1 Introduction

In the context of this task, I was asked to segment an anatomical structure of my choice from head/neck CT scans, and compare the result against a reference. Here, I choose to segment the external auditory canal.

1.1 Data

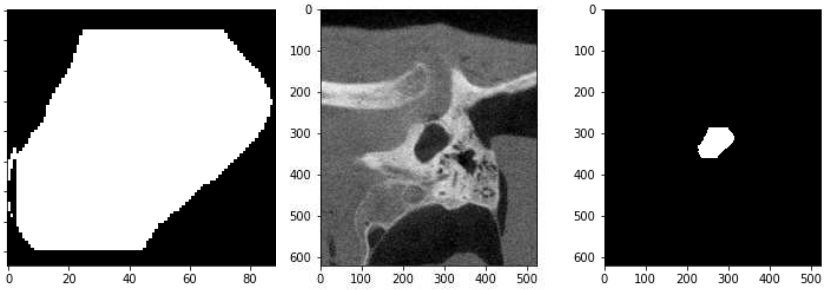
The data consist of 9 volumes coming from either the left (L), or right side (R) from 6 different patients (750, 772_722, 780, 791, 796, 800). For each volume there is the scanner acquisition (image.nii) along with the reference from different anatomical structures. All the data are in NIfTI file format. The first thing that I did was to place the reference segmentations on the volumes in order to see how the different structures are related with each other in space.

1.2 Reference segmentation mappings to the CT

To do so, I used nibabel toolkit for python. Each file contains the affine matrix that transforms the positions in the image in the physical space. Furthermore, it contains the pixel dimensions that is useful to estimate the physical sizes of the anatomical structures of interest. If A array is the affine matrix of the reference segmentation and B is the affine matrix of the CT scan, then the transformation from a reference segmentation coordinate (i, j, z) to a scanner coordinate (i', j', z') is:

$$\begin{bmatrix} i' \\ j' \\ z' \\ 1 \end{bmatrix} = B^{-1} A \begin{bmatrix} i \\ j \\ z \\ 1 \end{bmatrix} \tag{1}$$

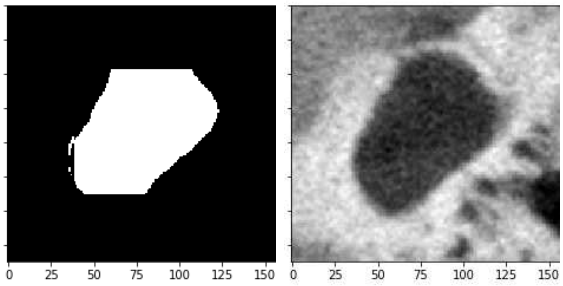
The following figure (Fig. 1) shows an example of how a slice from the reference segmentation of the external auditory canal is mapped to the CT scan using the previous approach in patient 800_L.



(a) Reference slice 38: no mapping

(b) Slice 216 of the original CT

(c) Reference slice 38: mapped



(d) cropped Reference slice 38: mapped

(e) cropped Slice 216 of the original CT

Figure 1: Example of a slice from the 800_L reference segmentation, and its correspondence to the original CT slice. The first row shows: (a) the 38 slice (middle of the segmented canal) of the reference, (b) the corresponding slice (216) in the CT scan of the patient, and (c) the mapped reference in the corresponding position. The second row shows the regions cropped to make the comparison easier.

1.3 Data organization, assumptions, and measurements

In my work, I grouped the data to training set and testing set. The training set consisted of patient 750, 772_722, 780, 791, 796, 800, Left or Right side, while the testing set included only the first patient 750_L. I used the training set to conclude about the dimensions, or the general shape of the auditory canal, and also to train the parameters for my method. Then, I used the testing set to compare my segmentation method against the reference one, and conclude about the performance.

In this work I made the following assumption. I assumed that the auditory canal is a uniform cylinder, with a relative constant diameter which is located perpendicular to one side of the acquired CT scan. At the same time, its dimensions (length and diameter) are approximately similar across the different subjects. So, I can use the reference of training set to estimate an average

length and diameter of the object of interest that I will look for in the test set. In order to ensure that the reference segmentation are corresponding in size, I examined the voxel dimensions given by the CT header files (pixdim nifti header parameter). I found that the voxel sizes are very close to 0.1 for all the volumes and dimensions (x, y, z, pixdim=[0.1, 0.1, 0.10000229]). Therefore, all the mapped reference segmentations are corresponding in size.

The following table (Table 1) summarizes the dimensions (Length and Diameter in pixels) of the segmented auditory canal in the training set. I measured the diameter of the midsection of the cylinder, or the middle slice of the cylinder.

Table 1: Auditory canal reference segmentation measurements

Volume	Length (pixels or slices)	Diameter (pixels in the midsection)
750_R	47 pixels or slices	63 pixels
772_722_L	75 pixels or slices	58 pixels
780_R	82 pixels or slices	51 pixels
791_L	117 pixels or slices	116 pixels
796_L	68 pixels or slices	73 pixels
796_R	107 pixels or slices	101 pixels
800_L	75 pixels or slices	72 pixels
Average	81.57	74.21

Therefore, in the test set (750_L) I can look for a cylinder that it will be perpendicular to one side of the CT scan. The cylinder will have length of approximately 81 slices and diameter of 74 pixels.

2 Methodology

The method that I used was adopted from my research during my doctoral studies. The core of the method is based on the tensor voting methodology [1]. I choose this method because it offers certain advantages against other methods. Firstly, it is an unsupervised method so it does not require training data to work. At the same time it can work under noisy conditions with limited amount of information in the form of fragmented components of the anatomical structures. For example, deep learning methods like U-net require sufficient amount of data in order to not under-fit the problem. At the same time simple methods like region growing would lead to leakage phenomena because the structure has gaps with low intensity.

2.1 Tensor voting

Tensor voting is a bottom-up approach for organizing neighborhood information based on perceptual principles from the Gestalt theory (proximity, similarity, and continuity laws). Essentially, tensor voting is another approach to model the human visual perception system that can be used to segment thin fragmented

structures. It formalizes the observation that high-level perceptual structures can be formed by grouping individual lower level structures. The method is an inference technique that has been showed to be very powerful in reconnecting fragmented information in perceptual curves (Fig. 3). Furthermore, it gives back rich information in the form of a saliency map, the curve direction, and the orientation uncertainty. With modification of its basic components it can resolve visual illusions [6], an aspect that only lately has been investigated by the deep learning community [4].

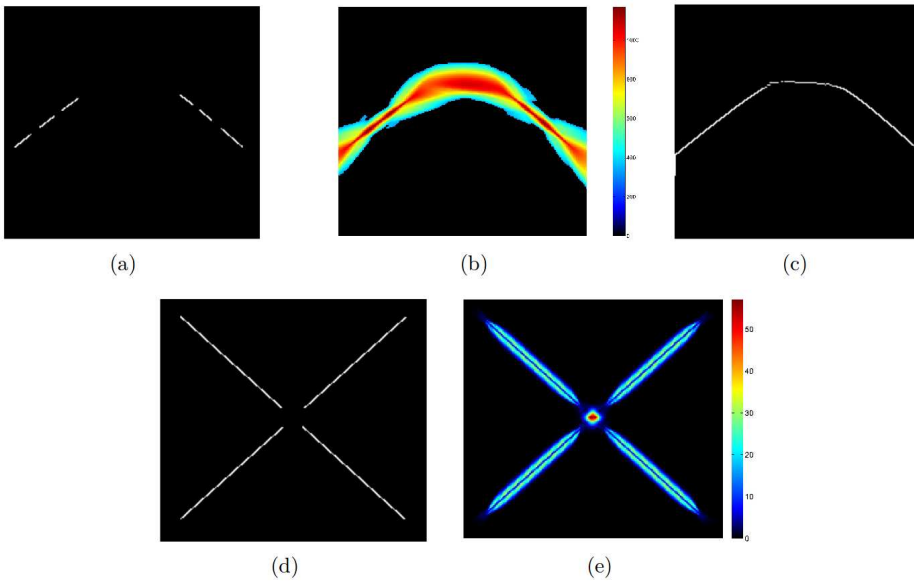


Figure 2: Tensor voting examples in fragmented inputs

Figure 3: The result of the application of the tensor voting approach in a fragmented input line (a-c) and a fragmented junction (d-e). The method works by computing a dense curve saliency map (b, e) by applying the appropriate steps. Furthermore, the orientation uncertainty λ_2 can be extracted (e). As a result, it reconnects the fragments into a single line and infers the existence of the curve perceptually in locations that the information is missing(c). It can also identify the location of junctions perceptually.

The method requires a prior in the form of fragmented components. For this reason, I applied a simple Sobel edge detector. Then, I cleaned the resulted map by removing those pixels that that have a Hounsfield Unit value of $T < 300$. My assumption here was that the contour of the auditory canal matches with the bone, so pixels with HU values more than the T will retain only the necessary information. The tensor voting was applied to reconnected the fragmented information, and extract the centerlines. Figure 4 shows an example in a slice around the ROI of the 800_L subject.

2.2 Deformable surfaces

The result of the tensor voting is a centerline map, where the priors are reconnected to form curves according to perceptual principles. Initially, I was planning to apply a simple flood fill operation to complete the segmentation, however the result is not complete and the contour not continuous. The reason is that there is influence from adjacent structures that increase the orientation uncertainty, fragmenting the final contour (it is visible in the orientation uncertainty map). Therefore, I applied a deformable model method to fully complete the result. I used the following method [5]. The method works by estimating an external force field from a segmentation map, the centerlines in our case, using the convolution with a vector kernel. Then, the scheme iteratively optimizes an initialized curve with the goal of fitting the target contour. The scheme is designed to work for surfaces too, so it is ideal for our case. Figure 5 shows an example where I applied the model on a representative slice from subject 800_L. An initialized spline, manually set in this case, is iteratively deformed based on some internal and external forces to fit the pre-segmented canal.

2.3 Automatisation and parameter tuning

In order to automate the process to be able to run across full volumes, I used a template matching technique. To do so, I analyzed the reference segmentations from all the subjects of the training set. Firstly, I manually found the middle slice of the auditory canal using the information from Table 1. Then, I found the centroids (C_x, C_y) from the segmentation. Table 2 summarizes the centroids coordinates per volume. Given the centroid location and the slice, I isolated a fixed and sufficient area around the center of the circle (55x55 pixels) that was common along all the subjects. Finally, I constructed an average template from all the figures (Fig. 6h). The following figure (Fig. 6) shows the isolated regions per volume, as well as the constructed average template. The template can be updated from other examples, or even other systems. For example, the trachea and the spinal canal have the same shape and morphology as the external auditory canal.

Table 2: Centroid location of the auditory canal midsection

Volume	Centroid row coordinate (C_x)	Centroid column coordinate (C_y)
750_R	527	364
772_722_L	171	341
780_R	549	198
791_L	241	279
796_L	254	368
796_R	156	300
800_L	322	267

Having constructed the template for the midsection of the auditory canal, I used template matching on the test volume (750_L) to find where is the best location to initialize the segmentation process. I used the average template, and I measured the normalized correlation between the image and the template per slice. Figure 7 shows the correlation map for an input slice. To generalize the approach to work for the whole volume, I found the location in 3D that gives the maximum correlation, which in this cases is the point with coordinates: [X=283, Y=355, Z(slice)=200]. This directly corresponds to the middle of the reference segmentation from the data.

The tensor voting has only a single scale parameter that is used to reconnect structures at different distance. There is a direct relation between its value and the searching distance, a small value equal to 15 is appropriate for this application. Alternatively, a multi-scale scheme can be applied to reconnect stuctures at variable distances [2] and remove the influence of this parameter. To tune the deformable model parameters I used a single representative slice from a volume of the training set. The framework is generally very flexible, unless extreme values are used for the model parameters. I noticed that the initialization is more crucial to the convergence of the model than the actual values of the parameters.

3 Results

For my experiments I used the 750_L volume to measure the performance of my segmentation method. Firstly, I used the identified middle point of the auditory canal to initialize a cylinder perpendicularly to the scanner. Its length was the average size of the canal of the training set (around 83 slices), and its diameter was set to around 60 pixels. I used the training set to set the parameters for my method, mainly the deformable model ones.

I compared the performance using the DICE measure as well as the the average contour distance. The DICE is defined as:

$$DICE = \frac{2\,|X \bigcap Y|}{|X + Y|} \tag{2}$$

where X is the reference and Y is the estimated segmentation, while $\bigcap$ denotes the common pixels. For the contour I measured the average Euclidean distance per contour pixel $\mu_{D_{Euclidean}}$.

Figures 8, 9, 10 show the result of the segmentation of my method against the reference in several slices along the cylinder. The first pair (Fig. 8) corresponds to the 40^{th} slice. For this pair the $DICE = 0.8229$ and $\mu_{D_{Euclidean}} = 456.97$ pixels. It seems that the reference is not a completely filled region and it contains some vertical lines exceeding the natural contour. I noticed this in all the volumes, and several slices of the test patient. It is also present in the original non mapped volumes. Overall, my method produces a contour that fits better to the auditory canal bone than the reference. For all the series of slices the average measures are: $\mu(DICE) = 0.7868$ and $\mu_{D_{Euclidean}} = 1336.9$ pixels. In the figures

I also show the slices that gave the best ($27^{th}, DICE = 0.8931, \mu_{D_{Euclidean}} = 340.57$) and the worst ($83^{rd}, DICE = 0.1964, \mu_{D_{Euclidean}} = 733.23$) values for the measures. The second pair (Fig. 9) gives the best results, and if the reference was not corrupted I would have get a $DICE$ very close to 1. For the third pair (Fig. 10) it seems that the reference segmentation covered less anatomical region than my method, probably because it stopped in a previous slice.

4 Discussion

In this report, I presented a new method for the segmentation of anatomical structures from CT scans. My method gives overall good results. However, there are some shortcomings:

1. The assumption that the length of the final cylinder is the average of training set is weak. There is a natural variability in the length of the cylinder that I do not take into account. In fact, for the test set the reference cylinder is longer than what I found.

2. Also, I initialize a general cylinder which again is not optimal, as elongated structures are generally curved along their long axis. A more adaptive initialization would overcome this. The following figure (Fig. 12) shows an example, where I integrated the method in a comprehensive toolkit with a GUI [3]. The initialization in this case is with adaptive cylinders that overall improve the convergence.

3. The assumption that the auditory canal is perpendicular to the scanner side is not favorable. Instead, I could use the relative location to the fiducials to find the angle of entry/exit of the canal.

4. Similar pixel dimensions across the different volumes were observed in this task. In reality scanner resolutions are usually different, therefore the segmentations should be described in physical dimensions.

For the methodological part, even though I argue that deep learning architectures, e.g. the U-net, require lot of data to be trained with, it is a good idea to try a CNN based method and compare its performance against the proposed one as well as the reference, even if it gets overfitted.

Regarding the segmentation of other structures. Some structures are more fine and elongated than the auditory canal. For example, I noticed that the facial nerve is a small elongated nerve inside a canal that traverses the patient's head from bottom to the top. At some point it curves moving in parallel to the head base. My method clearly will not work in this case, probably a method that it will rotate the plane will be more suitable. However, it could be interesting to investigate this direction.

The fiducials are easy to segment, as their purpose is to be easily distinguishable from the rest of the anatomical structures. They are usually made of titan or gold, so they have the maximum available HU units.

The Incus, Malleus, and Stapes, are bony structures that have a specific shape that I do not expect to be significant variant between the subjects. In this case, we can use an atlas to construct an average shape, and use this to adjust it and segment the new incoming bone structures.

References

[1] Argyrios Christodoulidis. *Segmentation and characterization of small retinal vessels in fundus images using the tensor voting approach*. PhD thesis, Ecole Polytechnique, Montreal (Canada), 2017.

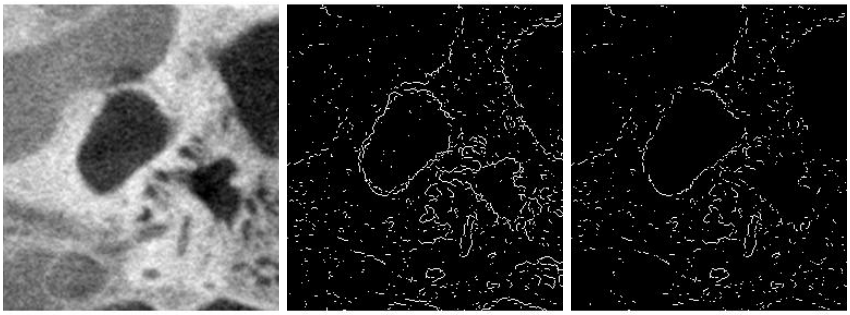
[2] Argyrios Christodoulidis, Thomas Hurtut, Houssem Ben Tahar, and Farida Cheriet. A multi-scale tensor voting approach for small retinal vessel segmentation in high resolution fundus images. *Computerized Medical Imaging and Graphics*, 52:28–43, 2016.

[3] Konstantinos K Delibasis, Argiris Christodoulidis, and Ilias Maglogiannis. An intelligent tool for anatomical object segmentation using deformable surfaces. In *Hellenic Conference on Artificial Intelligence*, pages 206–213. Springer, 2012.

[4] Been Kim, Emily Reif, Martin Wattenberg, and Samy Bengio. Do neural networks show gestalt phenomena? an exploration of the law of closure. *arXiv preprint arXiv:1903.01069*, 2019.

[5] Bing Li and Scott T Acton. Active contour external force using vector field convolution for image segmentation. *IEEE transactions on image processing*, 16(8):2096–2106, 2007.

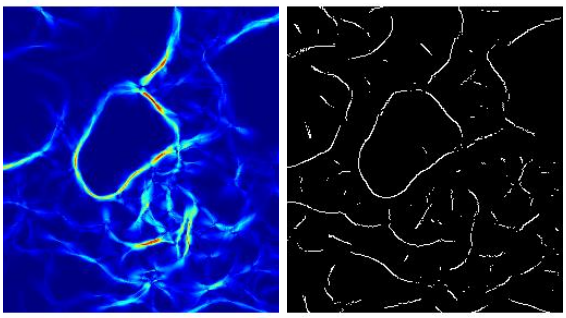
[6] Philippos Mordohai and Gérard Medioni. Junction inference and classification for figure completion using tensor voting. In *2004 Conference on Computer Vision and Pattern Recognition Workshop*, pages 56–56. IEEE.



(a) 800_L: Cropped region around the auditory canal

(b) Edge detection using Sobel to find the priors

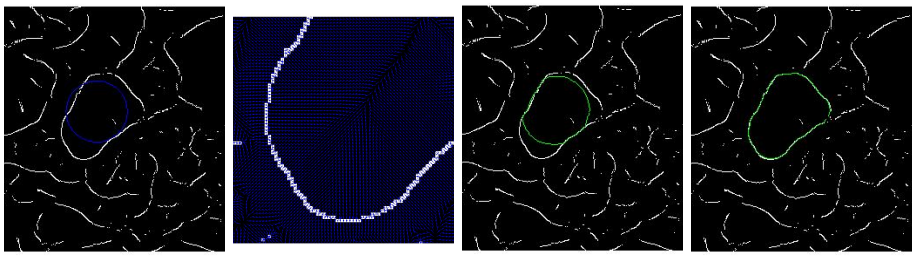
(c) Simple thesholding to isolate dense pixels



(d) Saliency map from the tensor voting

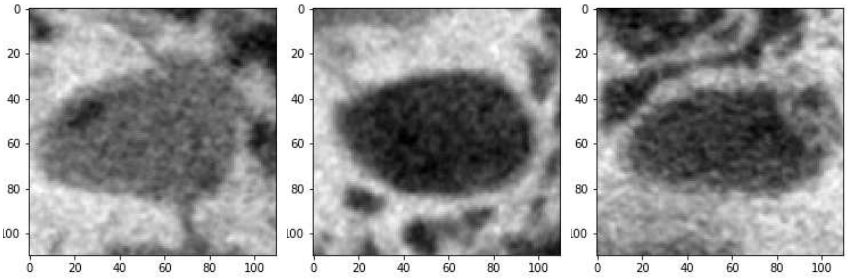
(e) Centerlines map from the tensor voting saliency

Figure 4: Example of the tensor voting application in an interesting region. The priors for the tensor voting are isolated by Sobel edge detection (b), and then the image is cleaned keeping the denser pixels (c). This is used as an input for the main tensor voting framework to extract the saliency (d), and subsequently the centerlines (d).



(a) Initialization (b) F_{ext}: vector field (c) 5^{th} iteration (d) Final result

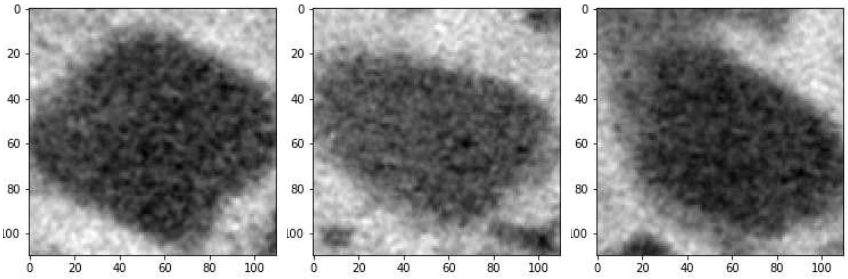
Figure 5: Example of the deformable model method on a typical ROI from subect 800_L. A spline is initialized close to the region of interest (a), then the external force F_{ext} is computed from the centerlines (b), and finally the contour envolves according to the framework (c, d) to extract a complete result.



(a) 750_R: Sampled canal midsection

(b) 772_722_L: Sampled midsection

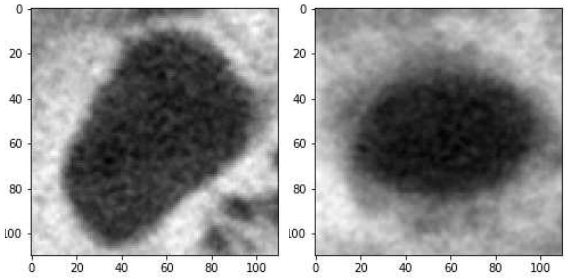
(c) 780_R: Sampled canal midsection



(d) 791_L: Sampled canal midsection

(e) 796_L: Sampled canal midsection

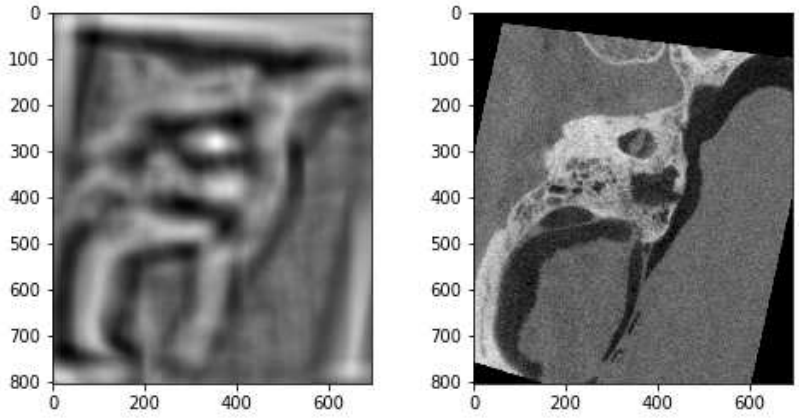
(f) 796_R: Sampled canal midsection



(g) 800_L: Sampled canal midsection

(h) Average representative template

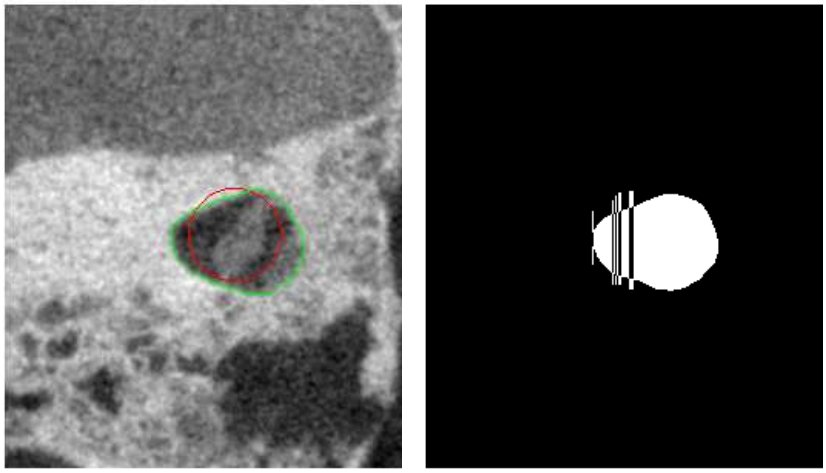
Figure 6: Sampled midsections from the training set, and the average template. The first seven figures (a-g) show the manually sampled slices, while the last (h) is the average representative template that I used in the test volume. The average can be used to new cases, or get updated.



(a) Template matching correlation map

(b) Corresponding slice in the original CT of the test set

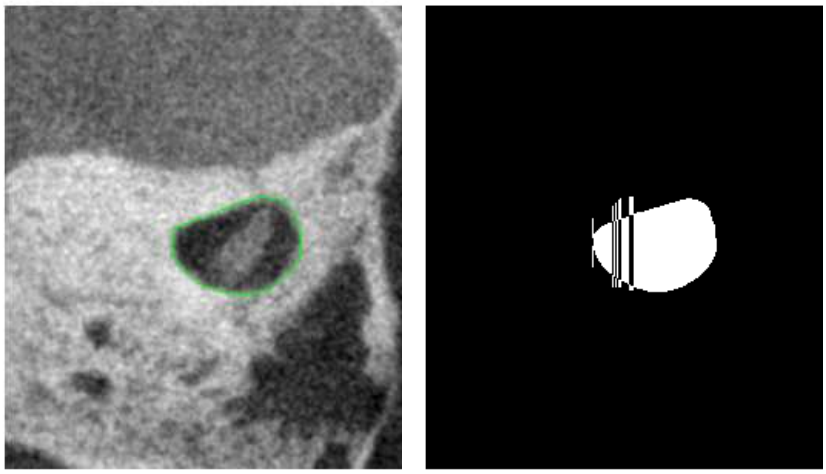
Figure 7: Example of the application of template matching in a single slice of the test set. The average template is applied across the slice and the correlation is measured (a). The maximum, which is the brightest spot, corresponds to the center of the auditory canal in figure (b).



(a) My method: Active contour evolution

(b) Reference final segmentation for the corresponding slice

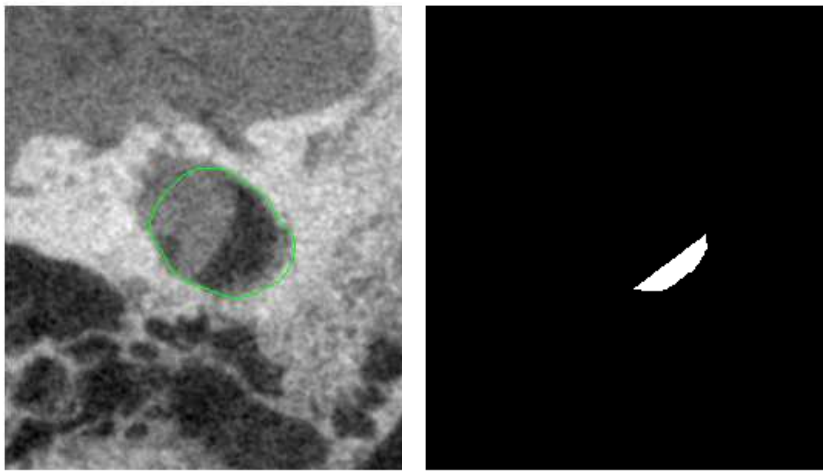
Figure 8: This is the 40^{th} slice from the test volume. The left figure shows the evolution of the contour, red is the initialization and green is the final converged position. The right figure is the reference segmentation for the same slice.



(a) My method: The slice that corresponds to the best $DICE$

(b) Reference final segmentation for the corresponding slice

Figure 9: This is the 27^{th} slice from the test volume that gives the best measures: $DICE = 0.8229$ and $\mu_{D_{Euclidean}} = 456.97$



(a) My method: The slice that corresponds to the worst $DICE$

(b) Reference final segmentation for the corresponding slice

Figure 10: This is the 83^{rd} slice from the test volume that gives the worst measures: $DICE = 0.1964$ and $\mu_{D_{Euclidean}} = 733.23$.

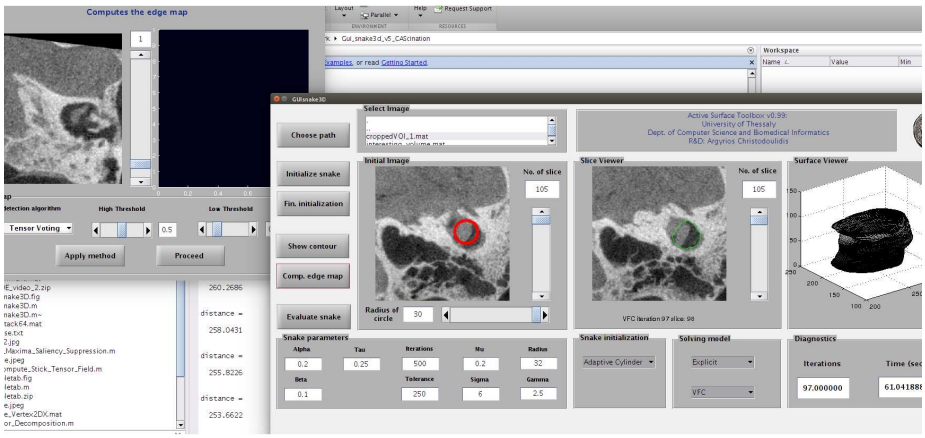


Figure 11: Example of a GUI for adaptive cylinder initialization

Figure 12: This figure shows the GUI that I had developed for image segmentation. It offers an adaptive way to initialize cylinders by modyfing the center of the spline across the different volume slices.