel?
No pixels more
Next pixel
<= threshold
Pixel_value?
> threshold
Sum = Sum + Pixel_value
Number_of_pixels ++
Segment_focus_rate.append(sum / Number_of_pixels)
Sum = 0
Number_of_pixels = 0
Max_focus_rate = max(Segment_focus_rate)
center_line_index = Segment_focus_rate.index(Max_focus_rate)

2. Second approach:

Based on the scientific paper ‘Robust Automatic Focus Algorithm for Low Contrast Images Using a New Contrast Measure’ written by Xin Xu, Yinglin Wang, Jinshan Tang, Xiaolong Zhang and Xiaoming Liu, published in 2011, we tried to enhance the discrimination power of our program, especially when dealing with low contrast images, suffering from a lack of sharpness, because they are easily influenced by noise.

The idea is to measure the contrast by applying the squared Laplacian:

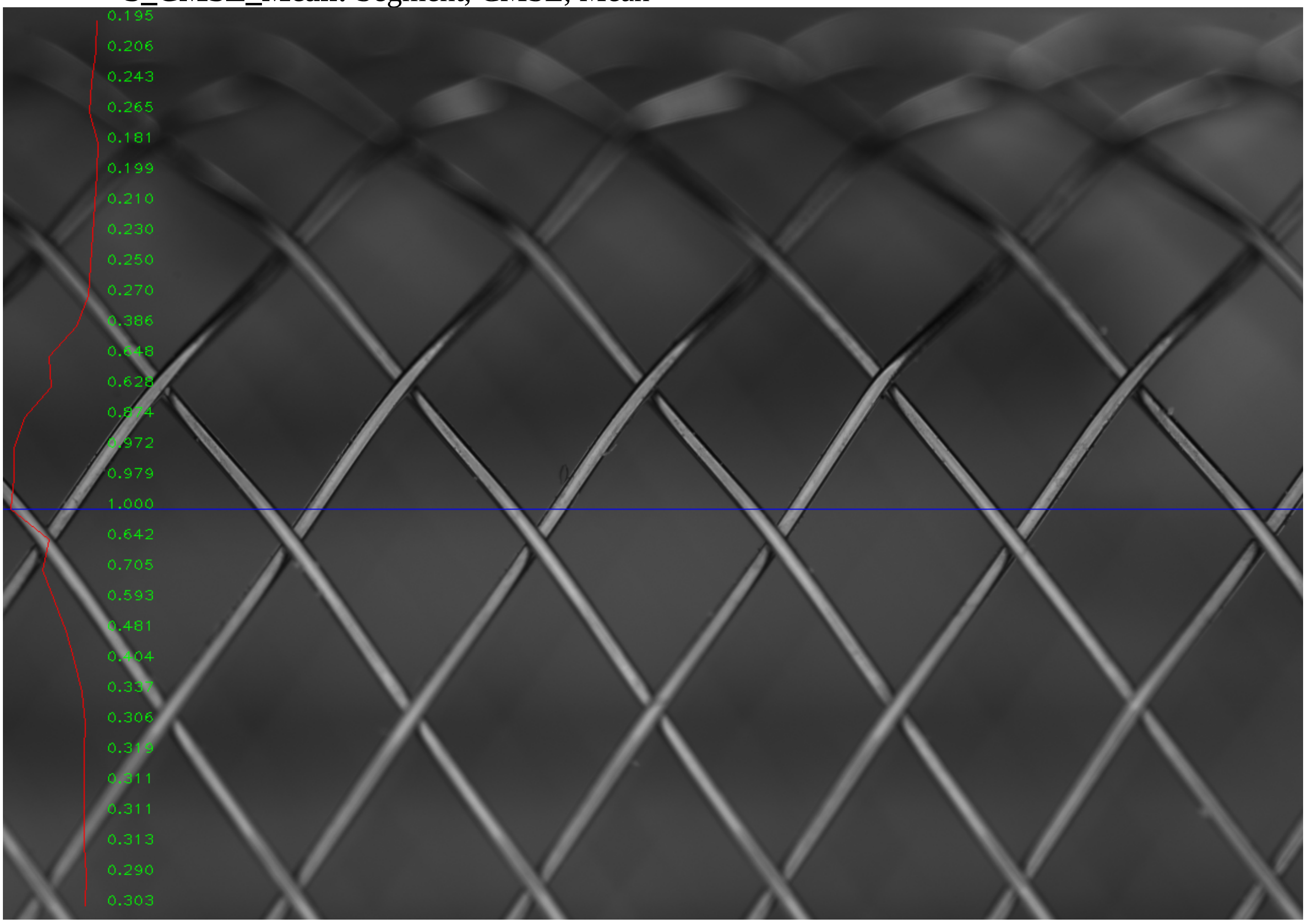
$$L(x,y)=\frac{1}{J*K}\sum_{x=1}^{J}\sum_{y=1}^{K}G(x,y)^2$$

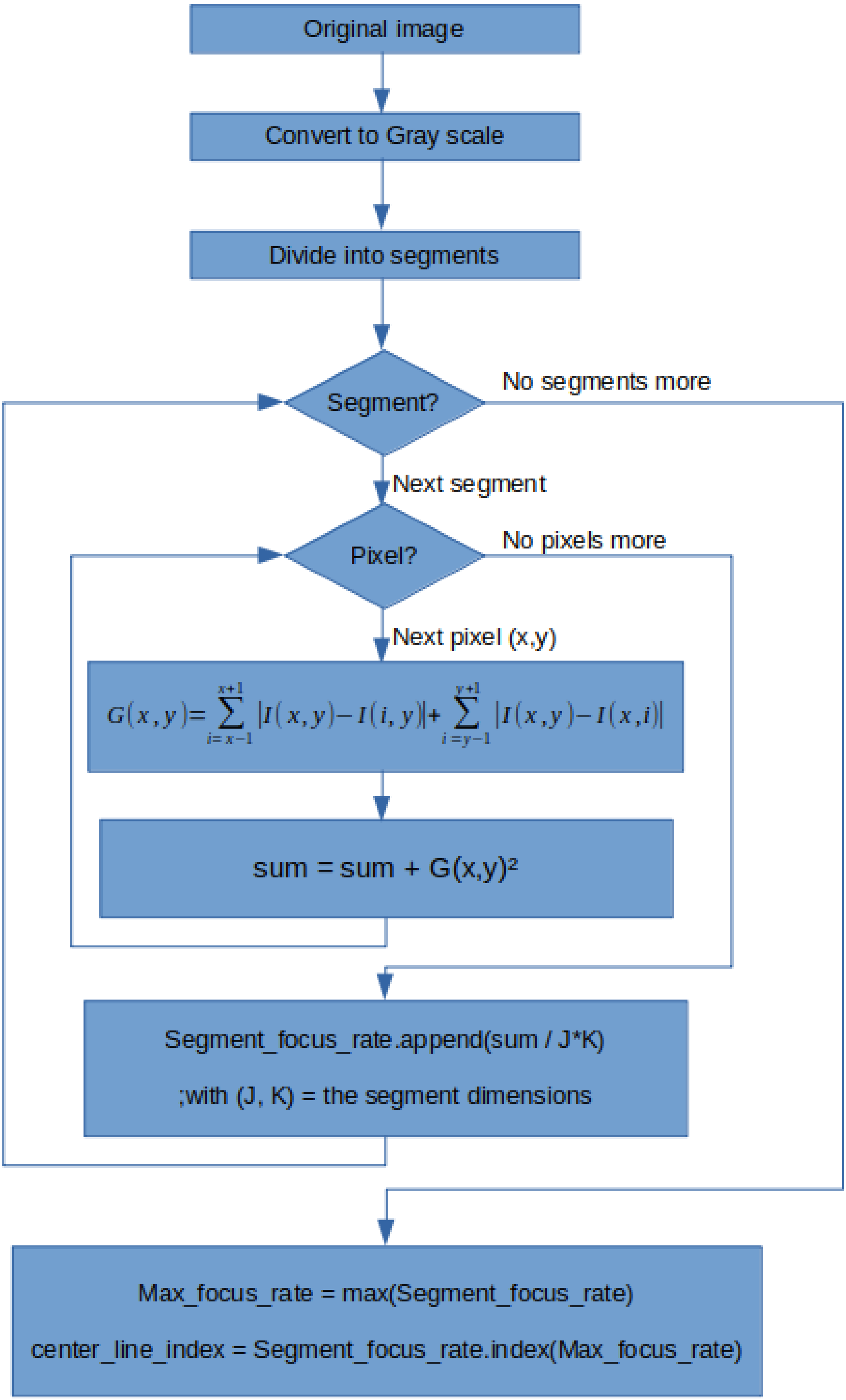
where G(x, y) is computed by:

$$G(x,y)=\sum_{i=x-1}^{x+1}|I(x,y)-I(i,y)|+\sum_{j=y-1}^{y+1}|I(x,y)-I(x,j)|$$

where I(x, y) is the intensity of the pixel (x, y), the parameters J and K are the height and width of focusing region in the image over which the contrast is evaluated.

S_CMSL_Mean: Segment, CMSL, Mean





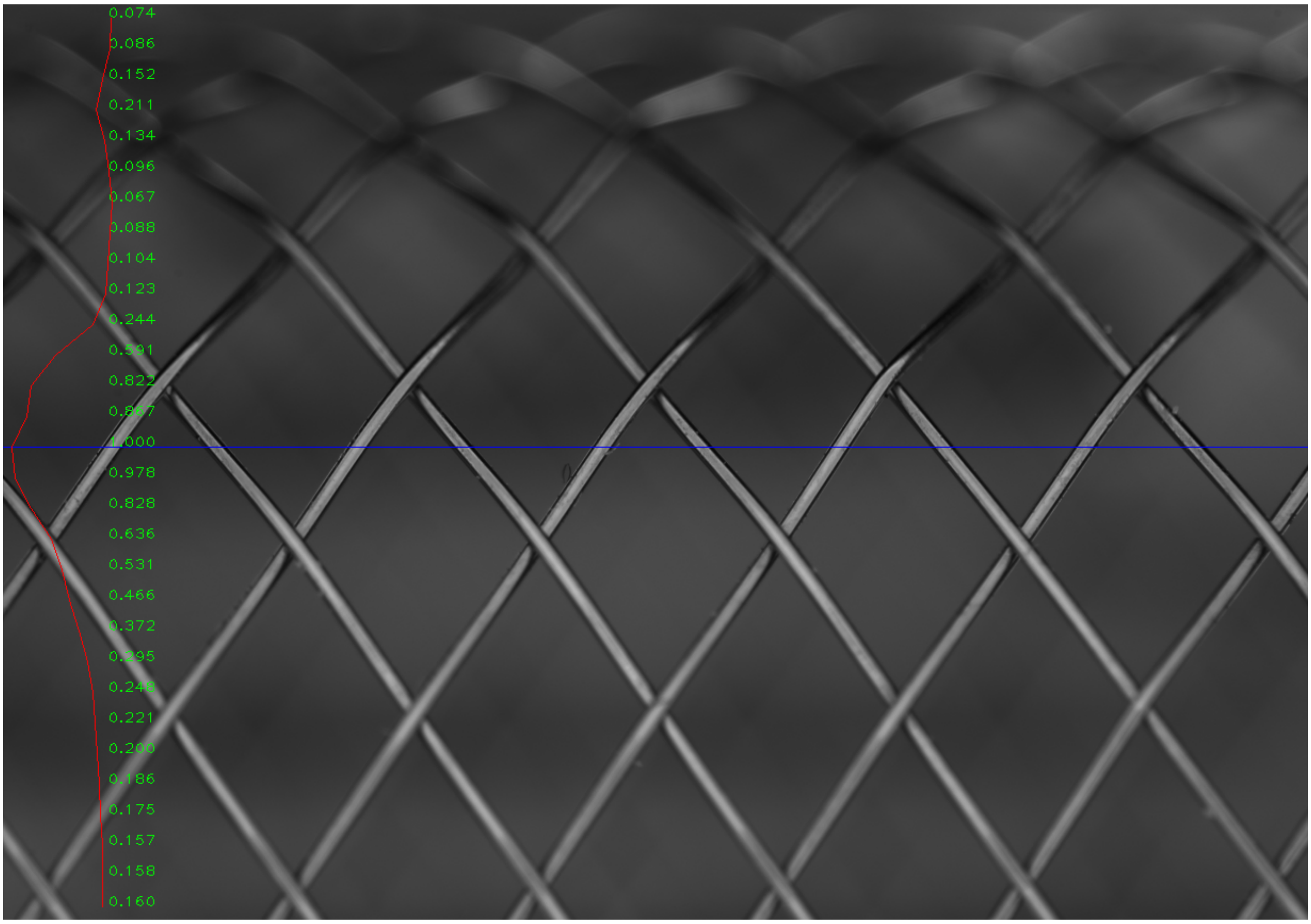
Original image
Convert to Gray scale
Divide into segments
Segment?
No segments more
Next segment
Pixel?
No pixels more
Next pixel (x,y)
$G(x,y)=\sum_{i=x-1}^{x+1}|I(x,y)-I(i,y)|+\sum_{i=y-1}^{y+1}|I(x,y)-I(x,i)|$
sum = sum + G(x,y)²
Segment_focus_rate.append(sum / J*K)
;with (J, K) = the segment dimensions
Max_focus_rate = max(Segment_focus_rate)
center_line_index = Segment_focus_rate.index(Max_focus_rate)

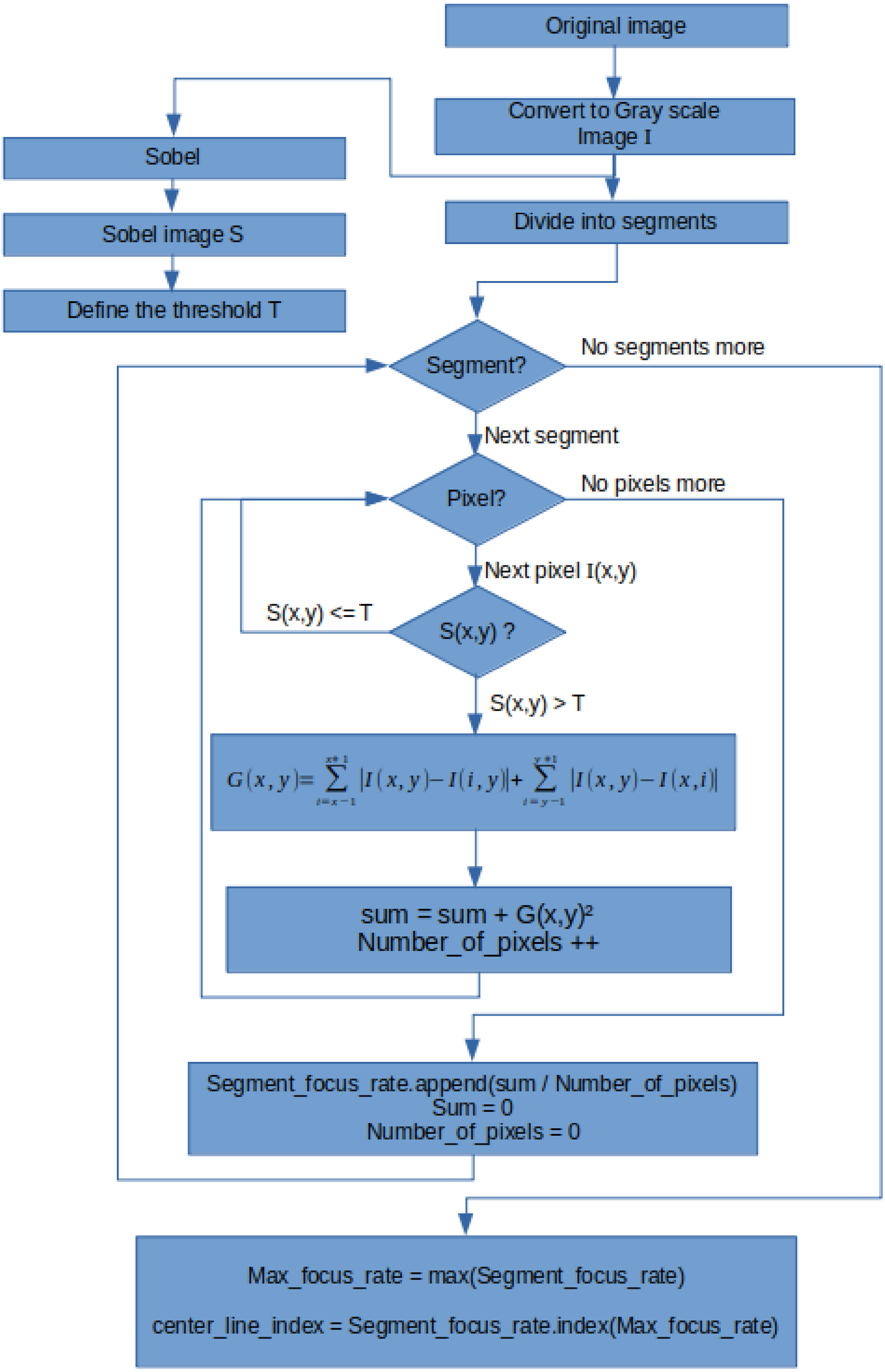
3. Third approach:

The problem of the intersections appears to be the same with the S_CMSL_Mean also. So we implemented Sobel filter to detect the edge pixels and then applied CMSL only on edge pixels.

The CMSL values for edge pixels in a segment will be summed and then divided by the number of edge pixels in that segment.

S_Sobel_CMSL_E_Mean: Segment, Sobel, CMSL, Edges, Mean




Original image
Convert to Gray scale Image I
Sobel
Sobel image S
Define the threshold T
Divide into segments
Segment?
No segments more
Next segment
Pixel?
No pixels more
Next pixel I(x,y)
S(x,y) <= T
S(x,y) ?
S(x,y) > T
$G(x,y)=\sum_{i=x-1}^{x+1}|I(x,y)-I(i,y)|+\sum_{i=y-1}^{y+1}|I(x,y)-I(x,i)|$
sum = sum + G(x,y)²
Number_of_pixels ++
Segment_focus_rate.append(sum / Number_of_pixels)
Sum = 0
Number_of_pixels = 0
Max_focus_rate = max(Segment_focus_rate)
center_line_index = Segment_focus_rate.index(Max_focus_rate)

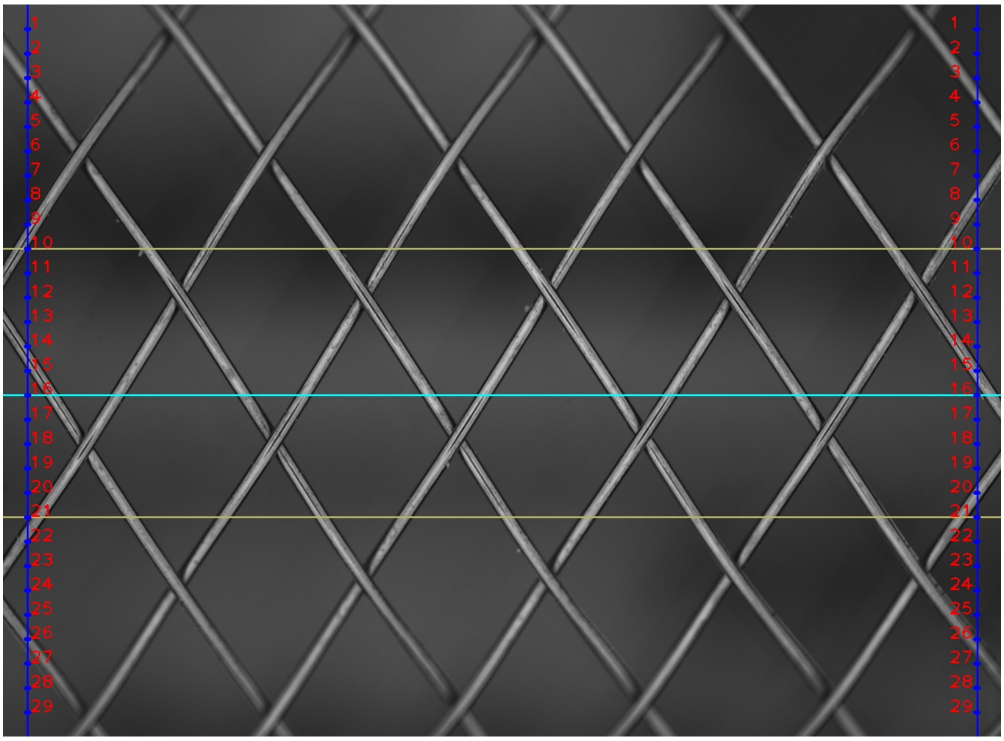
4. Survey:

In order to evaluate the three solutions and compare the results, we decided to lunch a survey in to have a set of reference results. We choose 10 camera positions, mentioned by the red rectangles in (15) and distributed to realistic positions of an acquisition scenario. To cover the whole circumference while the stent is rotation below the camera, we took 22 photos for each of the 10 positions. Then we randomly selected 2 images for each position.

To simplify the task, we re-scaled the images of (900x1228), which is 30% of the the original size (3000x4096), and we scaled to right and the left side from 0 to 30, which represents the segmentation of the image.

The people were asked to identify the area with an acceptable sharpness, determine the top and the bottom lines of this area (The two yellow lines), and decide where the center line could be (The cyan line).

Example survey image:



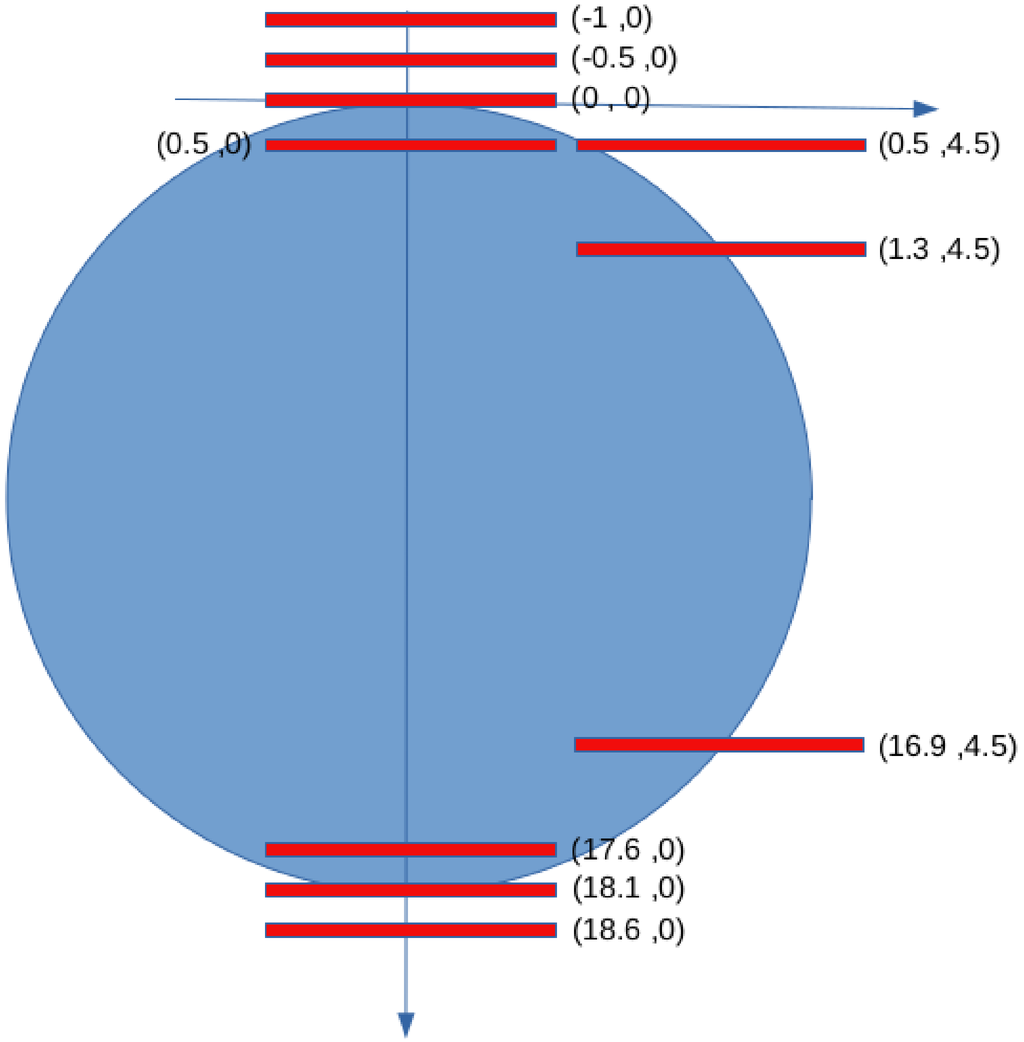
(15)

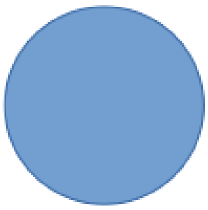
The evaluation of the data serves to create a measure that enables a good automated comparison of the different solutions and to define the focus rate acceptance which will help to identify the boards of the focus region.

As mentioned above, the focus rate in all three solutions is normalized between 0 and 1. We divided each segment focus rate by the highest occurring focus rate (Depends on the algorithm, the focus rate can be a sum or a mean). The center line of the segment with the highest focus rate = 1 is what we call the algorithm center line. The survey center line is the center line of the segment that the people consider as the segment with the highest focus rate.

If the algorithm works properly, the position of the algorithm center line and the position of the survey center line will be equal.

We closed the survey with 10 full answers. The survey was powered by google docs: https://forms.gle/fhAGVwAX9eqFhHHp7





: The circle represents the cross section of the stent. In this case, the stent has a diameter of 18 mm.

: The rectangle represents the Camera field of View (FOV), HFOV = 5.5 mm and DFOV= 0.5 mm

Survey results:

The results for the first image, as an example:

Top	=	{390,360,510,450,450,480,480,420,450,450}
Center	=	{480,600,570,540,600,540,570,600,630,660}
Bottom	=	{660,870,630,660,720,630,630,810,870,870}

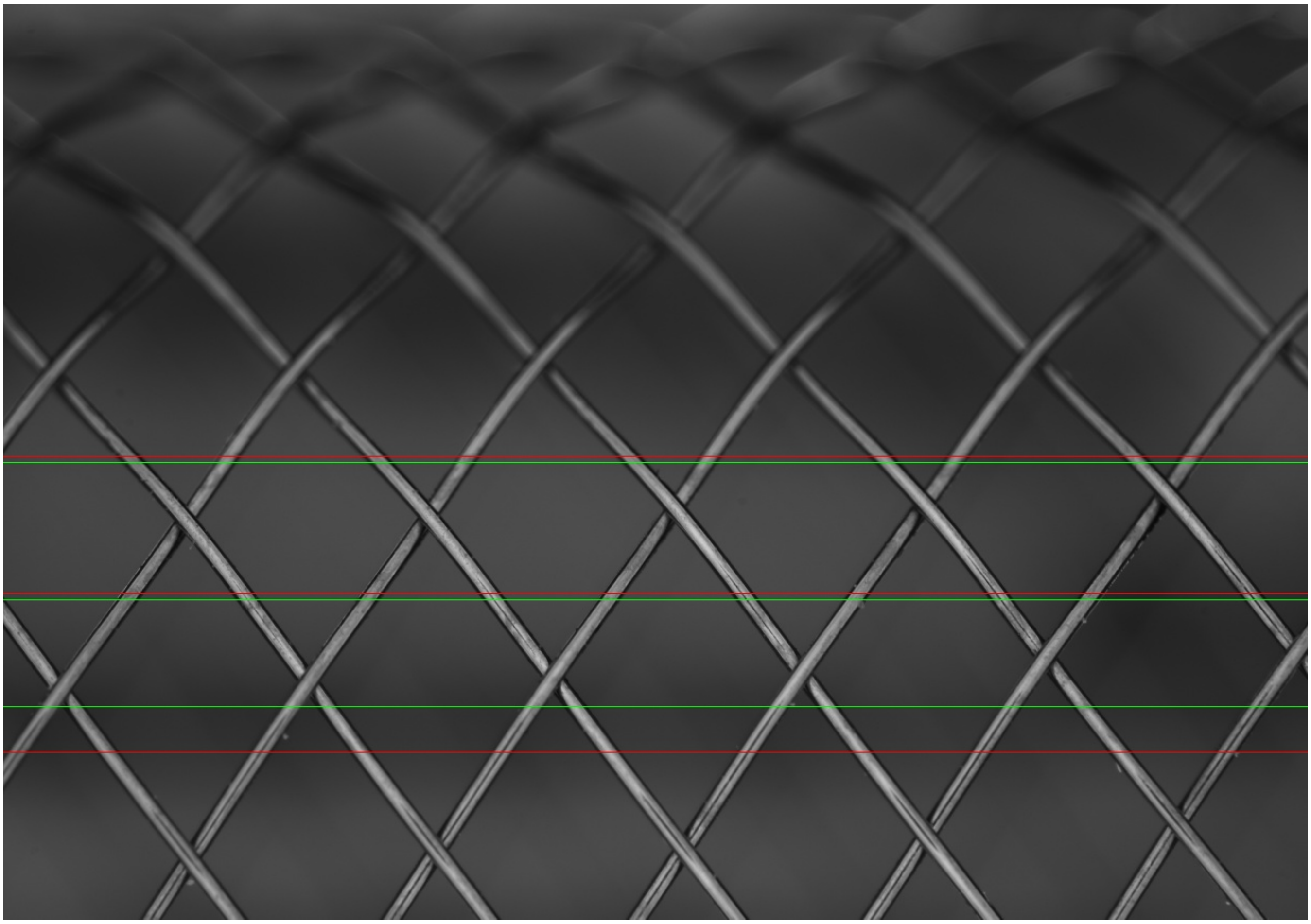
Mean values (Red lines):

Top = 444
Center = 579
Bottom = 735

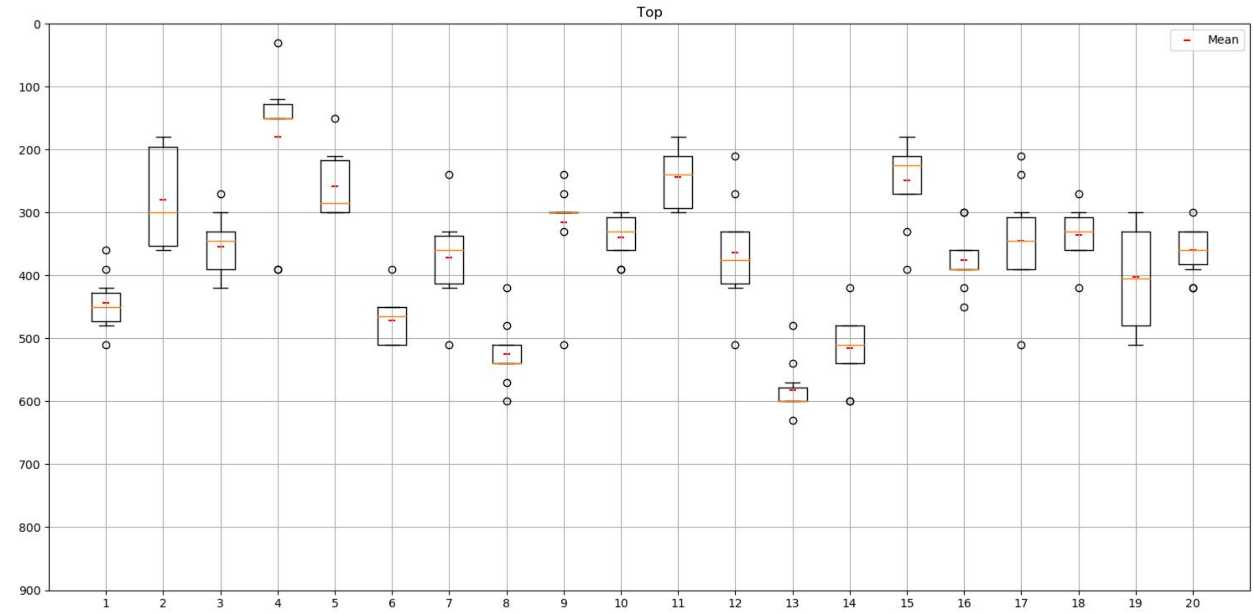
Median values (Green lines):

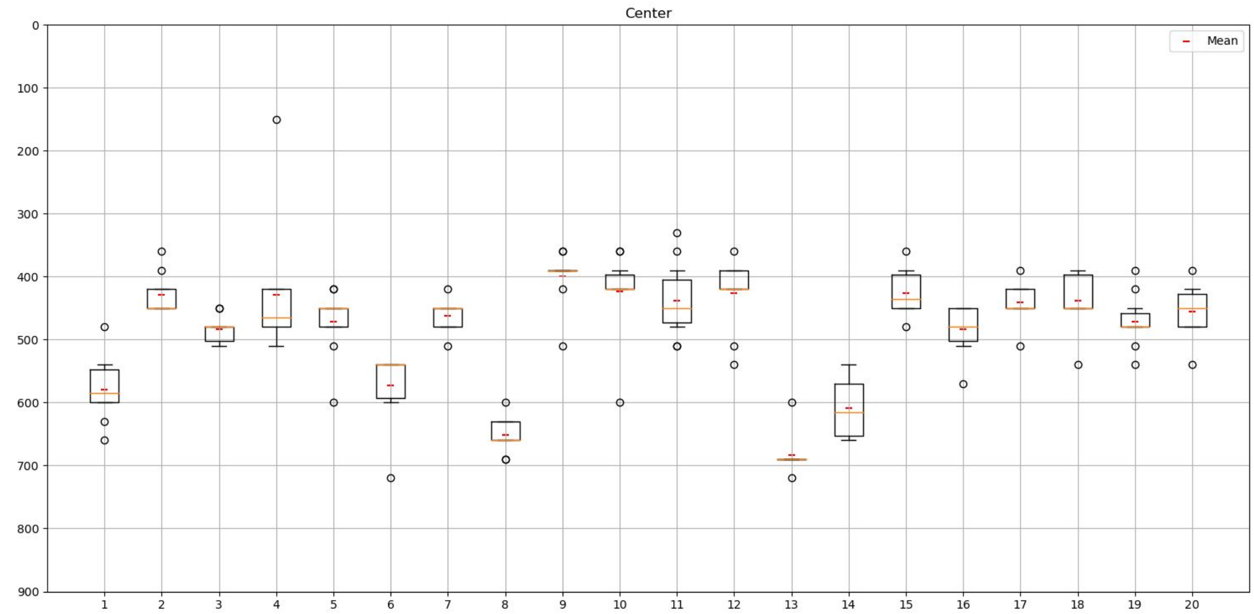
Top = 450
Center = 585
Bottom = 690

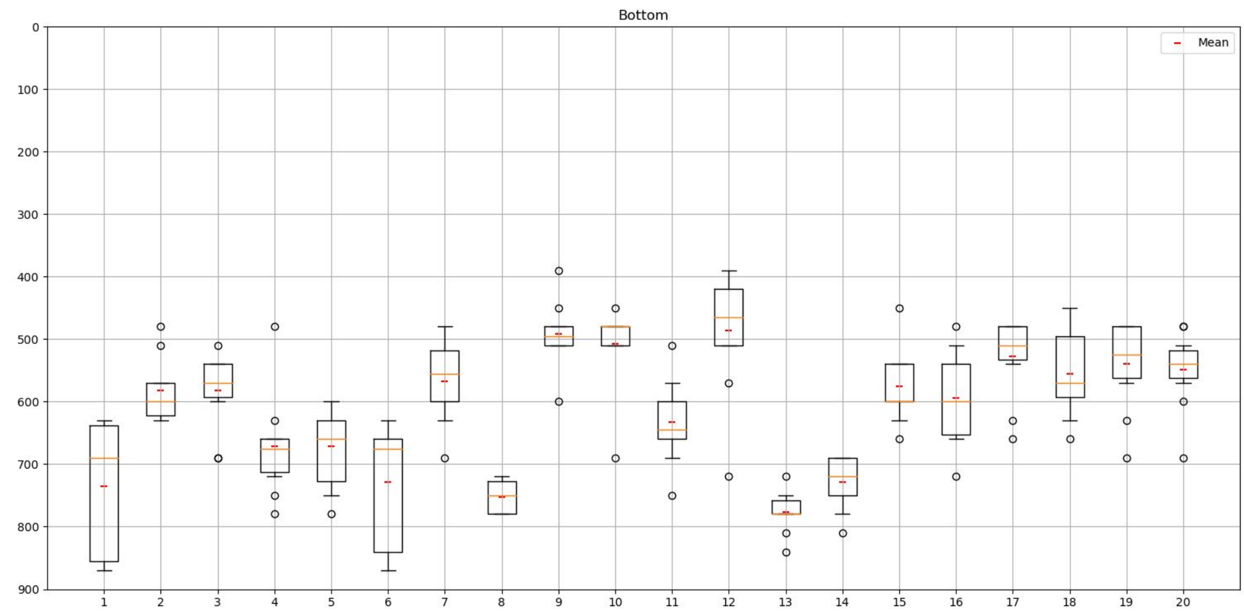
As a compromise, for the further calculations, we used the mean of (Mean values, Median values). So for the first image, as an example, Top = 447, Center = 583 and Bottom = 712.



(16)

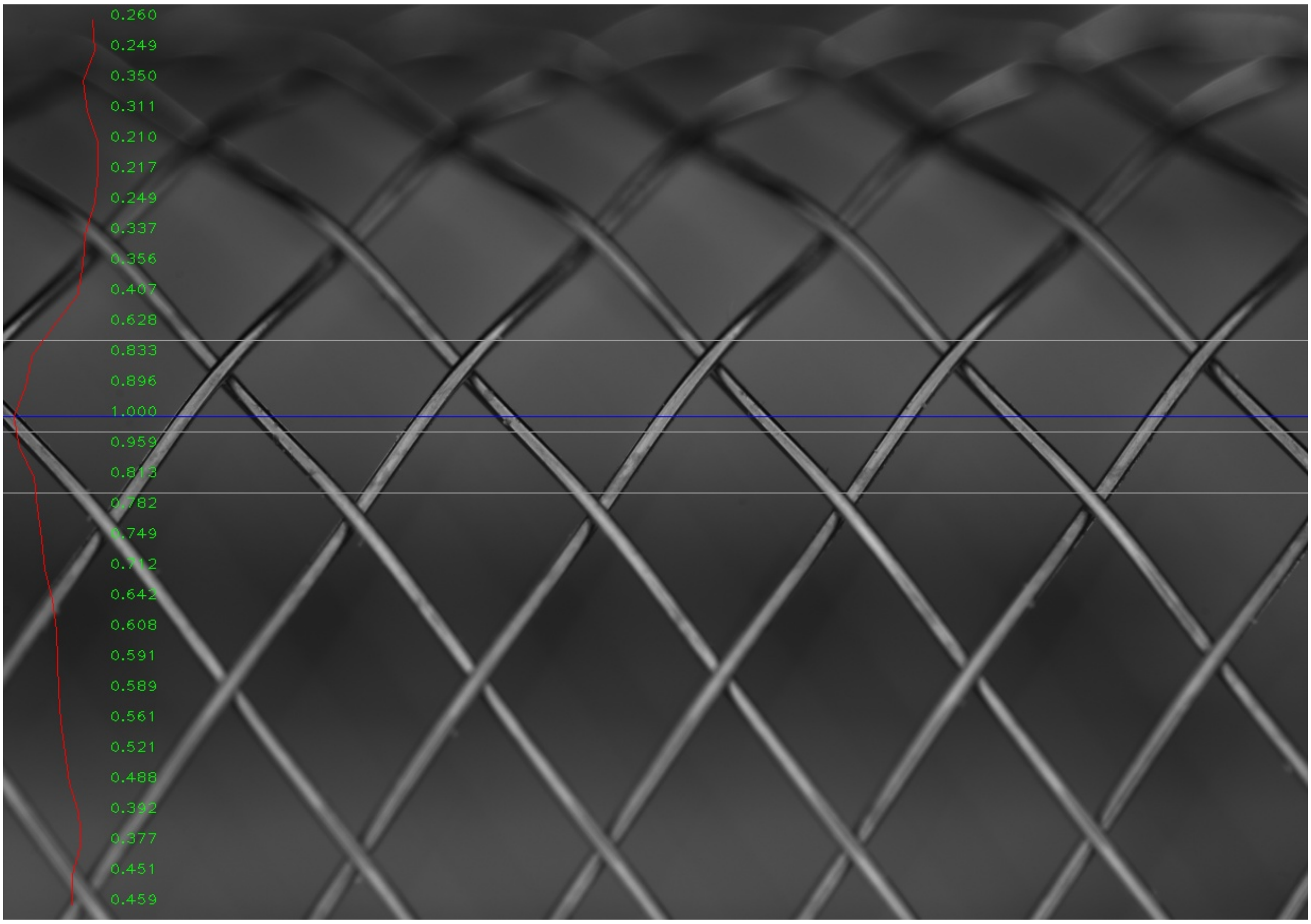

Top
Mean
0
100
200
300
400
500
600
700
800
900
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20


Center
Mean
0
100
200
300
400
500
600
700
800
900
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20


Bottom
Mean
0
100
200
300
400
500
600
700
800
900
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

To evaluate the three algorithms (S_Sobel_E_Mean, S_CMSL_Mean and S_Sobel_E_CMSL_Mean), we run each algorithm on the 20 survey images. Each algorithm gives, as an output, the position of the segment with highest focus rate (In (17), the blue line is the center of that segment) and the values of the focus rate in each segment of the image. Then we compared the position of the survey center line with the position of the algorithm center line (Blue line).

a. S_Sobel_E_Mean:

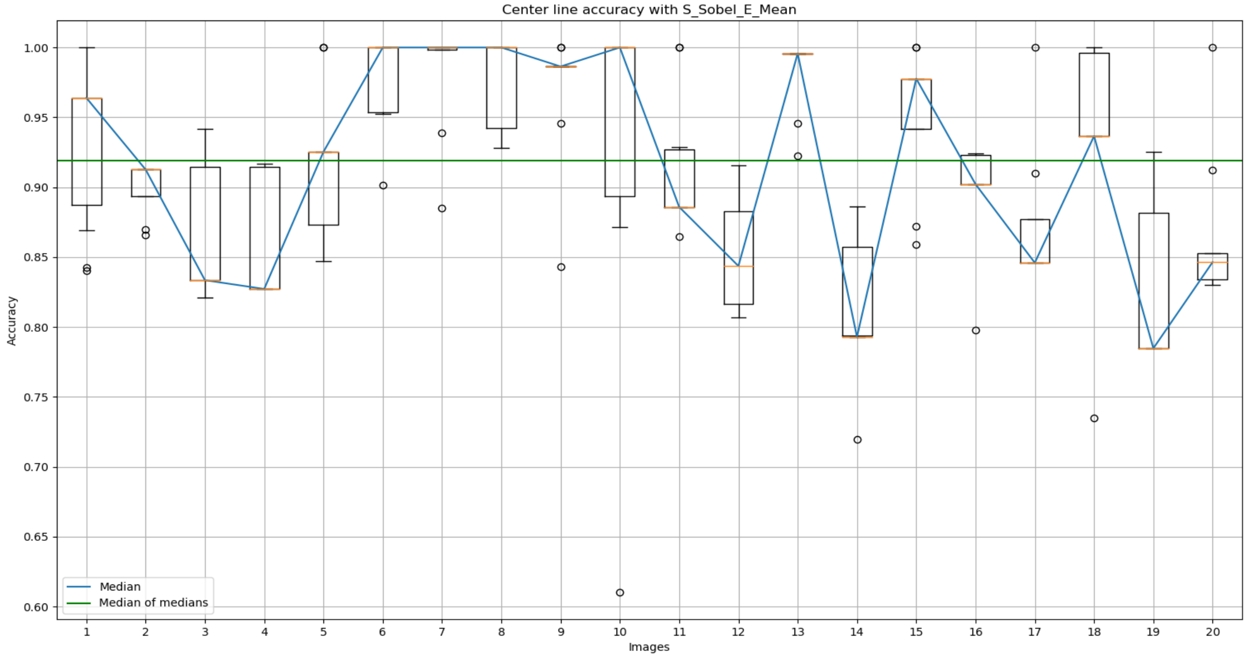


(17)

In (17), the gray lines are for the survey Top, Center and Bottom lines, the blues line is the center of the segment with the highest focus rate based on the S_Sobel_E_Mean algorithm, the numbers in green are the focus rate for each segment and the red curve shows the variation of the focus rate along the image.

The algorithm center line accuracy was studied based on the focus rate of the segment where the survey center line is located. This shows how precisely the algorithm meets the survey decisions.

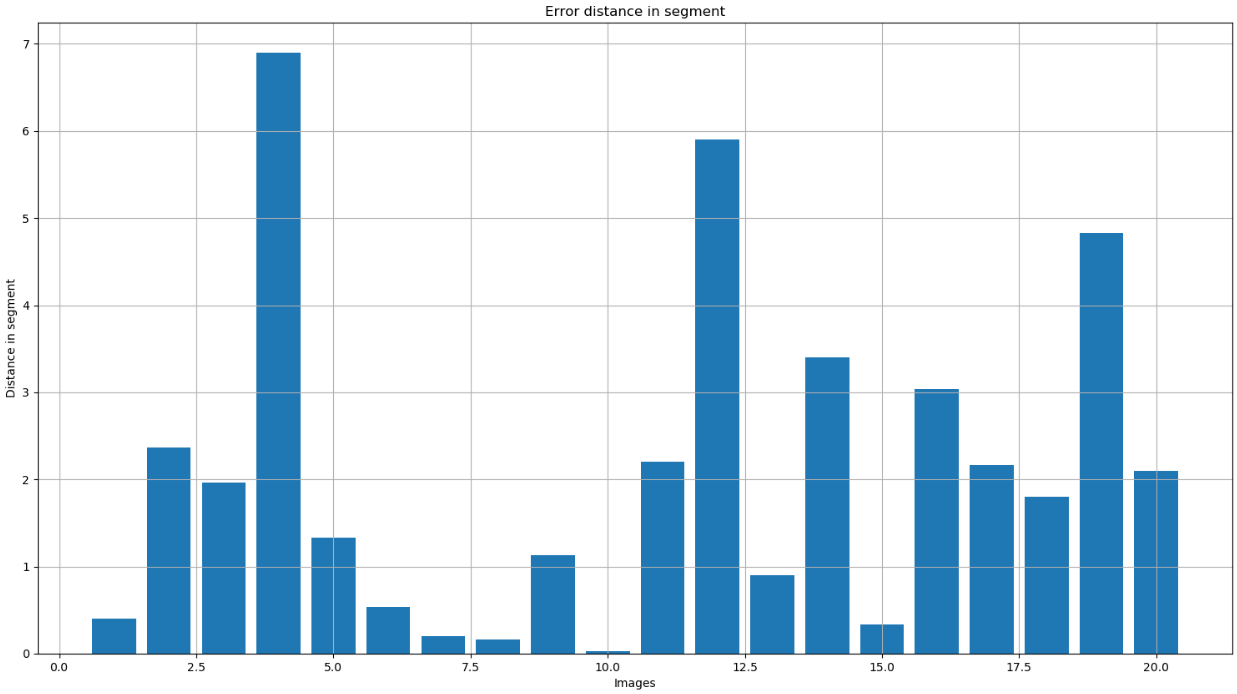
In (18), for each image, all the 10 results of the survey(The box-plot) were evaluated by the S_Sobel_E_Mean algorithm, that means we observe the focus rate of the segment were the chosen position is located. The orange lines are for focus rate of the segments where the survey center lines are located. The green line is the median of all the survey center lines focus rate.



(18)

We measured then the center line deviation in segments (The difference between the survey center line position and the algorithm center line position).

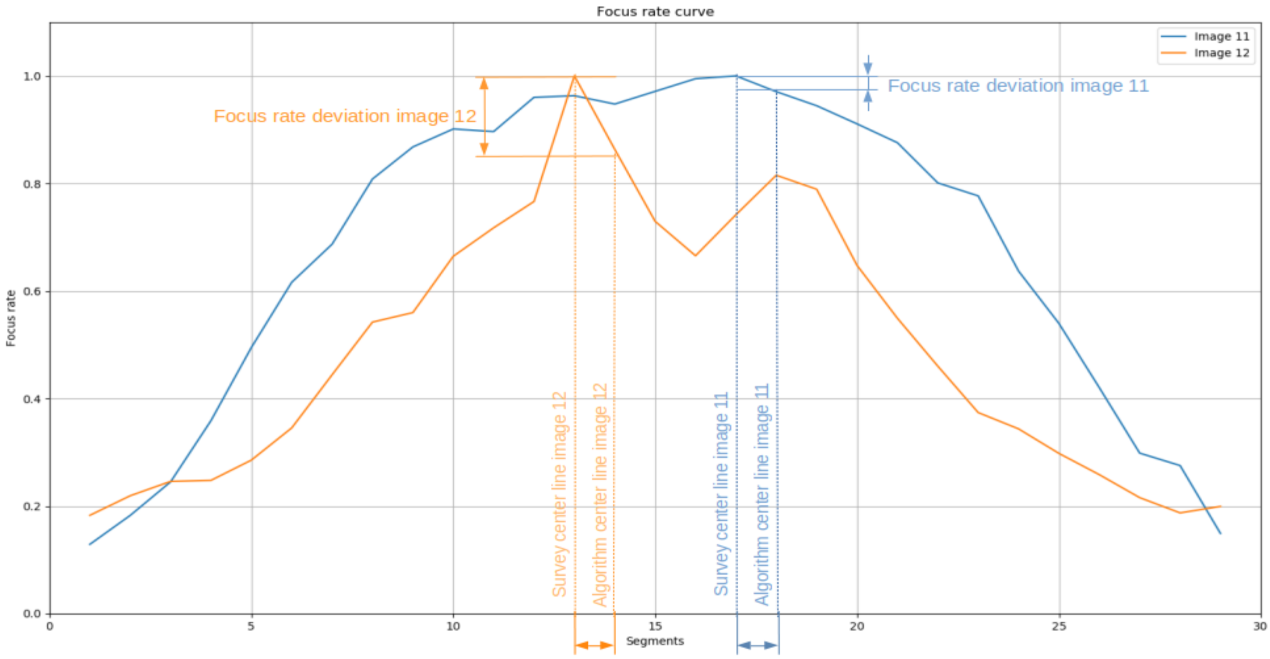
$$\textbf{Error in segment} = \frac{|Survey\ center\ line\ position - Algorithm\ center\ line\ position|}{Segment\ height}$$



(19)

However, the deviation in segments does not highlight its impact on the focus rate deviation between the survey center line and the algorithm center line.

In some images, a deviation of one segment has a bigger impact compared to a deviation of more than one segment in an other image. This due to the focus rate behavior in each image. The two graphs bellow are for the focus rates of two survey images 11 and 12:



The deviation is equal 1 segment for both images

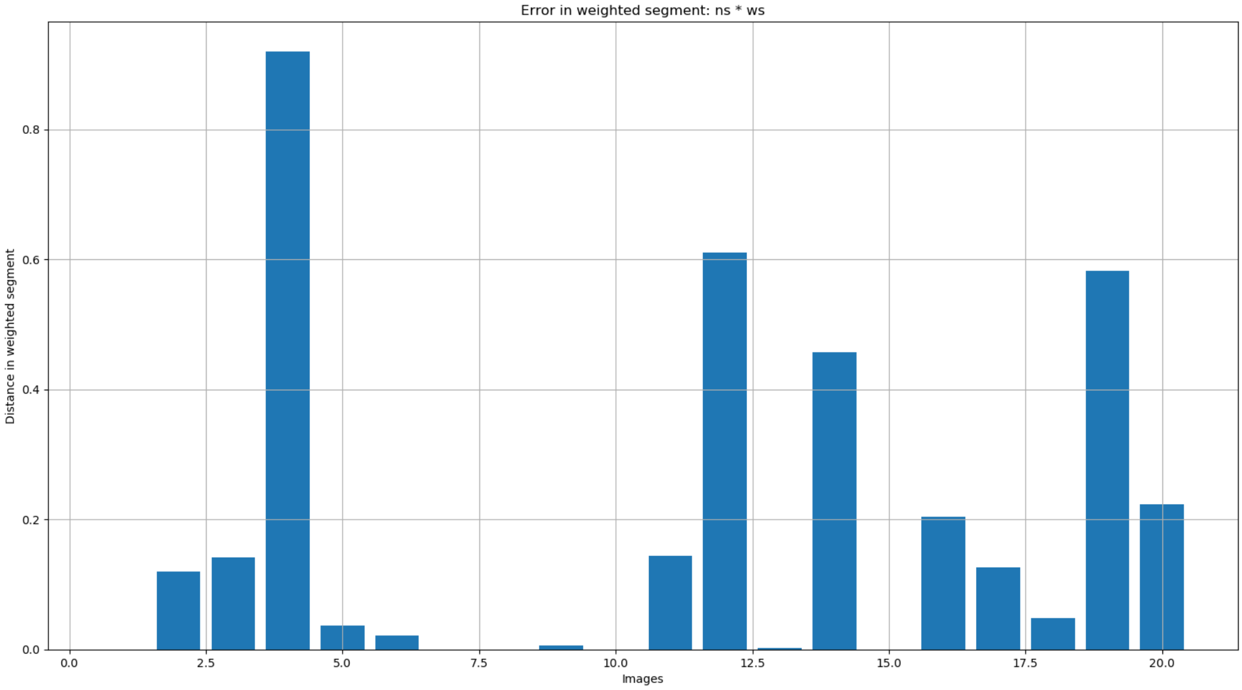
(20)

Let:

D : The deviation
Ss: The segment were the survey center line is located
Sa: The segment were the algorithm center line is located
f(s) = The focus rate in segment s

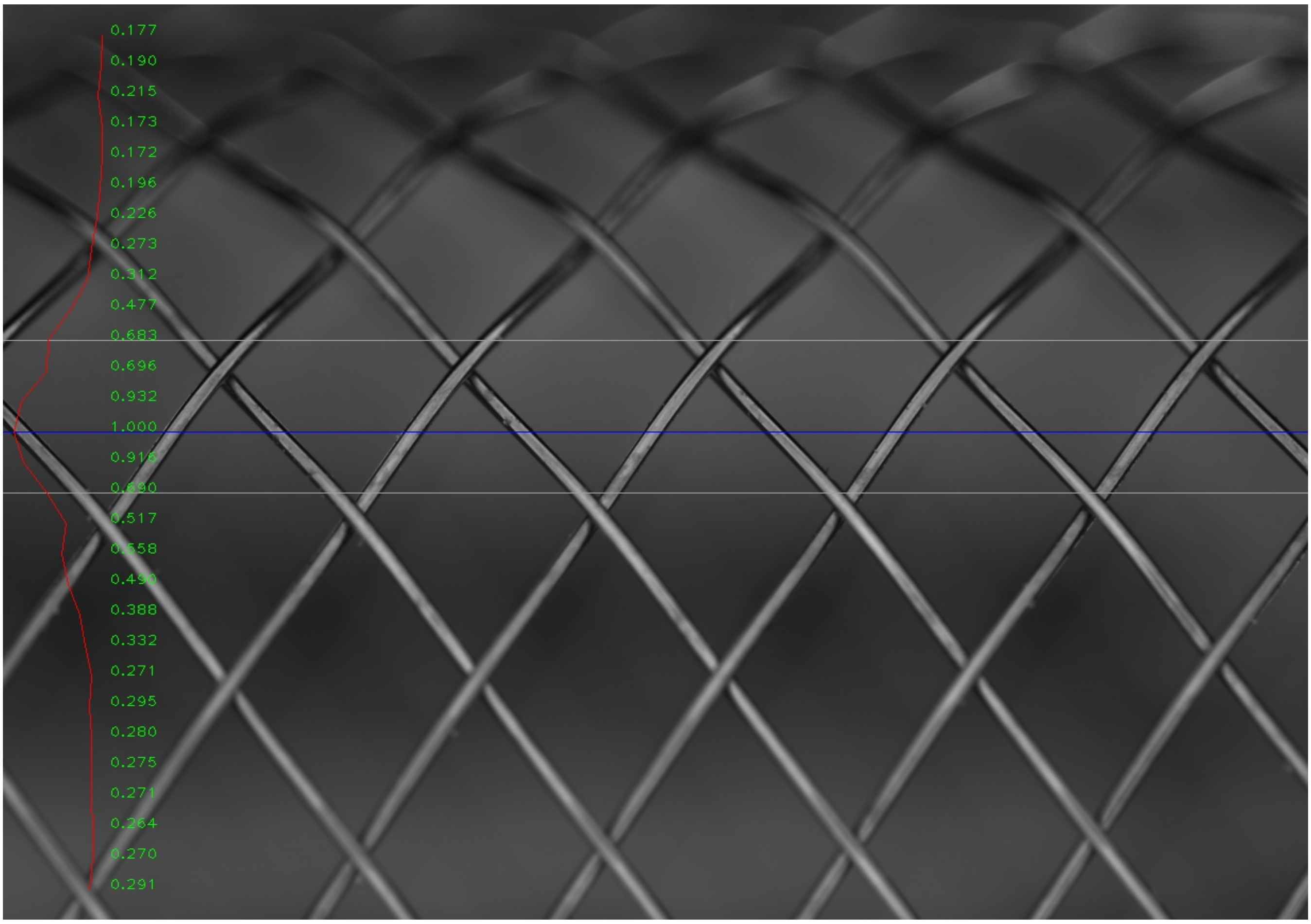
Then:

$$D = \sum_{s=Ss}^{Sa} 1 - f(s)$$

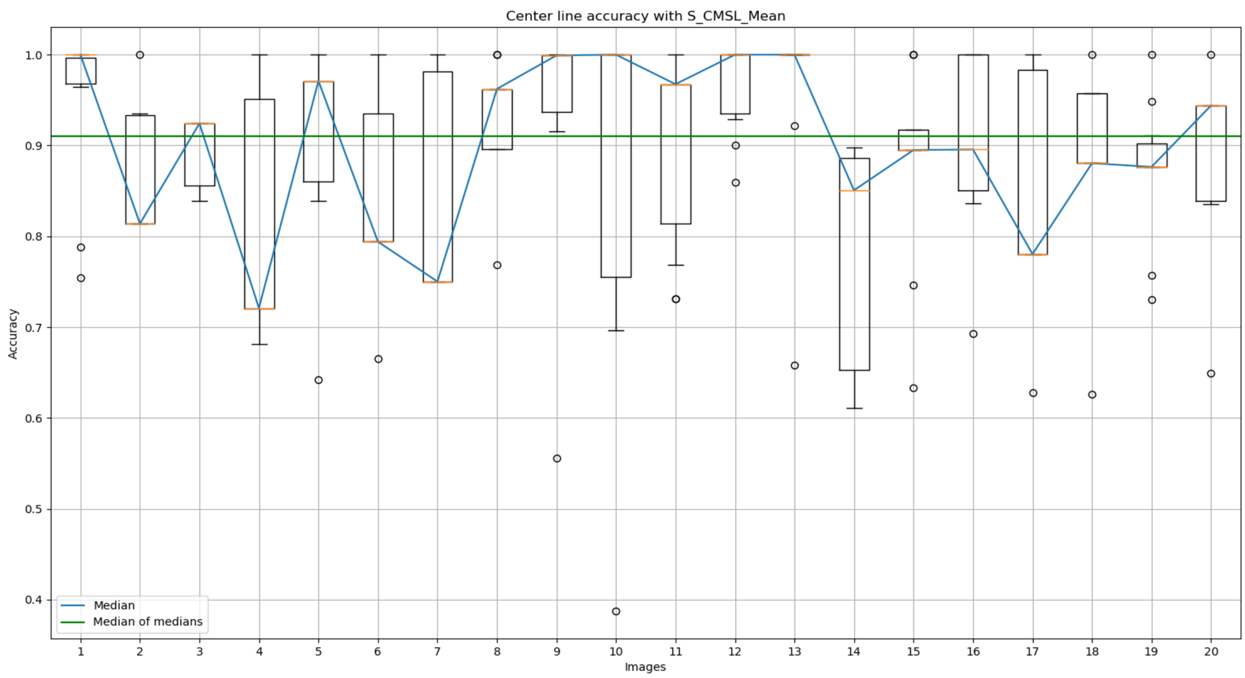


(21)

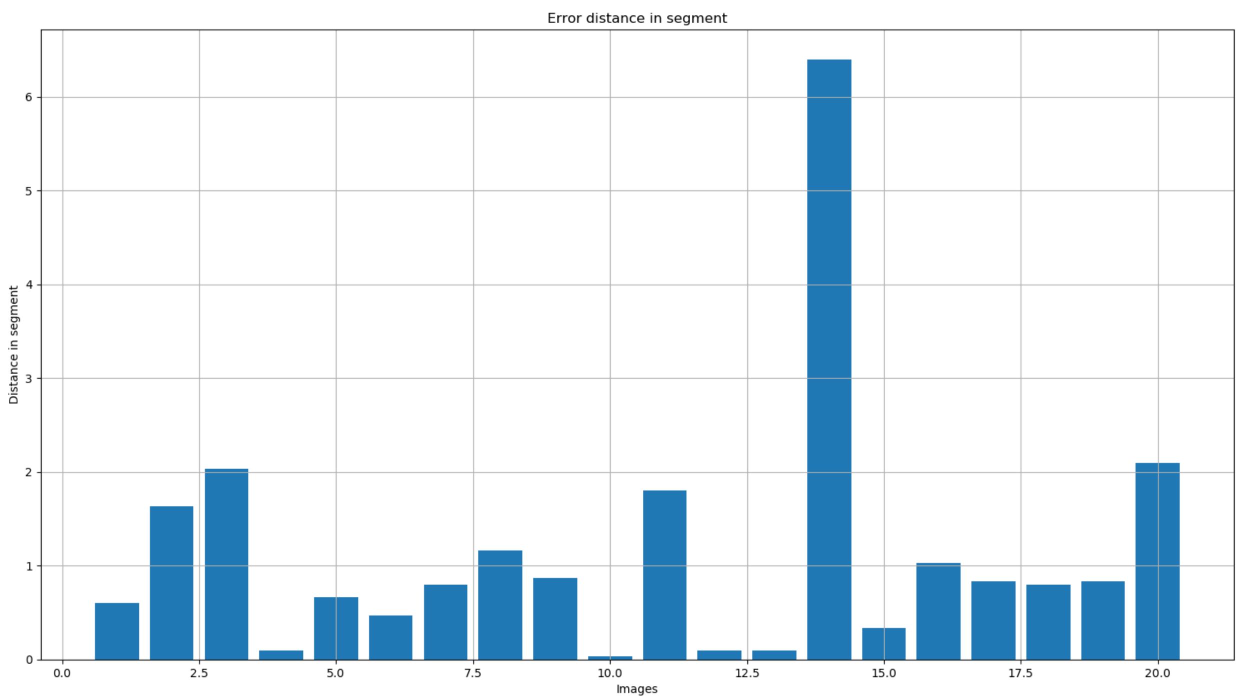
b. S_CMSL_Mean:



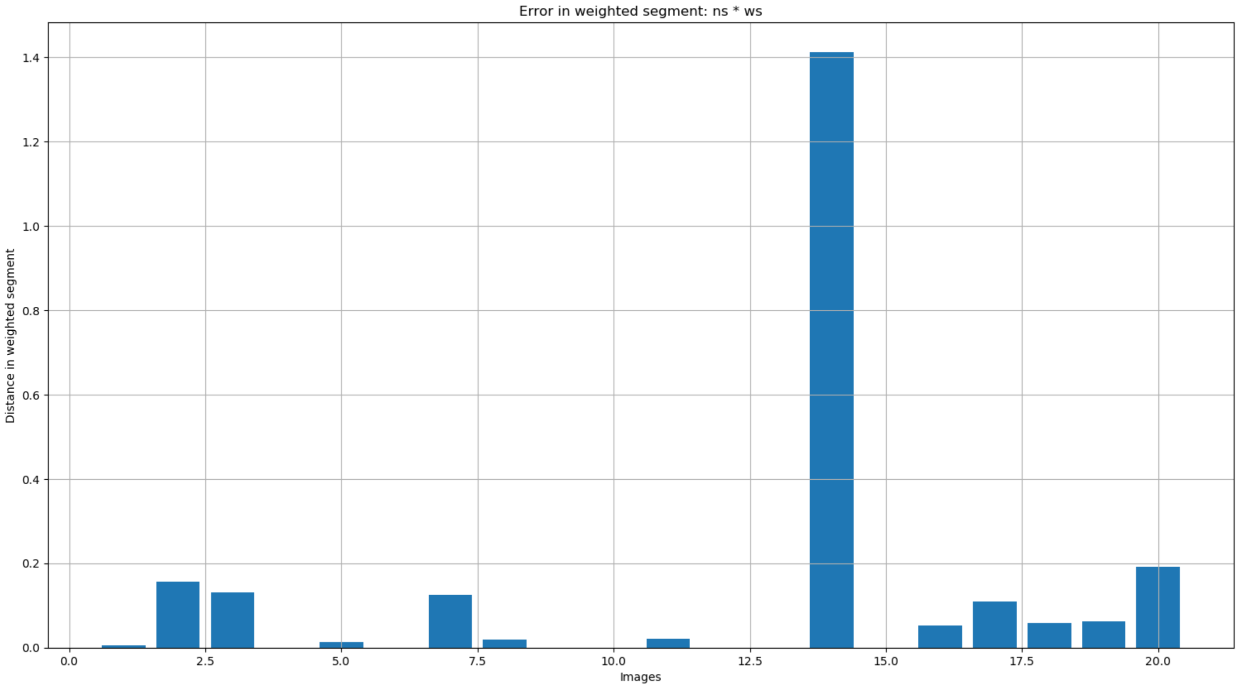
(22)


Center line accuracy with S_CMSL_Mean
Accuracy
Images
Median
Median of medians
1.0
0.9
0.8
0.7
0.6
0.5
0.4
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

(23)

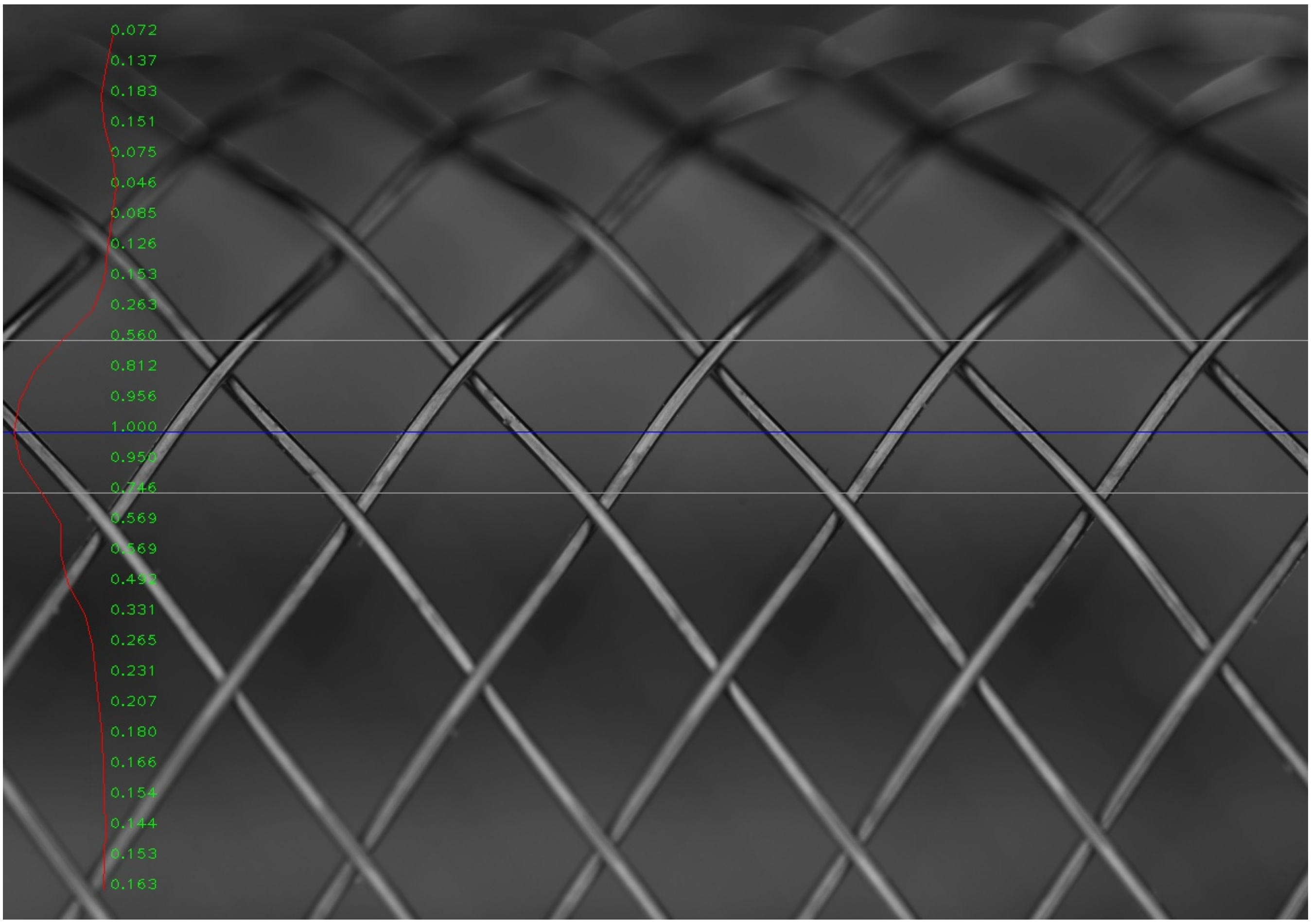

Error distance in segment
Distance in segment
Images
6
5
4
3
2
1
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0

(24)

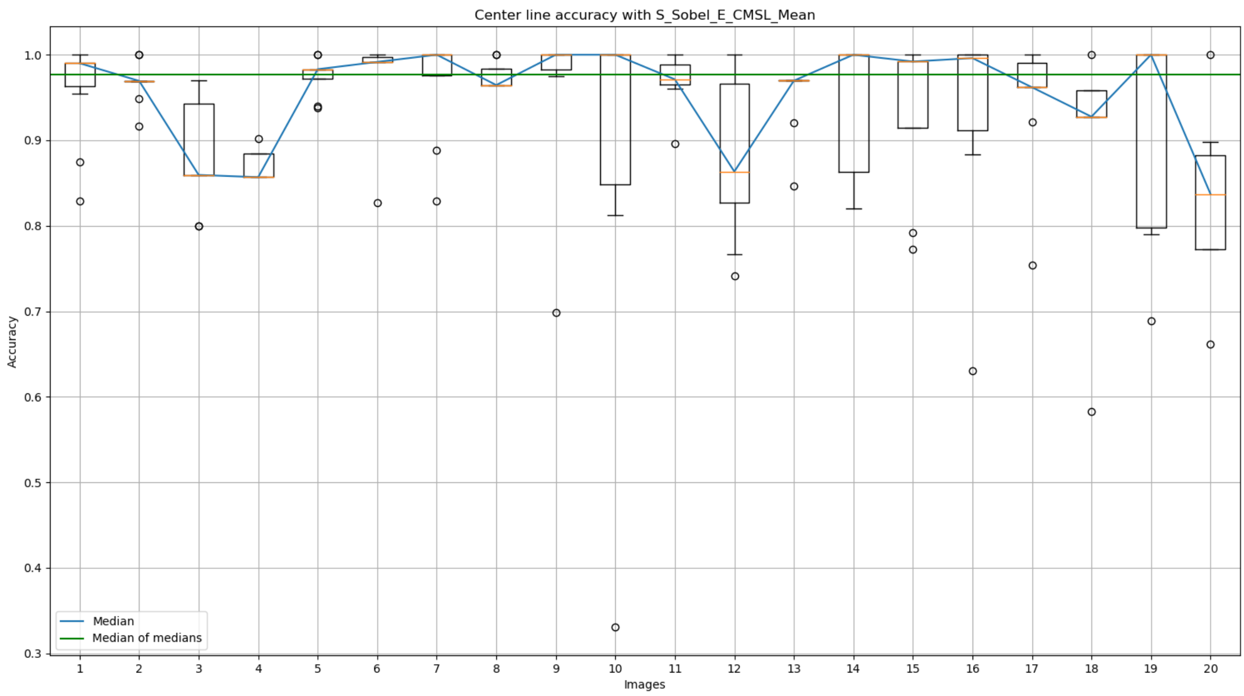


(25)

c. S_Sobel_E_CMSL_Mean:

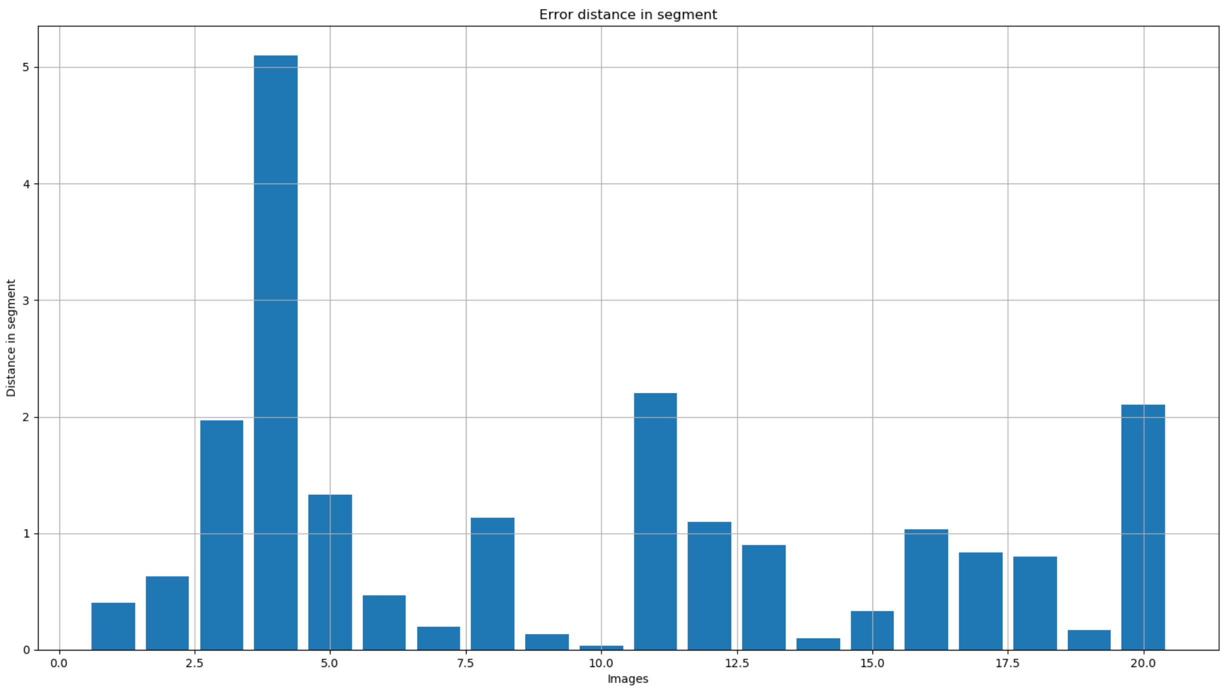


(26)



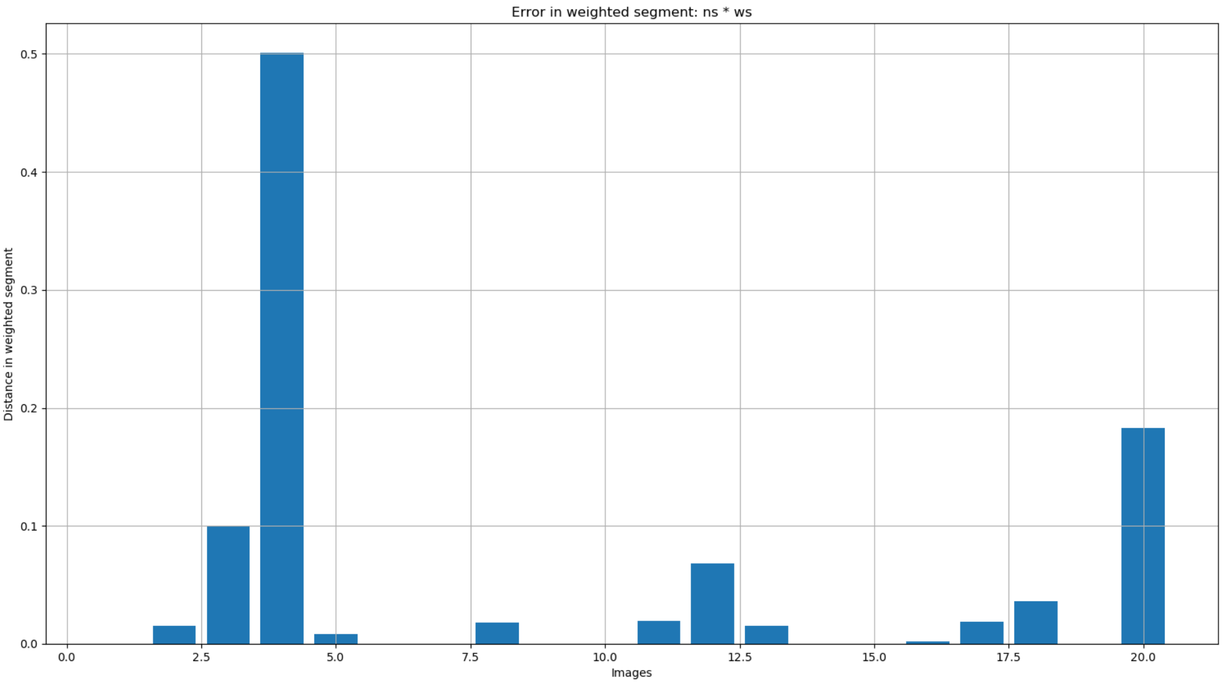
Center line accuracy with S_Sobel_E_CMSL_Mean
Accuracy
Images
Median
Median of medians

(27)



Error distance in segment
Distance in segment
Images

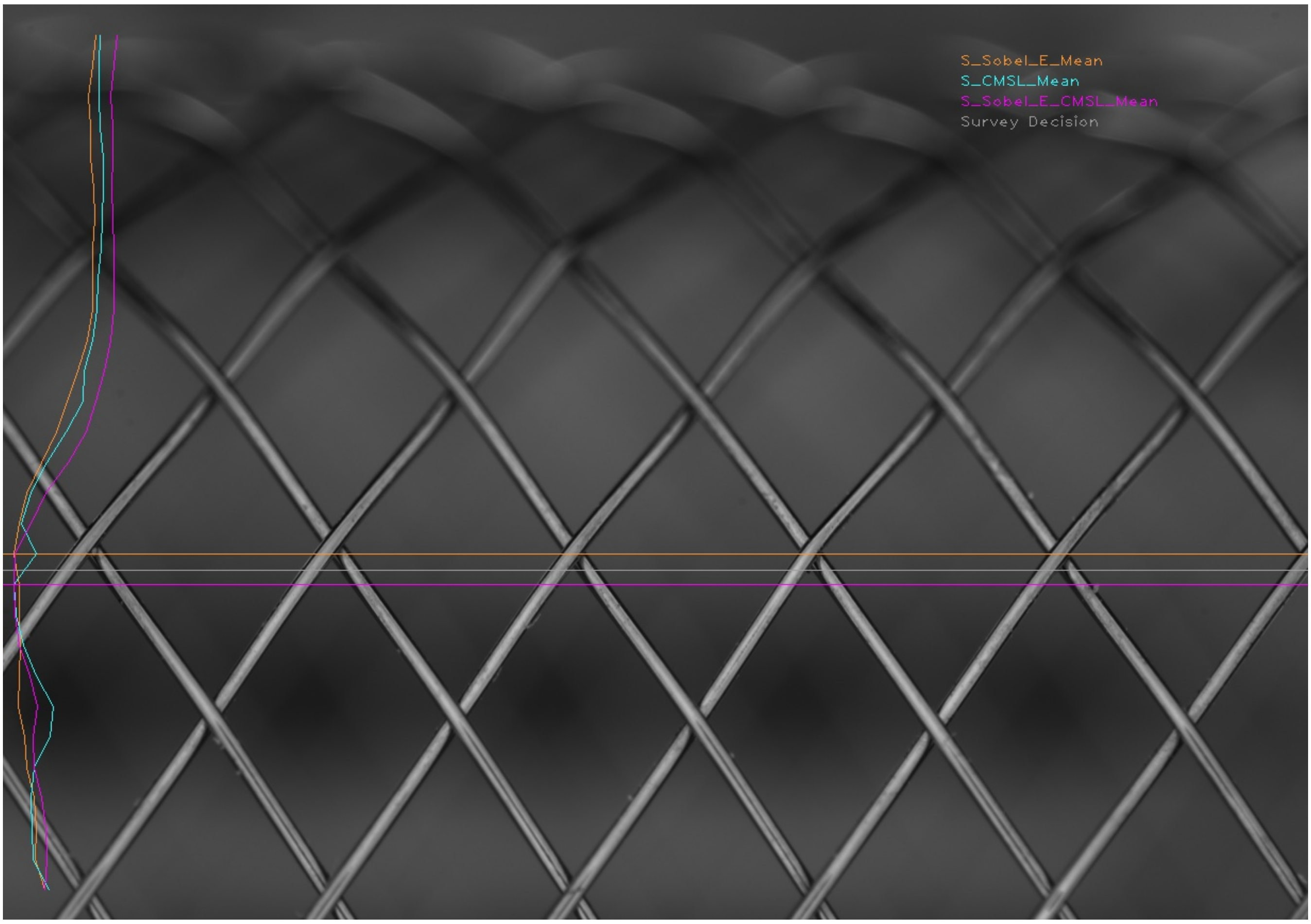
(28)



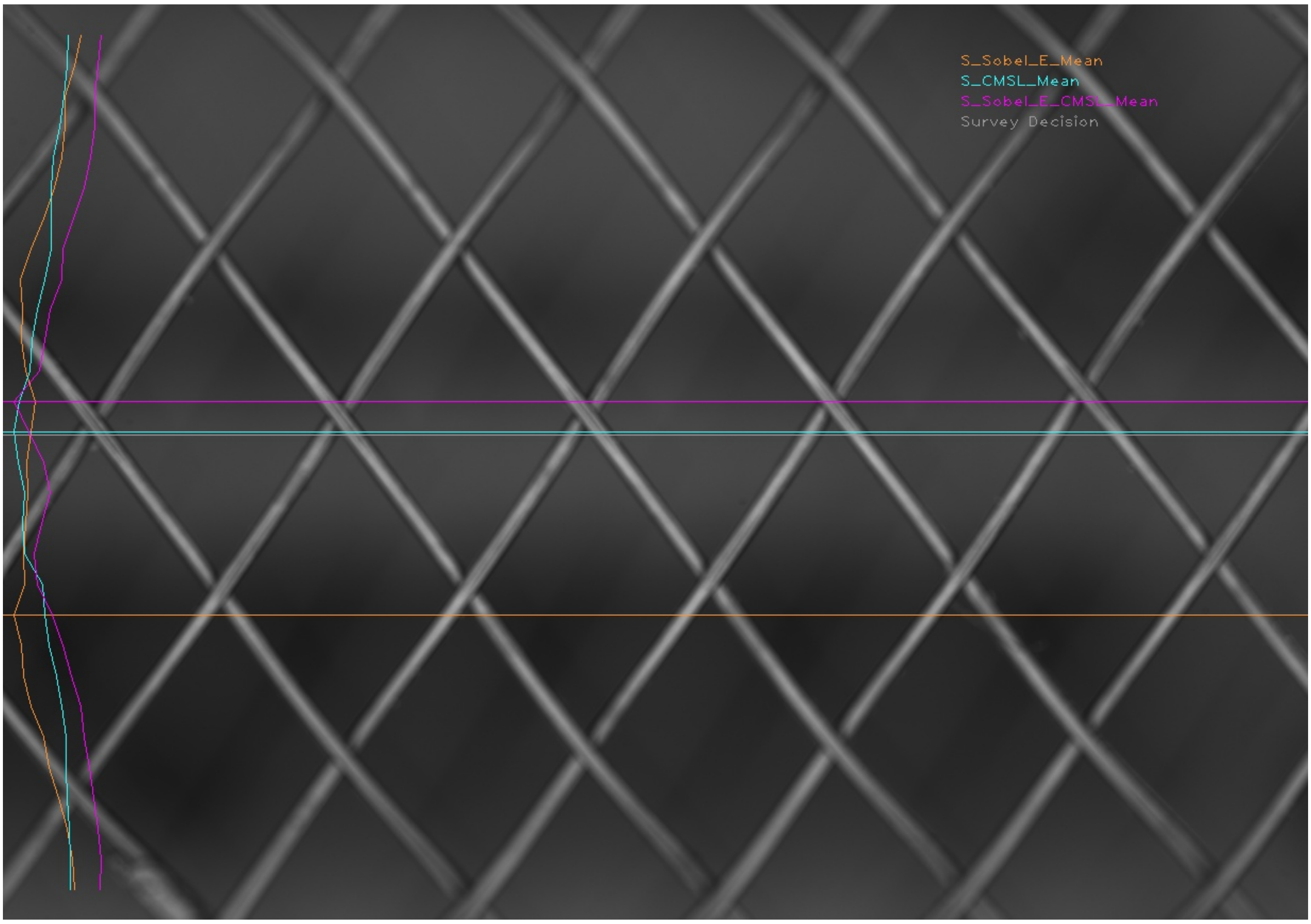
Error in weighted segment: ns * ws
Distance in weighted segment
0.0
0.1
0.2
0.3
0.4
0.5
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Images

(29)

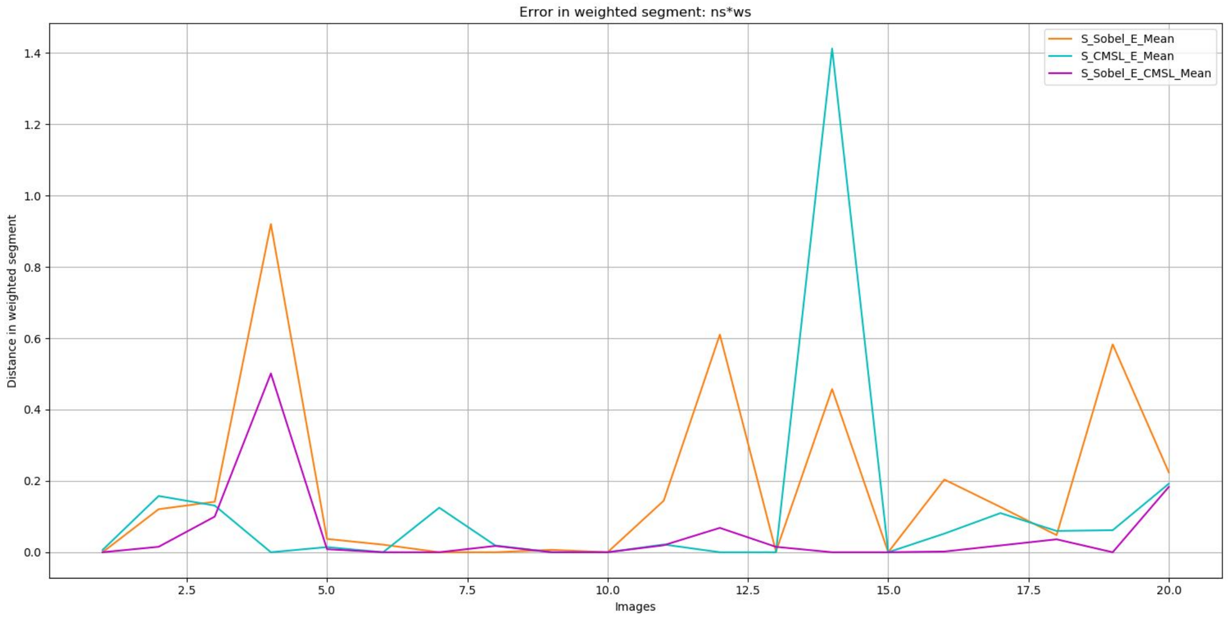
Survey image 6:



Survey image 12:



Error in weighted segments for the three algorithms over the 20 survey images:



(32)

The S_Sobel_E_CMSL_Mean shows a significant overall performance compared to the two other solutions. So we run it on a larger sample of images and try to detect problems if there are some.

The first problem that we faced is related to the use of the edge pixels in the calculation of the focus rate: Some unsharp segments has "hot-spot pixels". So, in that segment, we have a very small number of edge pixels but with a very high gradient magnitude $\rightarrow$ Important focus rate.

(33)

As shown in (33), the 1st and the 9th segments are unsharp. However, because of some hot-spots, they are considered as the segments with the two highest focus rates. In the next section, we are going to develop and discuss some solutions for this problem.

5. Improvement of S_Sobel_E_CMSL_Mean:

1st solution:

S_CMSL_Mean is stable against the hot-spots in the unsharp segment because it does not use the edge pixels in its calculation, but it is unstable in the segments were the wire intersections are. A solution can be a combination between S_CMSL_Mean and S_Sobel_E_CMSL_Mean, were S_CMSL_Mean will be applied on the whole picture and the S_Sobel_E_CMSL_Mean only on the sharp region as a correction for the segments were the wires intersections exist.

The challenge is to define the sharp region after S_CMSL_Mean.

2nd Solution :

We observe the morphology of the image and use the Opening and Closing (Erosion and Dilation) to remove the small pixel islands in the fore- and background.

Closing:

$$A \cdot B = (A \oplus B) \ominus B$$

Opening:

$$A \cdot B = (A \ominus B) \oplus B$$

$where \oplus denote$ the dilation and $with \ominus the$ erosion. A is the input image and B is a structuring element.

The dilation operation uses a structuring element for probing and expanding the shapes contained in the input image. The erosion operation, however, for probing and reducing.

Let E be a Euclidean space or an integer grid where A is well defined.

Erosion:

$$A \ominus B = \{z \in E | B_z \subseteq A\}$$

where B_z is the translation of B by the vector z, i.e., $B_z = \{b + z | b \in B\}, \forall z \in E$

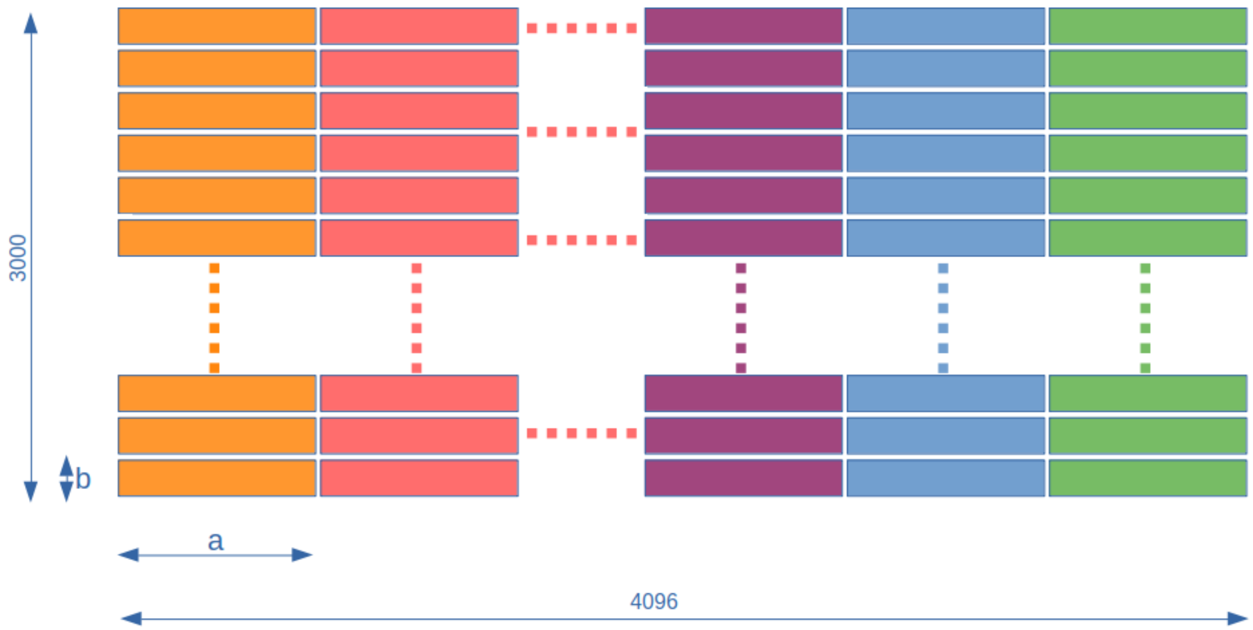
Dilation:

$$A \oplus B = {}_{b \in E} U\ A_b$$

where A_b is a translation of A by b.

3rd solution:

Instead of dividing the image in only horizontal segments, we can divided horizontally and vertically as shown in the example bellow:



(34)

For an image with size (3000 x 4096), we divided it into vertical segments (3000 x a) and then we apply S_Sobel_E_CMSL_Mean on each vertical segment by dividing it into horizontal segment with size (b x a).

Let F_{xy} be the focus rate in the x-vertical segment and the y-horizontal segment, v is the number of vertical segments = 4096/a and h the number of the horizontal segments = 3000/b.

For each segment S (b x 4096) the focus rate F_S is:

$$F_S = Median(F_{1S}, F_{2S}, \dots, F_{VS})$$