Report for Exercise 4 and 5

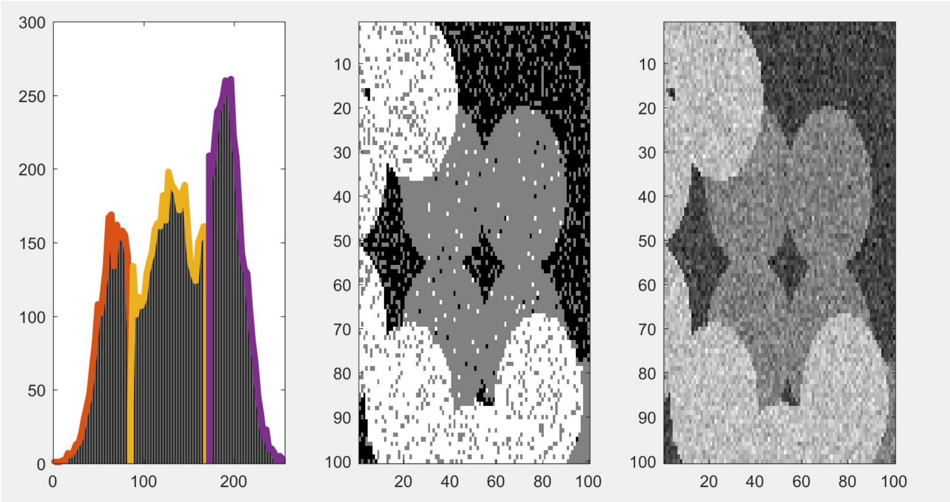
Gruppe
Phillip Brinck Vetter s144097
Jacob Jon Hansen s134097

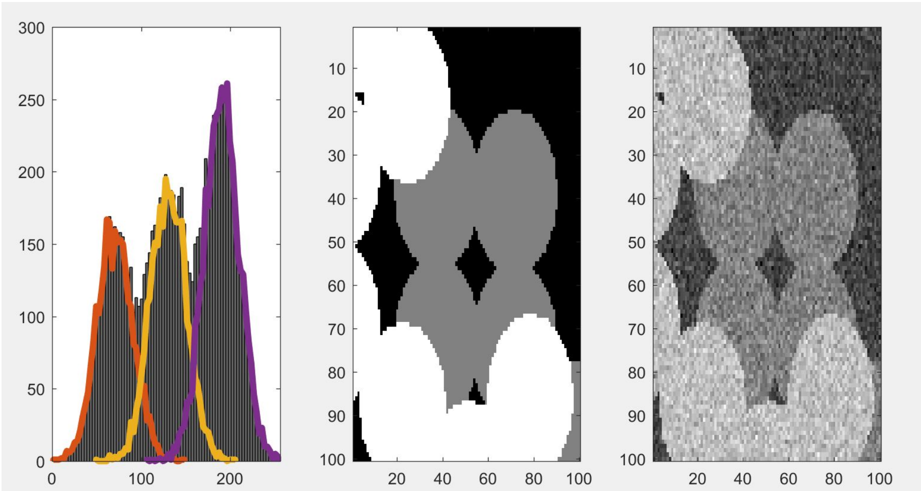
Markov Random Fields

We make a framework, which is able to encode prior information into the segmentation task. This is done by minimizing the energy of a configuration.
Likelihood = one clique potentials (look at the data), we can find a minimum in this regard. (likelihood of data belonging to F1 or F2 just from looking at the raw data)
Prior = Contextual prior that we define for 2-cliques, defines the likelihood of different configutarion.

Single pixel: We can find the likelihood of F1,F2 on each pixel alone based on some metric
Neigbourhood: We can define the cost of different configurations for a neighbourhood of pixel from our prior knowledge of the problem.

For the gender example:
Modelling: Likelihood (average of male and female height), cost of assigning different people to each class
Prior: The cost of different configurations for the 2-cliques

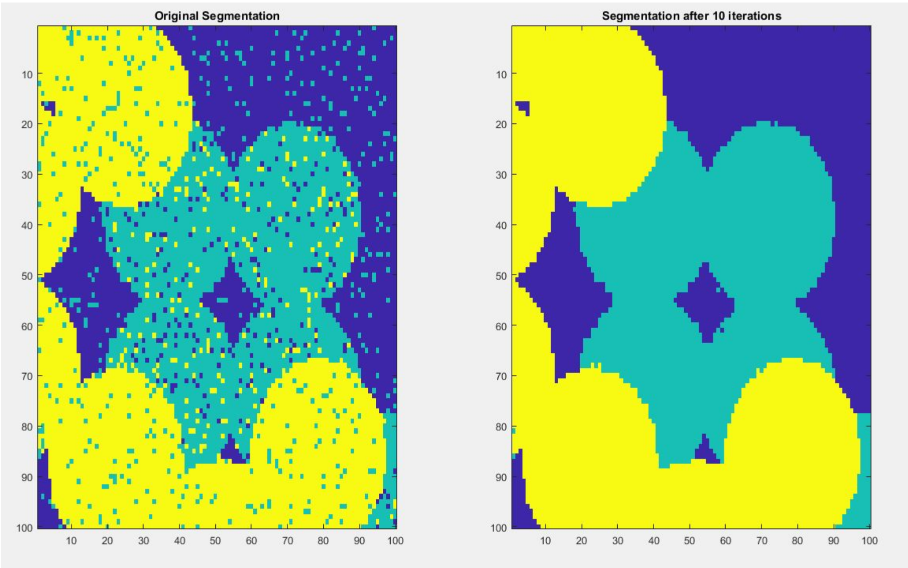




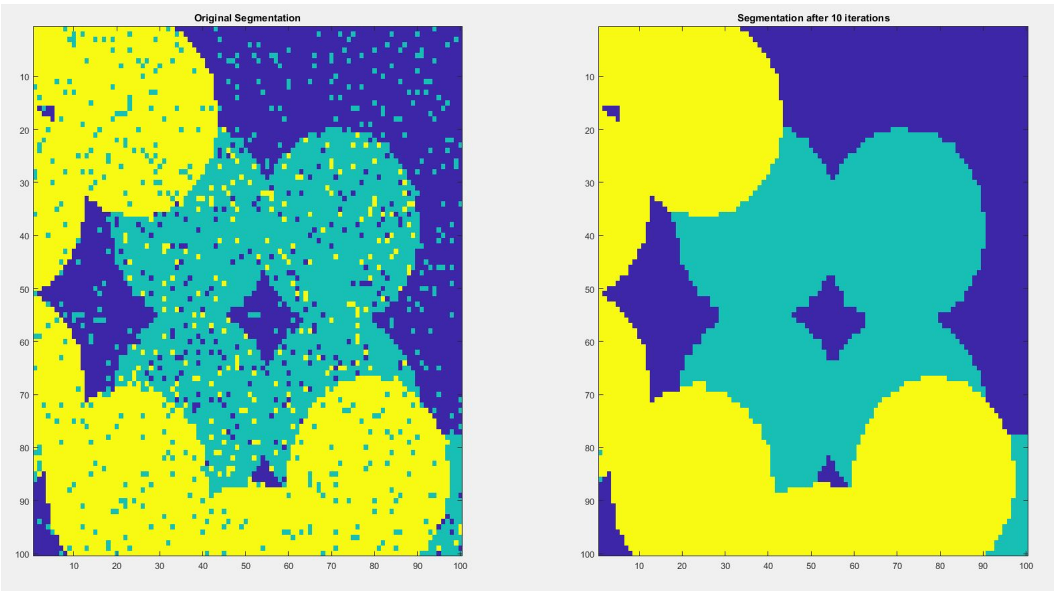
Above we see the resulting histogram of the two segmentations. One is thresholded and we see that adjacent pixels can easily be of different categories where the MRM segmentation punishes differences in neighbours therefore a more smooth segmentation is achieved.

Iterative Optimization:

Find out which label is optimal for every pixel in the image using energies and neighbouring pixels, try every label for every pixel and see what works best locally.
Update them all or update serially or update in pattern.
Find label energies again and repeat



Parallel updates: We will experience switching of pixels and it has a hard time of converging



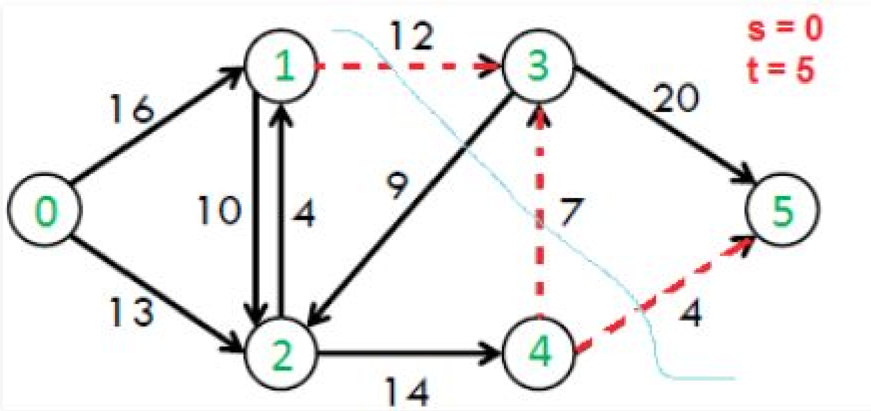
Non parallel updates: Update in checkerboard pattern, this has an easier time converging.

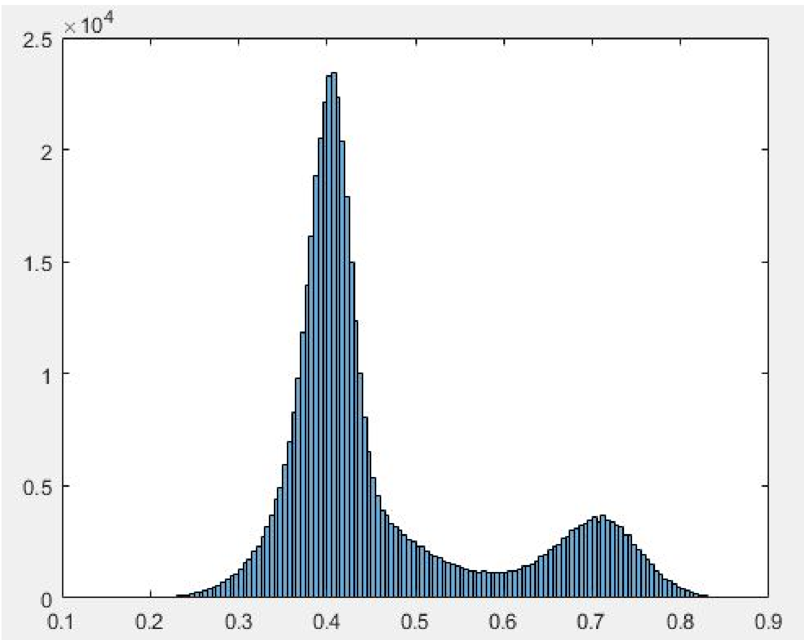
```
beta =

   100

Person 1 is estimated as M
Person 2 is estimated as M
Person 3 is estimated as M
Person 4 is estimated as F
Person 5 is estimated as F
Person 6 is estimated as F
```

```
beta =

    10

Person 1 is estimated as M
Person 2 is estimated as M
Person 3 is estimated as M
Person 4 is estimated as F
Person 5 is estimated as M
Person 6 is estimated as F
```

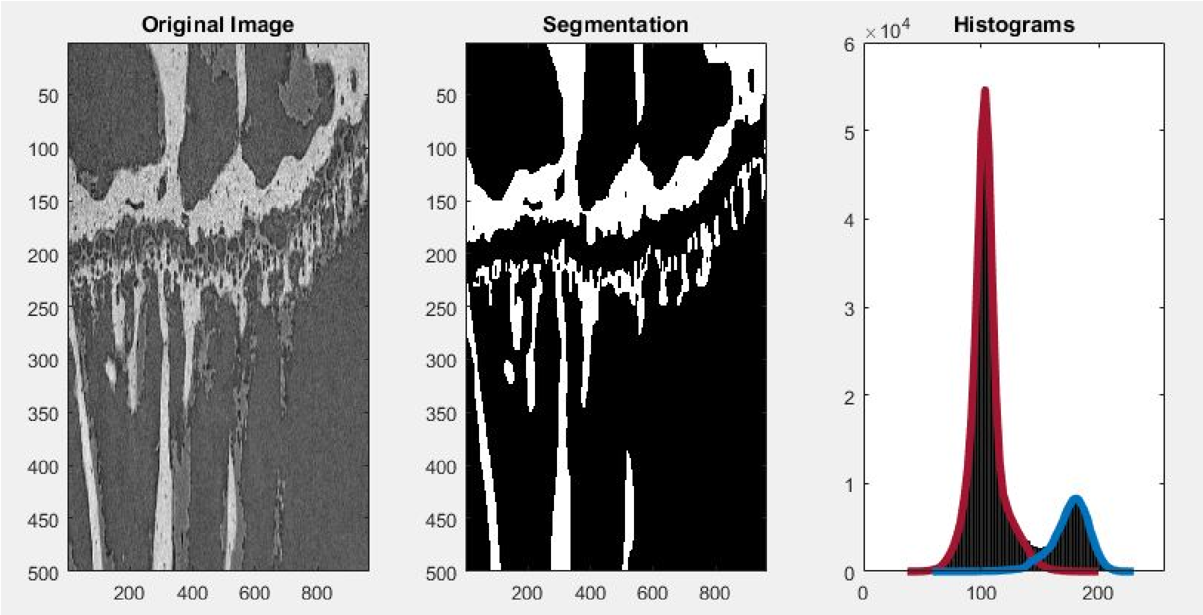
```
beta =

        1000

Person 1 is estimated as F
Person 2 is estimated as F
Person 3 is estimated as F
Person 4 is estimated as F
Person 5 is estimated as F
Person 6 is estimated as F
```

For different beta we get different optimal configurations with the graph-cut algorithm

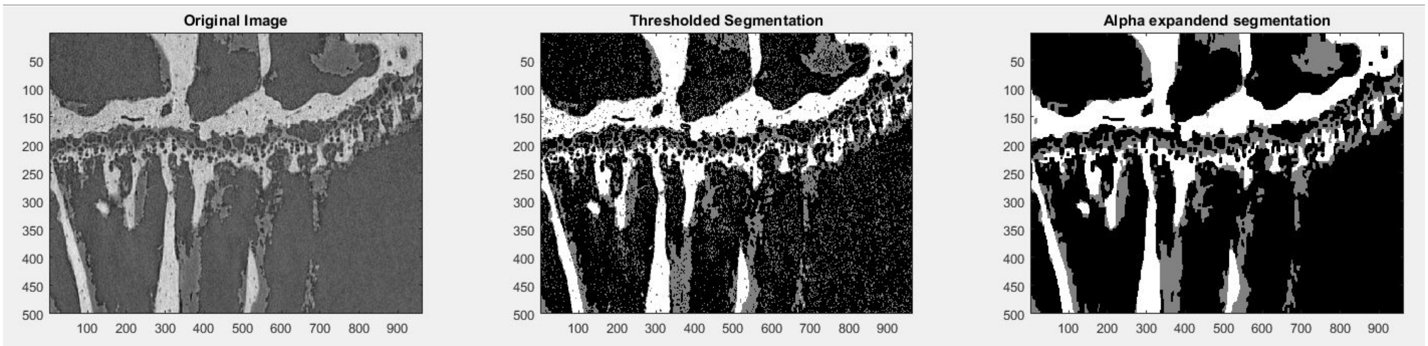




Bone image histogram, two classes has mean of 0.7 and 0.4



We construct the terminal and internal matrices and feed it to the graph cut (beta = 0.1) to get a good segmentation. The historgram of all the pixels are produced and then we use the segmentation as a mask on the image to produce the histograms shown in red and blue. We see that we get a really nice fit on the two classes and have removed most of the noise in the image, exploiting the prior knowledge of neighbours.



Here we initially threshold the image, then find the energies of the pixels on the segmented image. This is fed into the graph cut algorithm with beta = 4 to remove noise. We get a really good segmentation in the end.

Deformable Models

Crawling Amoeba

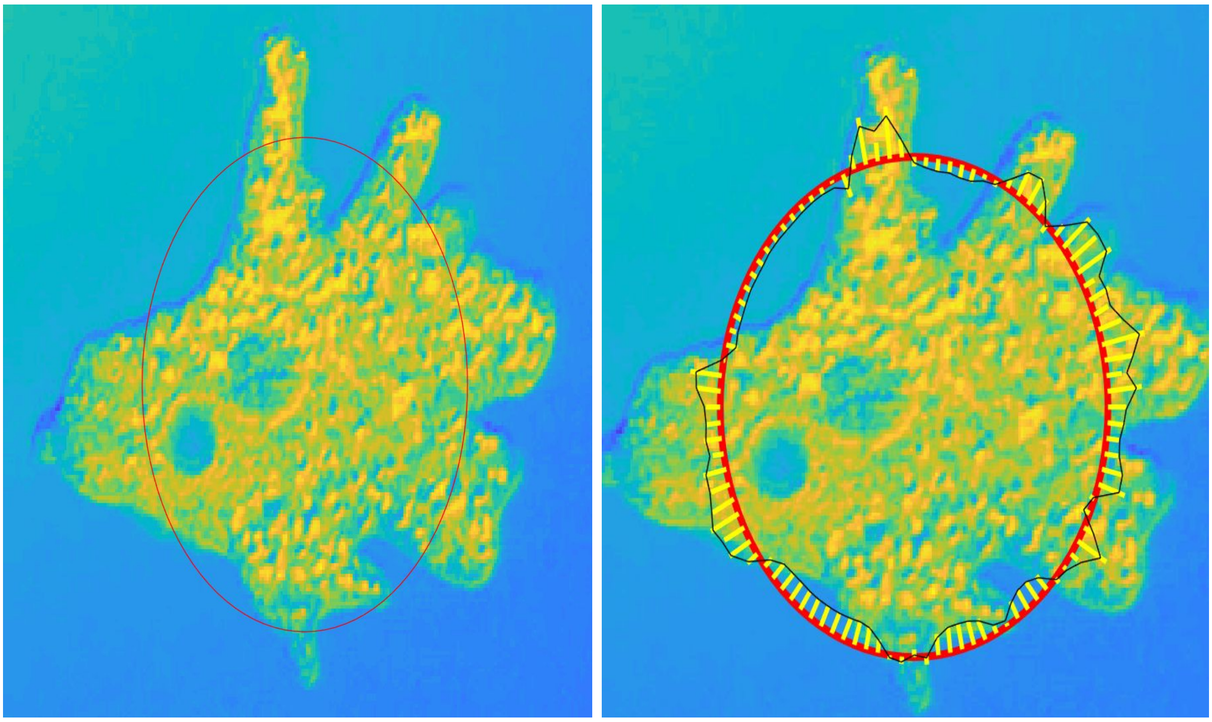


Figure 1: (Left) Initialization of the Snake at t = 0. (Right) The normal vectors of the points on the snake in which direction a force is applied. The force is applied in such a direction that the 2-norm of the energy in the Mumford-Shah framework minimizes.

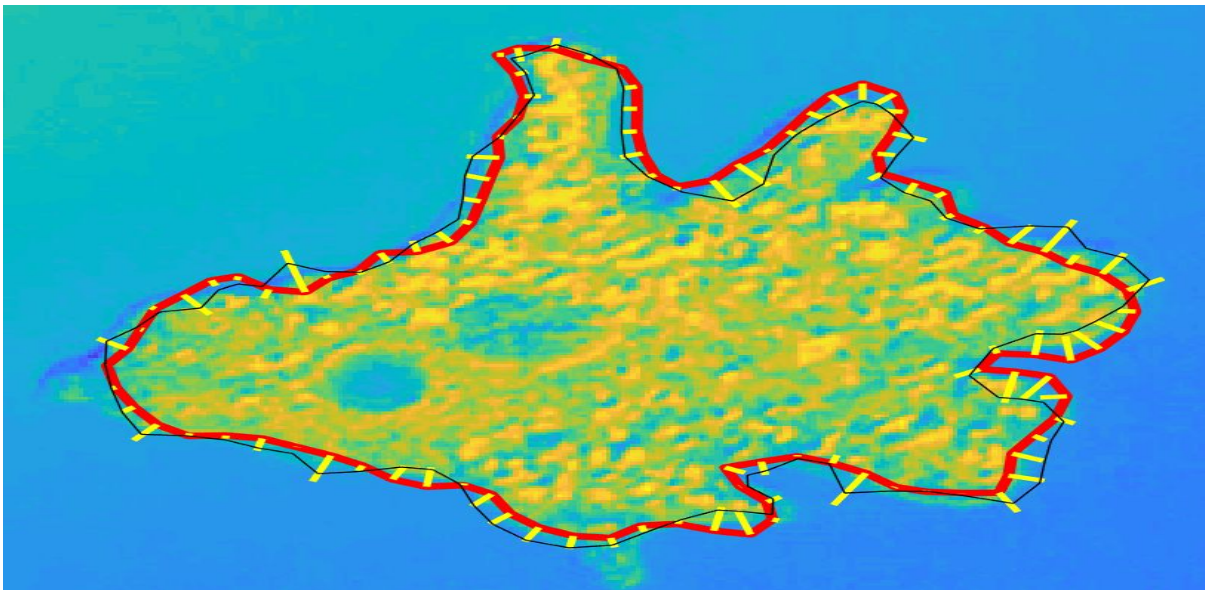


Figure 2: The curve (red) at any given time point and the curve (black) at the next time iterate. The position of the curve in the next iterate is regularized through the parameters α and β such that instead of the points moving to the end of the normal force-vectors, the points move onto the black curve.

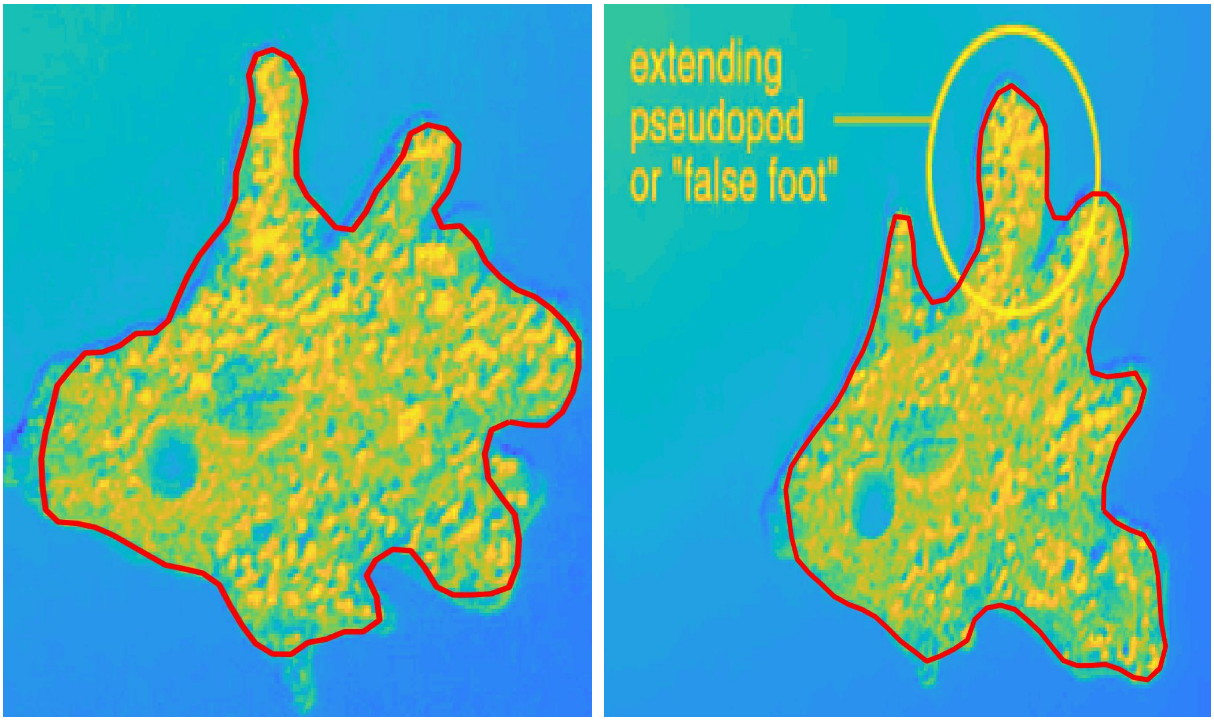


Figure 3: Frames of the movie at different times showing that the developed deformable snake adjusts to the amoeba as it moves around.

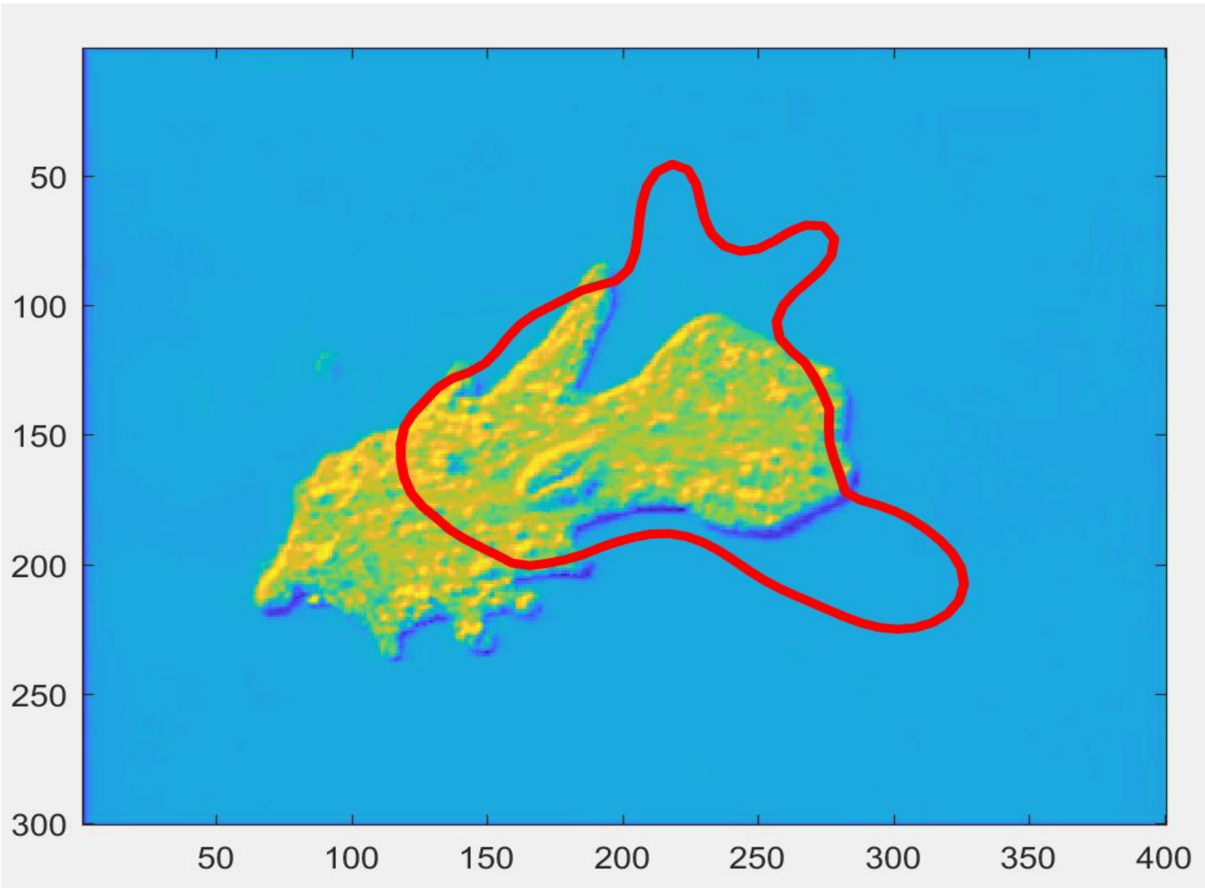


Figure 4: The amoeba suddenly changes shape at a point in the clip. The frame here captures the model at a point where it has not yet adapted to the new shape of the amoeba. The model is shown in **Figure 5** to adapt soon after this event has occured.

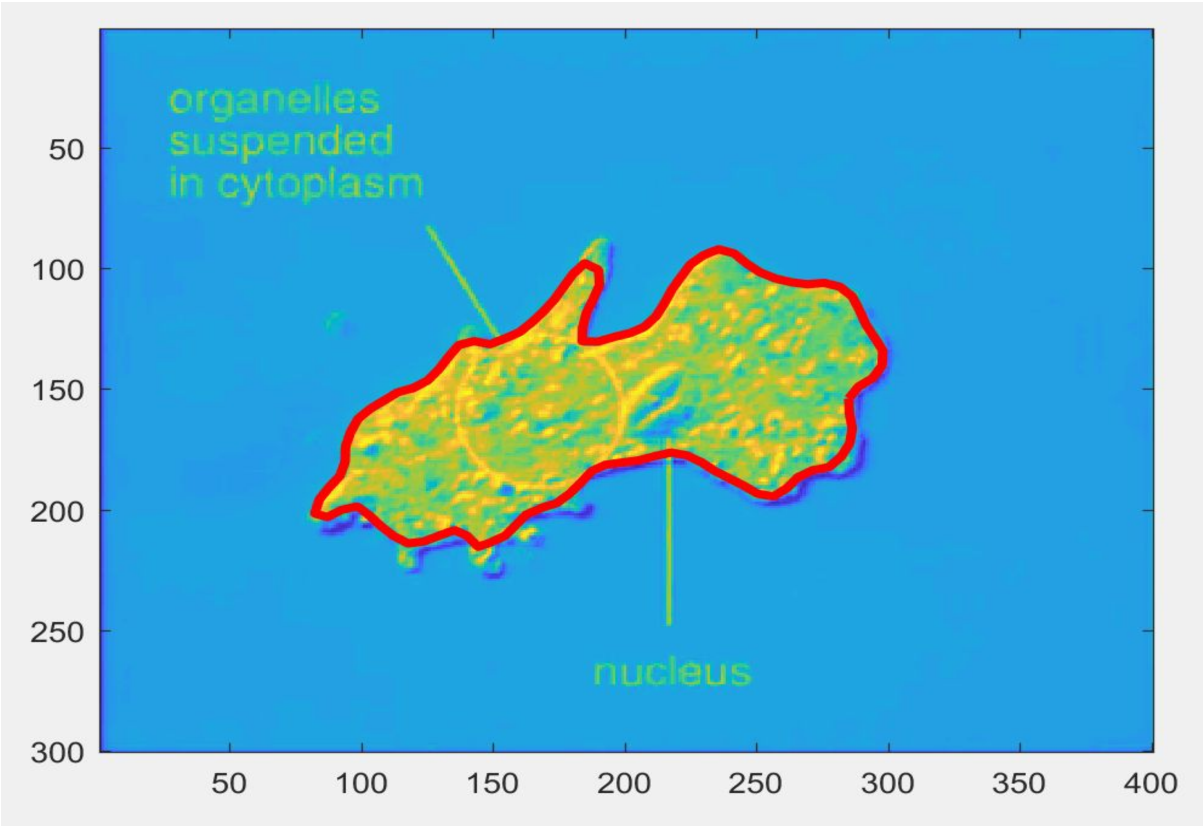


Figure 5: Model has successfully adapted to the new shape of the amoeba.

Figure 6: Last image of the clip. The model attempts to adjust to the shape of the “Garland” and “Taylor & Francis Group” letters.

Echnincsus:

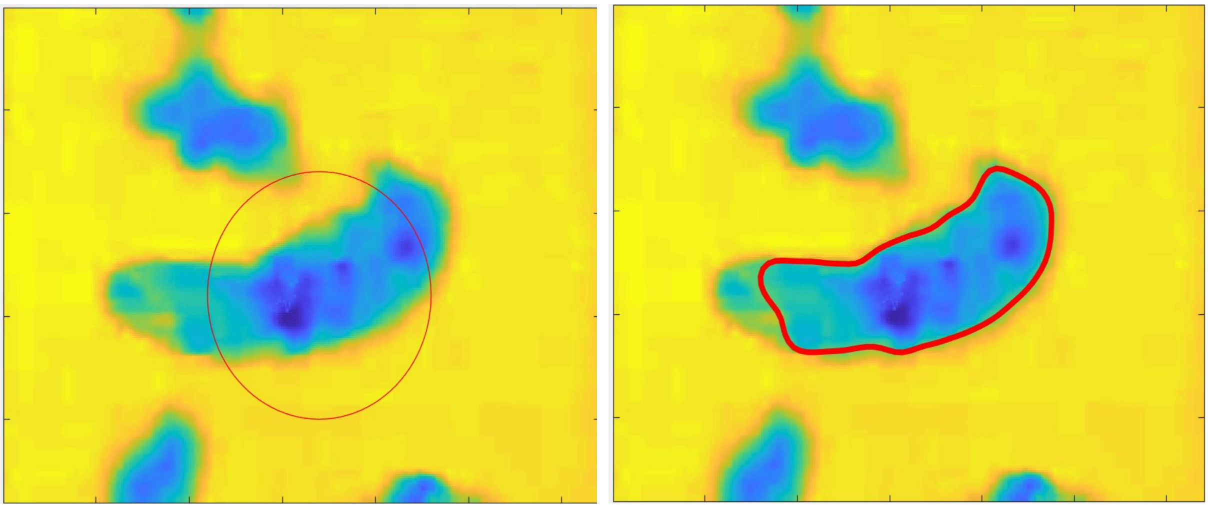


Figure 7: (Left) Initialization of the model. (Right) Model has adapted to the shape of the Echnincsus.

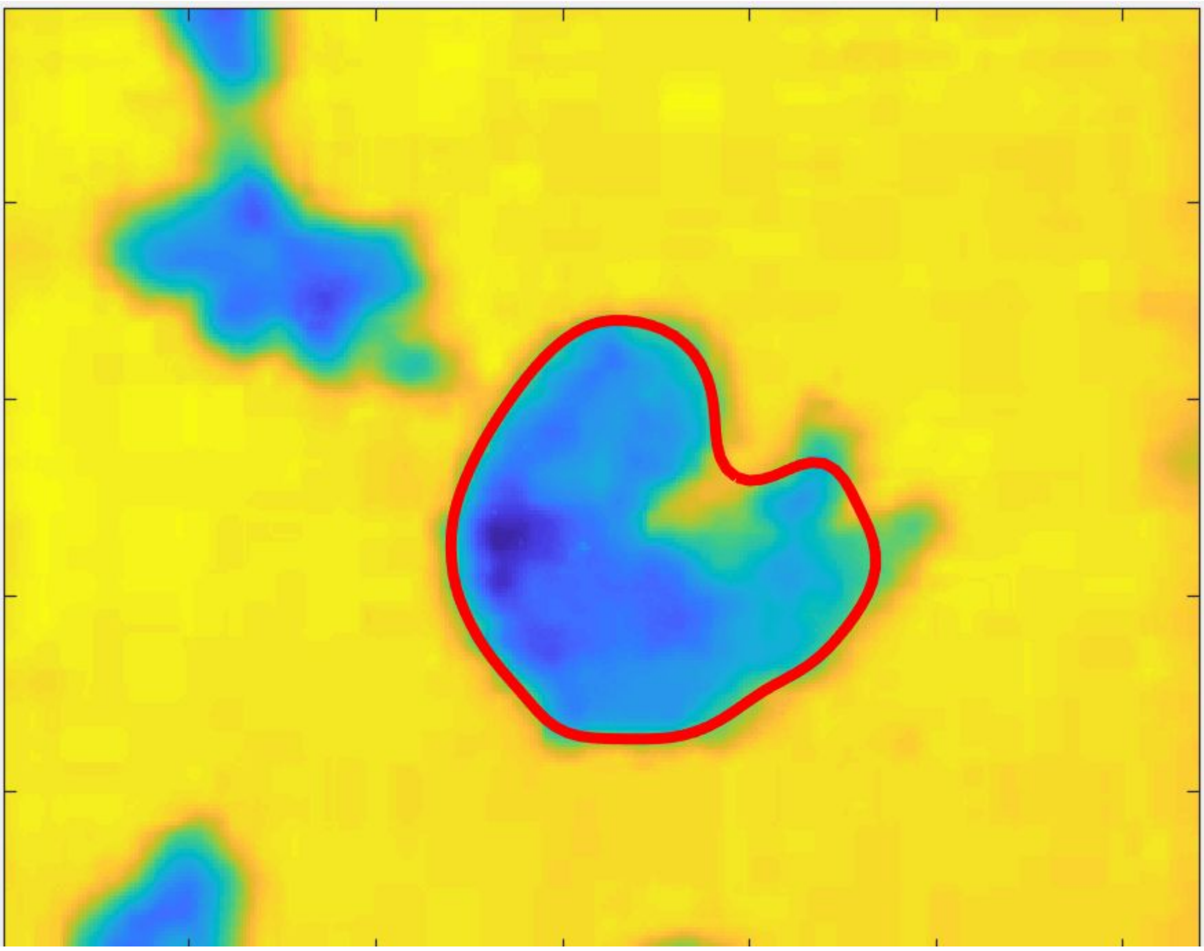


Figure 8: Model remains adapted to the shape of the Echnincsus throughout the movie clip.

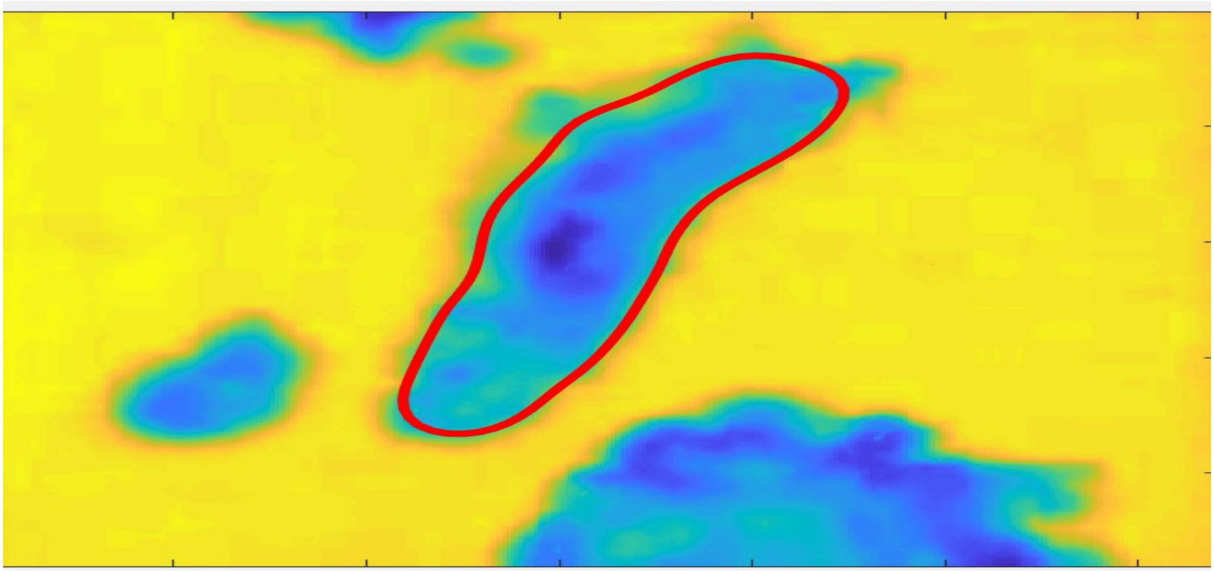


Figure 9: Model remains adapted to the shape of the Echnincsus throughout the movie clip.

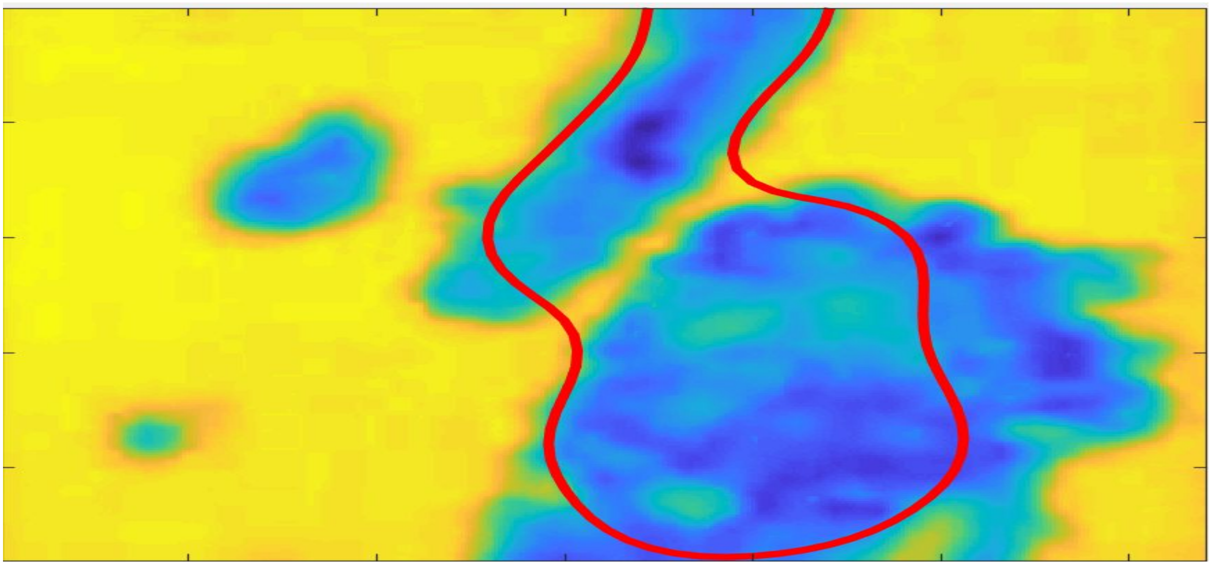


Figure 10: The shape of the model is broken as the Echninscus connects itself to another organism.

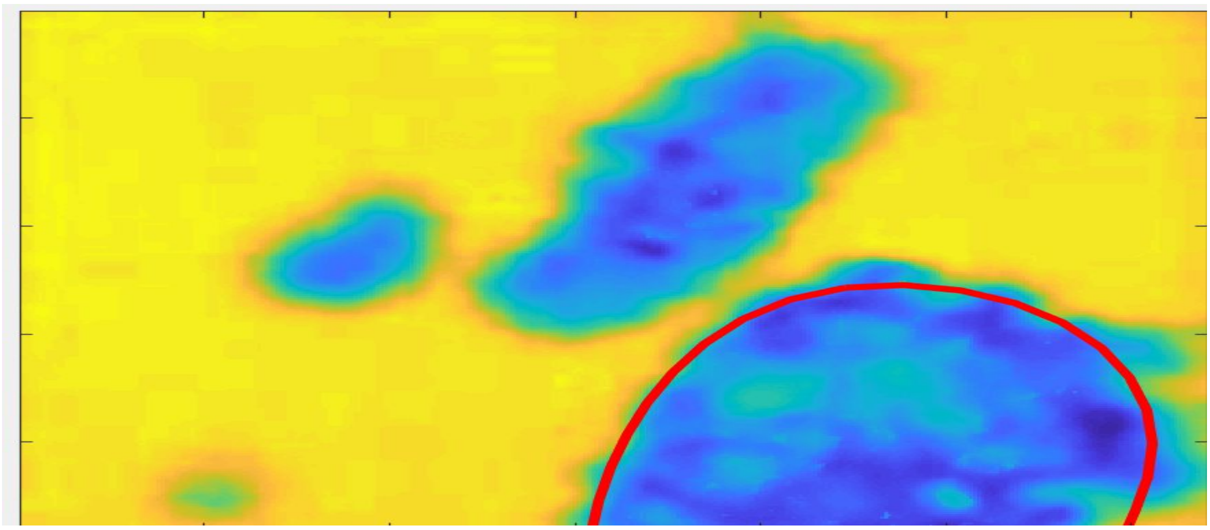


Figure 11: The shape disconnects from the organism that was originally tracked. It remains *stuck* at the bottom organism throughout the rest of the clip.

Nerves:

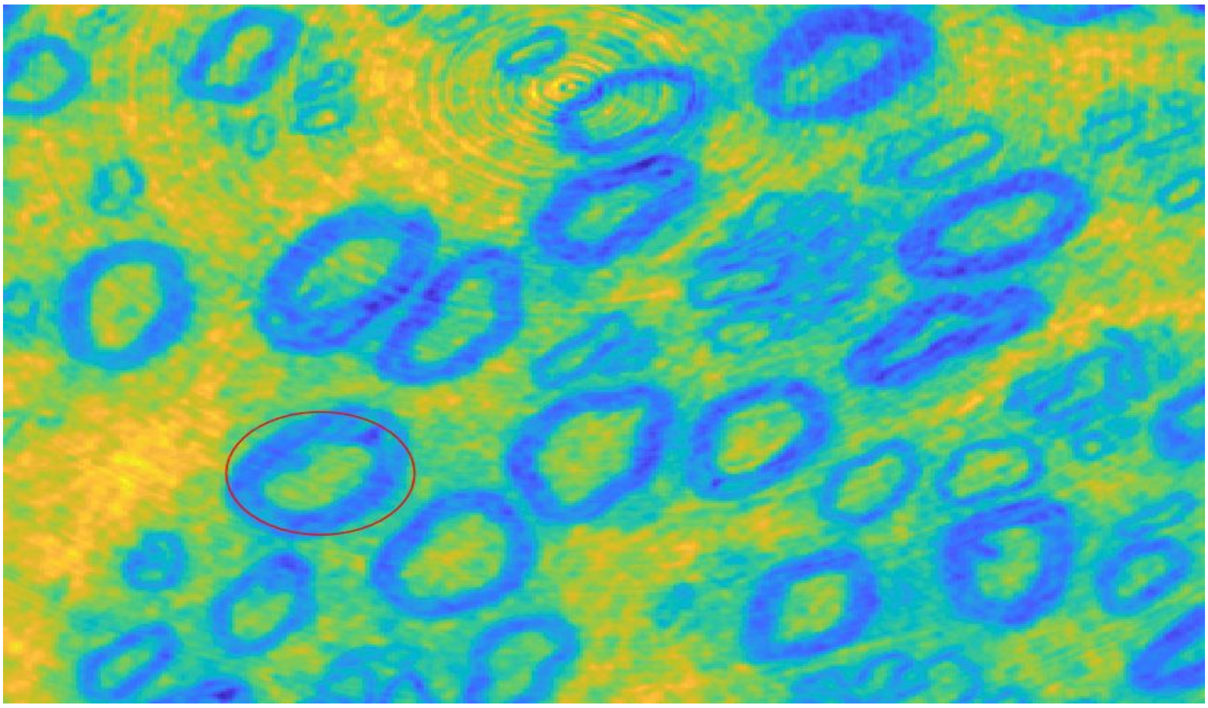


Figure 12: Initialization of the snake curve.

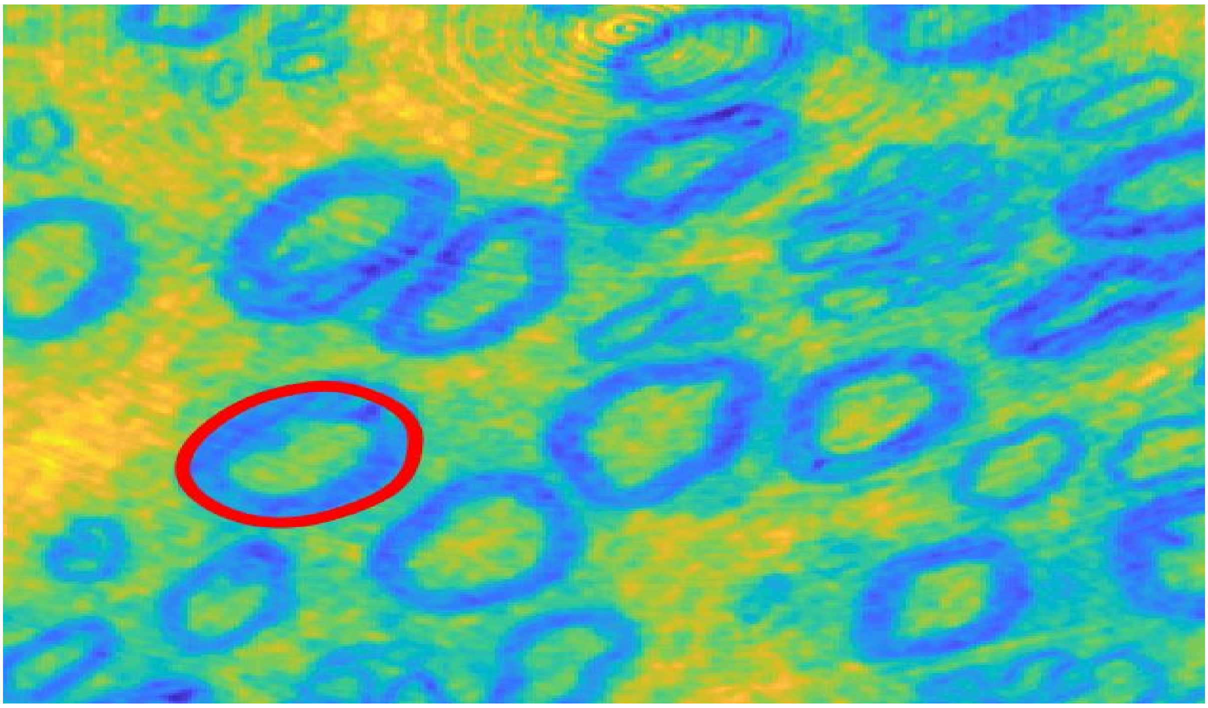


Figure 13: Tracking of the nerve cells in the image.

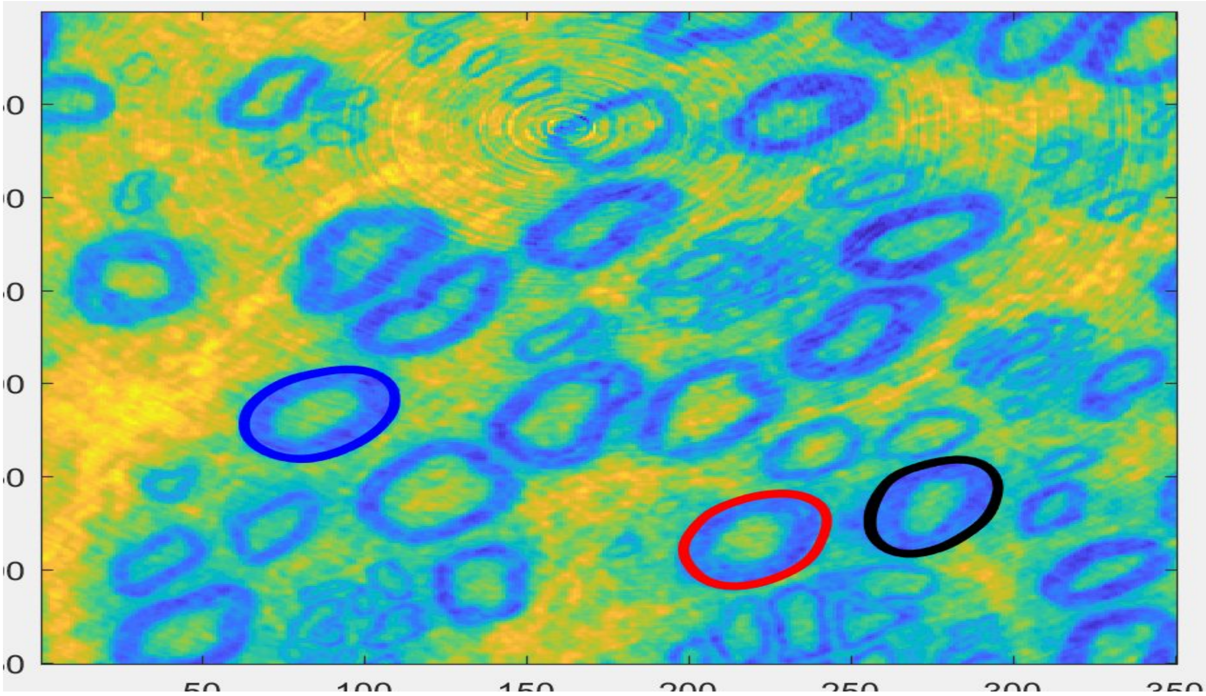


Figure 14: Tracking of the multiples nerve cells in the image by initialization various snakes.

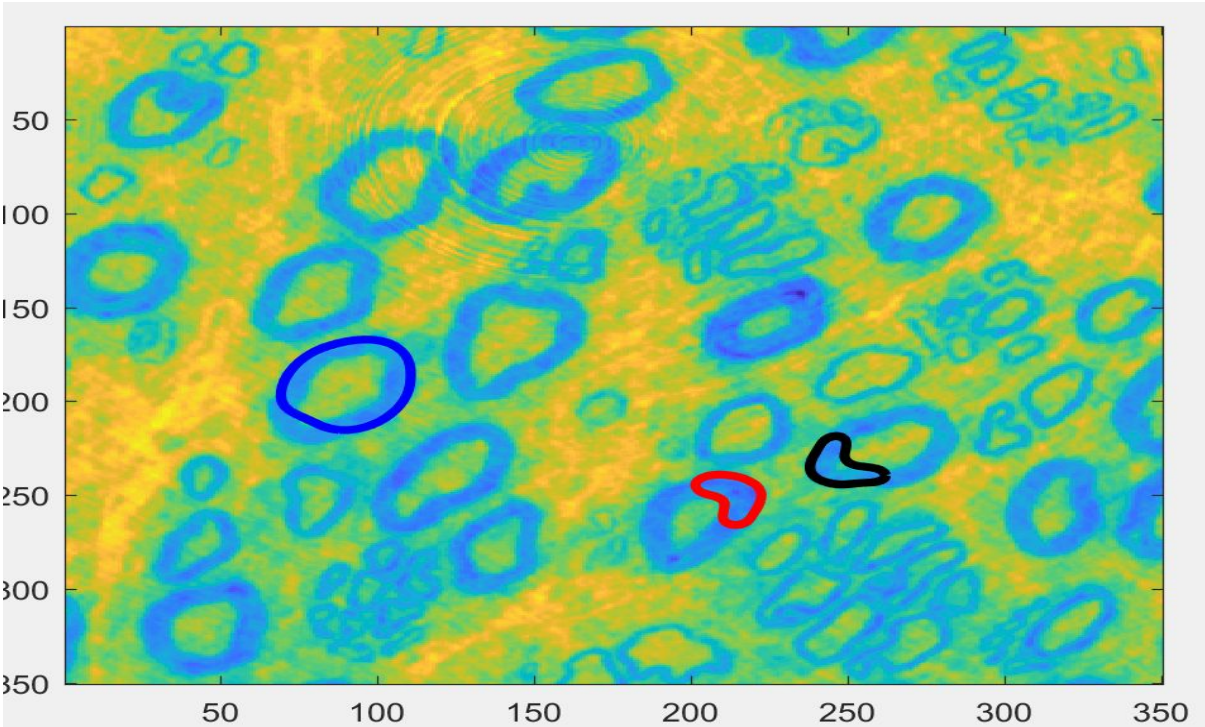


Figure 15: Tracking of the multiples nerve cells in the image fails towards the end of the sequence of slices

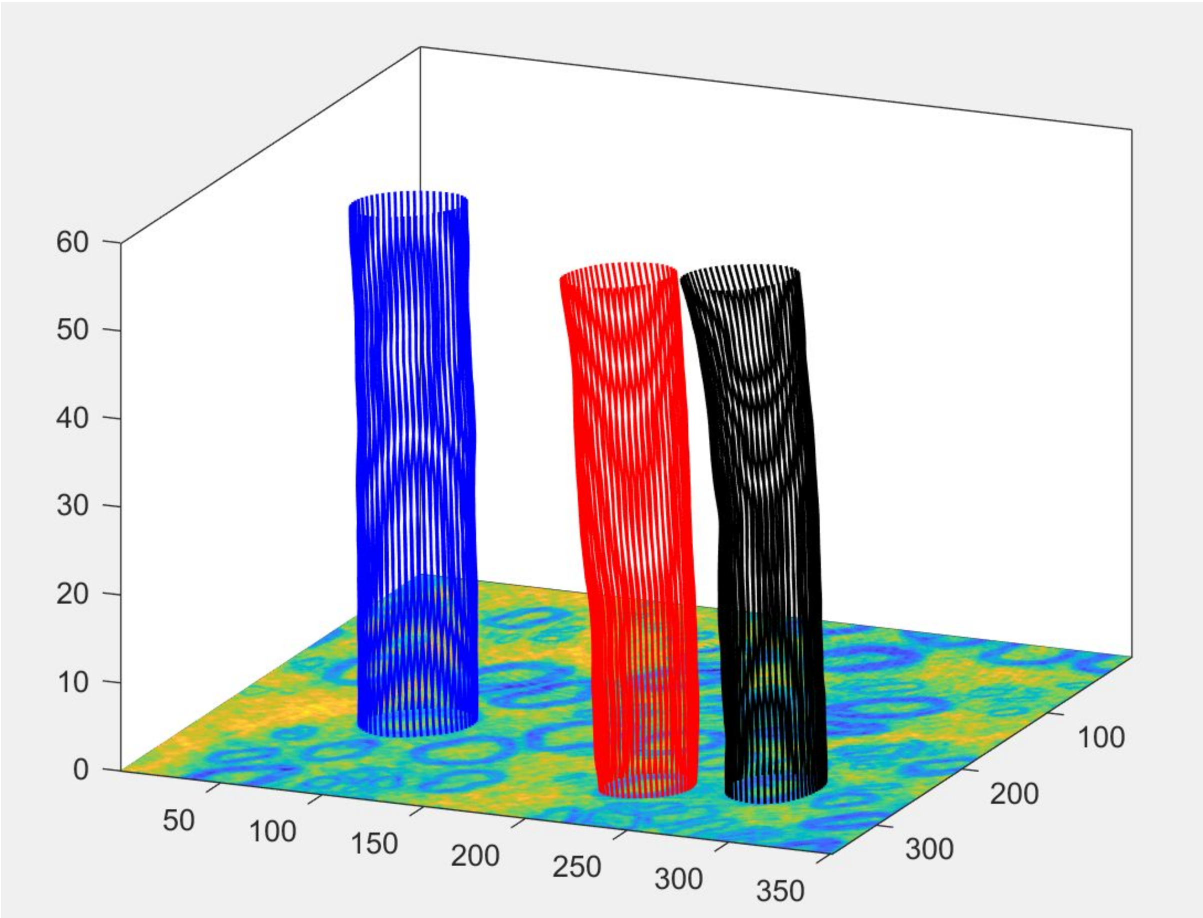
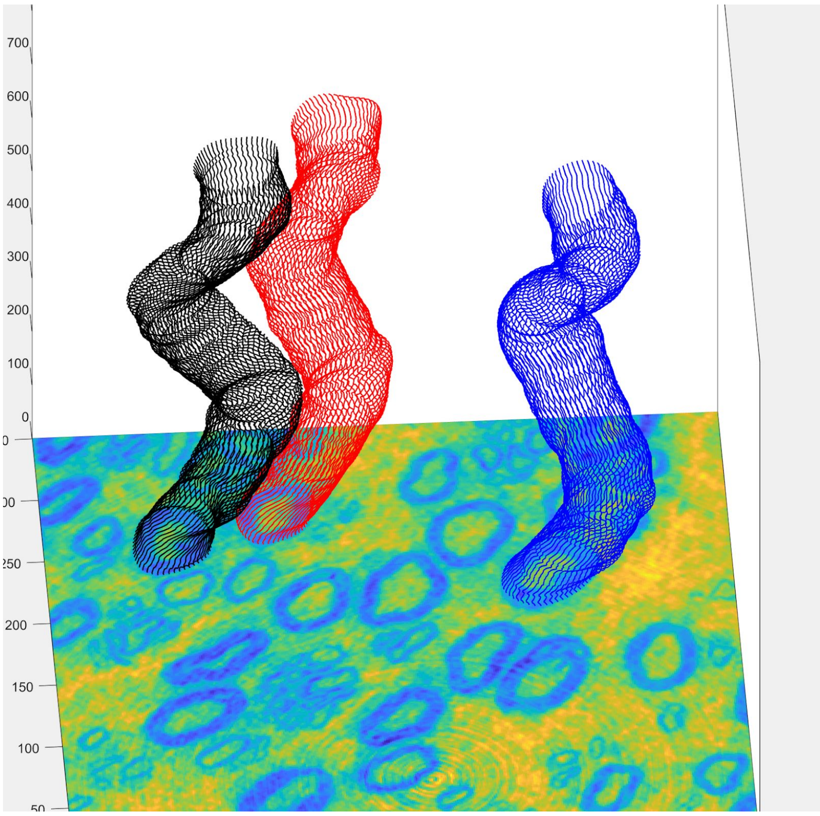


Figure 16:.3D-Reconstruction using individual slices. The surface is an interpolation between similar points on the snake in each clip of the frame.


700
600
500
400
300
200
100
0
250
200
150
100
50

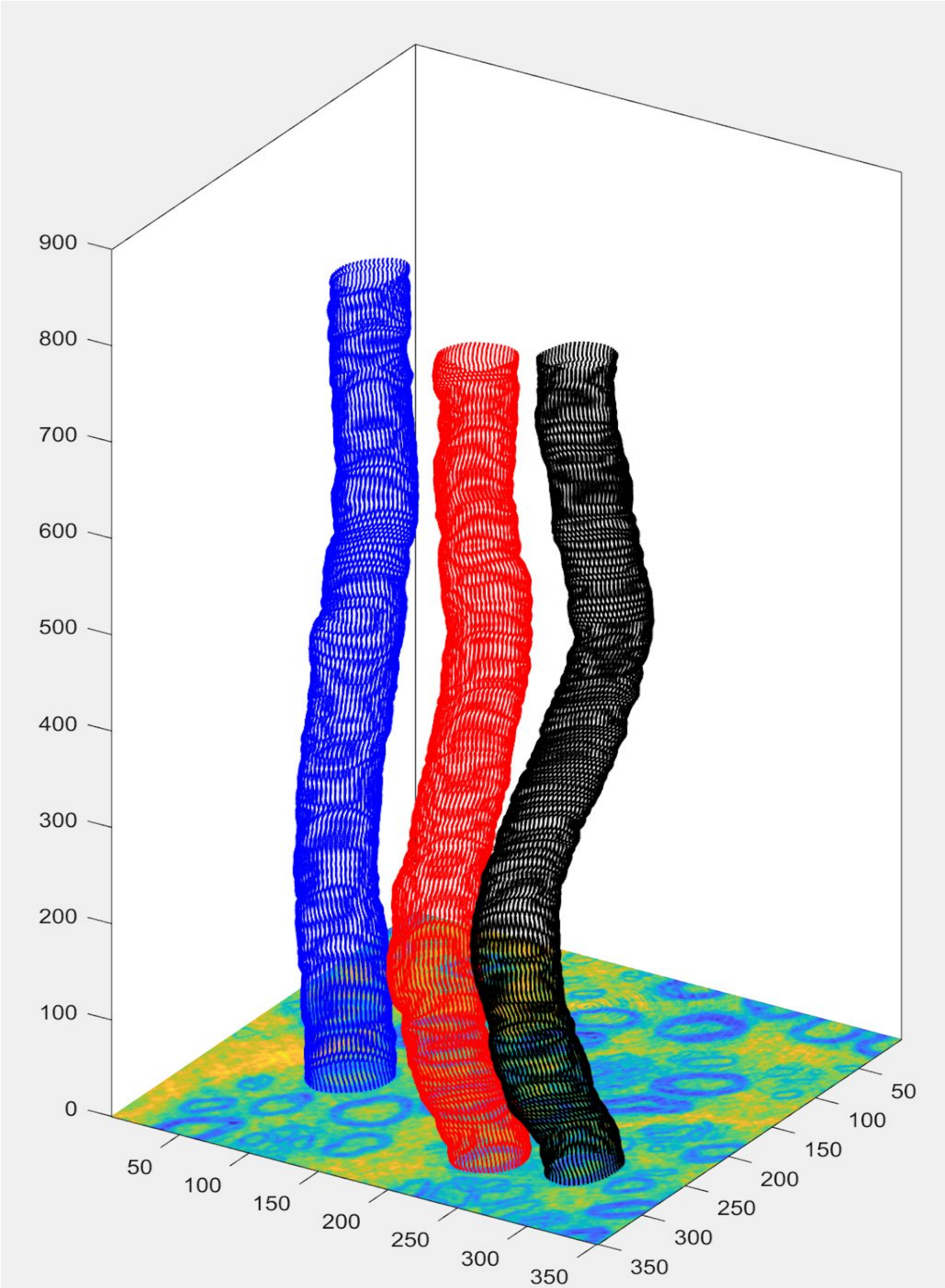


Figure 17:.3D-Reconstruction using individual frames. The surface is an interpolation between similar points on the snake in each clip of the frame.