LODZ UNIVERSITY OF TECHNOLOGY

Faculty of Electrical, Electronic,
Computer and Control Engineering

Bachelor of Science Thesis

Kidney segmentation in the DCE-MRI images
using machine learning algorithms

Fernando Espinosa Asensio

Student Number: 902506

Supervisor:
Artur Klepaczko, PhD

Łódź, 2018

Abstract page

The medical imaging applications are interesting due to the lack of intervention needed, as it is generally a non-invasive method. They let us have an idea of the interior of the bodies without worrying about issues such as infections, recovering time, etc. Lately with the increase in power of computers, more and more automatic or semiautomatic segmentation methods have become popular as they simplify the task of the radiologist and help doctors take decisions with value added images.

In this thesis we selected the kidneys as the organ to work with. We selected the K-means method, an automatic algorithm based on clusterization of the samples. There were studied 3 volumes, the datasets, from 2 different patients. While the result of the segmentations are not optimal, we can observe the influence in the clusterization of the amount of clusters arbitrary selected and the number of iterations (replicates) used. All the analytic work was carried on Matlab R2018a.

The results showed that there is variable number of clusters where the 3 main parts of interest are segmented optimally depending on the volume. For a bigger volume we will need a higher number of clusters. Also we saw that a low amount of clusters can lead to a deficient discretization of the different tissues and a high amount can lead to an over segmentation where samples pertaining to the same cluster are split across several clusters.

Regarding to the number of replicates, it was observed that low iterations can lead to segmentations not consistent and thus, it cannot be ensured an optimal output. An amount of 5 replicates/iterations was found to be a secure value to obtain consistent clusterizations.

Acknowledgments

I would like to thank my family in first place for giving me the support I needed along the study of my degree. Secondly, I would like to express my gratitude to my supervisor, who helped me with his knowledge and always provided me with a new point of view that I had not considered yet. Finally, I would like to thank that people who, in these 4 years, were close to me and in one way or another had influence in the person I have become.

Table of content

1. Introduction

1.1. Medical introduction

Location

The kidneys are one of the best-known organs in the body. They are two bean-shaped structures located in left and east side of the spinal cord, between T12 and L3 vertebras, on the back of the abdominal cavity, they lie at a slightly oblique angle. The liver causes certain asymmetry forcing the right kidney to a lower position and a smaller size than the left one. This one is placed below the diaphragm and posterior to the spleen while the right kidney is located also below the diaphragm but behind the liver. The adrenal glands are located on top of each kidney. Two layers of fat surround each kidney, the perirenal and pararenal fat.

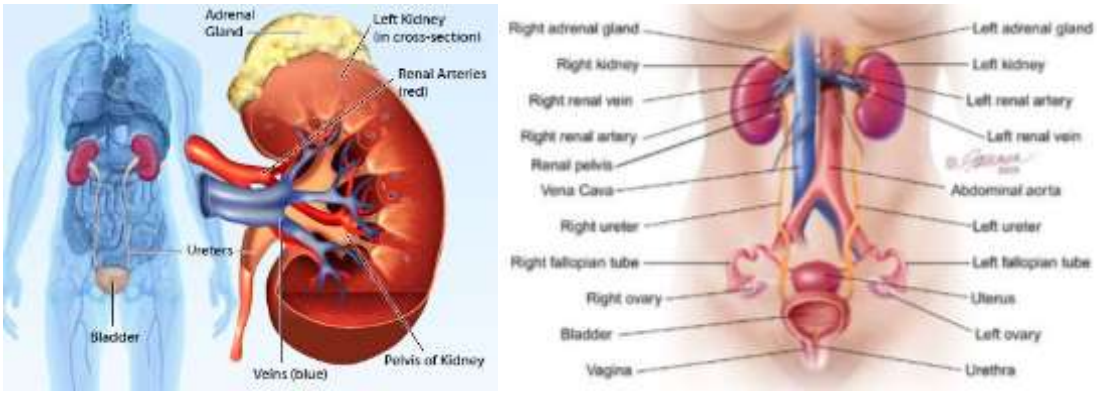


Figure 1. Kidney location. Source: https://macscience.wordpress.com/level-3-biology/homeostasis/

Internal structure

The parenchyma is divided into two main parts. The external is the renal cortex and the internal is the medulla. They are organised in lobules from 8 to 18, where the cortex surrounds the medulla. The functional unit, the nephron is placed along the 2 parts. It is here were the blood filtration is done. The outer part of the nephron is the renal corpuscle which is located in the cortex area, followed by the renal tubule in the medulla till the several calyx that collect the urine generated. The calyces lead to the pelvis, the 3rd main part of the kidney to segment [1].

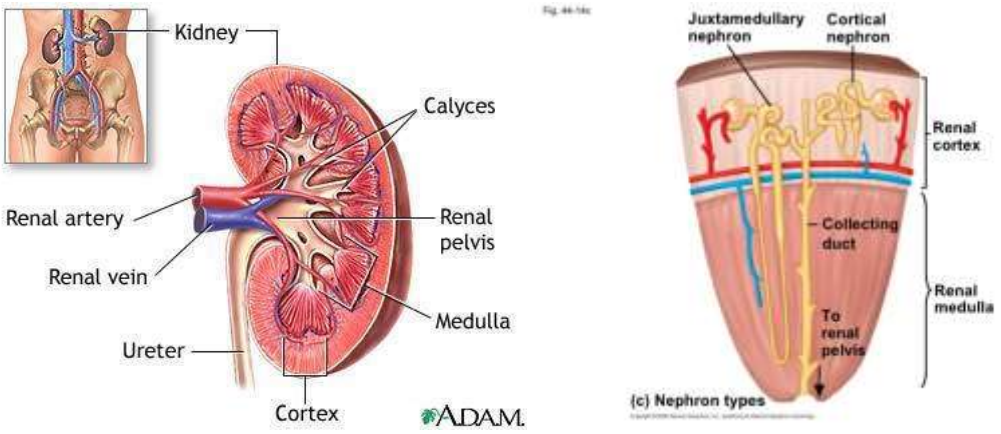


Figure 2. Kidney internal structure. Source: https://macscience.wordpress.com/level-3-biology/homeostasis/

To summarise, in the kidney we will look for the 3 main structures: cortex, the outer area; medulla, the middle area with triangular shape; the pelvis, the inner and closer to the dorsal spine part.

1.2 Contrast-Enhanced MRI (CE-MRI)

Magnetic Resonance Imaging (MRI) is a non-invasive imaging method that allow us to analyse the anatomy of interest without need of surgical intervention. It uses magnetic fields, magnetic field gradients and radio waves to compose an anatomical image. It does not use X-ray nor ionization to produce the image. This technique is less dangerous than computerized tomography (CT) or Positron Emission Tomography techniques (PET), due to non-exposition of the body to radiation.

Making a direct comparison to CT, MRI has a lower resolution, it takes longer to do a scan and the sound is louder. However, as we have already said MRI is not using radiation so it is safer for the patient.

The process simplified is to excite with radio frequency energy a certain atom element under an external magnetic field, after a small period those excited atoms emit the energy as radio frequency and we are able to locate those atoms with the magnetic field gradients. Generally the atom excited is hydrogen, massively present in water and fat, and that is the reason why in this type of images we will observe those structures with high content of these atoms/molecules.

Here takes relevance the contrast enhanced methods. They allow us to increase the signal produced in the tissue of interest where the agent, Gadolinium, will arrive. This is possible because the agent used shortens the T1 constant of the tissue, which is the base of the CE-MRI method. Another use of the contrast agent in this case is to track the way it follows inside the patient specifically inside the kidney to be able to differentiate the voxels belonging to one of the 3 parts of the kidney mentioned in the last paragraph of the previous section. It could be used as well as parameter to analyse the perfusion.

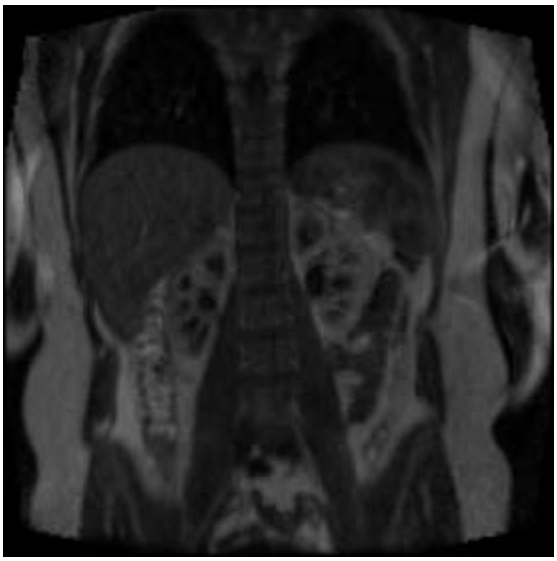


Figure 3. FF01, Vol1, slice 15, frame 15.

2. State of the art

2.1. Segmentation methods

There can be distinguished three main groups of segmentation techniques depending on the level of preparation required by a person.

1. Manual segmentation: Generally performed by a radiologist. It is usually the most precise method but it is also the longest to obtain as the person has to select one by one each of the voxels belonging to each part of the kidney [2].
2. Semiautomatic.
    a. Thresholding: computationally not demanding method. After selecting a specific brightness value, pixel with higher or lower intensity are set to an arbitrary value. [3]
    b. Region-based techniques: regions with maximum homogeneity are localized. [4, 5]
    c. Level-set model: focused on detecting object boundaries through speed function for curve evolution. [6, 7]
3. Automatic:
    a. Gaussian mixture model: based in probabilistic parameters. [8]
    b. Fuzzy set: which takes the advantage of the variation within the segments of the objects. It combines the most probable (located near each other and/or similar in the intensity values) regions into one segment. [9]
    c. Graph-cuts: uses energy minimization, maxflow mincut theorem. [10]
    d. K-means: groups the pixels/voxels of an image across an arbitrary number of clusters based on the similarities in the variables. [11, 12]
    e. Surface deformation [13].
    f. Neural networks [14].

2.2. K-means

This method analyses a 2D matrix where each row is the sample and each column is the variable, in this case the time frames. This method uses the variables that separates better the samples and tries to group them selecting random centroids and calculating, usually, the Euclidean distance to the centroid so the samples are linked to the closest one. After this, the mean is recalculated and again the samples are associated to the closest. This process is computed in each iteration. The last iteration should keep the centroids in the same place and thereby the samples stay in the same group.

3. Methods

The datasets analysed come from 2 patients, each of them have been explored 2 times with one week of difference between measurements for the same patient. They will be used to test the performance of the K-means segmentation method versus a manual segmentation provided by a doctor.

For obtaining the results it has been used 3 self-made scripts:

3.1. ROI selection

The goal of the first script is selecting a narrowed volume where the kidneys are contained to be able to perform the K-means segmentation of the two kidneys together or each kidney per separate. The input of the ROI_Selection5_VOL1 function are the directory of the .nii files and the file containing the ROI itself. As output, we can get up to 8 variables:

- w1: 2D matrix containing the raw intensity of each voxel (as rows) from the firs kidney along the number of frames captured (each column)
- c1: vector featuring the index of the voxels of the ROI of the first kidney.
- Z1: 2D matrix containing the standardized intensity of each voxel (as rows) from the firs kidney along the number of frames captured (each column)
- w2: 2D matrix containing the raw intensity of each voxel (as rows) from the second kidney along the number of frames captured (each column)
- c2: vector featuring the index of the voxels of the ROI of the second kidney.
- Z2: 2D matrix containing the standardized intensity of each voxel (as rows) from the second kidney along the number of frames captured (each column)
- w12: 2D matrix containing the raw intensity of each voxel (as rows) from both kidneys along the number of frames captured (each column)
- Z12: 2D matrix containing the standardized intensity of each voxel (as rows) from both kidneys along the number of frames captured (each column)

The performance of the code has been optimized using the latest functions introduced in matlab such as the command niftiread, from 2017b version.

3.2. K-means segmentation

The goal of the second script is to perform a segmentation using the k-means function with the choice by the user of performing it in a single volume, containing the data from both kidneys or otherwise running the method for each kidney separately. Either way the output will be a single volume containing the clustered data from both kidneys. With a little tweak, the user could be able to obtain each kidney volume per separate, but for this research is convenient a single volume.

The performance of the code has been optimized using the latest functions introduced in matlab such as the command volumeViewer, from 2017a version, or K-mean function introduced before 2006a version.

3.3. Dice analysis

For obtaining the Dice coefficient, we will compare the manual segmentation with each of the clustered volumes. The manual segmentation is given in two volumes, one for each kidney, unlike the clustered data. The goal of the script is to perform the comparison of volumes containing both kidneys from the manual data, placed in one single volume, with the clustered data, included in a single volume too. The formula implemented is the one given below:

$$DSC = \frac{2TP}{2TP + FP + FN}$$

TP: true positive, voxels included in both volumes. Correct voxels

FP: false positive, voxels from the automatic clusterization not present in the manual segmentation. Wrong voxels

FN: false negative, voxel from the manual segmentation not present in the automatic clusterization. Missing voxels.

3.4. Parameters of interest

As k-means has a random initialization we will run 3 times the k-means script in order to confirm the importance of different values such as the number of clusters and of different number of replicates. The replicates is a parameter from the k-means function that sets the number of times the clusterization is made, with different initialization data each time as it is a random method.

The replicates provides us a more consistent clusterization because the initialization of the centroids has a big impact in the output. The number of runs using the same parameters will help us see the impact of the initialization because our goal is to create an optimized algorithm which output will be consistent every time it is executed, as it will give more confidence about the clusterization.

4. Results

4.1. Patient FF01 - Vol_1

First of all the result of the segmentation in 3 and 4 clusters will be shown with the purpose of justifying the lack of need of calculation of the Dice coefficient.

It is true that, in figure 4, it can be slightly recognized the shape of the kidney but we cannot see any good segmentation of the 3 structures of the kidney itself. Therefore, it does not make sense to perform a run to get the Dice Coefficient.

Segmentation in 3 and 4 clusters

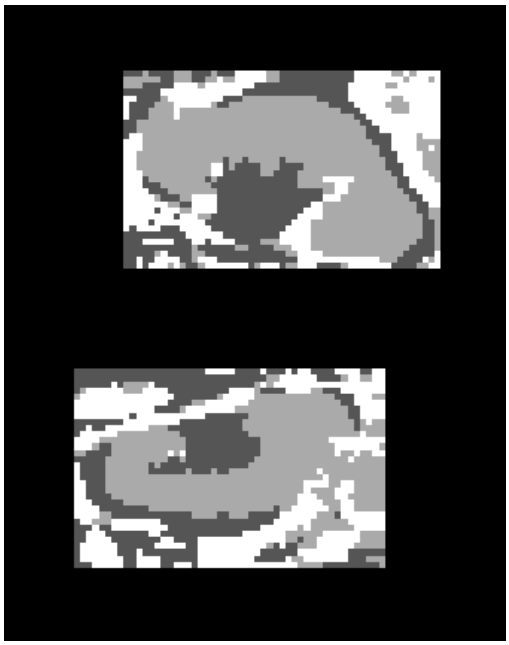


Figure 4. K-Mean segmentation in 3 clusters of FF01 Vol_1

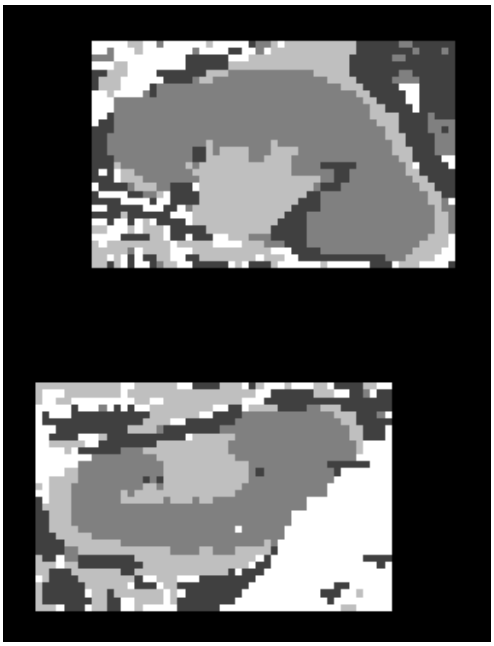


Figure 5. K-Mean segmentation in 4 clusters of FF01 Vol_1

Now we will start with the analysis of the influence in the segmentation of the number of clusters and the number of replicates in the K-means method.

Segmentation in 5 clusters

The first segmentation obtained with some sensible and consistent results was using 5 clusters for the K-means

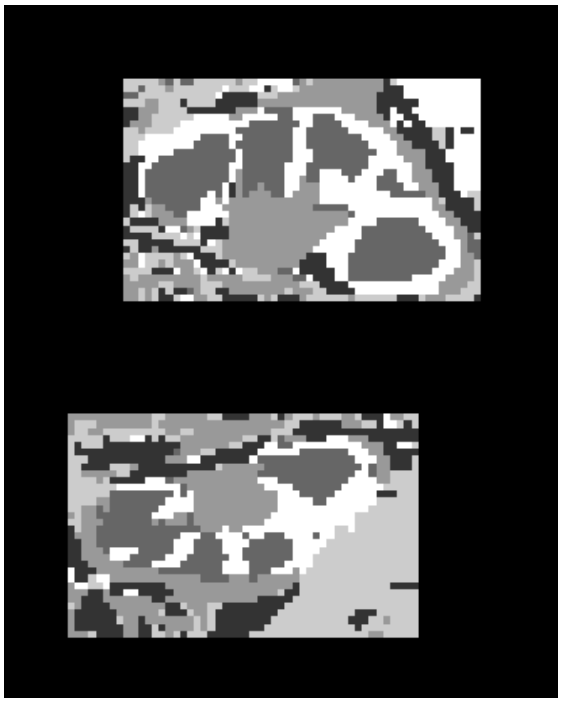


Figure 6. K-Mean segmentation in 5 clusters of FF01 Vol_1

In this case it is easily distinguished the medulla and cortex, but the pelvis is not so well segmented as it is included in the same cluster as the external surrounding tissue of the kidney.

As it is shown in figures 8, 10 and 12 the 3D volumes are confirming the previous statement. It is also noticeable the presence of voxels, in figures 6 and 7, odd to the medulla in the same cluster.

Medulla

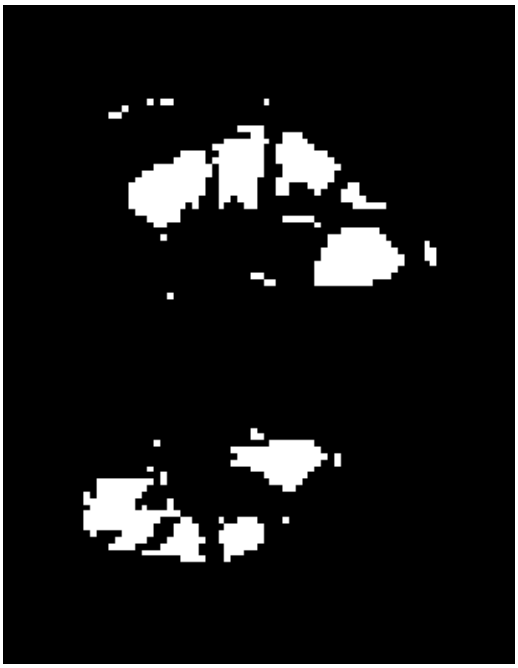


Figure 7. Medulla cluster, XY slice 15th

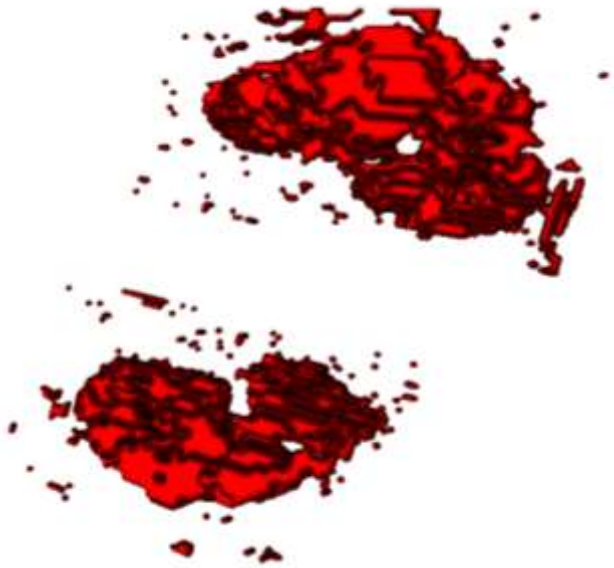


Figure 8. Medulla cluster, isosurface volume

Cortex

Figure 9. Cortex cluster, XY slice 15th

Figure 10. Cortex cluster, isosurface volume

Here it can be seen that in the upper left corner of the figure 6 it is included a big section of tissue alien to the cortex. This is a symptom of the need of a larger amount of clusters for the segmentation. In addition, there are some independent voxels spread all around the volume similar to the medulla cortex.

Pelvis

Figure 11. Pelvis cluster, XY slice 15th

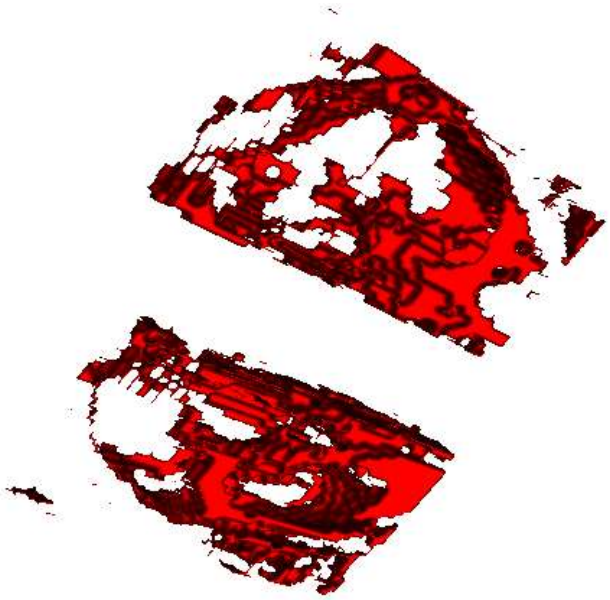


Figure 12. Pelvis cluster, isosurface volume

As previously anticipated, the pelvis cluster shows the worst result among the 3 kidney clusters with a total of 5 clusters for the segmentation.

Dice Coefficient

FF01_VOL1_Rx={pelvis cortex medulla}

Number of Replicates	1(run 1/run 2 /run 3)	2(run 1/run 2 /run 3)	3(run 1/run 2 /run 3)	5(run 1/run 2 /run 3)
Cortex	0.4912/ 0.5462/ 0.4946	0.4916/ 0.4935/ 0.5460	0.4935/ 0.5463/ 0.4916	0.4934/ 0.4917/ 0.4916
Medulla	0.6276/ 0.4501/ 0.6288	0.6276/ 0.6282/ 0.4502	0.6282/ 0.4501/ 0.6276	0.6283/ 0.6275/ 0.6276
Pelvis	0.1864/ 0.1864/ 0.1862	0.1864/ 0.1864/ 0.2349	0.1863/ 0.2355/ 0.1864	0.1865/ 0.1864/ 0.1864

Table 1. Dice coefficient analysis: FF01_VOL1_Manual vs K-means segmentation in 5 clusters

We observe an anomaly in the second run with a number of replicates equal to 2, in this case a cluster including both medulla and cortex was obtained as an output (figure 13).

Figure 13. Cortex and medulla in the same cluster. FF01_Vol1, 5 clusters, 1 replicate, second run

Again, in the execution with 2 replicates a deficient clusterization is observed.

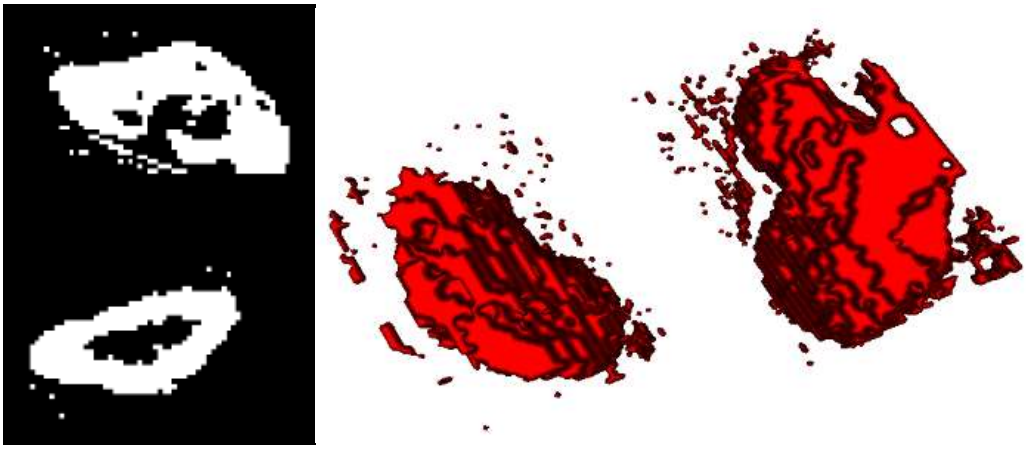


Figure 14. Cortex and medulla in the same cluster. FF01_Vol1, 5 clusters, 2 replicates, third run

Same as this two ambiguous clusterization appear in the second run with 3 replicates, as it can be observed in the figure 15.

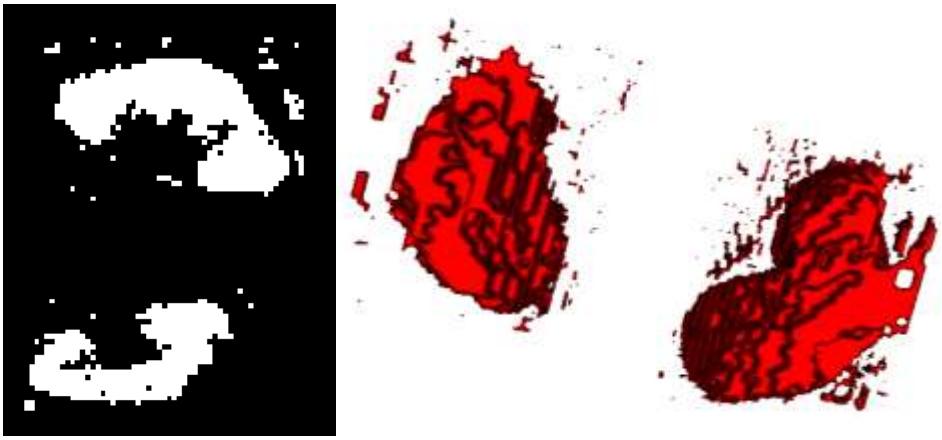


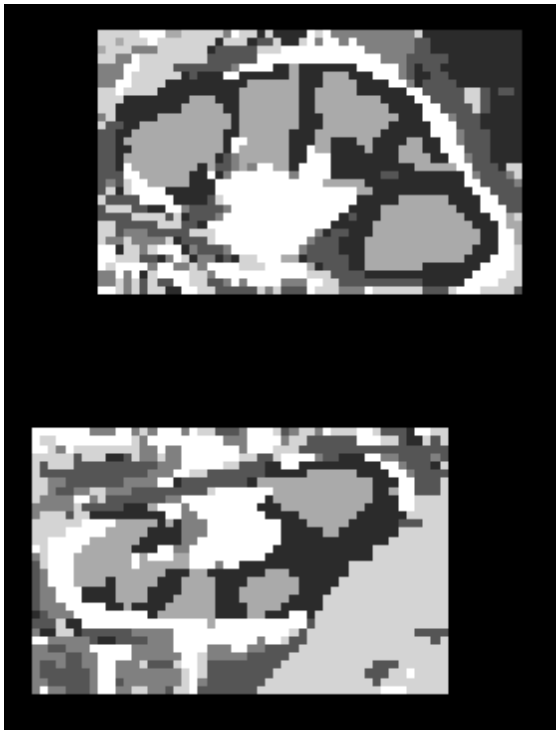
Figure 15. Cortex and medulla in the same cluster. FF01_Vol1, 5 clusters, 2 replicates, third run

After all the executions, it can be observed an apparently better segmentation of the medulla than the cortex. However, the cluster where the pelvis is included contains a

higher amount of voxels not related with the real pelvis area. An interesting event is that when the cortex and medulla are in the same cluster, the cortex is better segmented.

Segmentation in 6 clusters

After watching the results from the k-means segmentation using 5 clusters we will try to analyse the performance using 6 clusters. In order to provide a general view of the output obtained with this parameters, the clusterization with 5 replicates will be used as example.



In a raw stage of the clusterization, as shown in figure 16, it cannot be observed a big difference between using 5 and 6 clusters. Next, following the procedure from last section, the dice coefficients will be obtained and compared with the 5 clusters results.

Figure 16. K-Mean segmentation in 6 clusters of FF01 Vol_1

Medulla

Figure 17. Medulla. FF01_Vol1, 6 clusters, 5 replicates

Cortex

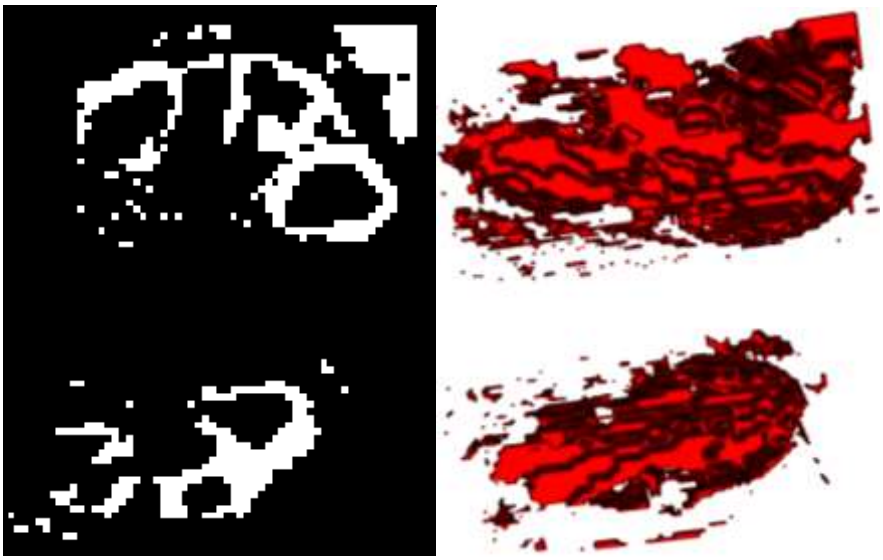


Figure 18. Cortex. FF01_Vol1, 6 clusters, 5 replicates.

Pelvis

Figure 19. Pelvis. FF01_Vol1, 6 clusters, 5 replicates.

Dice Coefficient

FF01_VOL1_C6_Rx={pelvis cortex medulla}

Number of Replicates	1(run 1/run 2 /run 3)	2(run 1/run 2 /run 3)	3(run 1/run 2 /run 3)	5(run 1/run 2 /run 3)
Cortex	0.5026/ 0.5053/ 0.5026	0.5074/ 0.5026/ 0.5078	0.5053/ 0.5024/ 0.5022	0.5056/ 0.5074/ 0.5021
Medulla	0.6440/ 0.6450/ 0.6438	0.6455/ 0.6438/ 0.6460	0.6449/ 0.6441/ 0.6444	0.6450/ 0.6455/ 0.6440
Pelvis	0.2232/ 0.2227/ 0.2232	0.2227/ 0.2232/ 0.2223	0.2227/ 0.2234/ 0.2234	0.2233/ 0.2226/ 0.2232

Table 2. Dice coefficient analysis: FF01_VOL1_Manual vs K-means segmentation in 6 clusters

With 6 clusters and this particular volume, the influence of the number of replicates cannot be noticed. Either way, the dice coefficients are improved in comparison with the clusterization of this same volume in 5 clusters.

4.2. Patient FF01 - Vol_2

The second volume correspond to a second scan, with a week of difference from the first scan, contained in the Volume 1. In this case the amount of voxels is bigger. This will impact the number of clusters needed to get a sensible segmentation.

Segmentation in 4 and 6 clusters

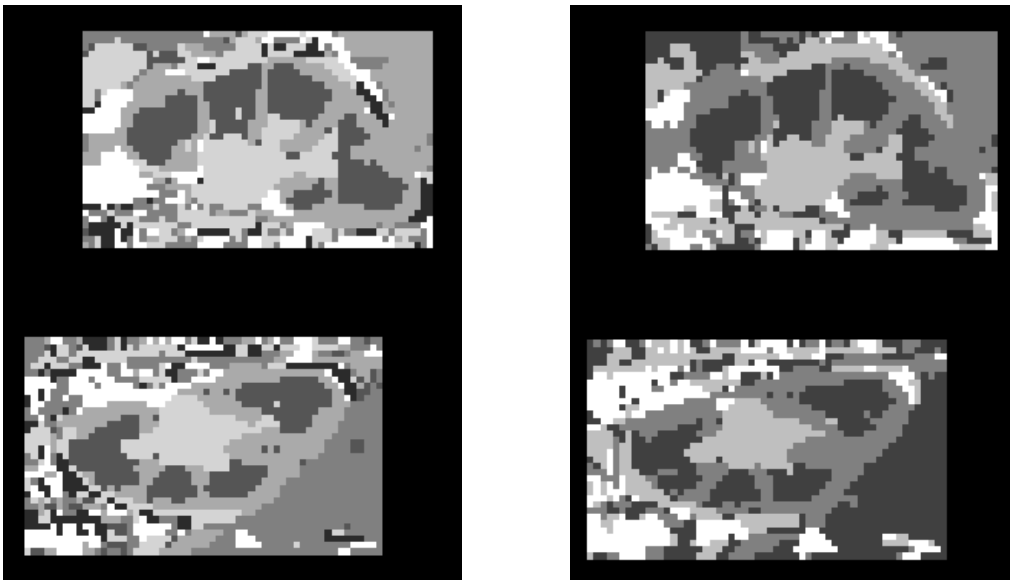


Figure 20. Clusterization of FF01_Vol_2 in 4 clusters (left) and 6 clusters (right)

In both case it can be observed that the cortex area is clustered with the area containing the spleen. This is the reason why there is no dice coefficient calculated with this number of cluster. In addition, this supports the statement that for a larger amount of voxels, a larger number of cluster is needed. The parameters used in this two examples where a number of replicates equal to 5, as it has been proved in the first experience that the results obtained with this amount of replicates only depends on the number of clusters and not on the number of random initializations.

Segmentation in 8 clusters

A sensible number of clusters can be identified from 8 cluster and higher. In the next step the 3 main clustered parts will be shown.

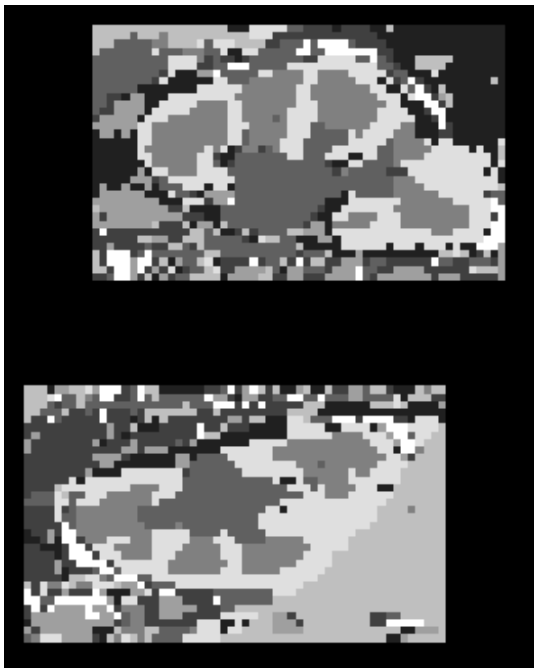


Figure 21. K-Mean segmentation in 8 clusters of FF01 Vol_2

Medulla

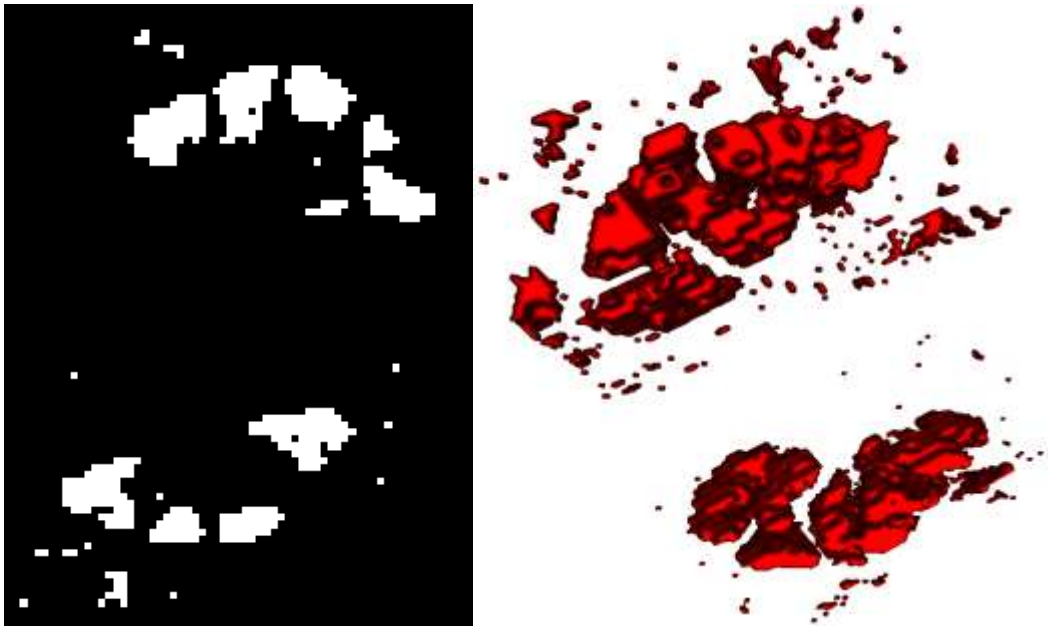


Figure 22. Medulla. FF01_Vol2, 8 clusters, 5 replicates

Cortex

Figure 23. Cortex. FF01_Vol2, 8 clusters, 5 replicates

Pelvis

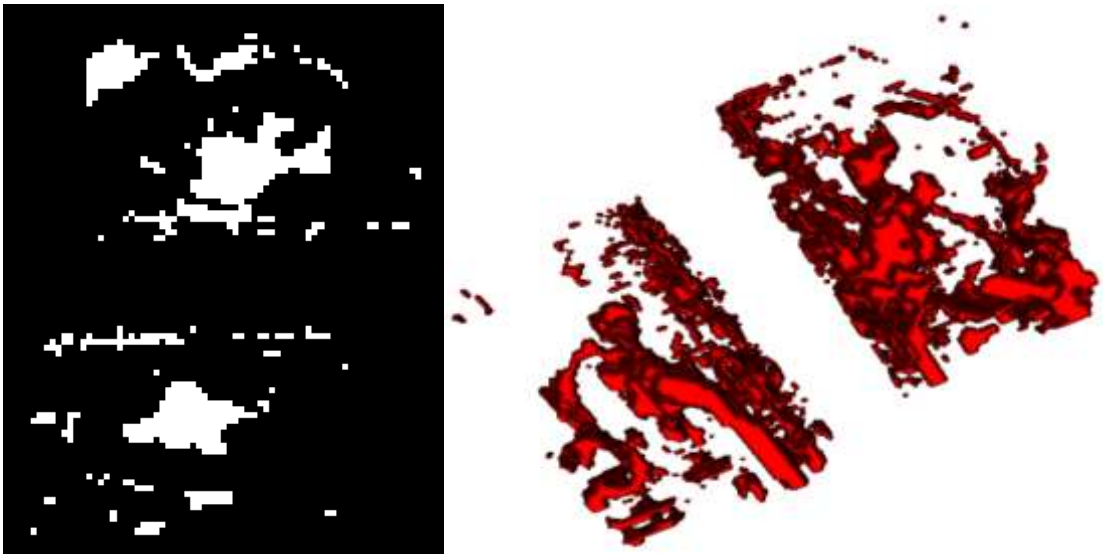


Figure 24. Pelvis. FF01_Vol2, 8 clusters, 5 replicates

Dice Coefficient

FF01_VOL2_C8_Rx ={pelvis cortex medulla}

Number of Replicates	1(run 1/run 2 /run 3)	2(run 1/run 2 /run 3)	3(run 1/run 2 /run 3)	5(run 1/run 2 /run 3)
Cortex	0.4609/ 0.4600/ 0.4883	0.5392/ 0.5393/ 0.4883	0.5394/ 0.4883/ 0.5391	0.5391/ 0.5392/ 0.5394
Medulla	0.6007/ 0.6024/ 0.5922	0.6223/ 0.6226/ 0.5921	0.6224/ 0.5921/ 0.6222	0.6224/ 0.6225/ 0.6226
Pelvis	0.2799/ 0.2763/ 0.2822	0.2688/ 0.2684/ 0.2820	0.2674/ 0.2823/ 0.2683	0.2675/ 0.2688/ 0.2681

Table 3. Dice coefficient analysis: FF01_VOL2_Manual vs K-means segmentation in 8 clusters

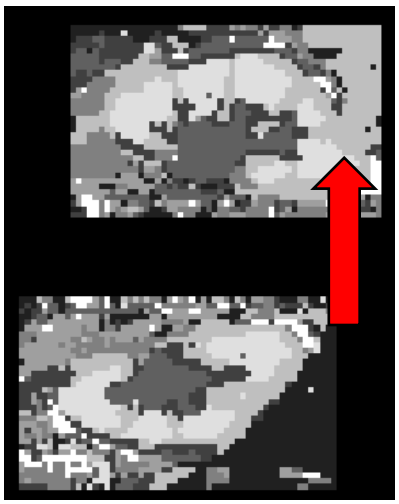


Figure 25. FF01_Vol_2, 8 clusters, 1 Replicate

```
>> Kmeans
Warning: Failed to converge in 100 iterations.
> In kmeans/loopBody (line 474)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

With only 1 replicate in the first run, it is not able to deliver a proper clusterization within 100 iterations of the K-means. In the output, figure 25 we observe that the cortex is clustered within the area related to the spleen. A red arrow indicates the junction of the cortex and the spleen in these figures.

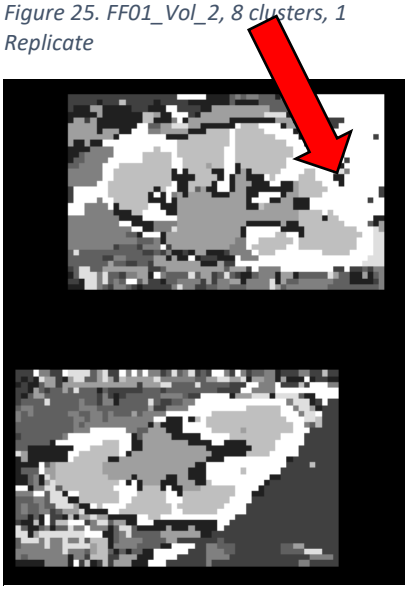


Figure 26. FF01_Vol_2, 8 clusters, 1 Replicate

In the second run, with only 1 replicate, we observe the same issue. This will compromise the value of the dice coefficient, as it can be observed in table 3.

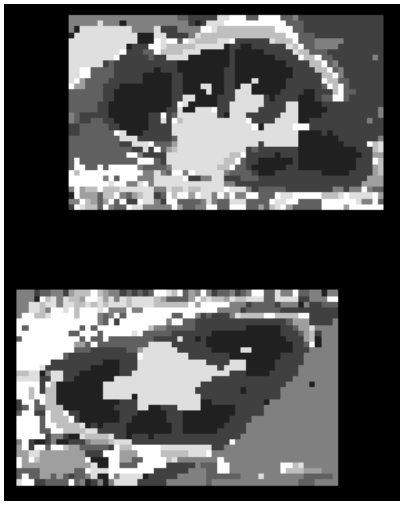


Figure 27. FF01_Vol_2, 8 clusters, 2 Replicates

```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 2.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

In the 3rd run with 2 replicates we encounter the same error as produced with only one replicate. In this case the 2 iterations of the K-means is not enough to give a good clusterization as output. In addition, with 2 replicates we observe that the dice coefficient is not consistent among the 3 runs (table 3)

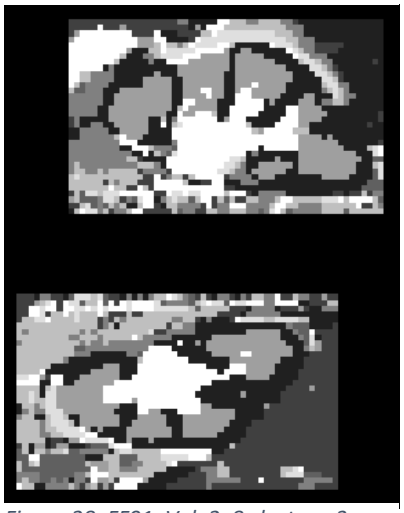


Figure 28. FF01_Vol_2, 8 clusters, 3 Replicates

```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 2.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

The error appears again with 3 replicates in the 2nd run, and so the clusterization is not properly done. Dice coefficient not consistent.

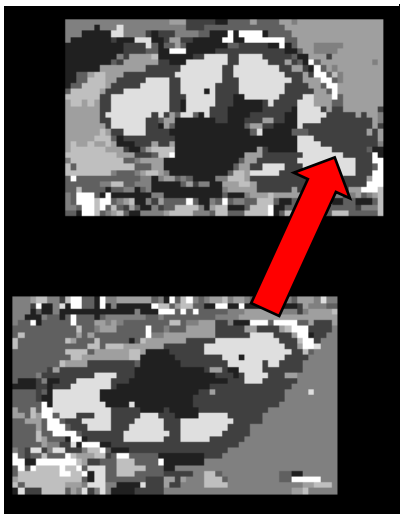
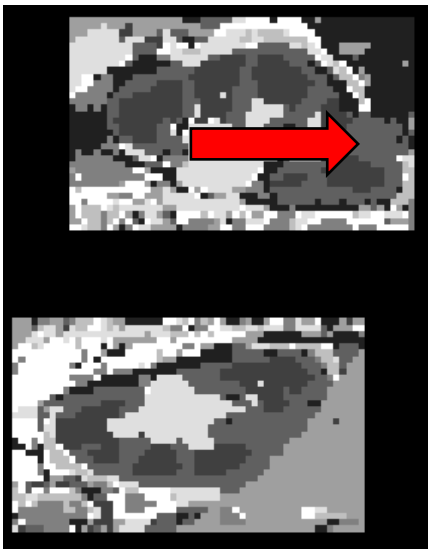


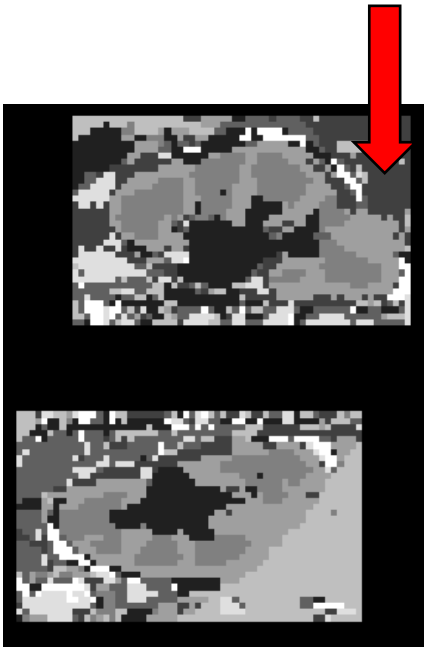
Figure 29. FF01_Vol_2, 8 clusters, 3 Replicates

```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 1.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

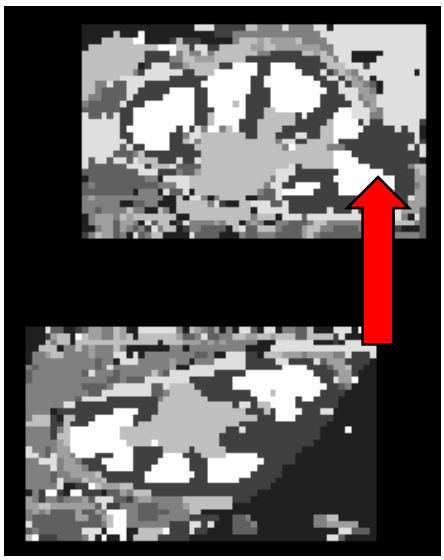
It also happens in the 3rd run with 3 replicates, but in this case the result of the clusterization is consistent with the performance of the run without failures in a K-mean iteration. Here it can be seen the relevance of having a higher amount of replicates/iterations.



a)



b)



d)

```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 1.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 363)
  In kmeans (line 31)
```

c)

Figure 30. a) FF01_Vol_2, 8 clusters, 5 Replicates, 1st run. b) FF01_Vol_2, 8 clusters, 5 Replicates, 2nd run. c) k-means iteration failure. d) FF01_Vol_2, 8 clusters, 5 Replicates, 3rd run.

As well as the error in the 3rd run with 3 replicates we can observe that despite the fact that there is failure in the 1st iteration, the dice coefficient is consistent with the best clustered results. This phenomenon was spotted in all the runs with 5 replicates.

Segmentation in 9 clusters

As well as with the Volume 1, the number of clusters will be increased in order to compare the accuracy of the segmentation and the influence of the number of clusters. Following with the procedure from previous experiences, the parameters set to illustrate the example of the 3 main parts of the kidney will be 9 clusters and 5 replicates. The overall clusterization is shown in figure 33.

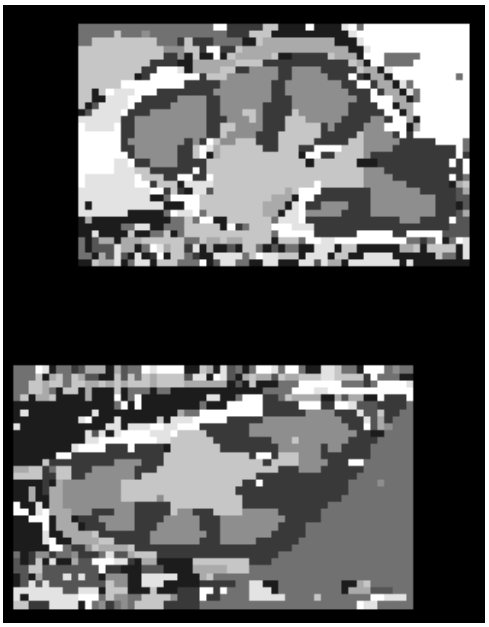


Figure 31. K-Means segmentation in 9 clusters of FF01 Vol_2.

Medulla

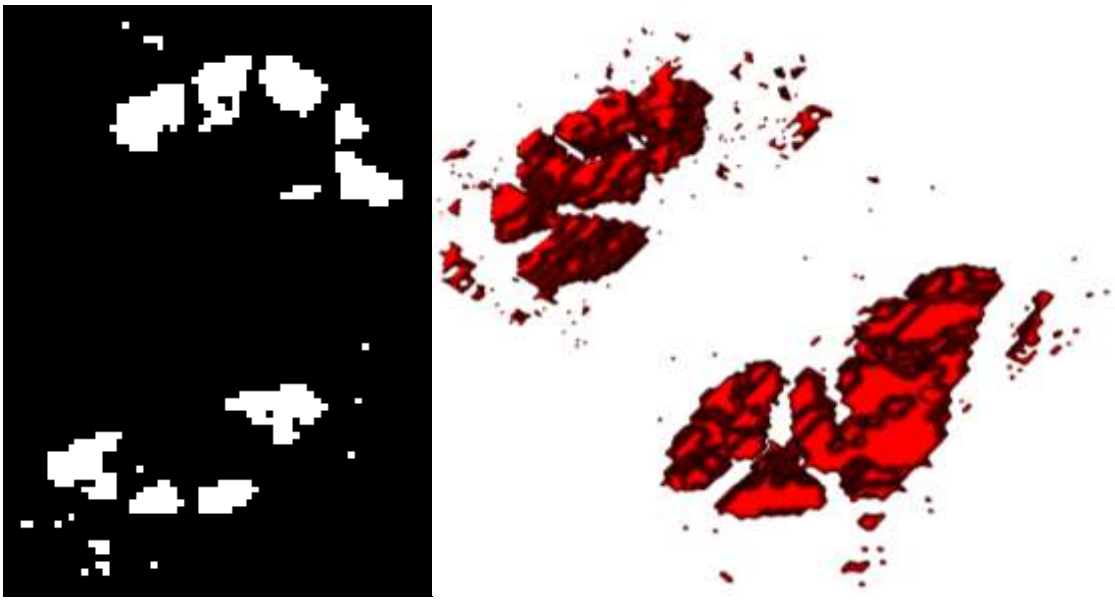


Figure 32. Medulla. FF01_Vol2, 9 clusters, 5 replicates

Cortex

Figure 33. Cortex. FF01_Vol2, 9 clusters, 5 replicates.

Pelvis

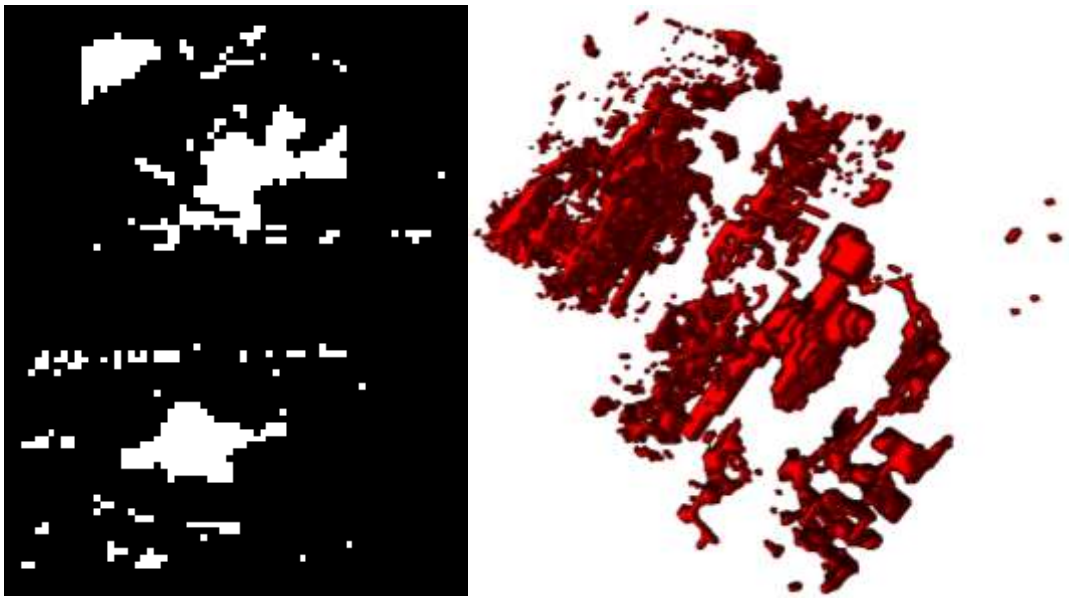


Figure 34. Pelvis. FF01_Vol2, 9 clusters, 5 replicates.

Dice Coefficient

FF01_VOL2_C9_Rx={pelvis cortex medulla}

Number of Replicates	1(run 1/run 2 /run 3)	2(run 1/run 2 /run 3)	3(run 1/run 2 /run 3)	5(run 1/run 2 /run 3)
Cortex	0.5310/ 0.4809/ 0.5322	0.5338/ 0.5340/ 0.5323	0.4940/ 0.5323/ 0.5341	0.5322/ 0.5403/ 0.5334
Medulla	0.6208/ 0.5953/ 0.6278	0.6284/ 0.6292/ 0.6281	0.5985/ 0.6280/ 0.5985	0.6280/ 0.6254/ 0.6217
Pelvis	0.2915/ 0.2974/ 0.2810	0.2806/ 0.2784/ 0.2815	0.2953/ 0.2815/ 0.2773	0.2814/ 0.2797/ 0.2903

Table 4. Dice coefficient analysis: FF01_VOL2_Manual vs K-means segmentation in 9 clusters.

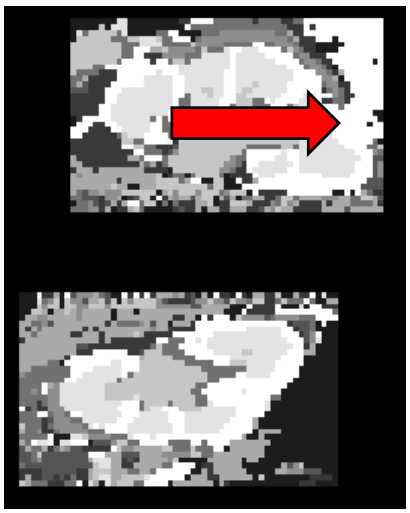


Figure 35. FF01_Vol_2, 9 clusters, 1 Replicate

```
Warning: Failed to converge in 100 iterations.
> In kmeans/loopBody (line 474)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

With only one replicate we encounter that in the 2nd run the segmentation is deficient, and again the cortex is in the same cluster than the spleen. It is also noticeable by the dice coefficient in table 4.

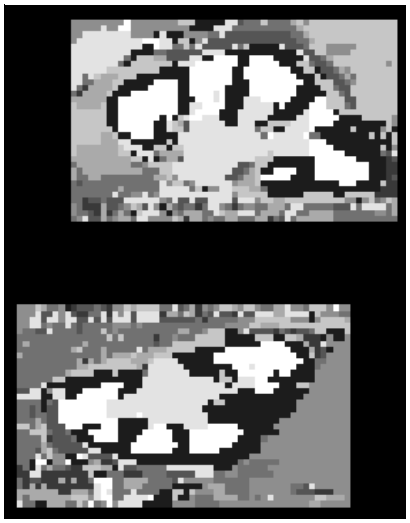
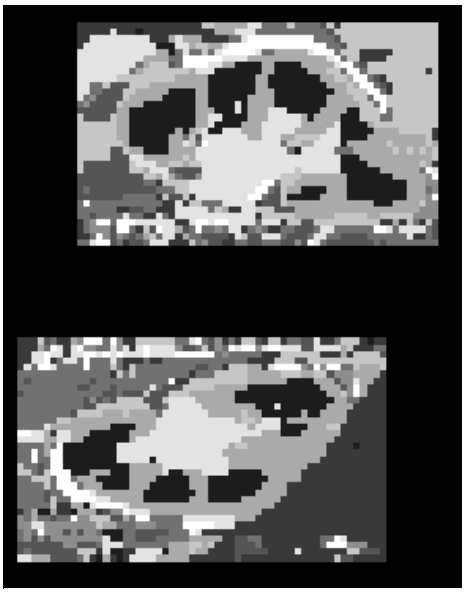


Figure 36. FF01_Vol_2, 9 clusters, 1 Replicate

```
Warning: Failed to converge in 100 iterations during replicate 2.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

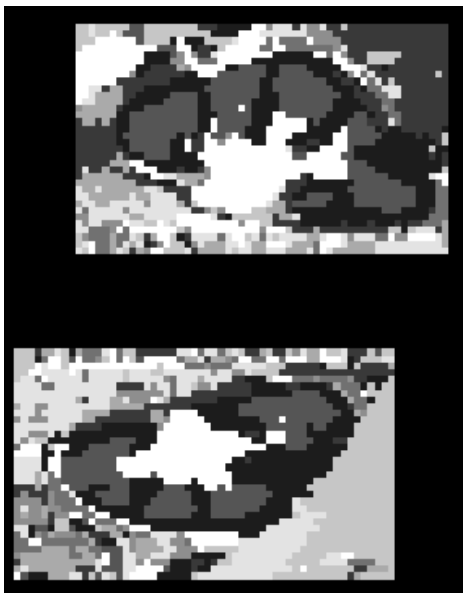
In the execution of the 2nd run with 2 replicates we observe that despite having a failure in the 2nd iteration of K-means, the clusterization is consistent with the best ones.



```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 2.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
Warning: Failed to converge in 100 iterations during replicate 3.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

In the 1st run with 5 replicates there were 2 iterations failures but the results were still consistent (Table 4).

Figure 37. FF01_Vol_2, 9 clusters, 5 Replicates



```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 2.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

The output of the 2nd run using 5 replicates gave proper results even though there was a failure in the 2nd iteration.

Figure 38. FF01_Vol_2, 9 clusters, 5 Replicates

After all the analysis of the dice coefficients we observe that in this particular case the performance between 8 and 9 clusters is similar compared with the manual segmentation. Other interesting result is the 1st run with 3 replicates which is consistent in the medulla but not in the cortex, it has been highlighted, in red colour, in table 4.

4.3. Patient FF02 - Vol_1

Now the volume scanned of the second patient will be analysed and segmented following the method already used with the previous datasets. First of all the clusterization in 3 and 5 groups will be shown to remind that, in this case, we need a higher number of cluster to get sensible results.

Segmentation in 3 and 5 clusters

3 Clusters

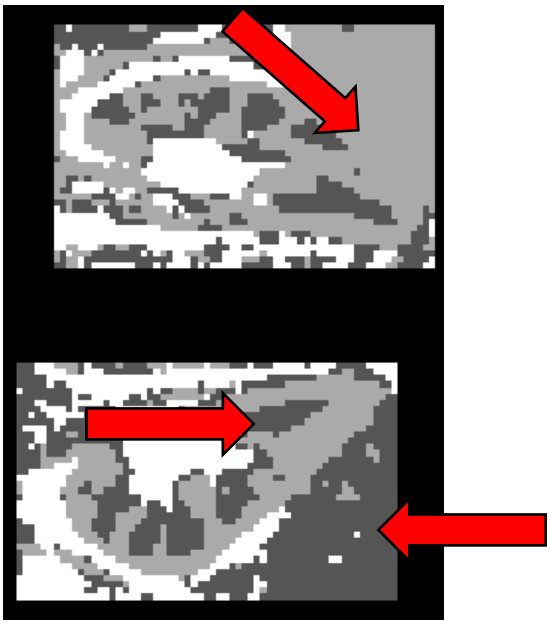


Figure 39. K-Mean segmentation in 3 clusters of FF02 Vol_1

In cluster number 1, which contains the medulla, is also including voxels from the liver. In cluster number 2 it can be seen that the spleen is included in the same cluster than the cortex. Finally, the cluster number 3 shows that as well as in previous segmentations, the pelvis cluster includes a lot more voxels not related to it.

5 Clusters

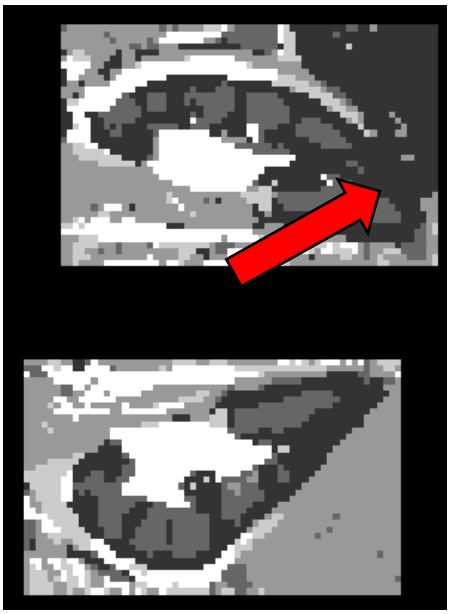


Figure 40. K-Mean segmentation in 5 clusters of FF02_Vol_1

In this case with 5 clusters, the liver and medulla are separately clustered but the spleen is still included in the same cluster as the cortex. Coming up next, the number of clusters will be increased to 6 and it could be observed that with that number, the spleen is also segmented separately.

Segmentation in 6 clusters

After some quick tests, a number of clusters set to 6 was the starting point of obtaining sensible results as is can be observed in figure 41. In addition, the different clusters containing the medulla, cortex and pelvis will be shown.

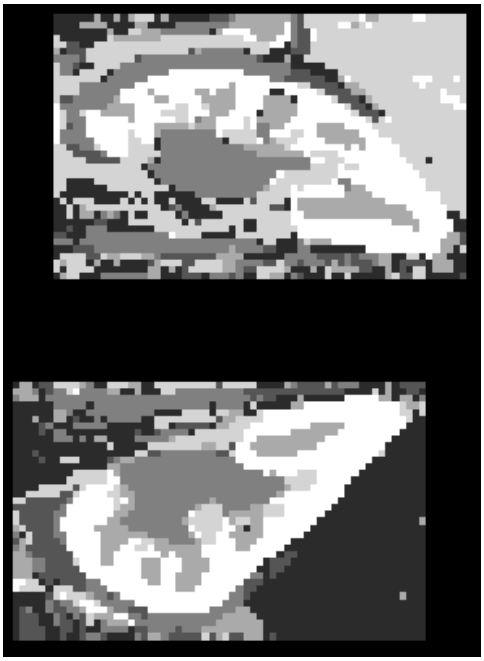


Figure 41. K-Mean segmentation in 6 clusters of FF02_Vol_1.

Medulla

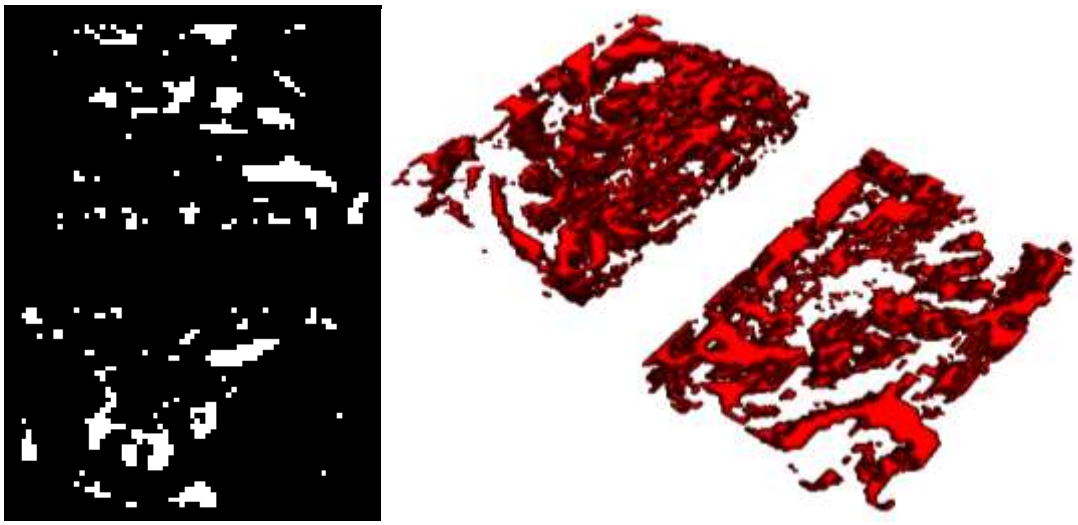


Figure 42. Medulla. FF02_Vol1, 6 clusters, 5 replicates.

Cortex

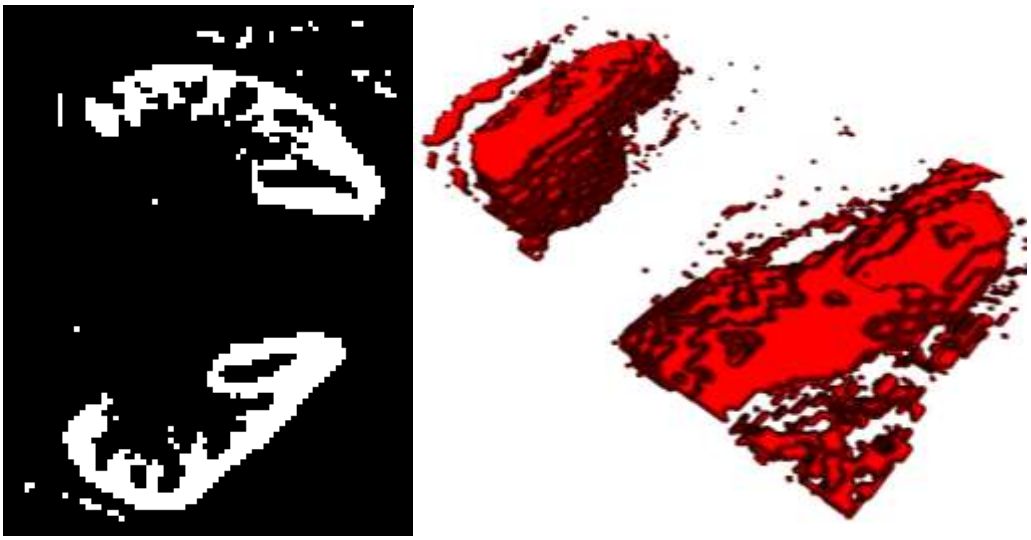


Figure 43. Cortex. FF02_Vol1, 6 clusters, 5 replicates.

Pelvis

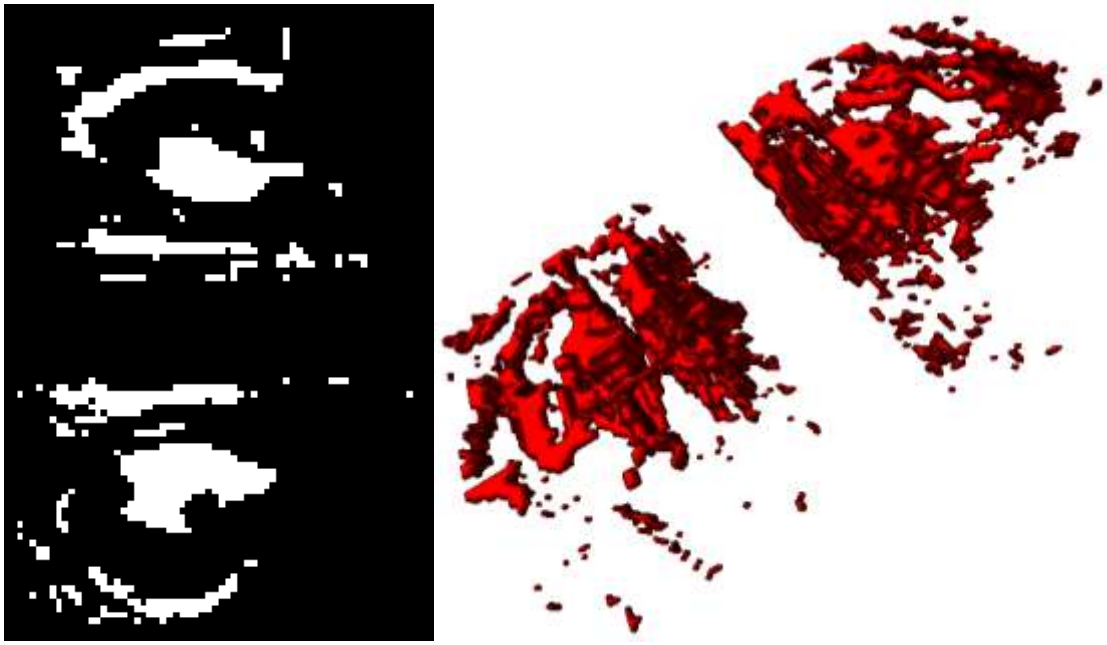


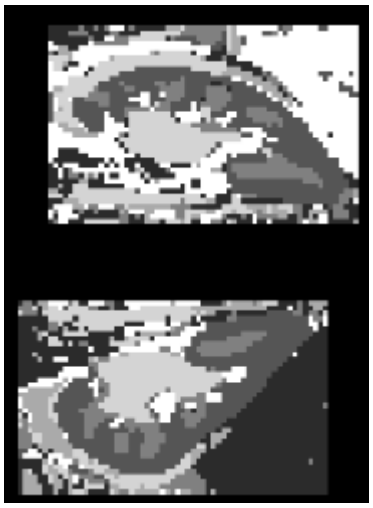
Figure 44. Pelvis. FF02_Vol1, 6 clusters, 5 replicates.

Dice Coefficient

FF01_VOL2_C6_Rx={pelvis cortex medulla}

Number of Replicates	1(run 1/run 2 /run 3)	2(run 1/run 2 /run 3)	3(run 1/run 2 /run 3)	5(run 1/run 2 /run 3)
Cortex	0.5111/ 0.5110/ 0.5111	0.5111/ 0.5111/ 0.5111	0.5111/ 0.5110/ 0.5111	0.5111/ 0.5111/ 0.5111
Medulla	0.2302/ 0.2310/ 0.2303	0.2302/ 0.2302/ 0.2302	0.2315/ 0.2312/ 0.2302	0.2315/ 0.2302/ 0.2312
Pelvis	0.1943/ 0.1945/ 0.1943	0.1943/ 0.1943/ 0.1943	0.1945/ 0.1945/ 0.1943	0.1945/ 0.1943/ 0.1945

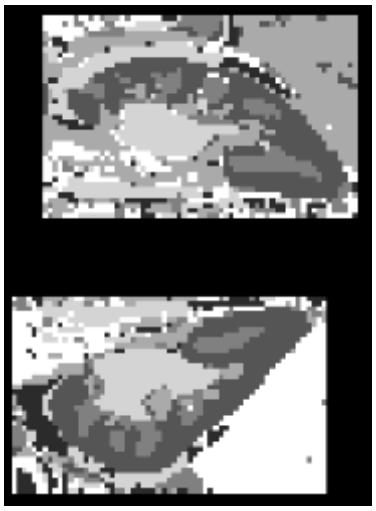
Table 5. Dice coefficient analysis: FF02_VOL1_Manual vs K-means segmentation in 6 clusters.



```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 3.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

In the first run with 3 replicates there is a failure in the 3rd iteration of the K-means. However, this does not affect the result of the clusterization and based on the Dice Coefficient from table 5, the segmentation is consistent.

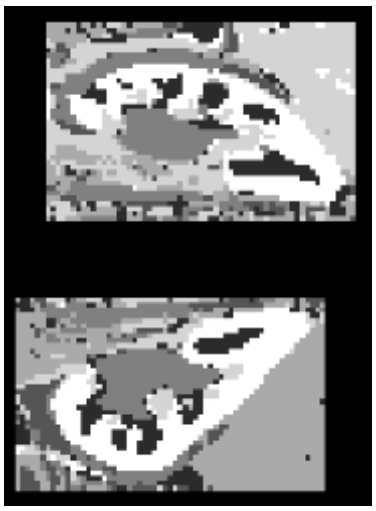
Figure 45. FF02_Vol_1, 6 clusters, 3 Replicates



```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 3.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

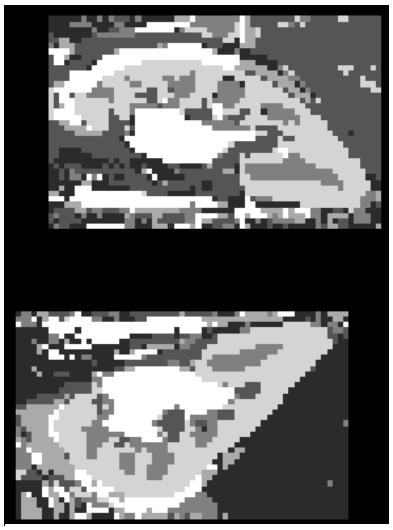
With 3 replicates in the 2nd run the failure happens again in the 3rd iteration. Once again, this does not affect the proper result of the clusterization as it can be observed in table 4.

Figure 47. FF02_Vol_1, 6 clusters, 3 Replicates



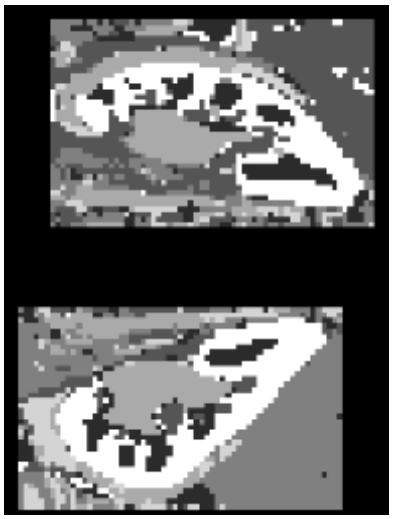
```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 2.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
Warning: Failed to converge in 100 iterations during replicate 3.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
Warning: Failed to converge in 100 iterations during replicate 4.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

Figure 46. FF02_Vol_1, 6 clusters, 5 Replicates



```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 1.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

Figure 48. FF02_Vol_1, 6 clusters, 5 Replicates



```
Warning: Failed to converge in 100 iterations during replicate 4.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 31)
```

Figure 49. FF02_Vol_1, 6 clusters, 5 Replicates

In all of the 3 runs of with 5 replicates, up to 2 failures in the same execution can be spotted. Despite this, the coefficient reports that the clusterization has been done as it would have been without having these failures (Table 4).

Segmentation in 7 clusters

The final analysis carried on this thesis will be the segmentation in 7 clusters of the first volume from the second patient. Once again, to illustrate as an example of the 3 main parts of interest, 5 replicates will be used.

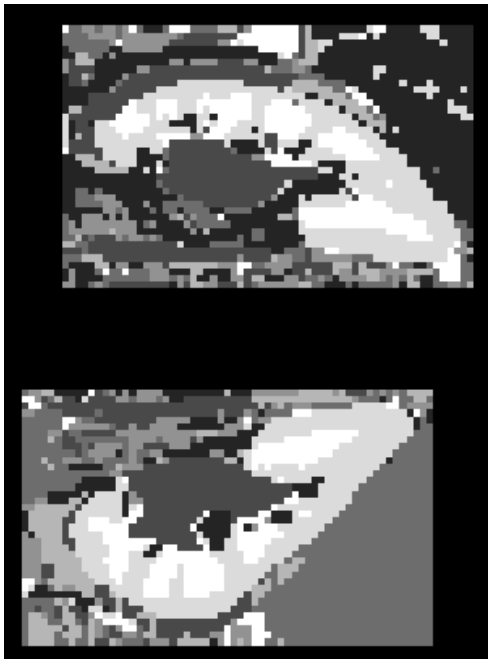


Figure 50. K-Mean segmentation in 7 clusters of FF02_Vol_1.

Medulla

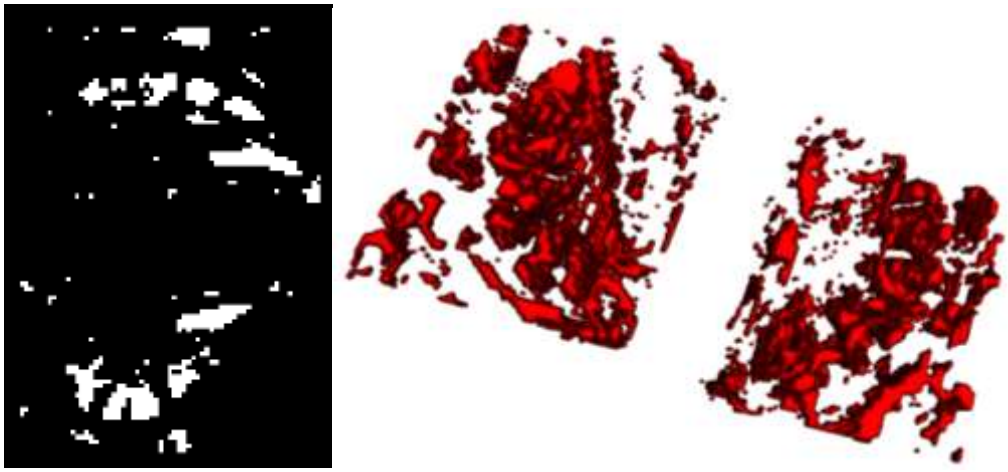


Figure 51. Medulla. FF02_Vol1, 7 clusters, 5 replicates.

Cortex

Figure 52. Cortex. FF02_Vol1, 7 clusters, 5 replicates.

Pelvis

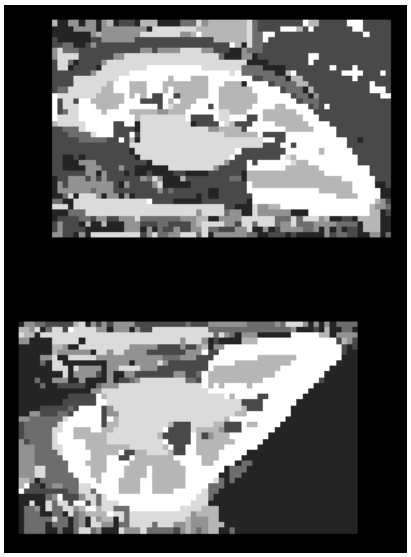
Figure 53. Pelvis. FF02_Vol1, 7 clusters, 5 replicates.

Dice Coefficient

FF01_VOL2_C7_Rx={pelvis cortex medulla}

Number of Replicates	1(run 1/run 2 /run 3)	2(run 1/run 2 /run 3)	3(run 1/run 2 /run 3)	5(run 1/run 2 /run 3)
Cortex	0.4973/ 0.4965/ 0.4939	0.4966/ 0.4968/ 0.4973	0.4968/ 0.4968/ 0.4965	0.4967/ 0.4968/ 0.4966
Medulla	0.3740/ 0.3754/ 0.2229	0.3756/ 0.3740/ 0.3740	0.3745/ 0.3753/ 0.3755	0.3745/ 0.3750/ 0.3755
Pelvis	0.2066/ 0.2073/ 0.1963	0.2070/ 0.2067/ 0.2066	0.2066/ 0.2072/ 0.2073/	0.2066/ 0.2073/ 0.2073

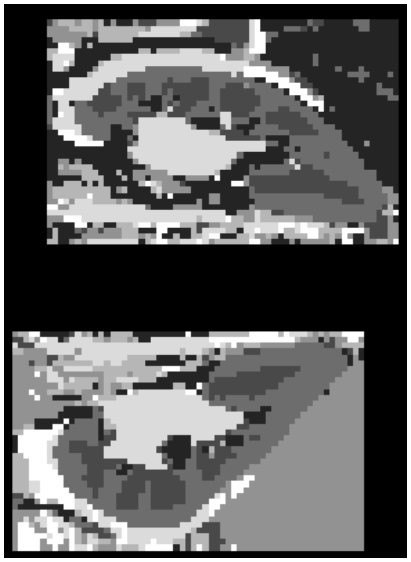
Table 6. Dice coefficient analysis: FF02_VOL1_Manual vs K-means segmentation in 7 clusters.



```
Warning: Failed to converge in 100 iterations during replicate 1.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 33)
```

Figure 54. FF02_Vol_1, 7 clusters, 2 Replicates

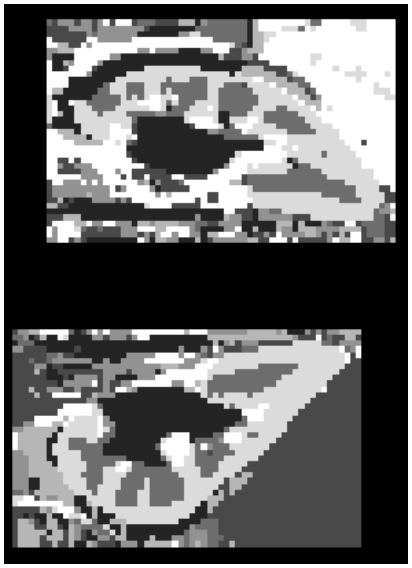
At the 1st run with 2 replicates there was a failure in the 1st iteration. However the result of the clusterization shows that it was done properly, consistent Dice Coefficient (table 6)



```
Warning: Failed to converge in 100 iterations during replicate 1.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 33)
```

Figure 55. FF02_Vol_1, 7 clusters, 2 Replicates

In the 3rd run with 2 clusters the event happens again, and as well as the 1st run, the Dice Coefficient is consistent with the best clusterization possible.



```
Warning: Failed to converge in 100 iterations during replicate 1.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 33)
```

Figure 56. FF02_Vol_1, 7 clusters, 3 Replicates

Once again, the failure spotted in the 1st run with 3 replicates does not affect the result of the clusterization.

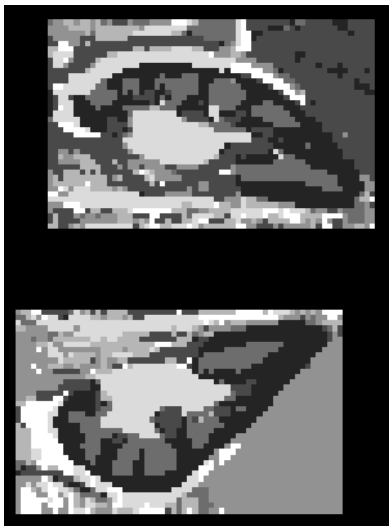


Figure 57. FF02_Vol_1, 7 clusters, 3 Replicates

```
Warning: Failed to converge in 100 iterations during replicate 2.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 33)
```

In the 2nd run we can observe the same event.

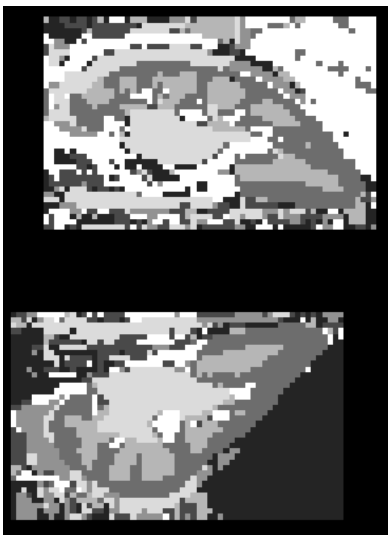


Figure 58. FF02_Vol_1, 7 clusters, 5 Replicates

```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 3.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 33)
```

As well as the 2 previous executions, the 3rd run with 3 replicates is properly done but with failure. Indicate that all of the executions are clustered consistently with the best segmentation possible.

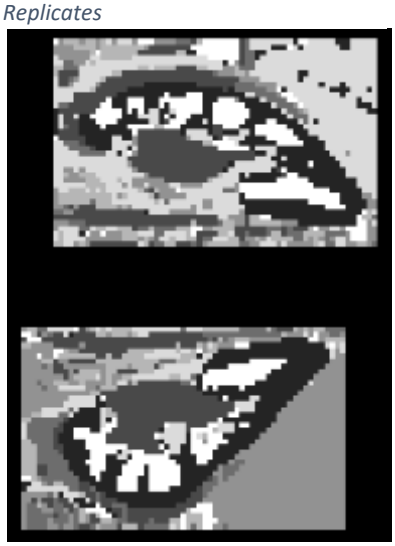
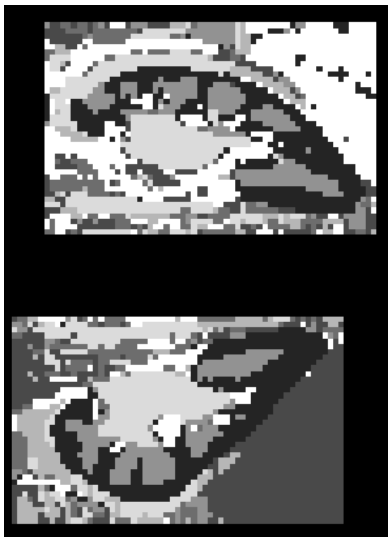


Figure 59. FF02_Vol_1, 7 clusters, 3 Replicates

```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 3.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 33)
Warning: Failed to converge in 100 iterations during replicate 5.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 33)
```

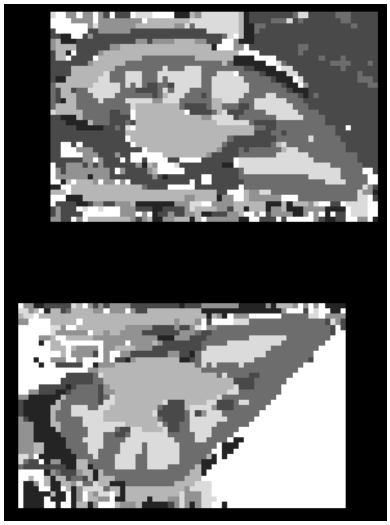
Failure in the 1st run with 5 replicates.



```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 5.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 33)
```

2nd run with 5 replicates.

Figure 60. FF02_Vol_1, 7 clusters, 5 Replicates



```
>> Kmeans
Warning: Failed to converge in 100 iterations during replicate 4.
> In kmeans/loopBody (line 476)
  In internal.stats.parallel.smartForReduce (line 136)
  In kmeans (line 343)
  In Kmeans (line 33)
```

3rd run with 5 replicates.

In the 3 runs with 5 clusters, the segmentation is consistent with the best reachable with 7 clusters.

Figure 61. FF02_Vol_1, 7 clusters, 5 Replicates

Comparing the dice coefficient between the segmentation with 6 and 7 clusters it can be seen an improvement in the medulla cluster, that is also worse clustered than the cortex. Another detail is the consistence of the segmentation along all the experiments, so we cannot see the influence of the number of replicates. The exception can be the 3rd run with a low Dice Coefficient for the medulla and a slightly lower value for the cortex cluster, both highlighted in red in table 6.

5. Conclusions

5.1. Problems and struggles

Manual segmentation FF02_VOL1

The storage orientation of the data from the patient was not the typical that can be found in a medical environment. However, for the purpose of the thesis it did not have any influence in the clusterization. Despite having the patient 1 manual segmentation volumes in the same orientation that the clustered data, the masks from the volume 1 of the second patient was set following the standard orientation. At the time of calculating the Dice Coefficient, I noticed that the values were around 0.05, far away from what could be sensible even for a rough clusterization. To fix it, I had to take the original volume and apply the following corrections as can be seen in figures 62, 63 and 64.

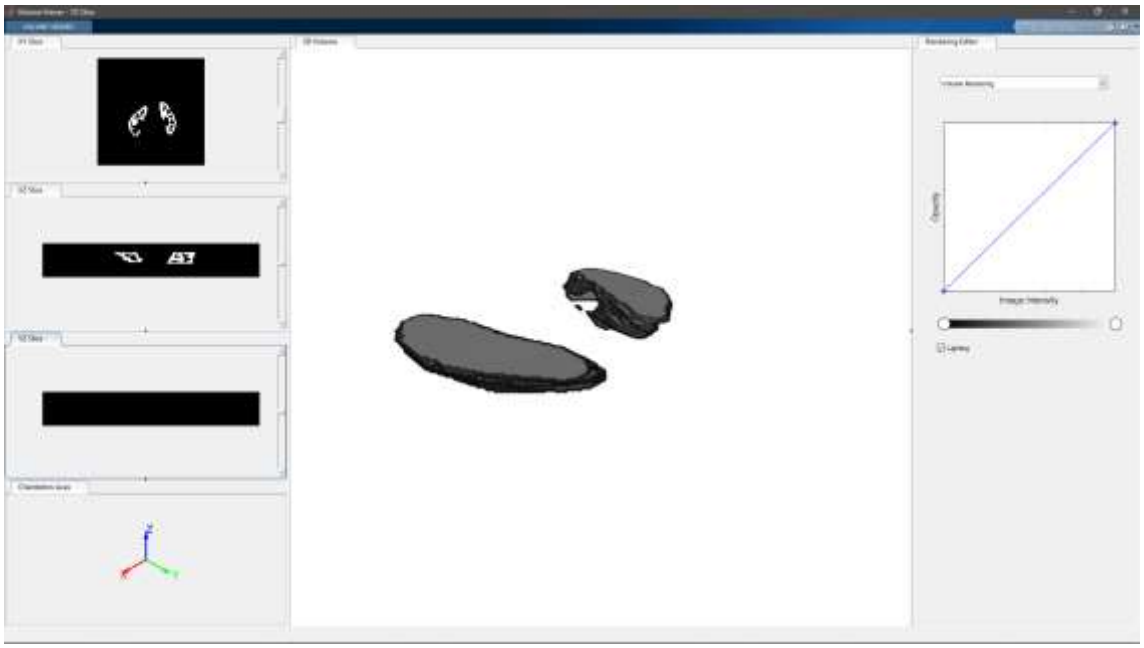


Figure 62. Original FF02_VOL1_Manual volume orientation.

The first step was to rotate -90º clockwise the axial plane or Z-axis. The result of this action is shown in figure 61.

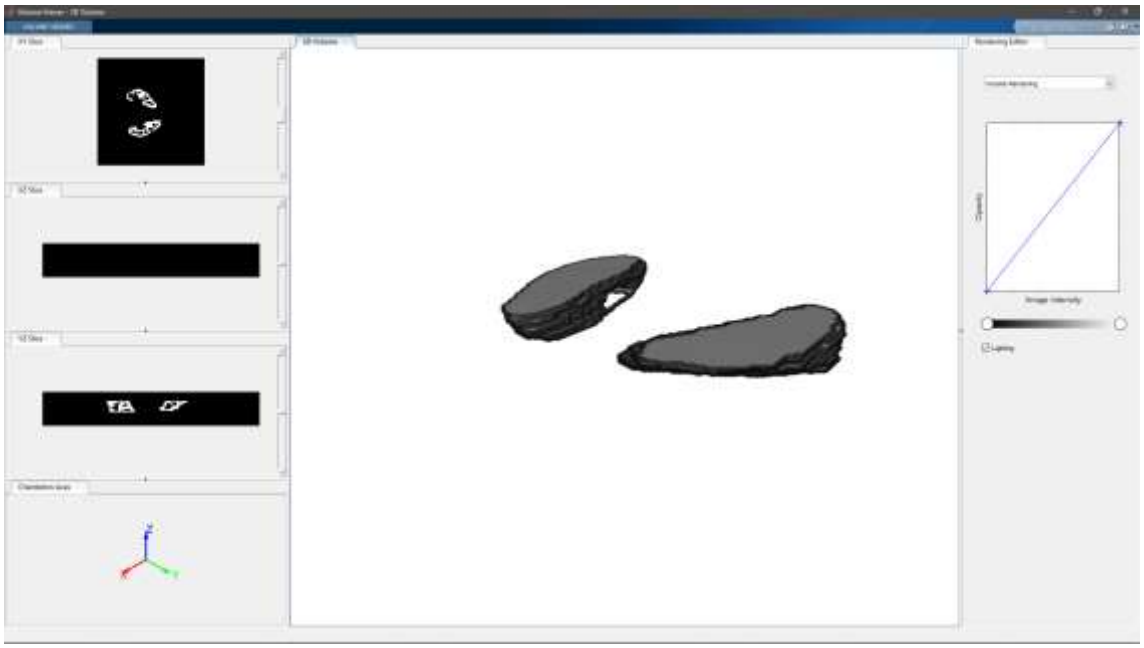


Figure 63. Volume after rotation along Z-axis.

The second step was to apply a mirror effect along the X-axis to the volume so it will have the same coordinates system than the patient dataset (figure 64).

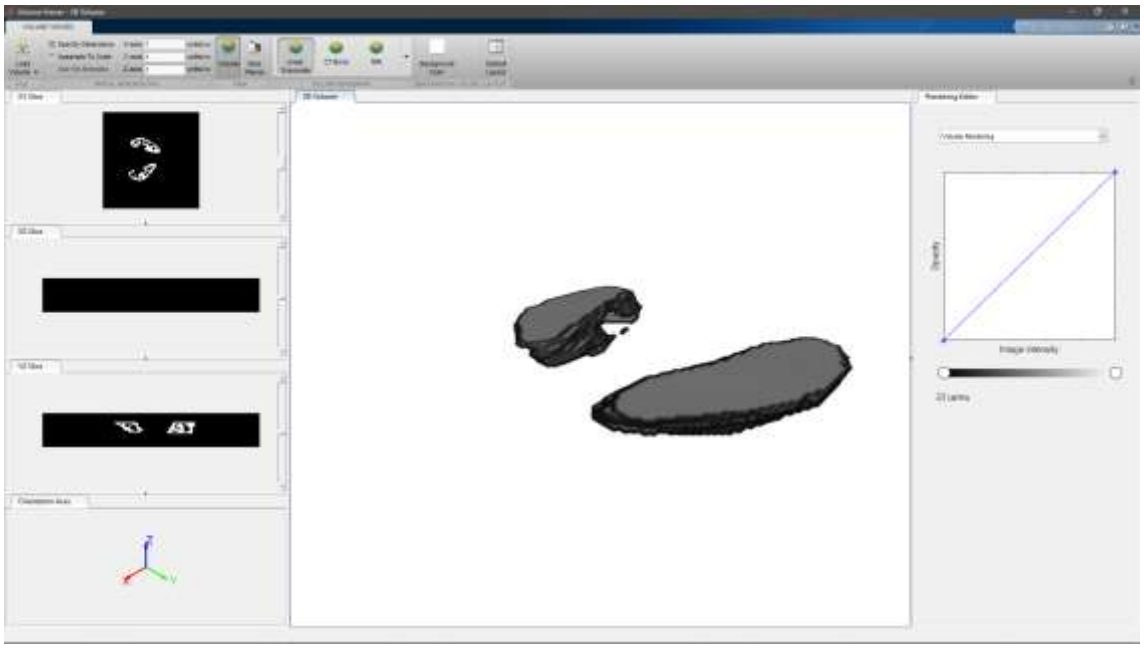


Figure 64. Volume after mirroring along X-axis.

5.2. Influence of the number of clusters

After all the executions using K-means we can observe a big improvement from 6 to 7 clusters in the volume belonging to the second patient. The Dice Coefficient is increased from 0.231 to 0.375; this is a 60% of improvement. It is true that those clusterization are far from being valid and we can extract that a larger number of clusters will provide a better segmentation until a certain number of cluster where we would observe an oversegmentation.

In the case of the volume number 1 belonging to patient number 1 the improvement in the dice coefficient is also noticeable but not as high as the case just analysed in the previous paragraph; from 0.628 to 0.645 it is about a 3% improvement in the medulla. The cortex is hardly improved too, from 0.492 to 0.505, another 3%.

For the volume number 2 of the same patient, the improvement is similar due to the similarities of both datasets, despite the fact that we need 8 or 9 cluster. This can be explained because a larger volume needs a larger number of clusters as it will include a higher number of different tissues in the ROI. Alternatively, the size of them could be different and in case of being bigger, a higher amount of different tissues can be recognised, as if we were zooming in the volume, 'increasing' the resolution in an indirect form. The values in this case vary from 0.622 to 0.625, about 0.5% for the medulla, and from 0.539 decreases to 0.535, about 0.7%.

To summarise, this results show the importance of the number of clusters and its impact to the segmentation accuracy. The goal of future rehearsal should be finding the optimal number of clusters with the exact same data since a low amount leads to an under segmentation that includes several voxels not belonging to the tissue of interest. On the other hand, if the amount of cluster is too large, we can suffer from overclusterization and have the tissue of interest split along different clusters. A parameter of interest from K-means will be the 'distance', minimizing the intracluster and maximizing the intercluster one.

5.3. Influence of the number of replicates

Focusing on the dataset volume 1 belonging to patient 1 (table 1) we can observe in the execution with 5 clusters that the only configuration of the parameters were consistent Dice Coefficients can be seen is with the highest number of replicates, 5. With 6 cluster, no difference along the Dice Coefficient calculation (table 2) was noticed. In the second volume (table 3 and table 4) the same event happens as in table 1. Only the Dice Coefficients obtained with 5 replicates are consistent. Finally, in the last dataset (table 5 and table 6) only one experience differs from the rest of the data, the case with only one replicates.

To conclude, a desirable value of 5 replicates will be the minimum amount to work with. In case that the K-means computation is high resources demanding, 3 replicates could be enough to have proper results.

5.4. Alternative approaches

During the development of the thesis, some interesting parameters about the intensity were found and could lead to a preprocessing of the data to help the K-means with the clusterization. First of all a lot of noise in the signal was spotted in form of frequent spikes and a not smooth plot.

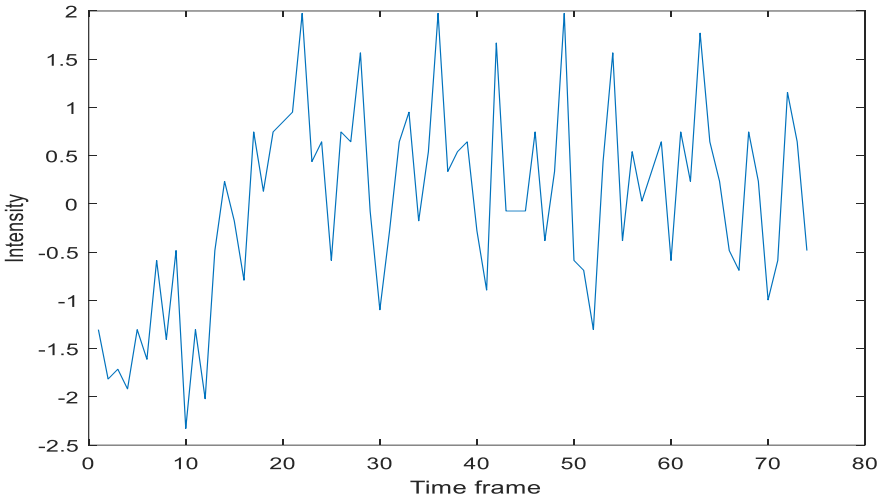


Figure 65. Example of voxel 1321 in FF01_VOL1, variable Z12

Once smoothed, we could use the difference in intensity between a couple of time frames, for example 5 frames. With this simple operation we could be able to maximize the fast spikes produced by the arrival of the contrast bolus. This will link all the voxels whose intensity is not variable along time, independently of the absolute value of its intensity.

Bibliography

1. MACscience. (2018). Homeostasis. [online] Available at: https://macscience.wordpress.com/level-3-biology/homeostasis/inside-kidney/ [Accessed 10 Jun. 2018].
2. Bae KT, Commean PK, Lee J., Volumetric measurement of renal cysts and parenchyma using MRI: phantoms and patients with polycystic kidney disease. J Comput Assist Tomogr 2000; 24:614-619.
3. De Priester JA, Kessels AG, Giele EL, et al. MR renography by semi-automated image analysis: performance in renal transplant recipients, J. Magn. Resonance Imaging 2001; 14:134-140.
4. Pohle R., Toennies KD, A new approach for model-based adaptive region growing in medical image analysis, Computer Analysis of Images and Patterns Journal, 2001; 238-246.
5. Abiria B, Parka B, Chandaranaa H, Mikheeva A, Leeb VS, Rusinek H, Performance of an automated renal segmentation algorithm based on morphological erosion and connectivity, Proceedings Of SPIE 9035, no. 1 (March 18, 2014).
6. Gloger O, Tonies KD, Liebscher V, Kugelmann B, Laqua R, Volzke H. Prior shape level set segmentation on multistep generated probability maps of MR datasets for fully automatic kidney parenchymal for volumetry. IEEE Trans Med Imaging 2012; 6:70-76.
7. Song T, Lee VS, Rusinek H, Chen Q, Bokacheva L, Laine AF, Segmentation of 4D MR Renography Images Using Temporal Dynamics in a Level Set Framework, Proc. IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 2008; 37-40.
8. Hodneland, Erlend & Hanson, Erik & Lundervold, Arvid & Modersitzki, J & Eikefjord, Eli & Zanna, A. (2014). Segmentation-Driven Image Registration-Application to 4D DCE-MRI Recordings of the Moving Kidneys. IEEE transactions on image processing: a publication of the IEEE Signal Processing Society.
9. Mushrif, Shreyas & Morales, Aldo & Sica, Christopher & X. Yang, Qing & Eskin, Susan & Sinowa, Lawrence.(2016). A novel intuitionistic fuzzy set approach for segmentation of kidney MR images. 1-6.
10. Ali A.M., Farag A.A., El-Baz A.S. (2007) Graph Cuts Framework for Kidney Segmentation with Prior Shape Constraints. In: Ayache N., Ourselin S., Maeder A. (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2007. MICCAI 2007. Lecture Notes in Computer Science, vol 4791. Springer, Berlin, Heidelberg.
11. X. Yang, H. Le Minh, K.T.T. Cheng, K.H. Sung, W. Liu, Renal compartment segmentation in DCE-MRI images, Med. Image Anal. 32 (2016) 269-280.
12. F. G. Zöllner, S. Li, J. Roervik, A. Lundervold, L.R. Schad, Segmentation of renal compartments in DCE-MRI of human kidney, in: Image Signal Process. Anal., 2011: pp. 744–748.
13. R. Chav, T. Cresson, G. Chartrand, C. Kauffmann, G. Soulez, J.A. Guise, Kidney Segmentation from a Single Prior Shape in MRI, in: Int. Symp. Biomed. Imaging, 2014: pp. 818-821.
14. Kubendran C., Malathi R.: A Neural Network Based Kidney Segmentation from MRI Images. International Journal of Advanced Research in Computer and Communication Engineering: Vol. 6, issue 8, 2017.