POLITECNICO DI TORINO

Master degree course
in Biomedical Engineering

Master Degree Thesis

Robust segmentation of Corpus Callosum in Multi-Scanner pediatric T1-w MRI using Transfer Learning

Supervisors

Prof. Filippo MOLINARI

Prof. Isabelle BLOCH

Co-Supervisor

Prof. Pietro GORI

Candidate

Giammarco LA BARBERA

ACADEMIC YEAR 2018-2019

To My Aunt Angela

Acknowledgements

I would like to use this opportunity to express my gratitude to everyone who supported me during this journey of six months.

First, I want to express my immense gratitude to my thesis supervisors: Prof. Isabelle Bloch at the Télécom ParisTech for her infinite kindness and availability, for all the opportunities she gave me and for guiding me throughout the project and giving me a new point of view to approach every kind of problem; Prof. Filippo Molinari at the Polytechnic of Turin, for the passion he transmitted to me regarding the medical imaging world, for having convinced me to do this thesis project abroad and for his solidarity and closeness despite the distance; Prof. Pietro Gori at the Télécom ParisTech for his patience and support, for being a friend rather than a chief, for his greatness in explaining new topics and opening me to new horizons.

I would also like to thank all of the people at the IMAGES team at Télécom ParisTech for having created a work environment in which one can breathe friendship, feel welcome and above all, share interest in all possible future developments in the world of images. In particular, a remark goes to the Italian group: Alessandro, who I could always count on and who has been a point of support throughout these months; Alessio, who has helped me become integrated to this new reality; Emanuele, who is always ready to provide me with new ideas.

I thank all the people I met during these months: friends from all over the world who soon became a family that will always have a place in my heart, someone more than others.

I also would like to thank all my friends in Italy: the old ones, the new ones and those of a lifetime. They are my prize in this life and through their friendship the distance is minimal.

Last but not least, I must express my very profound gratefulness to my huge family: my Mother, my second heart, for being by my side in all those moments in spite of everything; my Father, who has always been my source of inspiration and push to always aim high; my Sisters, for always trying not to make me miss home, confiding in me, giving me laughs but also staying close in the dark moments; my Grandparents, Uncle Fifo, Aunts, Uncles, Cousins, my little Cousin Pietro in particular, for being united more than ever in these particular months in which at the end a new star in the sky is born.

All this work of six months goes to that Star.

Thank you all.

CONTENTS

Introduction

In this work we present a robust method to segment the Corpus Callosum in Magnetic Resonance Images (MRI) based on Convolutional Neural Networks (CNN) and Transfer Learning. The thesis was carried out throughout an internship of six months at LTCI (Laboratoire Traitement et Communication de l'Information) of the Ecole Nationale Supérieure des Télécommunications (TELECOM ParisTech), and is part of a much larger French-Brazilian joint project, named ANR-FAPESP Project STAP, in which main topic is spatio-temporal analysis of pediatric MRI.

The Corpus Callosum (CC) (Fig. 1) has been intensively studied in adults and it has been shown, for instance, that there is a relationship between its morphology and some neurological diseases [1] [2]. Fewer morphological studies have been conducted on children showing that variations in size and shape might be correlated with Multiple Sclerosis (MS) [3] or other inflammatory diseases [4].

T1-weighted MR scans are usually employed since the CC is entirely visible and distinguishable in the Mid-Sagittal Plane (MSP). Such studies require an accurate segmentation of the CC. Manual segmentation might be quite tedious, time-consuming and prone to inter-user variability. State-of-the-art segmentation algorithms for adults, such as CCSeg [5] and ART-yuki [6], are difficult to use with pediatric images due to the low contrast-to-noise ratio, short acquisition time (hence low resolution), presence of other parts of the body (e.g. neck), and higher anatomical variability related to the child development. CCSeg is a semi-automatic method that needs a tissue label map, often difficult to compute in pediatric images. The automatic method ART-yuki requires a high resolution to correctly extract the MSP which is rare in pediatrics. To overcome these difficulties, we propose an automatic method based on Convolutional Neural Networks (CNN), which is able to work with multi-scanner and multi-protocol T1-w MR pediatric images (Fig.2): we first train a CNN on adult T1-w MR images, then perform Transfer Learning on pediatric images with different scanners and protocols to improve generalization power.

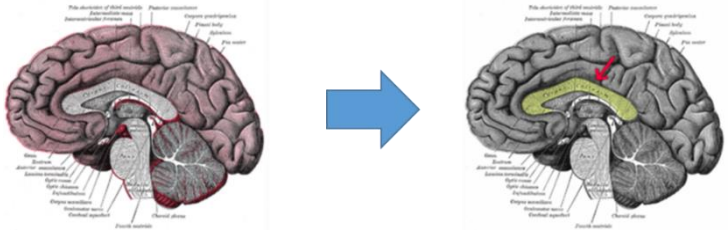


Figure 1. On the left a cross-section of brain, on the right the Corpus Callosum can be seen in the center. Adapted from **[7]**

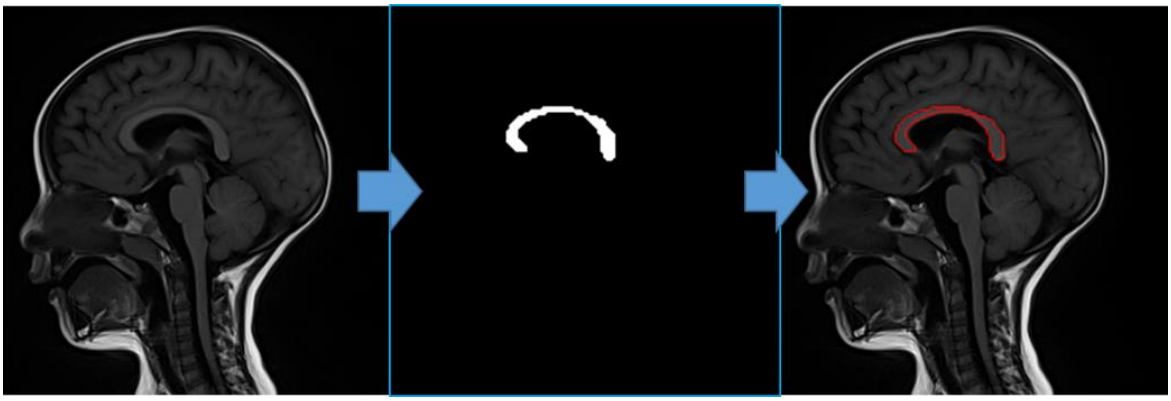


Figure 2. Example of the performance of our work

Pediatric radiologists of the Bicêtre Hospital of Paris (Doctor Catherine Adamsbaum and Doctor Gonzalo Barraza), who have collaborated in this work, suppose that the Corpus Callosum decreases in size and shape over time in Children who have MS, rather than in those affected by other inflammatory diseases, such as ADEM (Acute Disseminated Encephalomyelitis), transverse myelitis and optic myelitis. The second step of our research work is to verify these assumptions, thanks to the results obtained by our proposed method. This dissertation presents in the first chapter the theoretical foundation and the medical theories, while the second one shows the problems related to the existing segmentation methods and the tools that represent the state-of-the-art, finally illustrating the first examples of Artificial Neural Networks for the segmentation of the Corpus Callosum which have produced excellent results. The third chapter focuses in detail on Convolutional Neural Networks, starting from the first theories on Machine Learning and concluding with the highly effective approach of Deep Learning on using a pre-trained network to train a new one (Transfer Learning). In the fourth chapter we present our methodology which uses what is presented in the previous chapter. The fifth chapter is dedicated to results, evaluation and discussions. In the sixth one we deal with our morphological longitudinal analysis on the CC in MS patients. In the last one we present our conclusions and we suggest some future steps to improve the work.

Chapter 1

Biomedical aspects

In this Chapter we cover introductory topics related to our project. Firstly, we introduce the Corpus Callosum and its relationship with some neurological diseases. Then, we focus on MRI, with a brief explanation of the acquisition technique and a discussion on the difference between the types of MR images. Finally, we understand specifically the MR data used for studies on CC and hence for the segmentation.

1.1 The Corpus Callosum

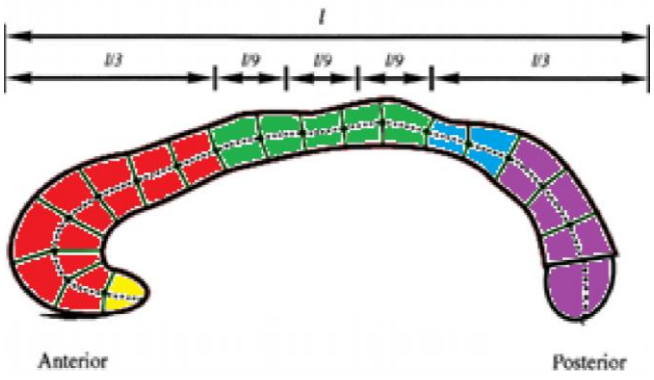


Figure 3. Subdivision of Witelson of the CC: yellow = rostrum, red = genu, green = midbody, blue = isthmus, purple = splenium. Adapted from **[8]**

The CC is composed exclusively of nerve fibers that transfer information between the lobes of the two hemispheres and thus co-ordinate them. According to the subdivision of Witelson, the Corpus Callosum can be divided in 5 parts (Fig. 3), considering the head looking to the left: rostrum, genu, midbody, isthmus and splenium [8]. It is hypothesized to play a fundamental role in integrating information and mediating complex behaviors. Structural changes in the Corpus Callosum may correlate with cognitive and behavioral deficits in neurodevelopmental disorders. Volumetric studies have reported reductions in the size of the CC in autism [1]; lesions in the children's CC were noted in patients with established epilepsy, in patients with seizures and in patients without seizures but with headaches, depression or altered consciousness [2]; Multiple Sclerosis (MS) patients undergoing a longitudinal (serial) study showed change in size and shape of the CC [3] [9].

The Agenesis (absence) of the Corpus Callosum is also important, it can be total or partial (splenius and body): it leads to children with inborn errors of metabolism, inability to couple stimuli coming from both hands or eyes, intellectual deficit and generalized seizures [10].
Examining Corpus Callosum on children, Magnetic Resonance is preferred because it is not harmful and not invasive, but it is difficult to perform because the baby is not standing still, the procedure lasts from 15 to 45 minutes and the machine is closed. CC is the largest white matter structure of the human brain, hence for this reason, MR (1.5 or 3.0 Tesla) images T1-weighted are preferred: White Matter presents a good contrast with Grey Matter. In MRI T2-weighted, homogeneity and contours are lost (because of the cerebro-spinal fluid, which can be suppressed by acquiring in FLAIR mode, usually not in standard protocols).
It must be remembered that until the end of the first year of life, the anatomical structures undergo further modifications, there is no sense in defining the "anatomical confirmation" obtainable with the MRI immediately after birth. A longitudinal study and over time is more indicated, as a precaution it is advisable after the second year of life, when the myelination process is completed (as it causes variations of intensity in the images during the period of the cerebral development) [11].

1.2 Magnetic Resonance Imaging

Before continuing our dissertation, it is important to remind and clarify briefly the key concepts of Magnetic Resonance Imaging. It is a medical image acquisition technique without ionizing radiation. The nuclei of certain atoms (for instance hydrogen, used for its abundance in tissues) are stimulated by a controlled emission of radio waves inside a strong magnetic field. As result, returning to their energy level the nuclei emit a measurable radio frequency (megahertz range). This latter is used to reconstruct images of the interior of the human body.
Consider a piece of tissue, inside we find a very large number of protons. The direction in the magnetic field space, at rest, is random. Statistically magnetization (M, total magnetic charge) is null. If we take the same set and from the outside we apply a static magnetic field (zero frequency), indicated with the symbol B_0, we can magnetize the matter and establish a preferential direction. The higher B_0 is (already 1 Tesla is elevated), the more protons try to put themselves in the same direction. Actually, the resultant spin of all protons tend to two orientations: parallel (same as B_0), in the state of minimum energy, and anti-parallel (opposite of B_0), in the state of maximum energy.

Parallel protons are slightly more prevalent than the antiparallel ones. This small prevalence produces the resulting magnetization M, parallel to B_0 and it is measurable. Furthermore, due to the effect of B_0, the axis of each proton rotates around the direction of the moment of B_0 (precession). Precession movements can be compared to the oscillations of a spinning top.

The magnetization vector alone is not sufficient to create an image because it is static and M has a slightly intense module. We exploit the interaction with the material to create an image. The method used to amplify the tissue response is called Resonance: thanks to an energy exchange between the subject and the external source at the right frequency, a system can acquire more energy than normal.

We irradiate the tissue with a radio-frequency electromagnetic field B_1 at the system's own frequency (determined by the precession, ω_0). The protons radiated to their ω_0 change their quantum state, increasing the overall energy, while others do nothing as it is not their resonance frequency. Specifically, all protons possessing a spin are precess around the axis of vector B_0 with a specific frequency of ω_0, given by Larmon's frequency:

$$\omega_0 = \gamma \cdot B_0 \tag{1}$$

where γ is a given particle's gyromagnetic ratio (it depends on the nuclide we are considering, for hydrogen atom is 42.58 MHz/T). By alternating B_1 at a given particle's Larmor frequency, we obtain resonance, inducing a sinusoidal current at ω_0 whose signal M we measure.

When B_1 is stopped, the protons rotate back to their original position, spinning around the axis of B_0. Hence, the measure signal decays because the field no longer alternates. This decay is called T1-relaxation: the protons are freed of excess energy giving it to the rest of the tissue (surrounding grid). Moreover, the proton's spins go slowly out of phase, no more precessing in-phase. This dephasing process is called T2-relaxation: the protons drift away the excess energy from each other, until they lose it. The value of T1 and T2 is that which is converted into an image.

By appropriately selecting the sequence of RF pulses it is possible to impose a specific dynamic on the spins system, so as to obtain the information from the MR signal. The parameters that influence the image result are Time to Repeat (TR) and Time of Echo (TE) times that can be long or short. By combining long or short TR and TE, you will have images weighted in T1 or T2. Generally, T1-w images have high values for fat and slow-flowing blood, while having lower values for tissues with high water content, the White and Gray

Matter have signals of intermediate intensity. In reverse, liquids or, in any case, highly hydrated tissues, appear bright white in T2-weighted images.

There are many types of techniques dependent on applied sequence of radio-frequency pulses, the most used are Spin-Echo (SE), Inversion Recovery (IR) and Gradient Echo (GE).

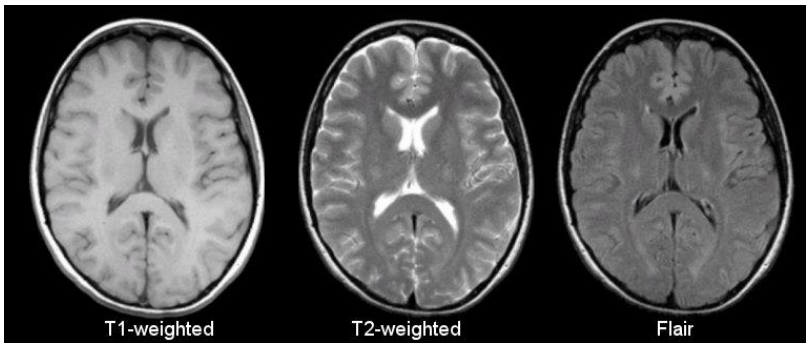


Figure 4. Comparison of T1-w, T2-w and Flair (CSF fluid is attenuated and made dark). Each slice is from the same region of the same patient. Reprinted from **[12]**

A stronger external magnetic field (for instance 3.0 Tesla) results in an increase of the quality of the acquired images but the acquisition area may be overheated (over human body limits) by the machine and the equipment required for such a field.

The 3D final image is composed by distinct slices acquired thanks to the use of a gradient magnetic field. As a consequence, MR images have much higher resolution on the capture's plane than on the others, which are reconstructed from the principal slices.

1.3 Understanding data and problems of segmentation

As mentioned above, the detection of contours of the Corpus Callosum in MR images is generally done in the T1-w images. In MRI, depending on the plane of captures there are three different types of images [13]: Axial (transverse images represent slices of the body), Sagittal (images taken perpendicular to the axial plane which separate the left and the right sides) and Coronal (images taken perpendicular to the sagittal plane which separate the front from the back), shown in Fig. 5.

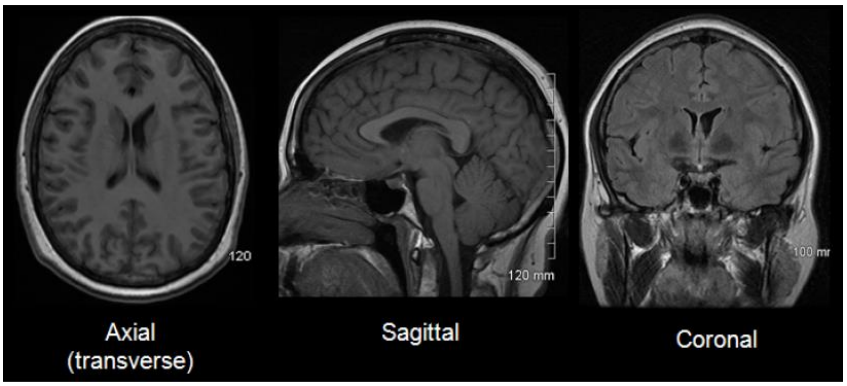


Figure 5. MRI Planes. Reprinted from **[13]**

According to radiologists, the best way to recognize the CC is in the Sagittal T1-w, where the object is entirely visible and distinguishable. A 3D study is not simple because lateral limits of the CC are not intrinsically defined. Indeed, the Corpus Callosum consists of nerve fibers that extend continuously from one hemisphere to another and these do not mark its borders. It is reasonable to work with 2D images, choosing the slice where CC has the best resolution and contours: the median slice (Mid-Sagittal Slice or Plane).

Such studies require an accurate segmentation of the CC. Manual segmentation might be quite tedious, time-consuming and prone to inter-user variability.

Foremost problems are related to the accuracy and reproducibility for the longitudinal study, as well as the possibility of automating the algorithm, for different reasons:

- independence of the algorithm with respect to the variation in the size and shape of the corpus callosum among the patients (no relevant differences in shape between female and male);
- adaptability to different modes (different T1-w MRI, e.g. Spin-Echo, Blade, Turbo, etc.);
- pixels/voxels of MRI are not homogeneous;
- robustness to a background that can vary in intensity both in the image itself and in its entire database;
- CC presents gaps in the contour, bottlenecks or bumps;
- one of the biggest problem is the presence of the fornix (Fig. 6), which is often in contact with the CC in the mid-sagittal MRI, which has the same brightness.

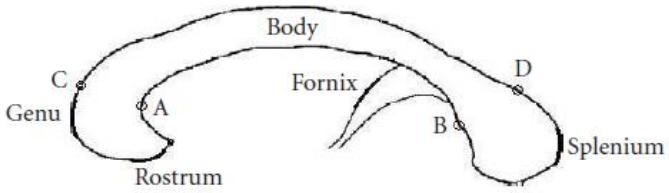


Figure 6. Case in which the fornix is in contact to the CC structure. Reprinted from **[14]**

In addition, pediatric MR images compared to adults ones present a smaller contrast-noise ratio and higher effects of partial volume because of the small dimensions (size) of the cerebral structures (overlapped structures) and because of the brief acquisition time (to limit more possible movements artifacts). To conclude, the smaller the child is, the more the body will be present in the MR image (Fig. 7) bringing further difficulties.

For everything we have dictated, nowadays, there are few studies and algorithms on segmentation of Corpus Callosum in the children's MRI.

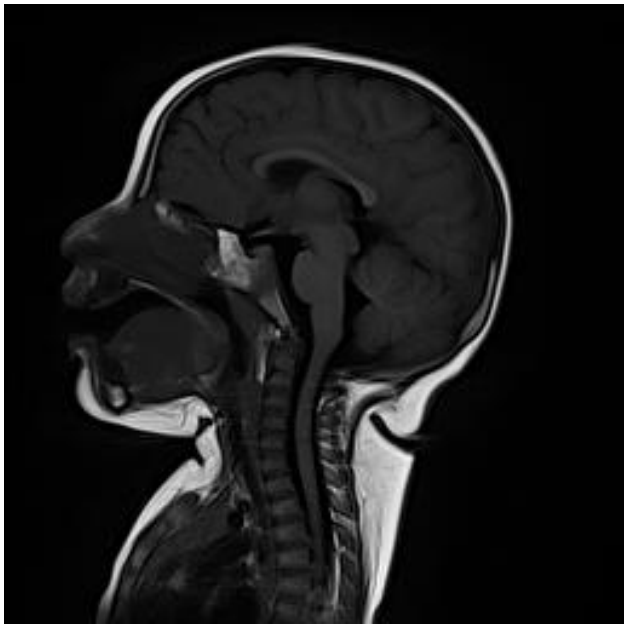


Figure 7. Presence of the body in the brain image of a child

Chapter 2

Literature review: existing segmentation methods

One of the challenges in Computer Vision is subdividing the image into regions that are homogeneous with respect to one or more characteristics of the image itself, as is done by human visual system of recognition. This task is called Image Segmentation and it tries to address this problem by making contours in the objects. The goal is to obtain the best possible contours in order to analyze different parts of the image separately. Generally speaking there is no optimal algorithm for segmentation: the choice depends on the characteristics of the image and the problems it presents. For what concerns our work, the aim is the segmentation of the Corpus Callosum, in particular the recognition of this object, in its two-dimensional form.

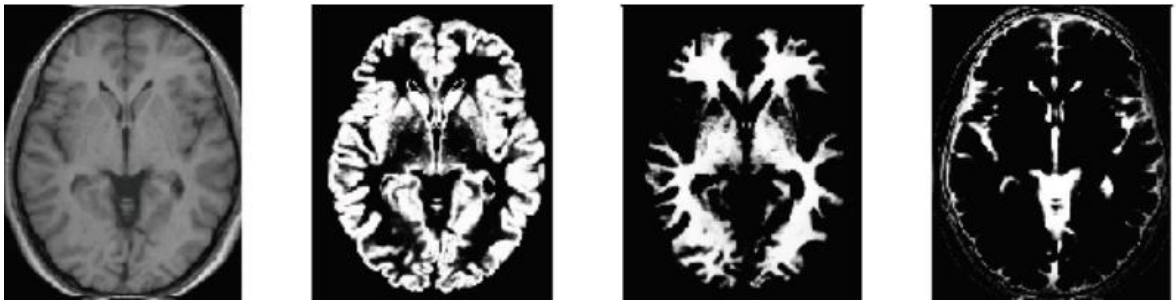


Figure 8. Example of Image Segmentation.
From left to right: MRI Brain slice, Gray Matter, White Matter, Cerebrospinal Fluid

Continuing our analysis, in this section we briefly present some of the solutions developed during these ages for the segmentation of the CC on adults, which present the problems showed in the previous section. Seeing their advantages and disadvantages, it will be easy to understand why they are hardly applicable on children. In general the approaches are based on active contour models or on models that use anatomical atlas.

2.1 Methods based on active contours

Active contours are deformable planar curves under the effect of internal forces due to the curve itself and external forces due to experimental data (the image in question). The balance between the effects of the two forces determines the adaptation of the model to shapes, objects or contours of the image. We distinguish them in parametric (SNAKES) and geometric (LEVEL-SET) models.

In the literature, many image segmentation techniques to properly segment the Corpus Callosum in the brain are based on active contours. Generally speaking, level-set is more

used than snake, because the latter presents difficulties in initialization due to noise and low contrast at the boundaries of the CC. For instance, [15] propose an unsupervised clustering using K-means combined with a subsequent level-set: it is a very simple and fast method, about 30-40 seconds per image. As indicated in Figure 9, the user clicks on a point in the region corresponding to the CC, which makes it possible to extract the associated cluster of pixels; various morphological operations are performed on this region in order to keep only the connected pixels belonging to the corpus callosum; the resulting volume then serves as an initialization for deformation by level-set converging towards the contour surface. However, there are multiple disadvantages: the intensity of the pixels in MR images is not homogeneous and moreover the intensities of the Corpus Callosum and of the White Matter are very close. For these reasons errors may occur upon the choosing of seeds as well as similarity criteria and stop criteria of clustering (as a region growing). One can doubt that this semi-automatic algorithm is intra-patient reproducible.

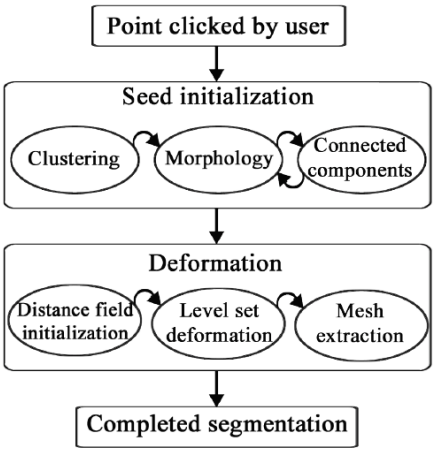


Figure 9. Pipeline of [15]. Reprinted from [15]

In [14], the user instead chooses three well distributed points in the CC; from these three points he can then draw two segments (Fig. 10) and then make a contour detection by Canny filtering on the image. The algorithm is fast (of the order of one second for an image). The main disadvantage lies in the initialization which requires a precise procedure for the segmentation to be correct. The contours of the corpus callosum must also be clear enough to be detected by Canny's filtering.

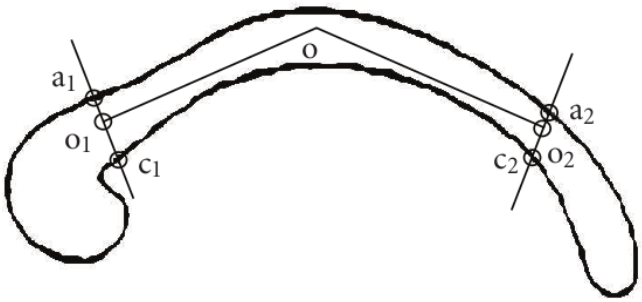


Figure 10. Initialization for the method in **[14]**. Reprinted from **[14]**

In 2012, at TELECOM ParisTech, a new model was drawn based on [16] and [17]. In the first one, the authors prove the existence of a unique and stable formulation of geometric active contours. In particular, they show that their model allows a stable detection of the contours even when their gradients have very variable values. This detection is based on a geodesic approach that consists in finding the points of strong contrasts and smoothing the curve that connects them. In [17] they introduce a local regional approach to level-set. The clue is to change the curve by specifically using the information of the pixels close to the active contour, hence the idea is adapted to the case where the initialization is not too far from the final contour (it covers roughly the shape to be segmented). As a result, only the information around the active contour is interesting to make it converge properly. Taken independently, the two methods presented above, both have their advantages and disadvantages. For example, the [16] is based only on the observation of the gradient of the image but if the object to be segmented is not uniform then the detection of other gradients can interfere; while the second is based on local averages of the intensities and it will produce errors when the outside of the contour is very heterogeneous. The method proposed by TELECOM ParisTech in 2012, and inspired by the two above, is therefore a union of a global method and a local one: the level-set is not calculated in the entire image, but in the area near the initialization as a local approach. However, the method has many limitations: it requires manually choosing the mid-sagittal slice and executing a manual initialization, and requires a choice of set parameters. There is the possibility to correct manually the result of the segmentation to recover the non-high performance however this tool is very hard to control. For all these reasons, the method has been discarded to search for a more robust and automatic one.

A state-of-the-art segmentation algorithm for MRI of adults based on active contours is **CCSeg**. It is an open-source C++-based application that allows automatic, as well as user-interactive segmentation of the Corpus Callosum. It is based on [5] and [18] and we can

define it as a computation mapping method for improving active shape model segmentation. Initialization is done by an atlas-based automatic tissue segmentation via an expectation maximization. An atlas is provided for both adult and pediatric images. It employs a constrained elastic deformation of Flexible contour model [5] that is an extension of Szekely's Fourier based Active Shape Model [18], as seen by following the pipeline in Figure 11.

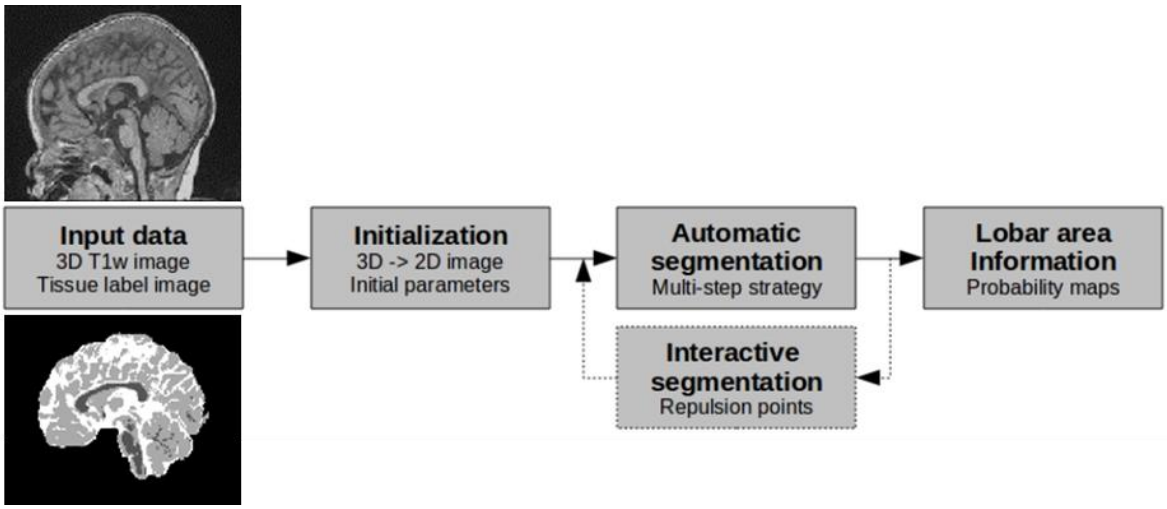


Figure 11. Pipeline of CCSeg. Adapted from **[5]**

The Equations of the Active Shape Model are:

$$E(\boldsymbol{p}) = E(\boldsymbol{r}(t,\boldsymbol{p})) = E_{\mathrm{I}}(\boldsymbol{r}(t,\boldsymbol{p})) + E_{\mathrm{D}}(\boldsymbol{r}(t,\boldsymbol{p})) \tag{2}$$

Where:

$$E_{\mathrm{I}}(\boldsymbol{r}) = \int_0^{2\pi} P(\boldsymbol{r}(t,\boldsymbol{p}))\,\mathrm{d}t, \tag{3}$$

with

$$P(\boldsymbol{r}(t,\boldsymbol{p})) = -\left|\nabla I(\boldsymbol{r}(t,\boldsymbol{p}))\right|, \tag{4}$$

and

$$E_{\mathrm{D}}(\boldsymbol{r}(t,\boldsymbol{p})) = \int_0^{2\pi} \dot{\kappa}^2(t,\boldsymbol{p}) \cdot \left\|\dot{\boldsymbol{r}}(t,\boldsymbol{p})\right\|^2 \mathrm{d}t\,. \tag{5}$$

CCSeg works efficiently with high quality images by providing an easy way to draw a segmentation of the CC. If the initialization of the average CC outline is off, it is not difficult to center on the figure (possibility by updating the section or using the repulsive points). Starting with the 3D T1-weighted image, the Mid-Sagittal Plane is easily defined by default as the average center slice of the image. More precisely, an average image of several center slices (plus/minus 2 slices by default) is computed to define such a 2D plane. This averaging

step results in a reduced intensity of the fornix and thus enhances the success rate of the segmentation procedure. The main problem is that it works only on volumetric images (to extract MSP) and it absolutely needs a high quality Tissue label image (not easy to have with low quality images as pediatric ones), as shown in Fig.11. In addition, it requires some tests on the parameters, especially for images with presence of neck or chest.

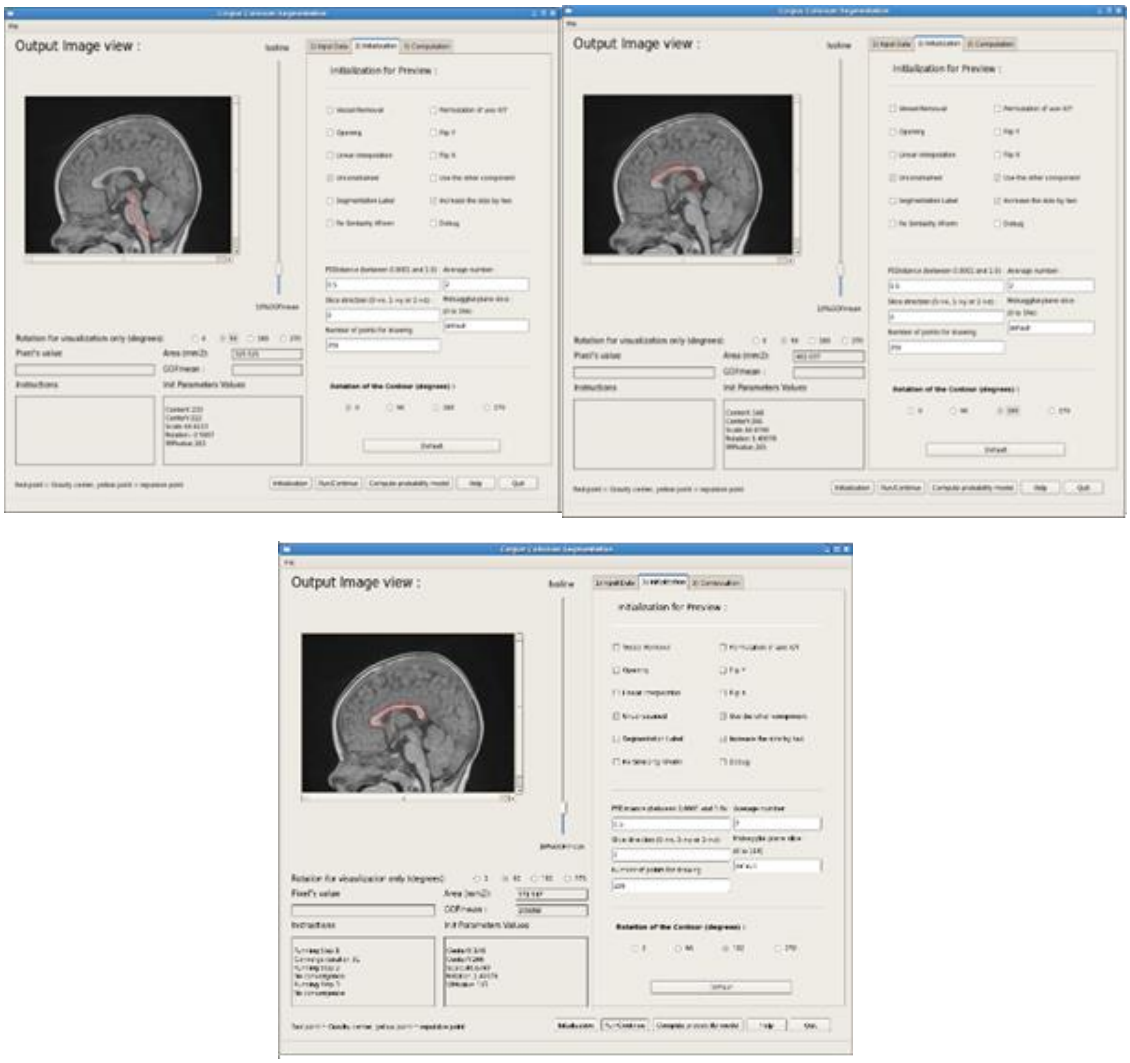


Figure 12. Graphical User Interface of CCSeg with a computation example

2.2 Methods based on anatomical atlas

The idea is to use a large dataset where important regions of interest in the brain are well delineated by expert radiologists. This large dataset gives a good understanding about pixels, value intensity, edges, etc. With this information, one can get a new MR image, that is not presented in the dataset, and match it with the atlas image and identify the region of interest. In [19], the authors proceed by spatially normalizing all the structural images to the same stereotaxic space thanks to a T1-w template image, and then following with a segmentation of the normalized images into Gray Matter, White Matter and Cerebral Spinal Fluid. In the part corresponding to the WM, a region of interest supposed to contain the Corpus Callosum, is selected. In this "box", the authors then do a partitioning operation (clustering). The largest region corresponds to the CC. By doing an inverse transformation, they obtain the segmentation of the Corpus Callosum in the original image. This method is rather simple and gives good results according to the article. It does not seem to be that fast and there are problems due to fornix. The article is however not very detailed and some stages (the number of partitions for example) are somewhat vague.

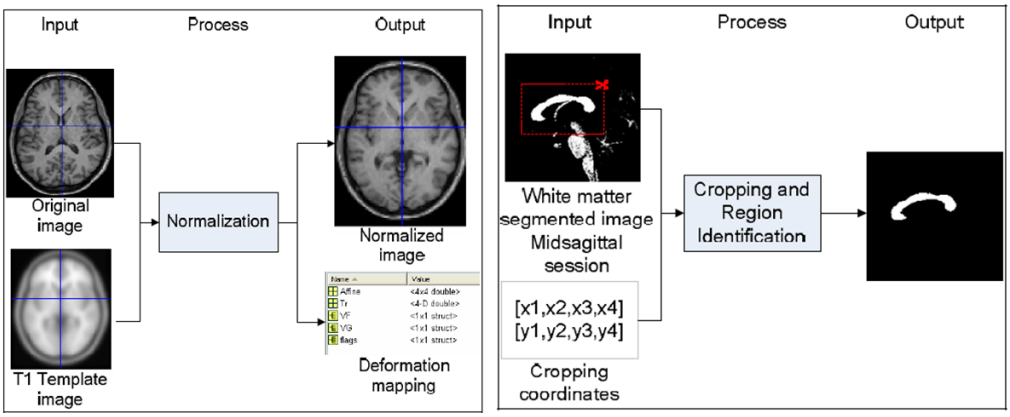


Figure 13. Some passage of the pipeline of [19]. Reprinted from [19]

One of the best results was presented in [20] and then improved in [21], where active contour methods are also used. Through the “adaptive mean shift clustering technique”, the image is first clustered into various homogeneous areas, representing various brain tissues. The CC area is then detected based on area analysis, template matching, in conjunction with shape and location analysis. The boundary of obtained CC area is extracted and evolved under the mechanism of Geometric Active Contour model, for final segmentation of CC structure. This method overcomes the problem of user-guide initialization and has very good performance in terms of segmentation accuracy, but it needs a large atlas database (hard to get as the radiologists should be very involved in the project) in order to perform a good template matching to identify the CC area.

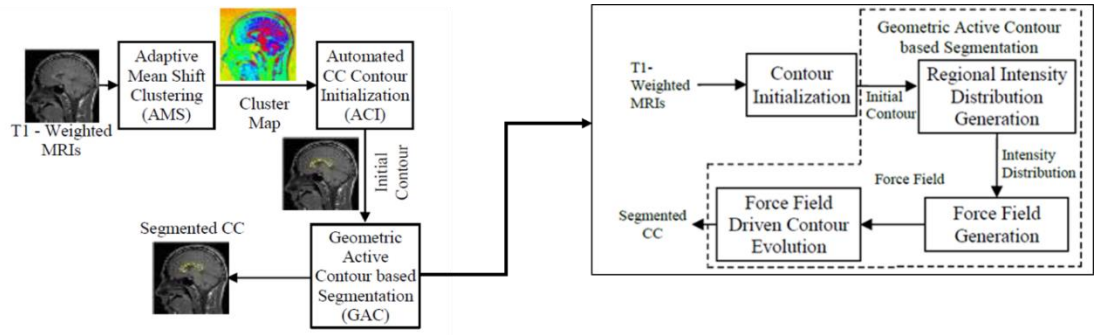


Figure 14. Pipeline of the method presented in **[21]**. Adapted from **[21]**

The most used open-source application tool based on atlas is **ART-yuki** [22] [23], considered as the state-of-the-art for the automatic segmentation algorithms for CC of adults' MRI. Therefore, it is an automated Corpus Callosum segmentation program under the Automatic Registration Toolbox (ART) project [6]. It performs fast, robust and fully automatic segmentation on 3D T1-weighted structural MRI scans (Fig. 15a). Initially the program does an automatic detection of the AC (anterior commissure) and PC (posterior commissure) landmarks: we expect for images representing anatomy that the relative position, rotation and size of healthy organs is restricted in a similar way as their elastic deformation. In the case of the CC, the AC-PC line is generally accepted as it is a well-detectable geometric feature of the Mid-Sagittal images, which represents such a standard coordinate system [18]. The AC-PC line is illustrated in Figure 15b.

Entering into detail, ART-yuki performs by the following steps [22]:

1. automatic MSP detection as the plane with maximum bilateral symmetry;
2. automatic location of AC-PC line on MSP;
3. reconstruction of a standard MSP image (512x512 with 0.5x0.5 mm^2 pixels) from the original MRI volume thanks to tri-linear interpolation;
4. automatic selection of a subset of 49 atlases from a set of over 600, based on correlation between atlas and test image inside a rectangular sub-image containing the CC on the reconstructed MSP of step 3;
5. projection of the 49 CC labels onto the test MSP using ART's non-linear registration approach [24];
6. averaging of the projected labels to obtain a fuzzy segmentation of the CC on the test MSP;
7. thresholding to yield the final binary CC segmentation (the threshold level used the Fisher's linear discriminant ratio [25]).

ART-yuki provides excellent results with AC-PC aligned high quality Sagittal images like those available on the two publicly multi-site and multi-protocol datasets: OASIS [26] and ABIDE [27]. These databases also contain images of children. The focal problem of the program is related to the high resolution to correctly extracting the MSP, which is rare in pediatric images. Besides, without a correct alignment of the MRI (something that often happens with old exams) it cannot work properly producing images with all pixels with a value of 0 because AC-PC landmarks are incorrectly identified.

ART-yuki is used in this dissertation as state-of-the-art for the segmentation of the CC to evaluate the performances of our proposed method, as will be showed in Chapter 5.

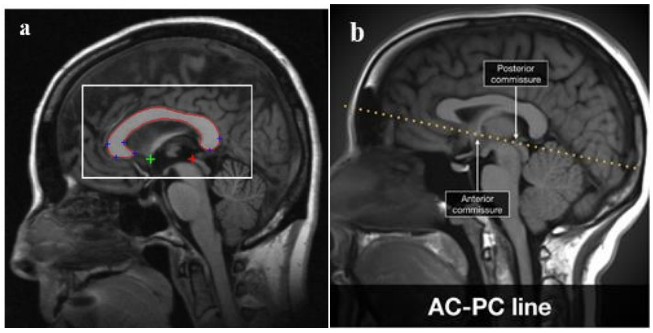


Figure 15. Some operation of Art-yuki
a. Example of detection of CC landmark on the MSP. Reprinted from **[22]**
b. AC-PC line. Reprinted from **[28]**

2.3 Artificial Neural Networks for Image Segmentation

Recently, Artificial Neural Networks, explained in detail in the following chapter, has started to be used to overcome the issues of standard approaches in automatic segmentation on medical images. Thanks to the work proposed in [29] regarding cell membrane segmentation, where the authors classify the centers of image patches directly using a Convolutional Neural Networks (CNN), a variety of researches ([28] [30] [31] [32] [33]) have advanced towards image segmentation through Machine learning and ANN. A Neural Networks can learn by itself the main features extractions presented in the input image. This eliminates the need for handmade features extractions, experts, atlas for template matching or previous studies. Furthermore, this method allows for generalizing, using multi-scanner and multi-sites (different protocols) images as well as images in different step-times: this enables us to proceed with a longitudinal (serial) study on the Corpus Callosum. Last but not least, there are different advantages in computational power and dataset extraction, which permit to implement an automatic algorithm.

To show some examples, [34] adopts a discriminatory learning approach in which the researches use a subset of the expert segmentations available from a single data set to train

a Convolutional Encoder Network (CEN), which performs the extraction of characteristics and the formation of the model parameters in a joint way. Whereas in [33], the authors introduce a CNN with anatomical information reaching a Dice index of 95.22% (the closer the measurement is to 100%, the better the spatial agreement between two images).

It is possible to find different codes online for adults, such as U-Net [30], a CNN for biomedical image segmentation, faster and more accurate of state-of-the-art methods according to [31] and [32]. One of the best performing research based on U-Net is [28], we have taken inspiration from its main idea, improving pre- and post-processing step, in particular focusing on normalization. Having to segment children's images we add the technique of Transfer learning. This method uses a pre-trained saved network as a generic model, transferring features acquired to a new NN, speeding up and improving performance (we will analyze this deeper in the next chapter). The combination of these two methods (U-Net and Transfer learning) was the base for our work on robust segmentation of Corpus Callosum in multi-scanner pediatric T1-w MRI.

Chapter 3

Convolutional Neural Networks

As stated in previous section, new approaches for the segmentation of the Corpus Callosum rely on the use of Convolutional Neural Networks. Defining clearly what we are talking about, first of all, we should say that a CNN is a computational model for Deep Learning. DL transforms raw data (e.g. pixel values) into a feature vector from which the learning system can detect and classify patterns from the input data, without any human intervention. Stacking Convolutional Layers one by one, each one will be responsible to transform the input data (from the previous layer) to a different non-linear representation, easier to be understood for the problem in question (e.g. image segmentation). To clarify it better, it is necessary to proceed neatly and slowly, starting from what Machine Learning (ML) is, what algorithms based on it do and what means speaking of Neural Networks as model for ML.

3.1 Machine Learning

"Could a computer go beyond what we know how to order it to perform and learn on its own how to perform a specified task?" [35]. This query has moved the way of thinking about programming, allowing the birth of Machine Learning, key to Artificial Intelligence (AI).
If in classical (heuristic) paradigm we give data and rules as inputs, to process information, and we have answers as output; in contrast, in ML as inputs we have data and the expected answers from these data, and we have returned the rules as output (Figure 16). The algorithm generates its own logic based on the data entered. The generated rules can be applied to new datasets to bring to answers as well as the original ones. It is important to check how the algorithm can evaluate data that it has never seen before. Performance measurements are executed using a set of data called Test set. This set of data is usually different from the set of characteristics used to train the system (Training set) so that more precise and independent evaluations can be made on how effective the training has been.

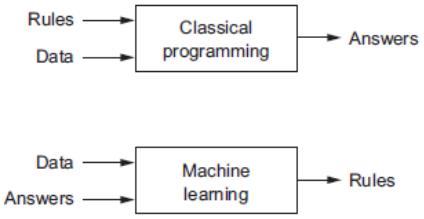


Figure 16. The two different programming paradigm. Reprinted from **[35]**

ML is essentially a form of applied statistics aimed at using computers to statistically estimate a complex function. Prof. T.M. Mitchell in 1997 [36] defined ML as "an algorithm that learns from experience *E* concerning a class of problems *T* with a measure equal to *P*, if its performance on problems *T*, measured by *P*, increases with experience *E*".

It is important to remember that Machine Learning works only if the problem is really solvable with the data available and if the Training set is sufficiently large and representative. The negative aspect is that it is not possible to have supporting evidence for a certain result, in fact, very often ML methods are defined as Black Box.

Many tasks can be learned through the use of ML techniques. Some of the most important are transcription, translation, synthesis, denoising and classification. The latter plays an important role for the work of this thesis and will be dealt with in detail when we will cover Deep Learning and Convolutional Neural Network.

Returning to what was said by Mitchell [36], for tasks such as classification, *P* is evaluated by measuring the accuracy of the model, the percentage of examples for which the model processes a correct output. Another parameter of reference can be the error rate, defined instead as the proportion of examples for which the system processes a wrong output. Usually the experience *E* corresponds to an entire dataset, a collection of examples. An example of a dataset can be a collection of images to be analyzed, as our MRI dataset.

Ordinarily, Machine Learning algorithms are divided in two main categories: Unsupervised Learning and Supervised Learning. The difference between these two types of learning concerns the information you have on the Training set.

3.1.1 Unsupervised Learning

The model aims to find a structure in the input provided, without the inputs being labeled in any way or without any answers. It is conceptually a sort of clustering: it subdivides the elements into homogeneous subgroups. To clarify, as if in the Figure 16 the "Answers" were not there as input in the "Machine Learning" box. This makes the learning process more complex and difficult to understand, because we do not know what features the model considers.

3.1.2 Supervised Learning

In this model we provide examples in the form of possible inputs and the desired outputs and the goal is to extract a general rule that associates the input with the correct output. If the training set knows, in addition to the values that the variables assume for each element,

also the class to which that element belongs, then a Supervised Learning can be used: this aims to reproduce the correct classification as faithfully as possible. The real class to which the elements belong to is called "desired output". It is therefore easy to understand how to manage learning and it can be done by managing the error that the network makes in classifying the elements by reducing it as much as possible. The error is equal to the difference between the output obtained from the network and the desired output.

3.2 Neural Networks

In the field of Machine Learning, the Artificial Neural Networks, or simply Neural Networks (NN), are a family of models inspired by biological neural networks.

What makes neural networks particularly efficient is that they have a very high distributed processing capacity that allows them to quickly process large amounts of data. Neural networks resemble the brain for the learning process and for connections, which in the case of Artificial Neural Networks become the weights. Weights are the element of the network that contains knowledge: the fruit of learning is stored in the form of weight, associated with the various arcs that connect neurons.

The procedure that allows you to carry out the learning process is called Learning Algorithm (LA) and works by modifying these weights. Starting from initial weights usually defined in a random way, the objective of the LA is to modify them iteratively until the learning process does not end, through different stop conditions.

When we talk about a neural network we actually have three basic elements:

- **Neurons**: they are the "processors" within the network, the computational unit; they effectively process the information and data and are characterized by an activation function dependent on the type of network; the type of neuron changes according to how the network is implemented.
- **Architecture**: neural networks are based on a structured architecture organized differently according to the types of learning and the type of network.
- **Learning Algorithm**: it is the element that most characterizes the NN. It is a procedure that allows you to carry out the learning process and work by modifying the weights. As mentioned above, there are Supervised and Unsupervised Learning.

In this dissertation we focus on Supervised NN, because image segmentation is nothing else than a classification problem. The simplest Supervised Neural Network is The Perceptron, developed in 1958 by Frank Rosenblatt [37].

3.2.1 The Perceptron

It is a network formed by a single neuron to which various inputs are connected and this allows to classify these elements only in two classes. The Artificial Neuron is organized as in Figure 17: the model takes the inputs, equal to the value that the variables assume (x_i, elements of the classifier), and a bias b that controls how easy is to the Perceptron to put output as 1; it builds the weighed sum and uses an activation function to find the desired output y, depending on the calculated weighing sum.

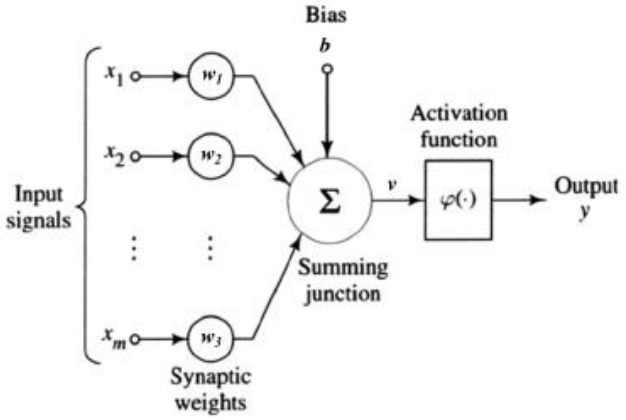


Figure 17. Artificial Neuron of Perceptron

Depending on the value that the weights assume, the sum will have a different value and consequently the output will change, even with the same inputs. There are several activation functions, such as linear, piecewise linear, step, sigmoid, etc. To give some examples: the model gives the weighing sum of 2 inputs as input to an activation step function, this last will return 0 or 1 according to the threshold set in the space of the combinations; if the inputs are 3, we have a surface (plane) that divides the space in 2 parts; more than 3 inputs will result in an inter-plane.

An epoch is a training phase characterized by the fact that all the elements of the training set are given as input to the network. Every time that all the elements of the training set have been entered as input to the network, an epoch ends. When we move on to the next epoch we need to give all the elements of the training set again as inputs and so on. Networks can be trained through the use of a few epochs, just as it may take a few hundred epochs to reach convergence and the trained network.

If $\hat{y}$ is the desired output, the LA will change weights until the difference y- $\hat{y}$ will become as small as possible. The difference that is obtained at epoch p is used as error e to calculate weights at epoch $p+1$, using the following equation:

$$w_i(p+1) = w_i(p) + \alpha \cdot x_i(p) \cdot e_i(p) \quad (6)$$

The Equation 6 is called Back-propagation and it is based on gradient descent. The α is the learning rate, it can be in the range from 0 to 1, and the higher is, the faster, but less precise the train will be.

To overcome the problem of classification in only two classes, Perceptron's concept has been extended over years [38]: the architecture is organized with a structure that sees several neurons divided into layers in which the neurons of a layer are connected with those of the next layer and not with the neurons of the same layer (shown in Figure 18). The layers between the input and the output layers are called Hidden layers, and their number depends on the complexity of the problem. When we give an input, this passes from the input neurons and it is processed until the result of the various elaborations reaches the output neuron (layer) that produces the output that will be seen. Every layer can make a decision at a more complex and abstract level than the previous layer. Meaning that a Multi-Layer Perceptron with a different number of Hidden layers can perform sophisticated decision making [38].

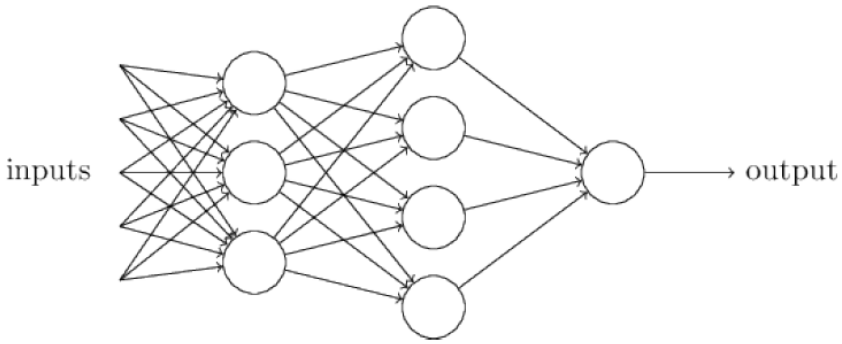


Figure 18. Multilayer Perceptron. Reprinted from **[38]**

3.2.2 Sigmoid Neuron

Another kind of neuron, used to tackle the problems of Perceptron, is based on a new function called sigmoid function:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{7}$$

where z is:

$$z = \sum_j w_j x_j + b \tag{8}$$

Sigmoid neuron can have any value in the range 0 – 1 as inputs, and small changes in its weight causes a small change in its output. In Figure 19, it is possible to understand how the function behaves.

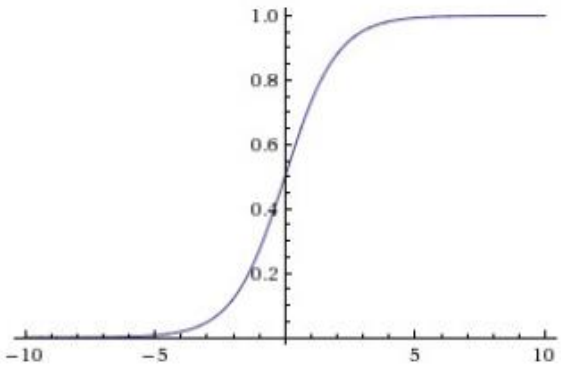


Figure 19. Sigmoid function. Reprinted from **[39]**

Another difference from Perceptron is the output: it is legitimated if within the range 0 – 1. When we want the output to be an integer (therefore 0 or 1, as in image segmentation), we use a threshold. For instance, all the outputs smaller than 0.5 indicate a "0", and the ones greater or equal than 0.5 indicate a "1" [38].

3.2.3 The Logistic Regression Model

Logistic regression is a regression model defined as: given an input variable x, the model tries to predict the output y as $\hat{y}$, using the model parameters (weights w and bias b). To measure how close the predicted result $\hat{y}$ is from the real output y, the loss function L is used. The aim is to find the best w and b to minimize L over a set of m training examples, in order to have the average error per pattern J (cost function) in Equation 9 as small as possible.

$$J(w,b) = 1/m * \sum_{m} L(\hat{y}^i, y^i) \tag{9}$$

Hence, we can see the cost J as a function of w and b in which we are looking for the minimum (see Figure 20), or rather the global optima of a convex function [40].

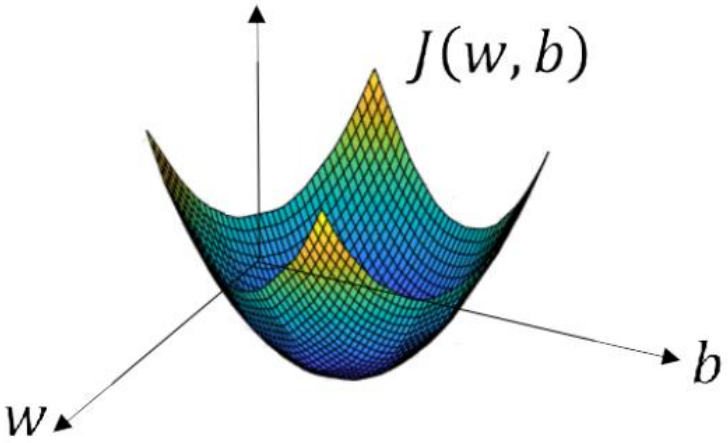


Figure 20. The cost function. Reprinted from [36]

There are different types of loss function that are usually used on NN, such as the quadratic loss function (or MSE function, that we will investigate later), the log loss function, the cross-entropy loss function and the dice coefficient loss function. The latter, also known as F1-Score [31], is the one we decided to use to classify pixel predictions on our work of segmentation of Corpus Callosum. It is a measure of how well we are doing the comparison between our segmented image and the ground truth, but it is necessary to explain it better. As illustrated in [28], we should imagine to have a 3x3 image as in Figure 21, in which the white square is the object to be segmented.

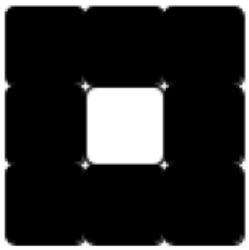


Figure 21. A 3x3 example image. Reprinted from **[28]**

Now, suppose that our segmentation model predicts the Figure 22 as segmented image.

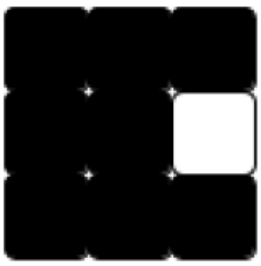


Figure 22. A 3x3 exmple image segmentation. Reprinted from **[28]**

To understand how far the segmented image is from the true one, we have to classify each pixel as True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). Taking for granted the knowledge of these concepts, we can build a contingency table in Figure 23:

	Object (y)	No Object (y)
Object ($\hat{y}$)	0	1
No Object ($\hat{y}$)	1	7

Figure 23. Contigency table

The accuracy (Equation 10) is good, about 77.7%, but the predicted object is completely different from the true one. It would be as if all the pixels of a CC are segmented wrong.

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{10}$$

To overcome this problem, we have two other measures, precision and recall, to understand if the model took the right decision for the classification of object's pixel.

$$precision = \frac{TP}{TP + FP} \tag{11}$$

$$recall = \frac{TP}{TP + FN} \tag{12}$$

For our example, precision and recall are both 0%. Combining these two measures, we have the Dice Coefficient or F1-Score [31], that can be seen as a harmonic average of both:

$$Dice = \frac{2 \cdot precision \cdot recall}{precision + recall} = \frac{2TP}{2TP + FN + FP} = \frac{2\,|A \cap G|}{|A| + |G|} \tag{13}$$

where A is the segmentation given by the model and G is the ground truth. To use this as loss function we have to set the gradient descent to go in the opposite way compared to it: a Dice Coefficient of 1 must correspond to a loss of -1 because we are making no mistakes.

3.2.4 Subdivision of the dataset

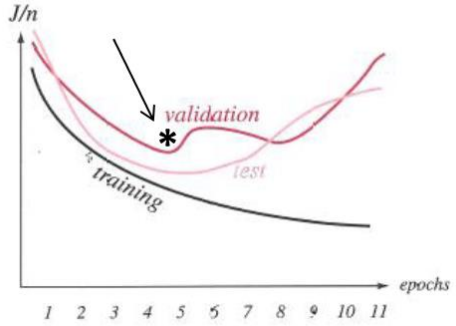


Figure 24. Example of a plot of the average error per pattern. The validation and test error are virtually always higher than the training error. Asterisk indicates when the error on the validation set increases.

When we have a data set, we can divide it into different parts, as said above for example in a training set and in a test set, and you can decide to train the network with all the training set elements. Most of the time, it is usual to divide the training set in other two parts: a part that is really training (training set) used to modify the data set weights, and a part of validation set, that is a part of the data set used as validation during learning in order to avoid overfitting. We want to avoid that the network is very well trained on the training set, but at the moment where there are elements outside the training set, its performance diminishes.

Observing the validation curve in Figure 24, the best performance is on the training set, but up to a certain point the error on the test set decreases as on the validation set. At the point indicated by the asterisk, the error on the training set continues to decrease, while the one on the validation set increases. This means that we have reached a point where the improvements on the training set actually represent a worsening from the point of view of the ability to generalize the network. In fact, we want the networks to behave well not only on the training set, but above all on the elements that will be provided to be classified later. The point identified is therefore the best compromise between the performance of the network with the training set and with other elements.

There is also a third data set (test set), which is used to understand how much the data set you are using gives good results. If the elements used for the training are adequate, taking test sets of different types, the performances should more or less remain; if instead there is a big difference between the validation set and the test set performances, it means that the training data set must be changed.

3.2.5 Performance of a NN

The performance, or rather the measures that allow the measurement of network performance, are linked to three aspects: accuracy, complexity and convergence.

Regarding accuracy, the primary objective is the generalization that indicates how the network behaves when it must classify elements that are not part of the training set (test set), meaning the network's ability to achieve a good performance on the elements that have not been a direct part of learning. As previously stated, the Dice Coefficient provides us with this measure.

Two other measures will be used to evaluate our test set. A first measure is the MSE (Mean Squared Error), which is nothing but a measure that binds the error that occurs between the output of the network, $o_{k,p}$ and the output associated with the training set, which is called the target, $t_{k,p}$. In Equation 14, p indicates the pattern (the element of the training set) and k indicates the output neuron.

$$MSE = \frac{\sum_{p=1}^{P}\sum_{k=1}^{K}(t_{k,p} - o_{k,p})^2}{P \cdot K} \tag{14}$$

The second measure is the Structural Similarity (SSIM) index. According to what was said in [41]: "the SSIM between an image x and another y is composed by a linear combination of a luminance comparison function $l(x,y)$, a contrast comparison function $c(x,y)$ and a structural comparison function $s(x,y)$" as in Equation 15 (α, β and γ are usually set up to 1).

$$SSIM(x,y) = [l(x,y)]^{\alpha} \cdot [c(x,y)]^{\beta} \cdot [s(x,y)]^{\gamma} \tag{15}$$

with:

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{16}$$

$$c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{17}$$

$$s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x + \sigma_y + C_3} \tag{18}$$

where μ_i is the luminance of the image i, σ_i is the standard deviation of μ_i, σ_{ij} is the correlation between the two standard deviation and all the three C are added to avoid division by zero.

MSE is simple to calculate and physically it has a clear meaning, but "it does not indicate the structure of the images when the error is being penalized" [41], while the SSIM gives higher value to the images that are higher quality. The example in Figure 25 helps to clarify the difference between these two measures that we will use to evaluate our results.

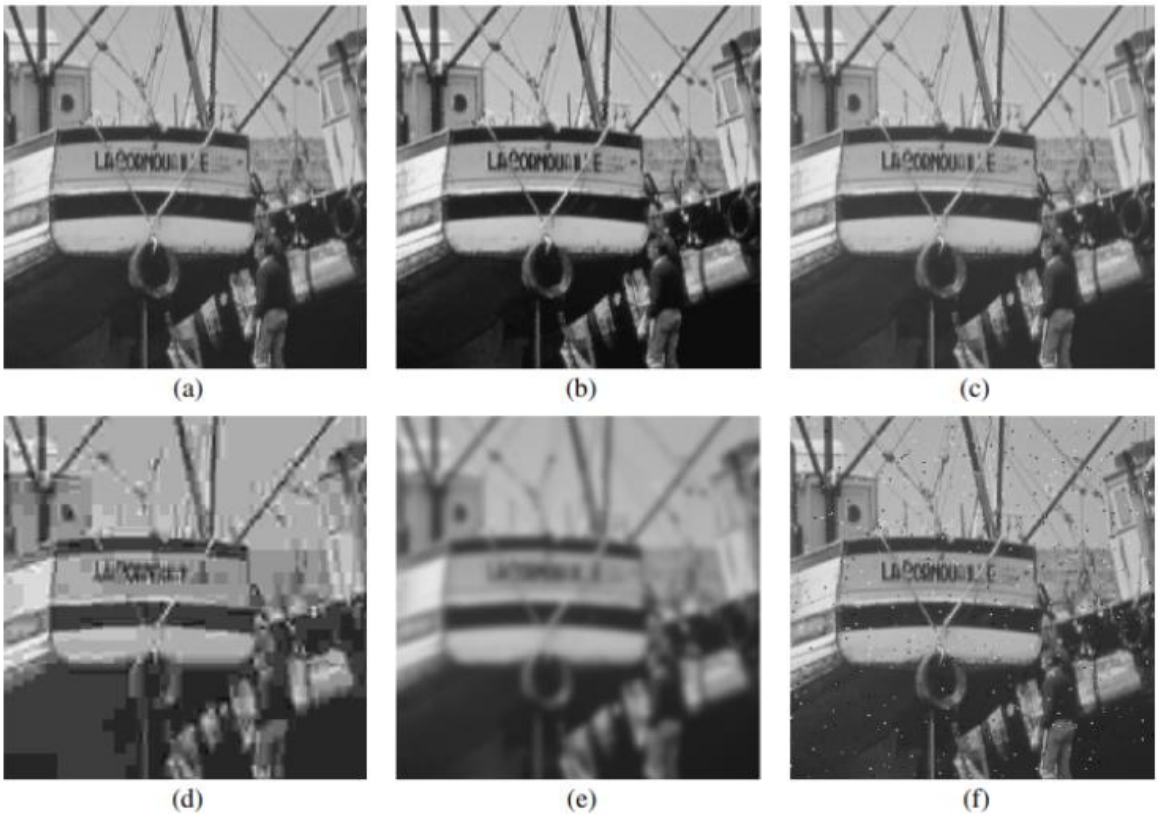


Figure 25. Comparison of the same original image with image containing different sorts of distortions. All the five comparisons have an MSE = 210, while each one presents a different Mean SSIM index. **(a)** Original image. **(b)** Mean SSIM = 0.9168. **(c)** Mean SSIM = 0.9900. **(d)** Mean SSIM = 0.6949. **(e)** Mean SSIM = 0.7052. **(f)** Mean SSIM = 0.7748. Reprinted from **[41]**

3.3 Deep Learning

Deep Learning (DL) is a sub-area of Machine Learning that uses Deep Neural Networks, which are equipped with many layers and new algorithms for the pre-processing of data for the regularization of the model.

The brain learns by trial and activates new neurons by learning from experience, in the same way of the architectures involved in Deep Learning: the extraction stages are modified according to the information received at the entrance. The development of DL took place in the 1980s, consequently and in parallel with the study of AI. In those years, computer technology was not advanced enough to allow a real improvement in this direction, so we had to wait until our days to see even more significant developments, thanks to the availability of data (Big Data) and the power of calculation (GPU). Once experts understood how to use 3D graphics cards (which are designed to make matrix multiplication extremely fast) instead of normal computer processors, working with Deep Neural Networks suddenly became practical and affordable.

Convolutional Neural Networks are a development of Deep NN that are specifically designed for Image recognition and segmentation, using a particular architecture.

3.3.1 Architecture of a Convolutional Neural Network

Each image used in learning is divided into ROI and each one of them will be processed by filters to look for particular patterns [42]. Formally, each image is represented as a three-dimensional array of pixels (width, height, color) and each of its sub-sections is put in convolution with the chosen filter. In other words, by filtering each filter along the image, the internal product between the filter itself and the input is calculated. This procedure produces a set of feature maps (activation maps) for the various filters and this type of layer is called Convolutional Layer. In the analysis of an input image that has hundreds or millions of pixels, we are interested in identifying significant features or important details that are very small compared to the total size of the image, contained for example in about ten pixels. Having a two-dimensional image I as input, a two-dimensional kernel W can be used:

$$S(i,j) = (I * W)(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n) \tag{19}$$

Given m input and n output, a matrix product would require $m \cdot n$ parameters and the algorithm a $T(m \cdot n)$ processing time. Restraining the connection number for each output to a number k, smaller than m, the matrix product would require only $k \cdot n$ parameters and a processing time of $T(k \cdot n)$. Gain in efficiency becomes extremely important when k is smaller than m by many orders of magnitude. This will make it possible to significantly reduce the number of weights to be updated and to speed up network training operations.

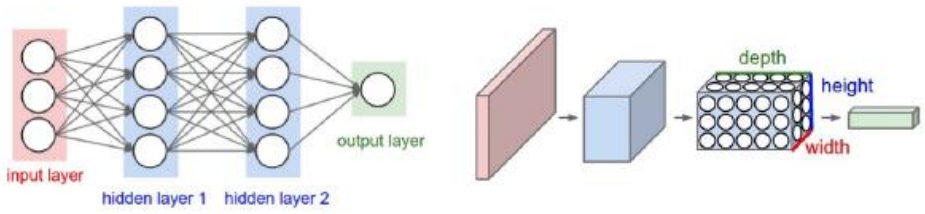


Figure 26. On the left: a classic 3-level neural network with a fully-connected stratified architecture. **On the right:** a network consisting of Convolutional Layers arranges its neurons in three dimensions. Reprinted from **[39]**

In Figure 26 (on the right), we can see an example of Convolutional Layers in which it is highlighted that CNN have neurons arranged in 3 dimensions: width, height and depth [39]. The depth of an output volume is a parameter that controls the number of neurons in the Convolutional Layer that are connected to the same region of the input. The set of these

neurons is called the depth column. This is given by the number of filters we use in the Convolutional Layer; hence the more filters you choose the larger the number of features describing the input that the network can learn at this level. The dimensions of the filters must also be defined, and if these are to be translated on the whole image or by performing a sub-sampling (stride parameter).

By overlapping the various feature maps of the same image portion, we obtain an output volume.

An additional main feature of CNN is the use of shared parameters meaning the use of the same parameters for more than one function in the model: each slice of the filter volume has the same weights and only one bias [28]. In a traditional network, each element of the weight matrix is used exactly once during the calculation of an output and it is no longer reused. The parameter sharing obviously leads to the sharing of weights: the value of a weight applied to an input is linked to the value of a weight applied at another point in the network. In a CNN, each kernel member is used in all input positions. If a pattern can appear in a part of the image, it can appear anywhere, so the idea of units in different locations that share the same weights allows them to detect the same patterns in different parts of the input array. Mathematically the operation performed by the feature map is a discrete convolution (it is a more affordable operation than the matrix product, also in terms of required memory).

After the Convolutional Layer we have a sub-sampling layer, which takes care of further subdividing the processed image in sub-sections, reducing its dimensionality to proceed in detail. This is called Pooling Layer, hence it aggregates information into the input volume, generating smaller feature maps, giving an invariance compared to simple input transformations while maintaining significant information for the purpose of subscribing the patterns contained in the input data analyzed. One of the most popular sub-sampling is that of MaxPooling [43]. In particular, this type of sampling partitions the image into a series of rectangles that are not overlapped, and returns from each rectangle the pixel corresponding to the maximum value point.

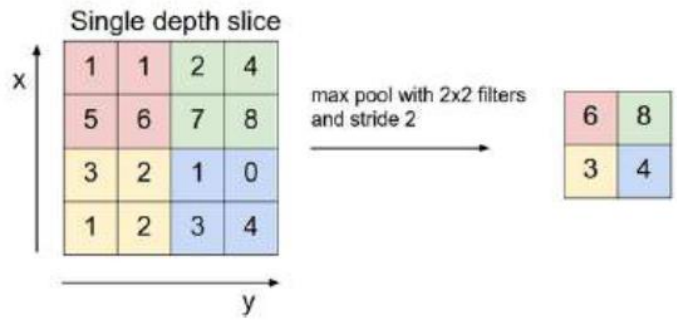


Figure 27. Example of applying a MaxPooling Layer on a 4x4 matrix. Reprinted from **[43]**

These two types of layers will alternate throughout the body of the network. You can find other types of layers, interleaved with Convolution and Pooling Layers. The last layer is the Output Layer: it has as many neurons as possible labels and for each label it returns the probability that the tested sample belongs to that label.

3.3.2 The Training phase of a CNN

In Deep Learning, we are dealing with large amounts of data and the gradient descent algorithm is time consuming, which elongates the time required to finish just the first epoch [28]. The training phase of a CNN starts by dividing the training set in K groups called batches, of size B, processing each one at a time, and after obtaining the output, it calculates the cost function J for each of them. This is minimized by means of algorithms such as the SGD (Stochastic Gradient Descent) method or Mini-batch Gradient Descent, and is also used for back propagation. In this way, we only do K updates for w and b per epoch.

SGD consists of sending as input only one sample per batch ($B = 1$ and $K = m$ = number of training examples). We refer to stochastic gradient because this one is approximated at each iteration with the gradient calculated on a single addend of the cost function (corresponding to a dataset element). As the algorithm runs through the training set, it performs the update for each training example. Multiple steps can be done on the training set until the algorithm converges. It is much more difficult to converge proceeding one sample at a time.

In Mini-batch gradient descent, it is possible to choose any value for B, so $K = m/B$. It is advisable to use a batch size of 2^n ($n = 1, 2, 3, 4, \ldots$) because it helps allocation in CPU/GPU memory [28]. At the moment it is the users' favorite choice [35]: it makes the training phase faster and with a better convergence than SGD.

3.3.3 Batch Normalization

When training a CNN, each feature can have different scales and this can directly affect the gradient descent. The ideal would be to center all the features with zero mean and unit variance, Batch Normalization tries to solve this [28]. If we have $Z^{[l](i)}$ as the output neuron of the l-th layer, we can normalize in this way:

$$\mu = \frac{\sum_{i=1}^{n} Z^{[l](i)}}{n} \tag{20}$$

$$\sigma^2 = \frac{\sum_{i=1}^{n} (Z^{[l](i)} - \mu) * *2}{n} \tag{21}$$

$$Z_{norm}^{[l](i)} = \frac{Z^{[l](i)} - \mu}{\sqrt{\sigma^2 + \epsilon}} \tag{22}$$

Note that ϵ is to avoid division by zero. To have fixed mean and variance for any change occurring in weights and bias during the whole training, we must do another step introducing two new parameters, σ and β. So to train our NN reasonably fast and avoiding to operate brutally we have to use Mini-batch gradient descent and Batch Normalization in forward propagation as in the following Equation for each hidden layer:

$$\tilde{Z}^{[l](i)} = \sigma Z_{norm}^{[l](i)} + \beta \tag{23}$$

3.3.4 Rectified Linear Units (ReLU) activation function

ReLu is a widely used activation function [44] for Convolutional Layers because, for example, unlike the hyperbolic tangent or the sigmoid, it does not present saturation regions in its dynamics, preventing the network from stagnating in a situation where the variation of the inputs acts very little on the variation of the outputs and in which the network learns extremely slowly (the gradient stagnates). ReLU is easy to calacute, the Equation is:

$$f(x) = \max\{0; x\} \tag{24}$$

and it is shown in Figure 28. Basically, this layer changes all negative activations to 0.

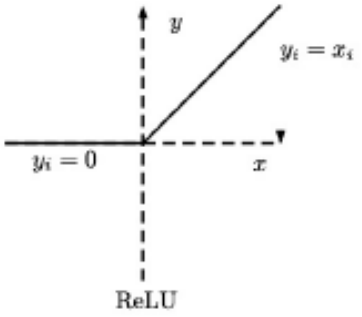


Figure 28. ReLU activation function

Apart from the advantage of avoiding the saturation regions, with the ReLU we introduce sparsity into the matrix containing the activations of the neurons, which begins to present several zeros within it. This facilitates and speeds up the error back-propagation mechanism as only those paths in which one neuron contributed to the downstream error formation will be analyzed, whereas a neuron that gives 0 as output it is as if it were turned off or deactivated.

A potential disadvantage of the ReLU is that when the neuron is not active, it always has zero gradient. This could cause neurons that are not active initially to never become active as gradient-based optimizers will never update their weights [45]. For this reason there are also variations of the ReLU function which avoid giving completely null outputs.

3.3.5 Hyper-parameters

There are some parameters which affect the search for the minimum more than others. We will call these Hyper-parameters of the model, and we will choose them opportunely in order to speed up the descent of the gradient to the final solution of the problem. As it will be evident during the elaboration, there are different hyper-parameters to select; among them, the learning rate is undoubtedly of great importance, as is the type of optimizer chosen.

In addition to the already seen Hyper-parameters such as Depth and Stride, we also have to choose the Padding. When we apply filters to an input volume, the spatial dimensions of the output would be smaller. We want to preserve information about the original input so that we can extract low level features. To make sure that the output volume remains the same spatial dimension of the input one, we can apply a zero padding: this pads the input volume with zeros around the border, until they allow us to have the output of the same size as the original input volume.

As we had exposed, the validation set helps us to select the best model, and therefore also the best hyper-parameters. To understand how close the results of the model are to the expected ones, we must pay attention to two important measurements: bias and variance. Bias is "how well your model fit the training data" [28] and variance is "how well your model can generalize in the validation set" [28]. Setting a baseline error, a good model is that in which both bias and variance are under this threshold.

There are other ways used to prevent NN from overfitting, for instance adding more data through Data Augmentation: it generates more training data from existing samples by augmenting them via random transformations that yield believable-looking images [35]. Some examples are translations, rotations, flips, crops and scaling. By applying just a couple of these transformations to your training data, you can easily and significantly increase the number of training samples.

To avoid overfitting there are also regularization techniques. One very functional technique is Dropout [46] that consists in selecting randomly in each layer some neurons to set to zero. This allows to have a simpler network, with less parameters, and the remaining neurons become more specialized in the assignment. With the addition of this technique on a Layer, we have a new hyper-parameter to be set: the probability of neurons to drop out from the process (Dropout rate). Another regularization technique that allows to speed up the training is Early Stopping. When the training and validation curves start diverging, the train is stopped and the model is saved.

Now that we know what a Convolutional Neural Network is, how it is composed and that depends on a huge number of parameters (number of layers, learning rate, activation functions, etc.), we can analyze an application example of this: U-Net.

3.4 U-Net

One of the most used CNN is U-Net [30], a Convolutional Neural Network to biomedical image segmentation, already used to segment the Corpus Callosum as previously explained. The purpose of U-Net is to separate the Convolutional Layers in two symmetric groups, or two paths: Contraction path (downsampling) and Expansion path (upsampling).

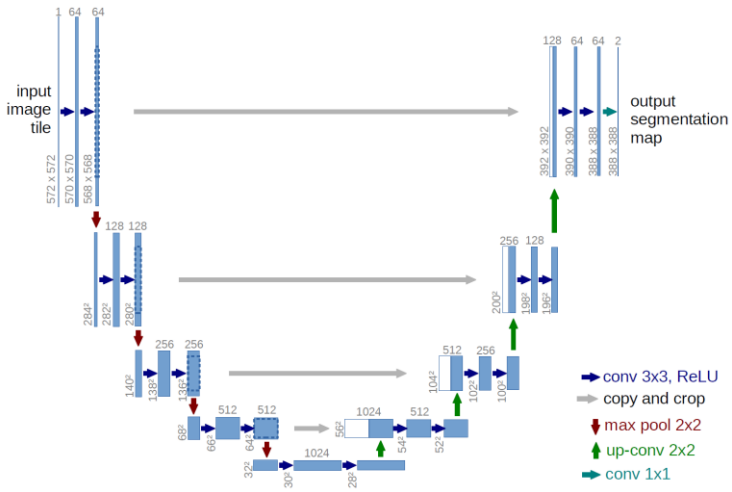


Figure 29. U-Net architecture. Reprinted from **[30]**

The contraction path applies convolutional filters and reduces the spatial size of the image through pooling layers. The upsampling group will get the reduced images and increase the spatial size of the image output until it has the same size of the input. This is done by an unpooling operation called MaxUnpooling. Thanks to the symmetry of the U-Net, we can keep in every downsampling operation the pixel position that was taken as the maximum in the receptive field and, when we will apply the unpooling operation in the respective upsampling layer, we will use that position acquired. Figure 30 clears up the operation of the MaxUnpoolling Layer.

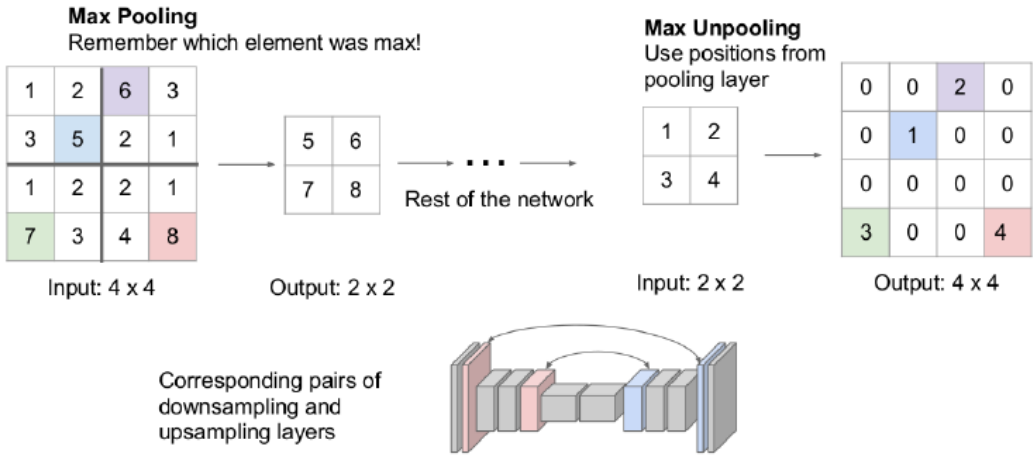


Figure 30. MaxUnpooling operation. Reprinted from **[39]**

This improves our training by reducing the size of the image in the downsampling layer instead of working with the original size in every layer as before, extending it to work with very few images and more precision segmentations. In order to localize, high resolution features from the contracting path are combined with the upsampled output thanks to the Concatenation Layer. This allows to copy the entire features map. A successive Convolution Layer can then learn to assemble a more precise output based on this information. One important modification introduced by U-Net is that in the upsampling part there is also a large number of feature channels, which allow the network to propagate context information to higher resolution layers. As a consequence, the expansive path is more or less symmetric to the contracting path, and yields a u-shaped architecture [30] (see Figure 29).

3.5 Using a pretrained CNN: Transfer Learning

Transfer Learning is a process that has helped to lessen the data demands to work with DL. The idea is taking a new network and learning on a set of correlated problems but applied to new tasks or new images which have insufficient training samples to learn a full deep NN. Taking a model in which weights and parameters are already trained (pre-trained model) on a large dataset, it is possible to transfer information to our smaller dataset. The more the dataset is similar to the pre-trained network, the more likely the TL's performance is good. This technique reduces the computational cost allowing us to do several tests without long training times. Lastly, it does not require hyper-parameters optimization.

There are two main technique for Transfer Learning using a pre-trained network: Feature Extraction and Fine-Tuning [35]. We will analyze the latter that was chosen for our research work. Rather than training the whole network through a random initialization of weights, we can use the weights of the pre-trained model (and freeze them) and focus on the more important layers (the ones that are higher up) for training. For instance, regarding U-Net an idea could be to unfreeze from the last layer of the Contraction path to the Output Layer, hence all the upsampling part (expansion path). This operation is called Fine-Tuning because "it slightly adjusts the more abstract representations of the model being reused, in order to make them more relevant for the problem at hand" [35].

To understand why we unfreeze only the last layers, we must remember that the first layers encode more generic features, it is more useful to fine-tune the last ones, which encode more specialized features. Moreover, this helps speed up the training. Finally, the more parameters we train in our small dataset, the higher the risk of overfitting.

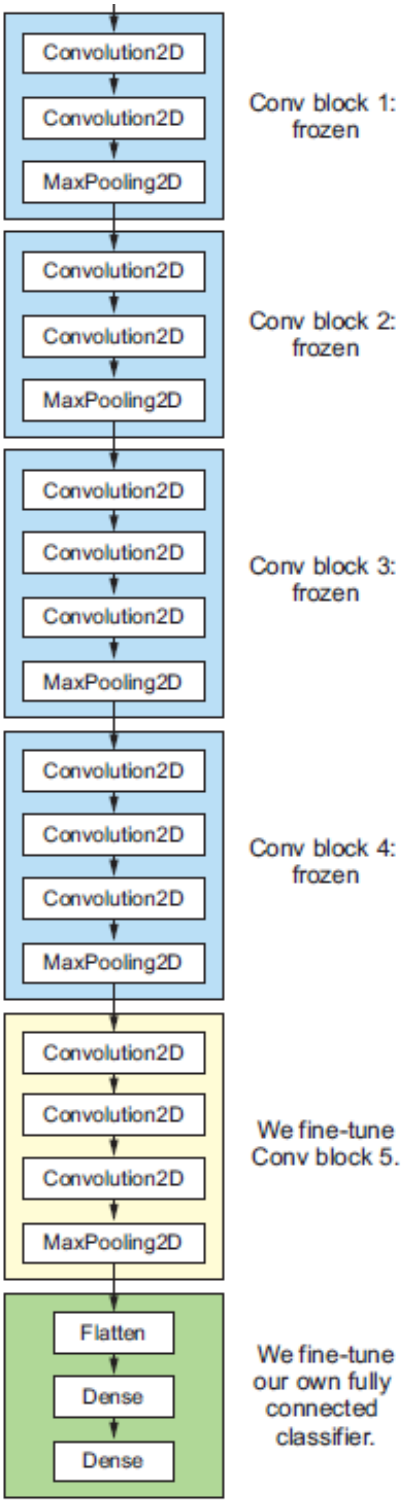


Figure 31. Example of Fine-Tuning. Reprinted from **[35]**

Chapter 4

Proposed method: Modified U-Net using Transfer Learning

In this chapter the methodology we use in this work is described. We will describe the datasets used, in particular the database gathered at Bicetre Hospital of Paris. Afterwards, we describe the CNN on adults: pre-processing steps applied, the data augmentation and the network architecture. Finally, we show the fine-tuning TL technique on pediatric images.

4.1 Database Project STAP

The project's database was gathered at Bicetre Hospital from its PACS archive in DICOM format. The collected archive contains:

- 300 healthy controls (the results of the examination was negative, in the report we find "normal MRI", "MRI without particularities" or "No anomalies", we have no volunteers in this case study);
- 41 patients, of which 33 with progressive Multiple Sclerosis and 8 with ADEM (Acute Disseminated Encephalomyelitis), undergoing treatment at the hospital.

All the acquired subjects are in the pediatric range (0 – 16 years old), images come from different French hospitals (multi-sites) with different scanner-models and acquisition protocols. All images have been anonymized. Unlike controls, which have only a singular time-step, patients present different time-step (six-month step by protocol, varying because it depends on the needs of the subject and the physician).

Concerning what we said, in the first step of our work we wrote a MATLAB program in order to automatically organize the database, and by taking inspiration by the well-known BIDS archive [47] and collaborating with medical experts, we achieved a functional hierarchical structure of the data. The result is shown in the following Figure 32. All images were converted in NIfTI format (Neuroimaging Informatics Technology Initiative): DICOM collection of slices are compressed in a single volumetric image, which contains header information followed by data [48], which is easier to use with CNN. Every file was nominated (in collaboration with radiologists) to facilitate the search and the understanding of the data for future references, according to the example:

"PS24M009Y_T1CORSE_15P.nii", in which the three separated alphanumeric strings mean:

1) STAP ID, patient sex (M = Male, F = Female), patient age → e.g. PS24M009Y

2) MRI series (sequence, plane, technique) → e.g. T1CORSE

3) TESLA (15 = 1.5 T, 30= 3.0 T, NF=Not Found), manufacturer of the MRI SCANNER (S=Samsung, P=Philips, G=GE Medical Systems, F=Fujifilm Medical Systems) → e.g. 15P

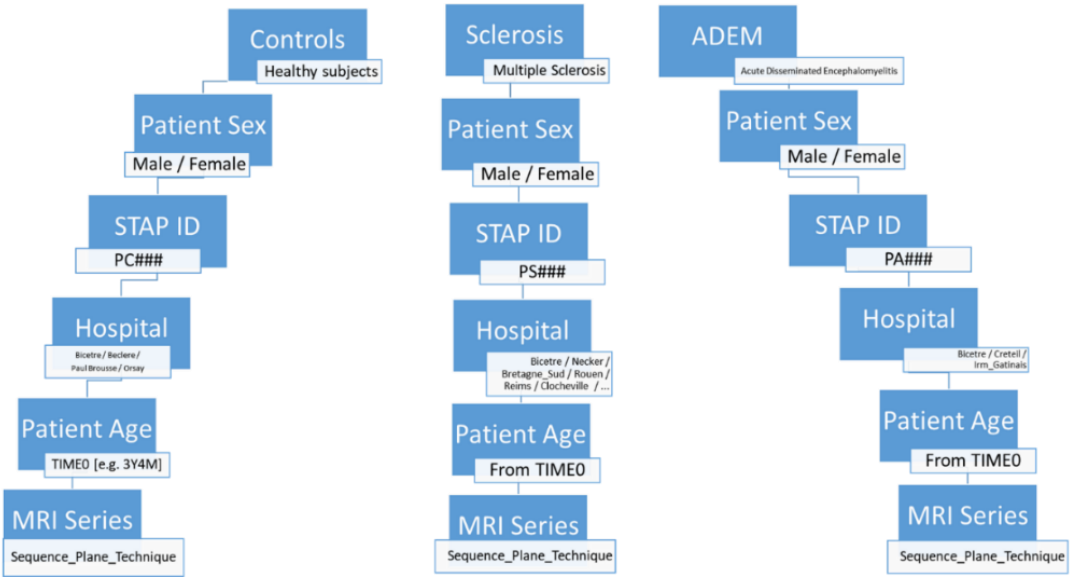


Figure 32. Organization of the archive of the Project STAP

The Standard MRI protocol at Bicetre Hospital is:

- AAHeadScout (Localization image)
- Axial T1
- Sagittal T1
- Axial T2
- Coronal T2
- Axial T2 FLAIR
- Coronal T2 FLAIR
- Diffusion-weighted MRI ADC
- Diffusion-weighted MRI TRACEW

It is possible to have other series, depending on what the radiologists are looking for or would like to see. On the contrary, we can gather less images depending on the quantity of movement artifacts that may have forced to interrupt the examination. We can see that Sagittal T1 is supposed to be available for all the patients (Normally they have these

dimensions: 256 x 256 x 30 or 192 x 192 x 21). Data acquired from various sources, as well as from the same source but at different time points, generally do not have similar intensity range. Moreover, presence of the pathologies can impact the tissue intensity behaviors.

All this exposed takes up what has been said in Chapter 1, including the choice of plan and the type of MR images to be used (see Section 1.3) and as well supports the idea of proceeding with Convolutional Neural Networks (according to Chapter 2 in Section 2.3).

However, the number of images is insufficient, also because not all subjects have a T1-w Sagittal MR image. It is advisable to train the network with the support of available online datasets, training the networks on adults and using Transfer Learning on children images.

The training set for the CNN on adults is composed of 1995 MSP of T1-w adults MRI with the corresponding CC segmentation, from two publicly available multi-site and multi-protocol datasets: OASIS [26] and ABIDE [27] (248 controls and 213 autism patients from ABIDE I, 426 controls and 393 autism patients from ABIDE II, 316 controls and 100 Alzheimer patients from OASIS 1, 234 controls and 65 Alzheimer patients from OASIS 3). All CC reference segmentations are obtained using ART-yuki and, if necessary, manually corrected (the tool selects automatically also the Mid-Sagittal Plane).

Examining these two different datasets:

- **ABIDE (Autism Brain Imaging Data Exchange):** this initiative has aggregated functional and structural brain imaging data, and now includes two large-scale collections called ABIDE I and ABIDE II. These are aggregations of datasets independently collected across more than 24 international brain imaging laboratories in order to be multi-sites, multi-scanners and with different acquisition protocols. The original dataset has all images in NIfTI format. As for MRI series present in the database ABIDE, for every patient we have a Resting State fMRI and a T1-w MPRAGE 3D. The latter is used in our work, because very similar to T1-w Spin-Echo in Bicetre's protocol.

- **OASIS (Open Access Series of Imaging Studies):** this is a project aimed at making neuroimaging data sets formed by Cross-sectional MRI NIfTI data in young, middle-aged, non-demented and demented older adults with Alzheimer's disease. The OASIS initiative is formed by OASIS 1, 2 and 3. Data were collected across several ongoing projects over the course of 30 years, trying to be as much as possible multi-sites, multi-scanners and with different acquisition protocols. The dataset contains over 2000 MR sessions which include T2-w, FLAIR, ASL, SWI, Resting State BOLD, DTI and obviously T1-w, that we used on ART-yuki.

We use 519 (resp. 120) children images from ABIDE I and 50 (resp. 50) manually segmented pediatric images (40 (resp. 40) controls and 10 (resp. 10) affected by MS) from STAP archive (6 different protocols and 4 French hospitals) for Transfer Learning and test, respectively. Manual segmentations were validated one by one by Bicetre's radiologists.

4.2 Automatize the choice of Mid-Sagittal slice

For Bicetre's collection images, the MSP slice is selected as the slice maximizing the similarity between the resulting two hemispheres, similarly to what is stated in [49]. As initialization the algorithm chooses the slice closest to the center of mass (the mean value across each dimension, where the image is not black, $i \neq 0$, see Figure 33), remembering that we work on gray scale MR images. Due to the poor depth of the non-volumetric images T1-w available to us (around 30 slides) with a slice thickness of 4 or 5 mm (low resolution), we have chosen not to interpolate to avoid losing information.

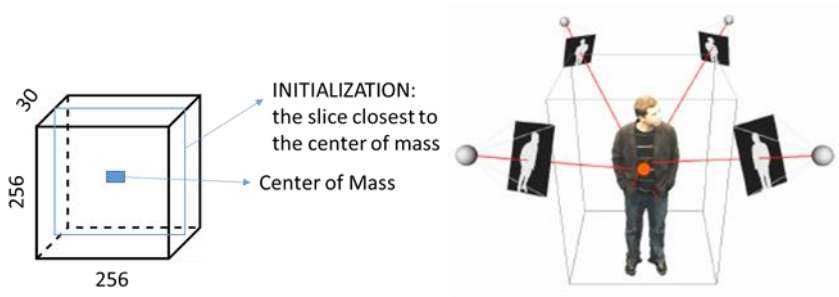


Figure 33. On the left: initialization of the algorithm. **On the right:** understanding the center of mass

The slice chose by the algorithm becomes our symmetrical plane. The image f is flipped respect to the symmetrical plane to obtain its reflection $e_{u,d}(f)$. Consequently, we use the following symmetry measure derived by the norm L_2:

$$\mu_{u,d}(f) = 1 - \frac{\|f - e_{u,d}(f)\|^2}{2\|f\|^2}. \tag{25}$$

The following step consists on an optimization: each one of the 3 slices before and 3 slices after (in order to analyze a total of 3 cm) is treated as before. Finally, the one with the highest symmetry L_2 is chosen as the MSP slice (Results shown in Figure 34 and 35).

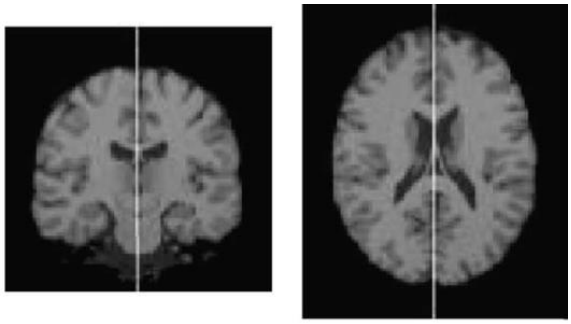


Figure 34. Result of the algorithm to automatize the choice of mid-sagittal slice. Reprinted from **[49]**

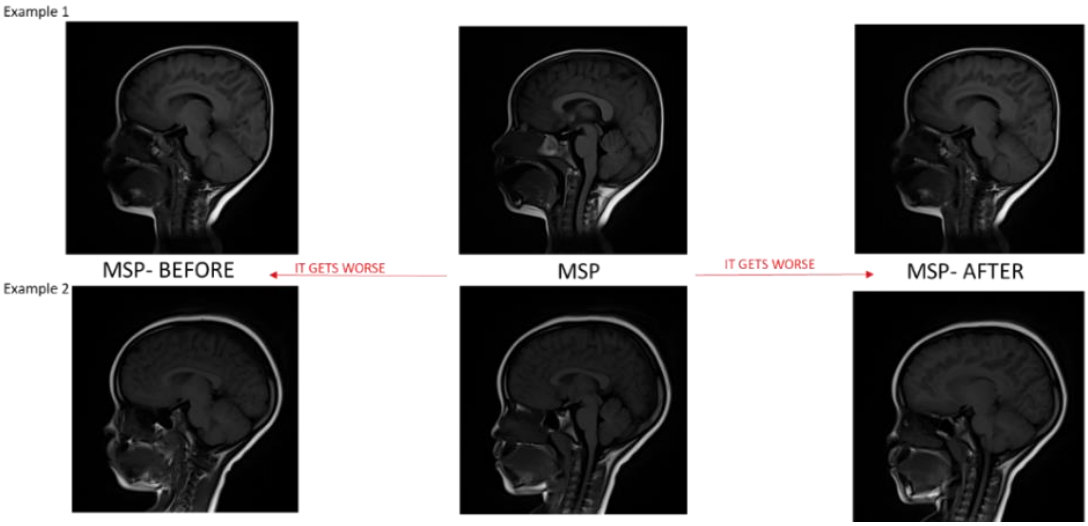


Figure 35. Differences among slice chosen, one slice before and one slice after. The Corpus Callosum is better defined in the one chosen automatically by our program.

4.3 CNN on adult T1-w MR images

We first train a CNN on adult images, based on the U-Net architecture [30], using mini-batch gradient descent with Adam optimizer, Dice loss and 10-fold cross-validation.

Adam (Adaptive moment estimation) [50] is an extension of the Root Mean Square Propagation, so it is a method with adaptive learning rate, which takes into account the moving average of the first and second moments of the gradient. We have not entered into the details of this study, because it is present in the standard architecture of the U-Net.

Conversely, in the original paper of U-Net [30] as loss function the Weighted cross-entropy is used, but we change it with Dice loss (widely described in section 3.2.3) because this has better results according to [51].

Cross-validation is used as a more sophisticated alternative to the validation set to estimate the optimal value of the parameters with which to execute a given learning algorithm. In this case we address internal cross-validation as we only act on the training set of 1995 T1-w MR images. This is divided into N blocks (10 for us) of equal size and the algorithm is executed N times with a fixed value of the parameters using each block in turn as a validation

set and the remaining N-1 blocks as a training set. By mediating the validation error on the N blocks we obtain the cross-validation error.

In each training epoch we shuffle the data to ensure that bias in the presentation order does not affect the final result.

The test set is formed by a total of 170 children's images, as we described in the previous section.

We use the Deep Learning framework TensorFlow-GPU [52] and the high-level API Keras [53] to implement the training, obtain the model and evaluate it on test set.

TensorFlow is an open source software library for Machine Learning, which provides tested and optimized modules useful in the implementation of algorithms for different types of perceptual tasks and language comprehension; while Keras is a simple, highly modular Neural Networks library, written in Python and capable of running on top of TensorFlow. It was developed with a focus on enabling fast experimentation, it supports CNN and runs seamlessly on CPU and GPU.

4.3.1 Data Pre-processing

Although the dataset is ready to be used, the following steps are done as data preparation to allow the use of the images on the neural network. After a conversion to a gray scale in which 0 is black and 255 is white (*uint8*) and a format *tiff*, the training images are first resized to 128x128 to make the training computationally less expensive and faster. Finally, images are normalized (Fig.37) with the two-stage histogram matching method [54] (the template histogram is also used for Transfer Learning and test). This is the most common approach for normalization, involving the matching of histograms. This process consists of two stages: the first one creates a template histogram, with landmarks of interest usually through averaging histograms in a reference population (training images); as for the second stage, for each subject in the study, the histograms of each subject are mapped "via a piecewise linear transformation to the template defined using quantiles as knots". Figure 36 shows what this technique consists of in a clearer way. This process is computationally fast and has proven helpful for lesion segmentation as shown in [54].

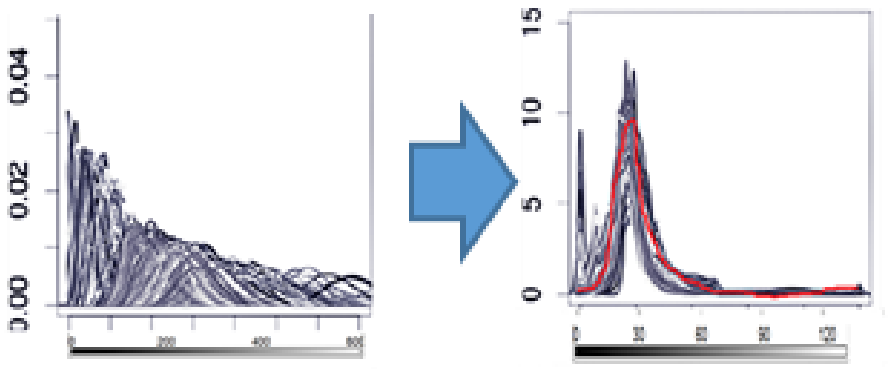


Figure 36. Two-stage histogram matching. Adapted from **[54]**

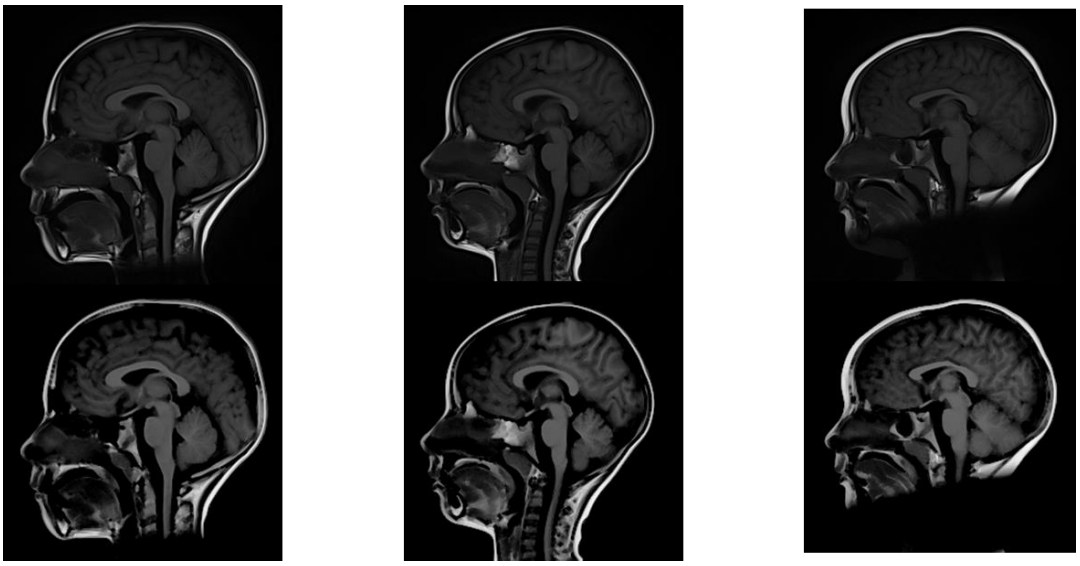


Figure 37. First row: examples of original images.
Second row: examples of normalized images.

4.3.2 Data Augmentation

To assure that the system will be robust enough with only 1995 training images, we then use data augmentation. In particular we use flipping, translations, rotations and scaling, as shown in Figure 38. To do translations and rotations, we use affine transformations in order to preserve points and lines, not changing the image structure. A horizontal flip is done because it is still possible to find old MRI acquisitions that do not comply with the AC-PC convention. The Scaling is used in this part of the work to facilitate Transfer Learning to children, having these subjects a head size lower than adults. In conclusion, in addition to the original image we have a flip left-right, 2 rotations, 4 translations and 2 scaling. In this way, we are augmenting the total training set (both the adult T1-w MRI and the corresponding CC segmentation) by a factor of 10, for a total dataset of size 19950.

DATA AUGMENTATION: 1995 x 10 = 19950
Original
Flip left-right
Rotate 30°
Rotate -30°
Translation (0, 20)
Translation (0, -20)
Translation (20, 0)
Translation (-20, 0)
Scaling 0,8
Scaling 0,5

Figure 38. Data Augmentation table

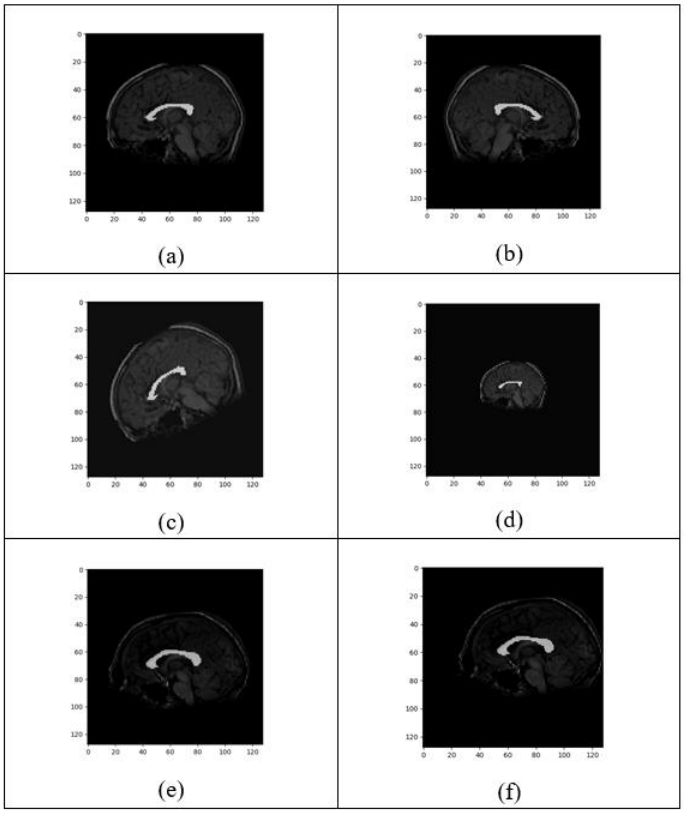


Figure 39. Examples of Data Augmentation of a T1-w MRI with overlap the corresponding CC segmentation. **(a)** Original image. **(b)** Flip left-right. **(c)** Rotate 30°. **(d)** Scaling 0.5 **(e)** Translation (0, -20). **(f)** Translation (20, 0)

4.3.3 Proposed CNN Architecture

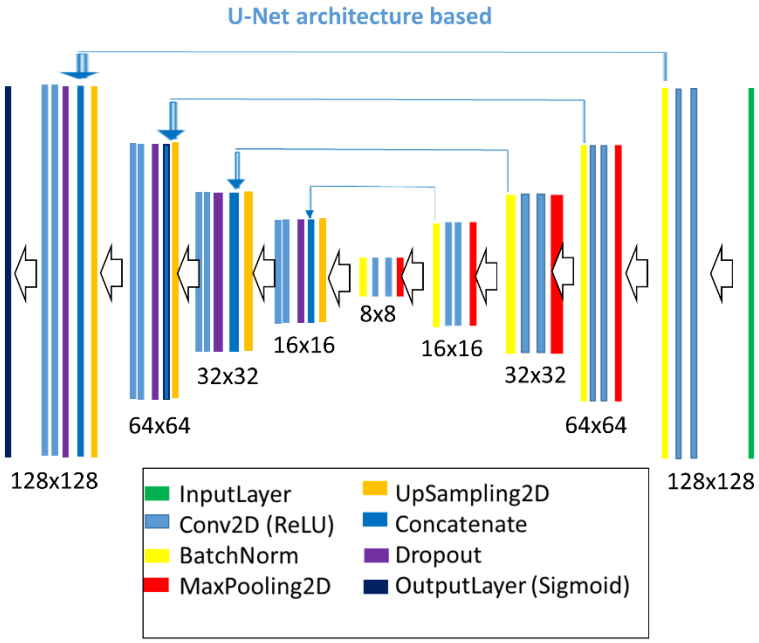


Figure 40. Our trained CNN model architecture

At this stage, with the dataset ready to be used, we can focus on our Convolutional Neural Network with an architecture based on U-Net (Figure 40). The proposed training model alternates between the Input Layer and the Output Layer:

- Convolutional Layer (for bidimensional images) with a ReLU activation function,
- Batch Normalization Layer (in downsampling path),
- MaxPooling (for bidimensional images in downsampling path),
- Concatenation Layer (in upsampling path),
- Dropout Layer (with a Dropout Rate of 0.5 in upsampling path),
- MaxUnpooling (for bidimensional images in upsampling path).

We choose the Sigmoid Neuron for the Output layer and we train the NN using a size of mini batch of 32, an initial amount of filter to start the training of 32 with a kernel size of 3x3. The number of epochs for training is 100. The trained model presents a total of 87,157,889 parameters, where 87,153,921 are trainable and 3,968 are non-trainable.
Most certainly, we present the model with the last configuration, considered by us the best among those tested and with good enough results, which we will analyze in Chapter 5.

Figure 41 shows the complete pipeline for the training of our proposed modified U-Net on OASIS and ABIDE (adults) database.

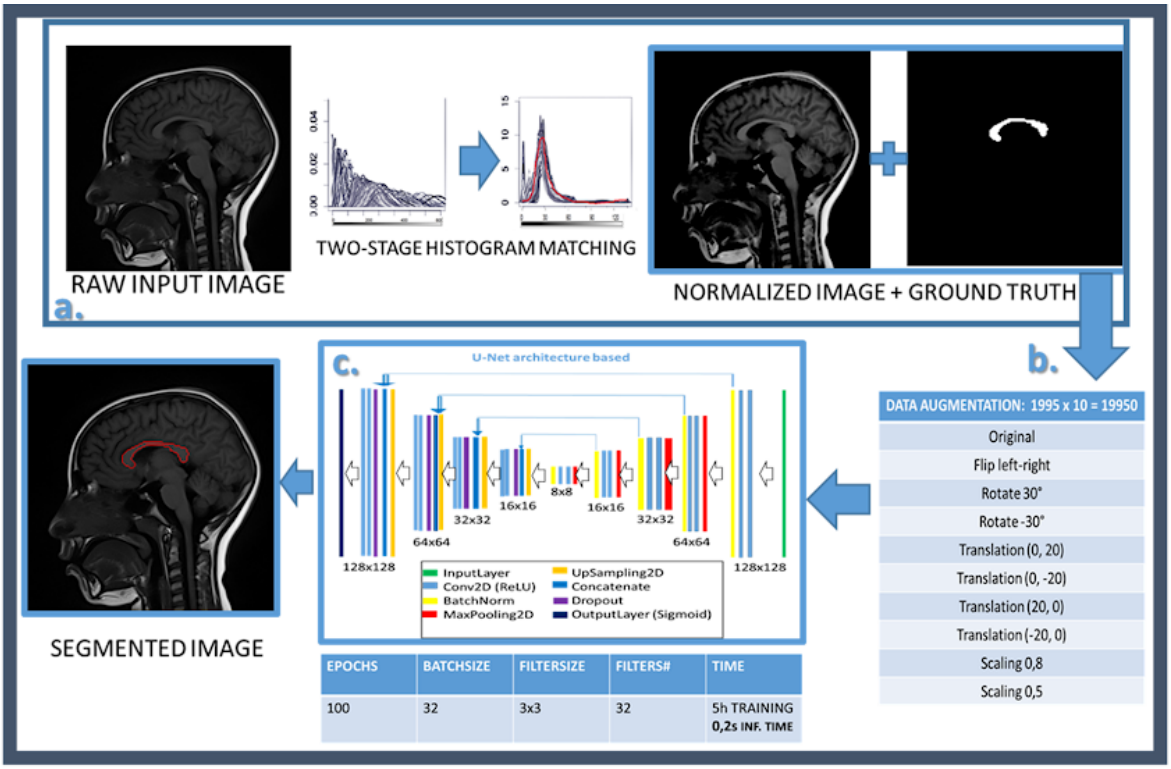


Figure 41. Framework of training on adults' database.
(a) Raw input images are normalized using two-stage histogram matching.
(b) Normalized images and ground truth are augmented through data augmentation.
(c) Model of modified U-Net for training (table shows hyper-parameters, training and inference times).

4.4 Transfer learning on pediatric images

Transfer learning is performed on the same proposed model architecture shown in Section 4.3.3 by only retraining the decoder weights (last layers, as explained in Section 3.5), using saved weights of pre-trained network in the encoder path, as in Figure 42.

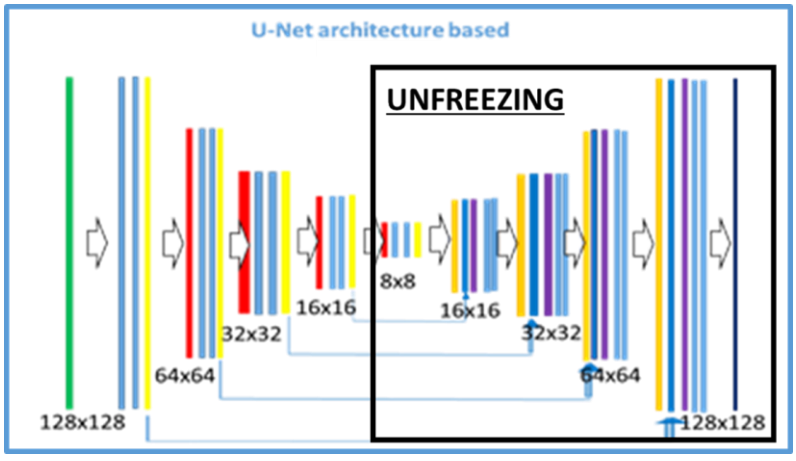


Figure 42. Fine-Tuning TL on the proposed model architecture

The 569 Children images are then normalized using the template histogram of the training phase on adults (Figure 43) according to [54] and what exposed in section 4.3.1.

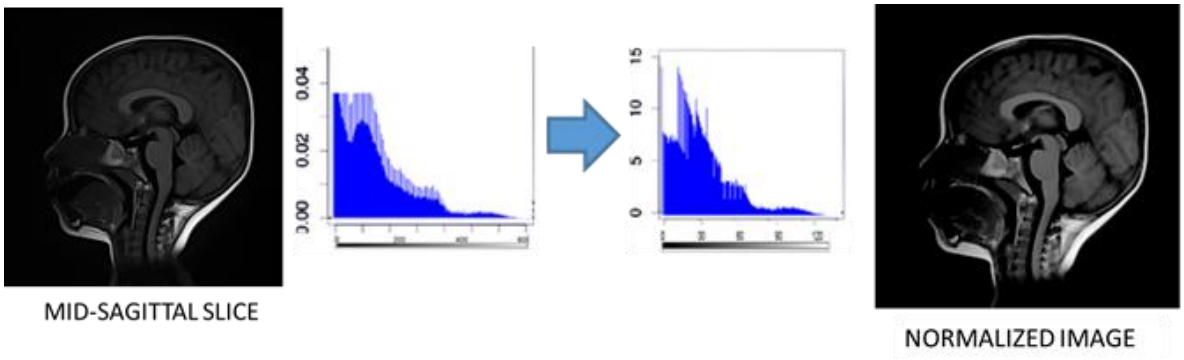


Figure 43. Two-stage histogram matching using template of pre-trained network

After a new data augmentation (as in Figure 38 in Section 4.3.2), the Transfer Learning with Fine-Tuning technique is performed on pediatric database formed by a total of 5690 images. In this way the trainable parameters become only 7,848,065, while all the others remain unchanged (non-trainable).

A simple post-processing due to the low resolution of some images (in particular the images gathered at Bicetre Hospital of Paris, in which Mid-Sagittal Slice was extracted by us) consists in selecting the largest connected component and applying a final morphological closing (a disk-shaped structuring element with a radius of 3 mm) on the output images from the network.

Figure 44 (next page) shows the complete framework of training of Transfer Learning on our pediatric database.

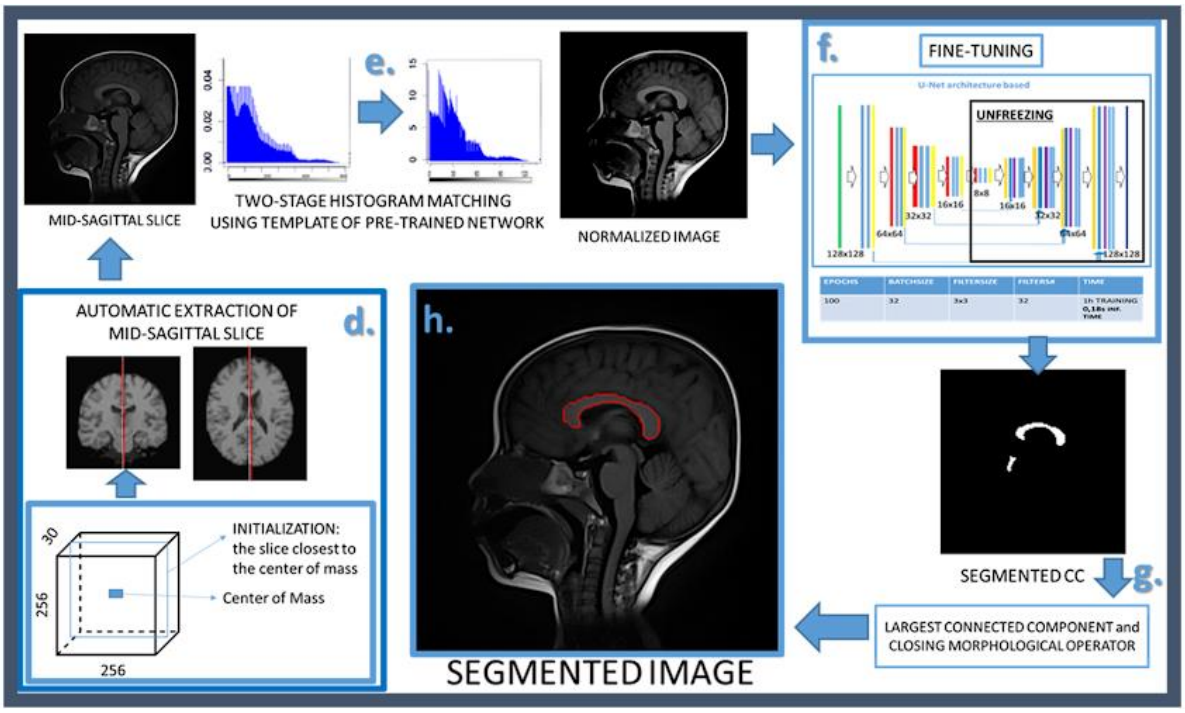


Figure 44. Pipeline of transfer learning on pediatric database.
(d) Extraction of mid-sagittal slice.
(e) Normalization of MSP using template histogram of pre-trained network on adults.
(f) After a new data augmentation (b), the transfer learning with fine-tuning technique is performed on pediatric database (table shows hyper-parameters, training and inference times).
(g) For some images, a simple post-processing consists in selecting the largest connected component, and applying a small closing morphological operator.
(h) Segmented image.
Framework executable for whatever new NIfTI images, using the saved weights.

Chapter 5

Results and Discussions

In this chapter we will describe our results on 170 children's images of the test set. We will show the qualitative results, focusing on the robustness of our proposed method, and the quantitative results, by comparing different CNN aproaches as well as ART-yuki results through indexes of measure (previously presented in Section 3.2.3 and 3.2.5). Later, we will point out area, perimeter and thickness profile of CC of Multiple Sclerosis patients, to discuss what is supposed by Bicetre's pediatric radiologists.

5.1 Qualitative results

Initially, we proceed by analyzing our results in a qualitative way. Comparing the reference images, considered by us the Ground Truth, with the predicted image from our CNN, we can immediately appreciate the good performance of the network (Fig. 45 in the next page).

From Figure 46 we can deduce that the goal to make our network able to segment the Corpus Callosum in children has been reached. We can see good segmentation of MRI with the presence of the neck or part of the body, and additionally babies with rotated head, which could be partially missing.

Figure 47 also illustrates the robustness of the proposed approach with respect to different protocols, making our work very applicable to real situations.

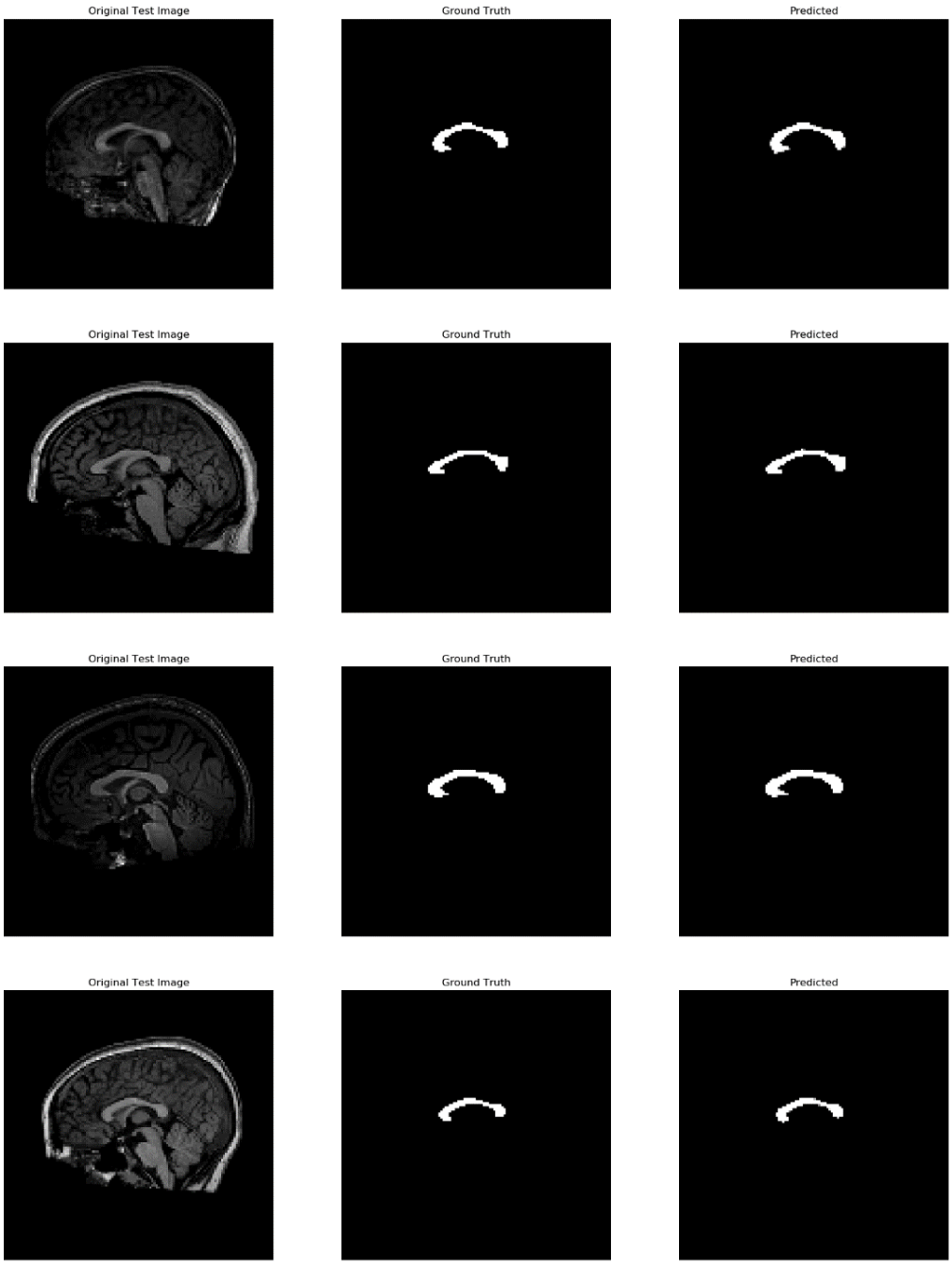


Figure 45. Examples of results from four ABIDE children images of test set.

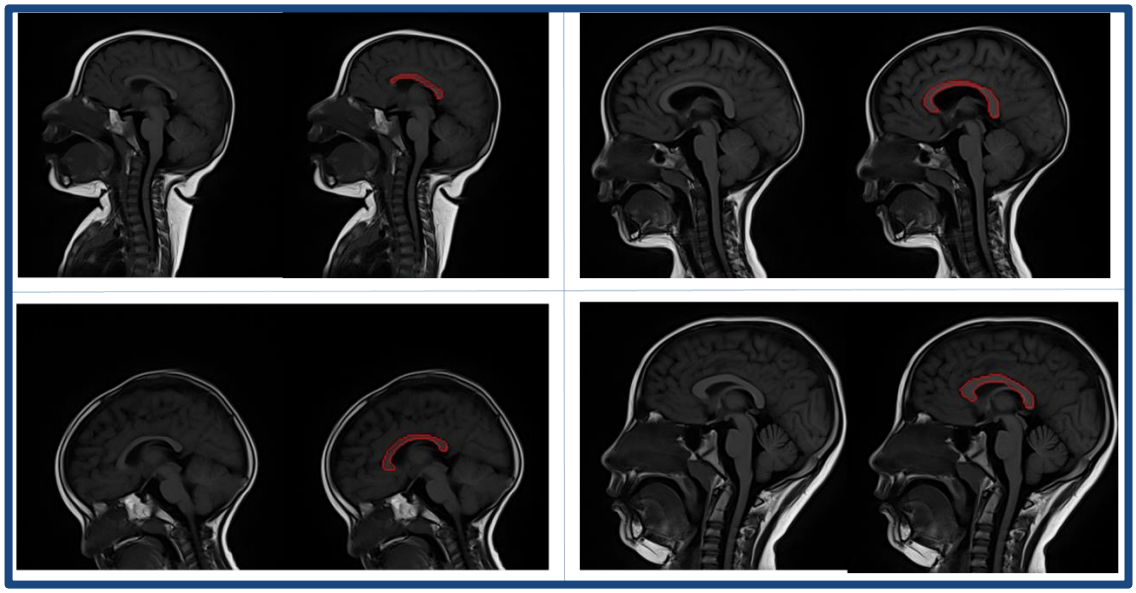


Figure 46. Some excellent results on particular Bicetre controls images:
T1w SAG BLADE 1.5 TESLA SIEMENS

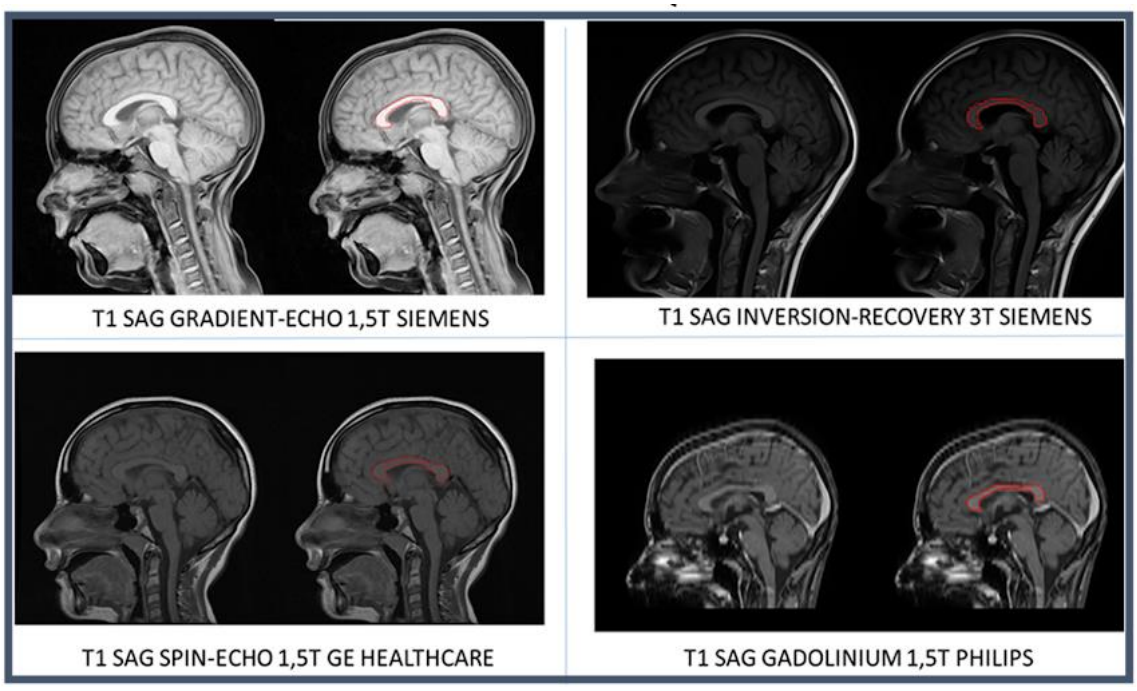


Figure 47. Ability of our network in segmenting correctly images from various kinds of MR scanners with different T1w techniques. (Left = Original Image; Right = Segmented Image)

5.2 Quantitative results

The table in Figure 48 illustrates the principal hyper-parameters of the pre-trained network on Adults. Remembering the total number of parameters and the number of the images after data augmentation, a training time of 5 hours in a NVIDIA GeForce GTX580 CUDA-enabled GPU is really satisfying. This network serves us only as a basis for Transfer Learning, as explained in the related chapter. Moreover, if we will want to use this CNN on other adults' images, the inference time takes only 0.2 seconds per image in a regular Intel®Core™ i7-7500U CPU. Figure 49 shows the loss (opposite of Dice index, -1 means we are making no mistakes) obtained during the training of our network on Adults images. Focusing on the validation loss curve, we can see that the model is a little bit unstable, but this instability does not cause an overfitting, the validation loss curve is close to the training one. Anyway, this is a preliminary step for our purpose: obtain significant results with the Transfer Learning on pre-trained network.

EPOCHS	BATCHSIZE	FILTERSIZE	FILTERS#	TIME
100	32	3x3	32	5h TRAINING **0,2s INF. TIME**

Figure 48. Table of the train on Adults images shows hyper-parameters, training and inference times

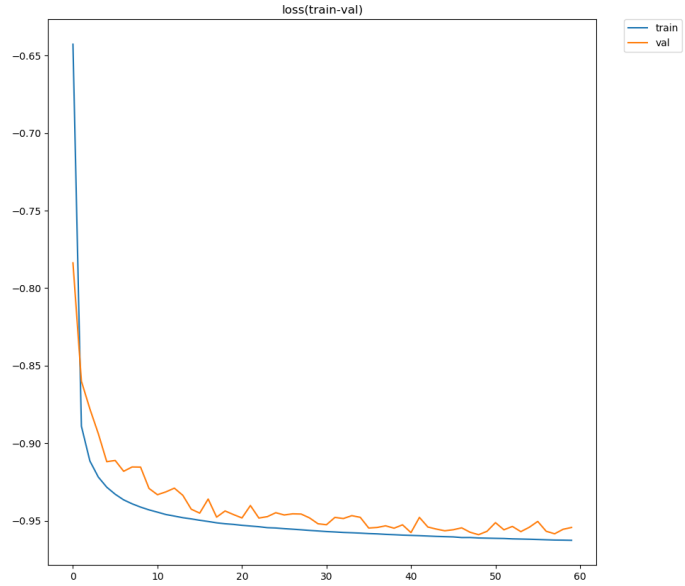


Figure 49. Loss of the train on Adults images (EarlyStopping 60 epochs)

We immediately see the results of the final CNN based on U-Net architecture applying TL. Figure 50 illustrates the principal hyper-parameters of the Transfer Learning network on the pediatric database. We know that in this case the total number of trainable parameters is

significantly lower as well as the number of the images after data augmentation, thanks to the use of the pre-trained weights. The table shows a reasonable training time of 1 hour (in a same GPU as before). Furthermore, the inference time takes only 0.18 second per image (in a same CPU as before), making the system feasible to be used in a real application as we will see in Chapter 6. Figure 51 shows the loss (as in Fig. 49) obtained during the training of the CNN trained with TL technique. Focusing on the validation loss curve, the model is very stable, without overfitting: it fluctuates in a range of 0.01. Still, there is reason to question why the training loss is higher than the validation loss. Firstly, Dropout is turned off at testing time. Besides, what we see is the average training loss over each batch of training data. The loss generally decreases from the first batches of an epoch to the last ones [35] whereas the validation loss for an epoch is computed using the model at the end of the epoch, resulting in a lower loss (anyway, we talk about a range less than 0.04).

EPOCHS	BATCHSIZE	FILTERSIZE	FILTERS#	TIME
100	32	3x3	32	1h TRAINING **0,18s INF. TIME**

Figure 50. Table of the TL on Children images shows hyper-parameters, training and inference times

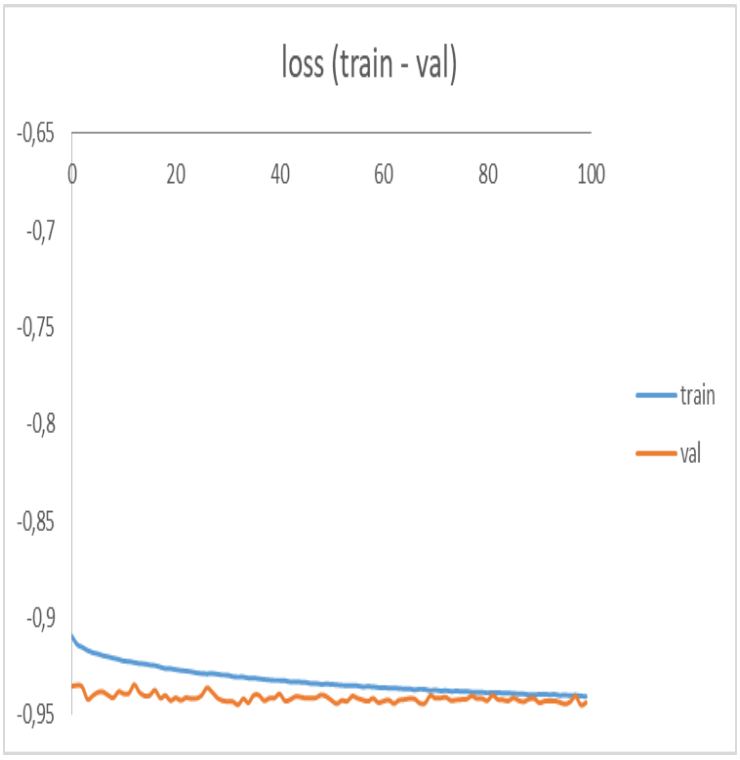


Figure 51. Loss of the TL on Children images (100 epochs)

In Figure 52 we evaluate, using what was addressed in Chapter 3 (Section 3.2.3 and 3.2.5), the proposed method on the test set composed of 170 pediatric images using Dice index (0.93±0.05), Mean Square Error (0.002±0.001 mm) and Structure Similarity index (0.99±0.01). Detailed results per image in Figure 53, 54 and 55, in which it is possible to notice that the first 120 from ABIDE database have better performance than Bicetre images, due to the higher resolution. These results are more than acceptable and are better than with a network trained only on children or using ART-yuki. This is probably due to the low resolution of some test images, as ART-yuki was not always able to correctly extract the MSP, as mentioned before in section 2.2.

Note that learning with both adults and children together, without Transfer Learning, leads to a Dice index of 0.87±0.04.

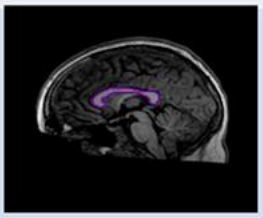
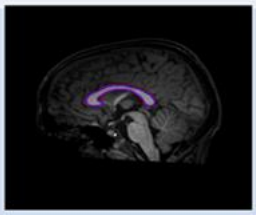
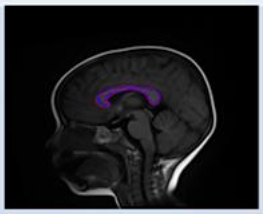
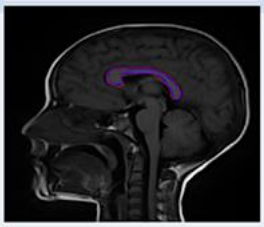
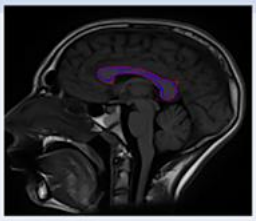
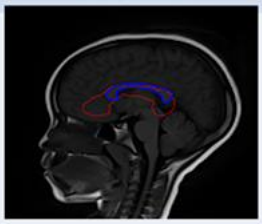
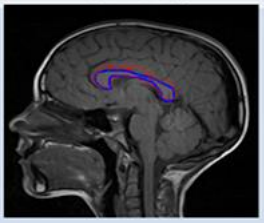
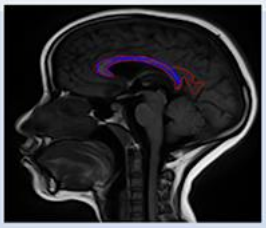
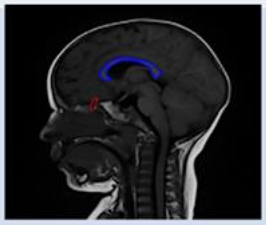
	TRAINING on ADULTS and TRANSFER LEARNING on CHILDREN	TRAINING directly on CHILDREN	ART-yuki
DICE LOSS AVERAGE ± STD DEV	**0.93 ± 0.05**	0.88 ± 0.07	0.72 ± 0.35
DICE LOSS MINIMUM MAXIMUM	**0.75** **1.00**	0.65 0.99	0.00 0.94
MSE (mm) AVERAGE ± STD DEV	**0.002 ± 0.001**	0.003 ± 0.002	0.007 ± 0.006
SSIM AVERAGE ± STD DEV	**0.99 ± 0.01**	0.99 ± 0.01	0.95 ± 0.01
BEST IMAGE			
AVERAGE IMAGE			
WORST IMAGE			

Figure 52. Quantitative comparison using Dice index, mean square error and structure similarity, done on a test set of 170 pediatric images to evaluate the two best networks and ART-yuki; Best, Average and Worst Image are sample images with respectively maximun, average and minimum Dice index. (Blue = Reference segmentation; Red = Predicted segmentation).

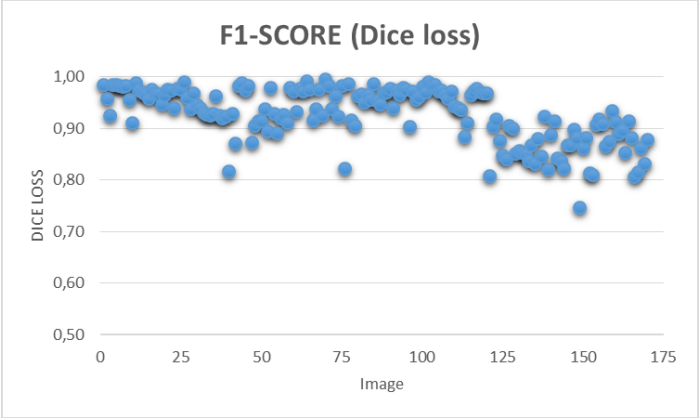


Figure 53. Dice loss index per image

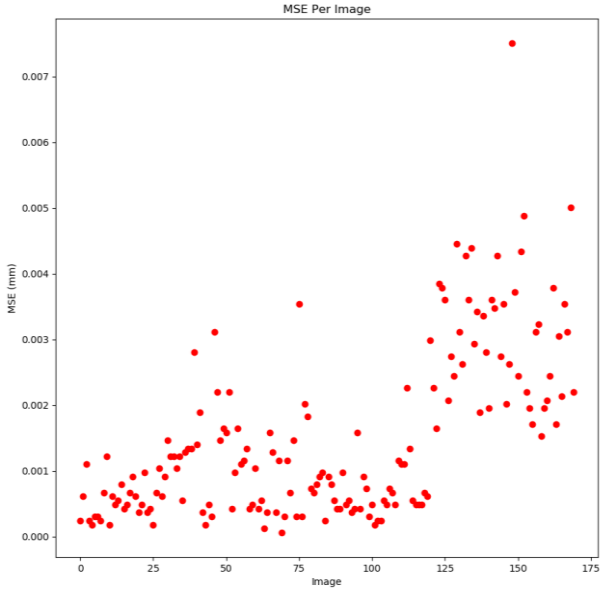


Figure 54. MSE per image

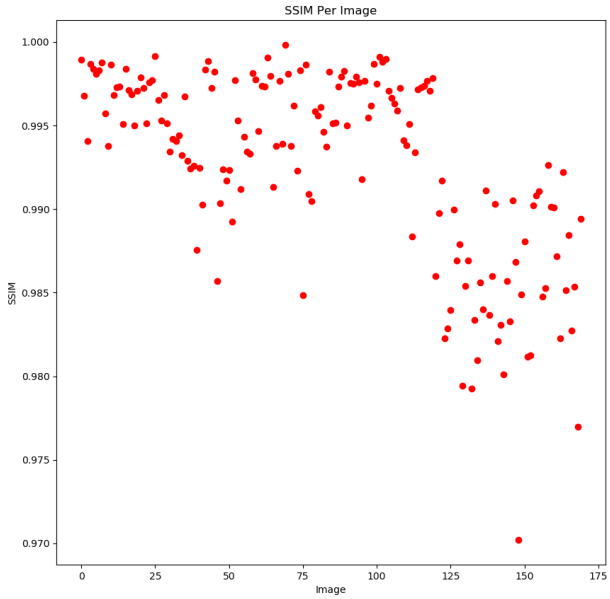


Figure 55. SSIM index per image

5.3 Errors and Particularities

There are particular cases in which our proposed method makes obvious mistakes. The first being contours do not fit in different part, and the second one brain parts that are not the Corpus Callosum are incorrectly segmented.

For instance, MR images with contrast agent such as Gadolinium: it improves the visibility of blood vessels, being mixed up with the CC (Fig. 56). Anyway, these MR acquisitions are not part of a standard protocol.

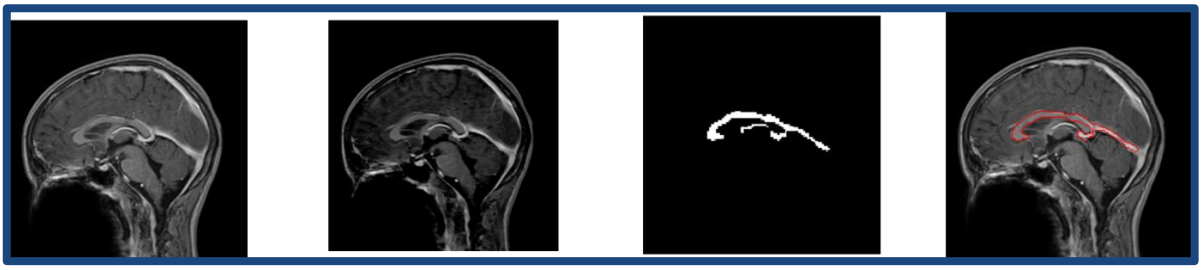


Figure 56. Example of an image with Gadolinium injection.
From left to right: original image, normalized image, segmented CC, segmented image

Another example concerns children MRI for those babies who have not yet completed the process of myelination. Before and after normalization, the Corpus Callosum is non-visible and non-detectable (Fig. 57) in the MSP.

Figure 57. Example of an image with non-visible Corpus Cassolum.
From left to right: original image, normalized image, segmented CC, segmented image

To our advantage, we can say that there is a clear difficulty also in the manual segmentation. The same holds for the case of images with very high motion artifacts (Fig. 58, next page). Even if after normalization we improve the visibility of the CC, the program shows difficulties to find the right contours. However, to recognize where this is and to approximate its shape is already a good result.

A particularity of our method is bring out the Corpus Callosum thanks to the normalization technique applied, that allows a good segmentation of the CC even when its brightness is equal to that of other structures of the brain, such as White Matter (Fig. 59, next page).

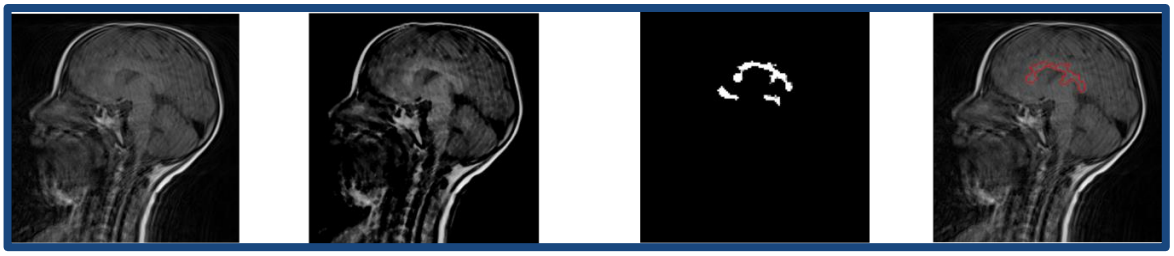


Figure 58. Example of an image with high motion artifacts.
From left to right: original image, normalized image, segmented CC, segmented image

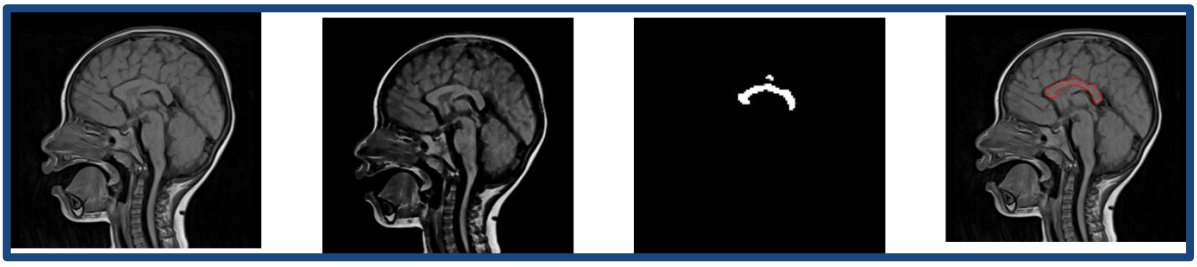


Figure 59. Example of an image with brightness of CC equal to White Matter.
From left to right: original image, normalized image, segmented CC, segmented image

Our program is not able to segment T2-w MR images, but we first hypothesized that it would perform well with FLAIR T2-w MRI. Figure 60 shows that this is not true, probably because of the different histograms leading to the disappearance of the CC (darker than in a T1-w MRI) with the Two-Stage histogram matching technique that we use.

Future work may involve further shape constraints or adaptations to different types of weighted sequences to help similar problems.

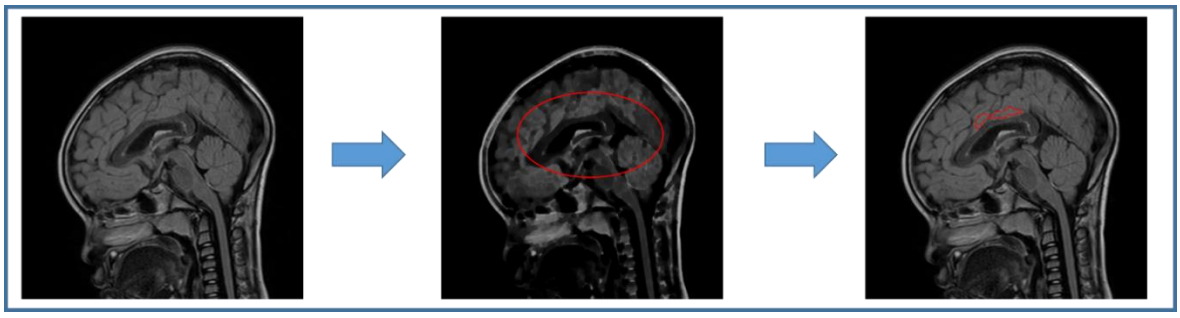


Figure 60. Bad segmentation with FLAIR T2-w MRI

Chapter 6

Morphometric study of the Corpus Callosum in the case of Multiple Sclerosis

The morphological study of the Corpus Callosum intercedes in the follow-up of many cognitive diseases like Multiple Sclerosis. The observation of the thinning of the Corpus Callosum can be an important information in the study of the evolution of a pathology. However, it is not always easy to identify deformations and even less to quantify them with the naked eye. We propose in this part a useful method on this subject.

The morphological study of the Corpus Callosum can have two different ends:

- we may want to relate the particular form of the corpus callosum to a pathology (Multiple Sclerosis [9] or Autism [14] [1]);
- one may also want to follow the progress of a disease by measuring a change in the shape of the Corpus Callosum (Multiple Sclerosis for example [9]).

In the second case we must highlight deformations with respect to a previously determined form of the CC. This is what Bicetre's radiologists hypothesize: the Corpus Callosum decreases in size and shape over time in Children who have MS, rather than in controls or those affected by other inflammatory diseases, such as ADEM.

6.1 Geometric measurements and index for the morphological analysis of CC

A simple way to characterize a change in morphology is first to calculate the perimeter and area of the Corpus Callosum in the MSP (as was done in [55]).

Another index used [56] is called "Corpus Callosum Index" (ICC) that compares the thickness of the genu, midbody and splenium with the length of the CC (see Figure 61). Normally this index is calculated manually, for this reason it is unused.

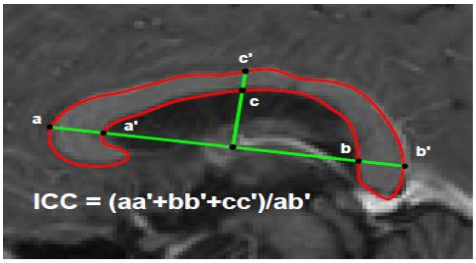


Figure 61. Corpus Callosum Index

As presented in the Chapter 1, it is possible to subdivide the CC in 5 parts according to the subdivision of Witelson (see Figure 3), this allows to define and to independently study the zones [8].

One more index is the Median profile, that combines the study of the shape of the Corpus Callosum with its subdivision [57]. The main idea is to extract the skeleton from the structure of the CC, which means to find the line that connects the two extremities and that passes equidistant from each side of the Corpus Callosum. Then, we can select a certain number of points on this median line and calculate the thickness of the corpus callosum in these places, having all the Thickness Profile.

Entering into the details, the algorithms of the selected geometric measurements for the study are:

- **Area**: we sum all the pixels of the binary segmentation mask and we multiply them by the voxel size.
- **Perimeter**: we take each pair of consecutive pixels of the contour and we calculate the distance between them. The sum of all these distances is multiplied by the voxel size of the image to have the real length.
- **Thickness Profile:** From the binary mask of the segmentation of CC, we will first extract the binary skeleton by morphological operations. The skeleton often consists of several branches. It will therefore be necessary to eliminate some of them to extract the median profile: the line that goes from one end to the other of the CC (in the posterior-anterior direction) and which passes in the middle of this one. It is the method of "skeletal pruning" by morphological transformations of hit-or-miss, which is presented in [57]. The way in which this method works is as follows. Once the binary skeleton is extracted, all the branches are detected by searching for the end points. The 24 terminations shown in Figure 62 are used and translated into each pixel of the image. If the element is exactly identical to the patch of the image to which it is compared, then the corresponding central pixel is set to 1. At the end of these operations we obtain a binary image with pixels at 1 which correspond to the endings of the branches of the skeleton. The pixels corresponding to the detected terminations are deleted. We repeat these operations until only two endings remain, meaning that there is only one branch left. Having kept all the eliminated pixels in memory, we then restore the eliminated pixels associated with the two remaining branches, resulting in our Median profile.

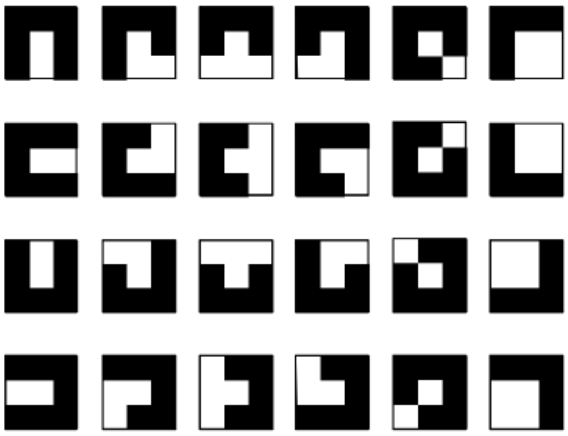


Figure 62. The 24 structuring elements for the detection of terminations. Reprinted from **[57]**

Afterwards, N equidistant points are selected on the median profile. From these points we then look for the closest points of contour, above and below the median axis. The distances between these points and their counterparts on the median axis are calculated. In this way we have two series of distances, the sum of the upper and lower thicknesses at each point makes it possible to obtain the total thickness profile. Figure 63 shows the complete framework for the calculation of the thickness profile.

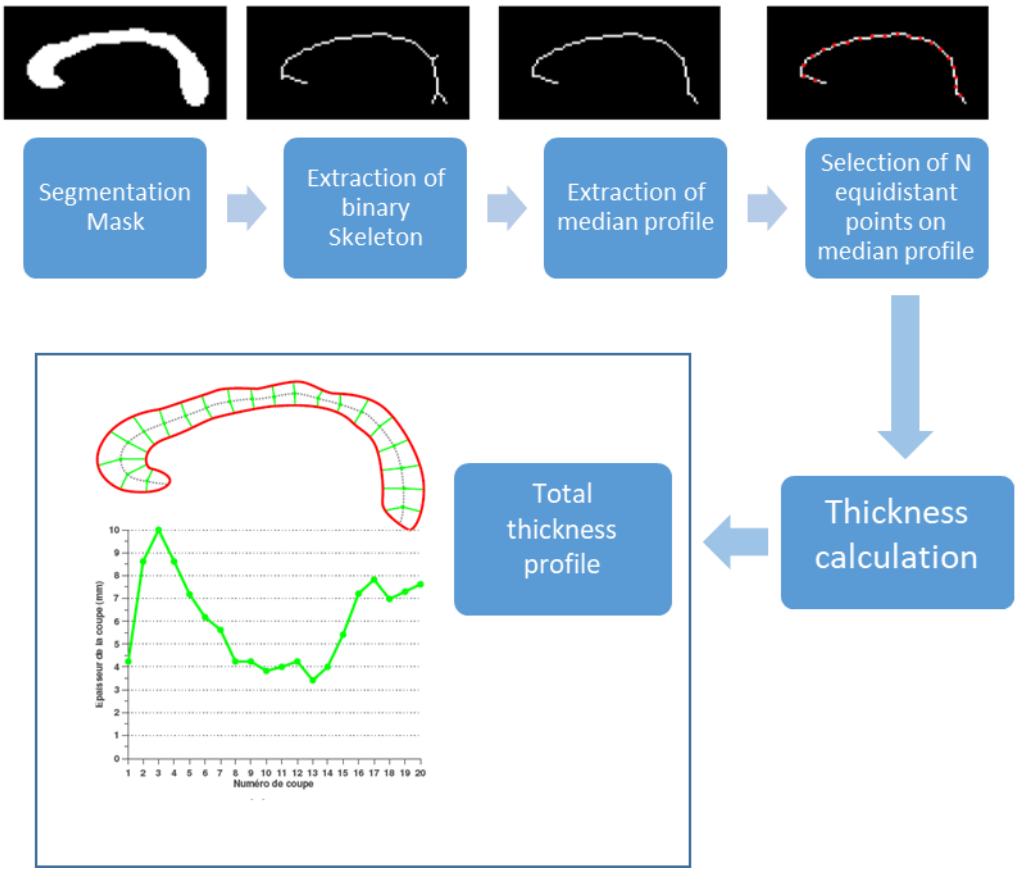


Figure 63. Pipeline of the total thickness profile calculation

6.2 Patients dataset

We have 33 MS patients with different time-steps. Unfortunately, the greatest part of them presents the T1-w Sagittal MRI focused on the spine, due to the particular inflammatory disease. In this case the head is usually cut off, without the presence of the Corpus Callosum, as shown in Figure 64.

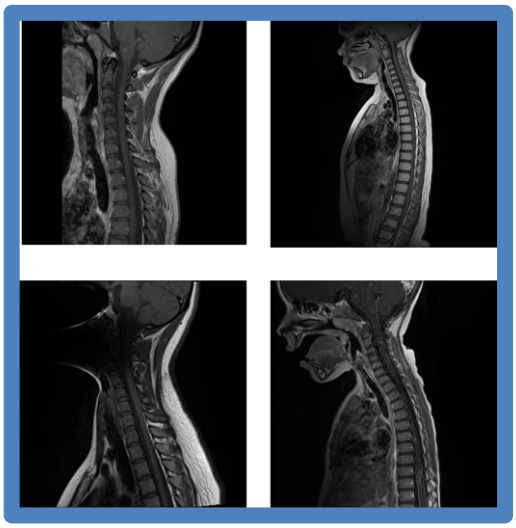


Figure 64. Examples of Spine T1-w Sagittal MRI

We can find some T1-w Sagittal focused on the head but for the reasons stated above they are not many. In addition, those one focused on the head are often with Gadolinium (problems with our program as said above).

A solution is to use the AAHead Scout Sagittal MRI. This is one of the three-panel localizer scans to register the anatomy to the coordinate system of the scanner, so the images have low resolution (160x160) and a large field-of-view. These images are not ideal but our program performs good segmentation on them.

The limitations stated above highlight the problems that arise when working with a real dataset.

Accordingly, we chose 14 patients in the STAP database affected by MS and 1 with Acute Disseminated Encephalomyelitis (ADEM), trying to select those with high quality T1-w Sagittal.

Measurements were made on both manually segmented and automatically segmented data.

In particular we made 4 manual segmentations (validated by experts) and we considered the average values for area and perimeter. We estimate the greatest difference, among the differences between the average and each image, as the maximum error that can be induced by the lack of precision of the images.

The automatic segmentation was made by our program. If the values of area and perimeter are in the range of the values of the 4 manual ones, we can consider these values as reliable; otherwise there is a segmentation error (it will be discussed in detail after). Furthermore, this allows to analyze the consistency of morphometric measurements with the medical diagnosis.

Concerning the lack of precision of the image, there are other two factors that have an effect on the results: the orientation and position of the slice can change a little from one exam to another (for the same patient), and the segmentation is done with a resolution of one pixel (approximately).

It is interesting to add that atrophy is not detectable a priori between examinations spaced less than 6 months.

6.3 Results of the morphological analysis

We used one of our ADEM patients to verify further the correct functioning of our program, and also to make sure that used indexes give us all the information, and finally that the idea of Bicetre's experts is confirmed. Figure 65 shows no atrophy of the Corpus Callosum. Area and Perimeter values increase with brain development, as expected. Even if the values between manual and automatic segmentation are slightly different, they follow the same trend (difference due to the lack of precision of the images).

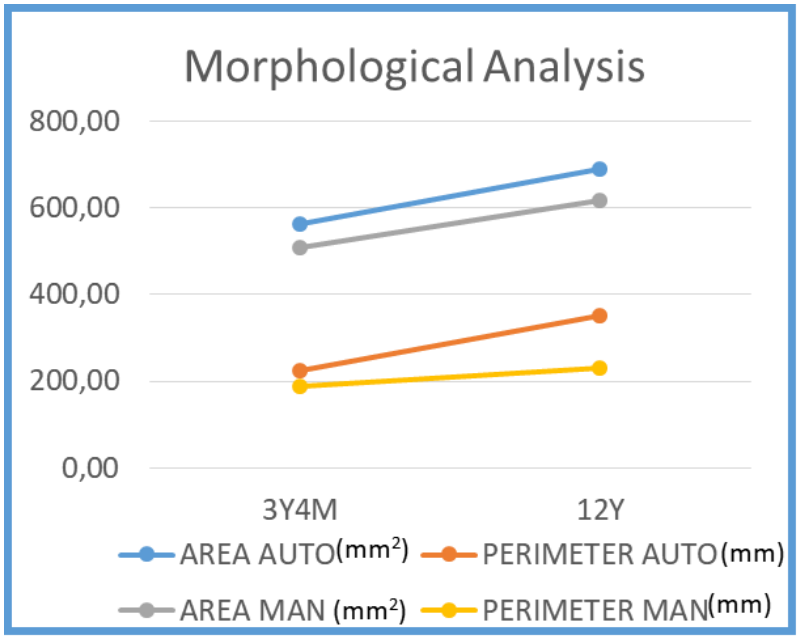


Figure 65. Area and Perimeter values (from automatic and manual segmentation) for the same subject at 3 years and 4 months and at 12 years.

Thanks to the thickness profile (Figure 66) we can evaluate which parts increase more than others, referring to the subdivision of Witelson that we reproduce in the Figure 67.

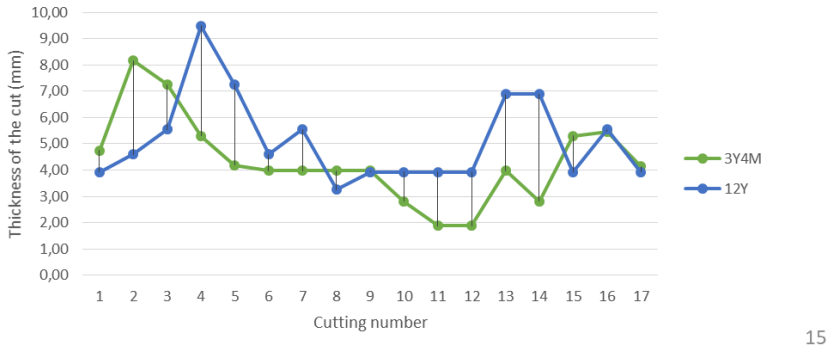


Figure 66. Thickness profile from automatic segmentation of an ADEM patient

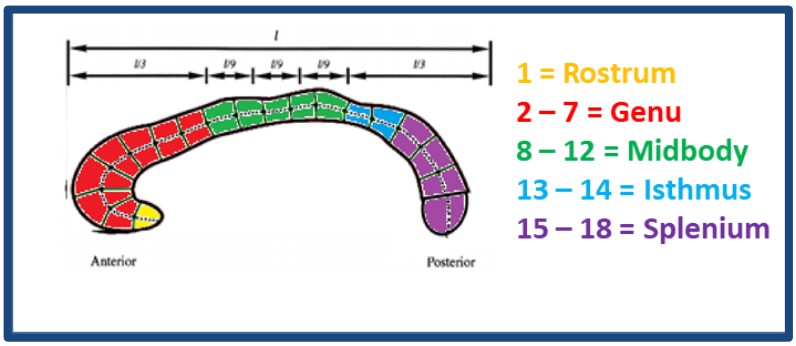


Figure 67. Choosing a N=18 in thickness profile algorithm, we can imagine pairing each cutting number with a part of the CC according to the subdivision of Witelson

We proceed in the same way with all the MS patients (results per patient in Appendix A). The table in Figure 68 shows the evolution of the area and perimeter for each patient with MS calculated in manual segmentations. The time interval between the first and the last exam that resulted in these tests is also mentioned. The differences below the maximum error are not interpreted significantly. Instead, the table in Figure 69 shows the results of the analysis on the segmented CC with our automatic segmentation.

Comparing the two segmentation we can affirm that:

- 7 patients (counting ADEM too) present the same trend, with difference in values in the range due to the lack of precision of the images.
- 5 patients present a comparable trend, some have different percentages because of a very low resolution images and because of the presence of Gadolinium that results in over-segmentation.
- Only 3 patients have an automatic segmentation completely out of the range, in which there is an evident segmentation error (visible over-segmentation).

These data confirm the good performance of our proposed automatic program despite the images present in our dataset, which are not the best for a delicate project of this kind.

	PA1	PS3	PS4	PS8	PS9	PS11	PS16	PS18	PS19	PS21	PS22	PS23	PS24	PS26	PS28
STEP-TIME BETWEEN FIRST AND LAST EXAM	8 Y 8 M (3Y4M – 12Y)	1 Y (15 – 16)	4 Y (9 – 13)	2 Y (12 - 14)	3 Y (13 – 16)	5 Y (11 – 16)	1 Y (13 – 14)	3 Y (13 - 16)	1 Y (15 - 16)	9 Y (7 - 16)	1 Y (14 - 15)	1 Y (15 - 16)	5 Y (10 - 15)	3 Y (11 – 14)	6 Y (10 – 16)
EVOLUTION OF **AREA** BETWEEN FIRST AND LAST EXAM (%)	+21,1	+3,9	-10,6	-17,2	-31,0	+3,7	+2,7	-12,8	+17,4	-0,6	-19,7	+2,5	-1,6	+12,7	+15,7
EVOLUTION OF **PERIMETER** BETWEEN FIRST AND LAST EXAM (%)	+22,7	+2,8	-0,9	-4,5	-6,0	+2,1	+2,5	-3,1	+6,0	-3,4	-1,4	+1,8	-2,1	-3,2	+2,7
LACK OF PRECISION FOR FIRST AND LAST (%)	4,4% 7,8%	8,2% 17,7%	9,5% 4,8%	2,9% 15,6%.	6,9% 6,6%	1,7% 6,0%	8,5% 0,9%	6,1% 7,1%	4,6% 10,6%	9,7% 8,0%	6,9% 3,4%	4,6% 6,6%	8,0% 2,0%	4,7% 11,0%	10,4% 3,0%

Figure 68. Morphological analysis of the segmented Corpus Callosum with MANUAL segmentation. Green = Significant decrease. Red = Significant increase. Yellow = No significant change.

	PA1	PS3	PS4	PS8	PS9	PS11	PS16	PS18	PS19	PS21	PS22	PS23	PS24	PS26	PS28
STEP-TIME BETWEEN FIRST AND LAST EXAM	8 Y 8 M (3Y4M – 12Y)	1 Y (15 – 16)	4 Y (9 – 13)	2 Y (12 - 14)	3 Y (13 – 16)	5 Y (11 – 16)	1 Y (13 – 14)	3 Y (13 - 16)	1 Y (15 - 16)	9 Y (7 - 16)	1 Y (14 - 15)	1 Y (15 - 16)	5 Y (10 - 15)	3 Y (11 – 14)	6 Y (10 – 16)
EVOLUTION OF **AREA** BETWEEN FIRST AND LAST EXAM (%)	+20,3	+27,15	+2,97	-7,84	-14,53	-16,49	+0,21	+1,30	+7,45	-6,84	-29,36	-7,11	+0,34	-5,20	+5,02
EVOLUTION OF **PERIMETER** BETWEEN FIRST AND LAST EXAM (%)	+15,5	-0,61	-2,33	-5,84	-8,35	+2,26	-7,33	-2,56	-3,84	-4,19	-7,48	-8,38	+12,89	-1,85	+0,18
LACK OF PRECISION FOR FIRST AND LAST (%)	4,4% 7,8%	8,2% 17,7%	9,5% 4,8%	2,9% 15,6%.	6,9% 6,6%	1,7% 6,0%	8,5% 0,9%	6,1% 6,6%	4,6% 10,6%	9,7% 8,0%	6,9% 3,4%	4,6% 6,6%	8,0% 2,0%	4,7% 11,0%	10,4% 3,0%

Figure 69. Morphological analysis of the segmented Corpus Callosum with AUTOMATIC segmentation. Green = Significant decrease. Red = Significant increase. Yellow = No significant change.

As a conclusion of our morphological analysis of the segmented Corpus Callosum for Multiple Sclerosis patients, we analyze the results in detail:

- **Perimeter:** it does not change significantly in every image of MS patients. It seems that in a part the CC decreases and in another part it increases. Physiologically, according to Bicetre's experts, it should decrease in the midbody and increase in genu and splenium. Instead, concerning ADEM patients, the perimeter increases significantly.
- **Area and Thickness profile:**

- in 5 patients the CC decreases: in Patient 4 and 18 uniformly, in Patient 8 mostly the genu, while in Patient 9 and 22 the isthmus and splenium mostly increase respectively (the number of the Patients, Patient #, is related to Figures in Appendix A and also to STAP ID of MS patients in Database of Project STAP).
- In 3 patients the CC increases: in Patient 26 uniformly, in Patient 28 the midbody mostly increases, while in Patient 19 both midbody and isthmus increase more than other parts.
- The other 6 patients show an unchanged CC area between the first and last exams, even though time-step of only one year in adolescent age (such as for Patients 3, 16 and 23) is likely not very informative.

The results do not seem to confirm the radiologists' hypothesis but it is worth remembering that there are too many variables that could have an effect on the results of this study (quality of the images, slice cutting, inclination of the head, etc.), as mentioned above.

Chapter 7

Conclusions and Future steps

During this thesis work we have developed an automatic and robust segmentation method for the Corpus Callosum.

The developed Convolution Neural Network allowed to obtain very good performance with MR images, in line with those present in the literature for adults, verified with the use of Dice index, Mean Square Error and Structural Similarity index.

It has been understood that the study of the Corpus Callosum is a great issue in the study of neurological diseases, and in particular of Multiple Sclerosis. The development of tools assisting physicians in the longitudinal study appears necessary and indispensable for the progress of the knowledge and treatment of these pathologies. This dissertation also provided an overview of past and current research on the segmentation for the morphological study of the Corpus Callosum.

Our proposed method permits accuracy and reproducibility for longitudinal studies, to be independent with respect to the variation in the size and shape of the CC among the patients. In addition, it is robust in respect to the background of MR images, to the gaps in the contour and to the presence of the fornix.

Furthermore, the use of Transfer Learning was perfectly suited to the project, guaranteeing excellent results also with children's images. The difficulties of a smaller contrast-noise ratio, brief acquisition time and presence of the body in the image are overcome. The possibility of using our tool with children also allows medical studies, like those on progressive diseases that are developed from an early age, to be done with less difficulty.

All this is done automatically, without handmade features extraction, atlas of experts for template matching or use of previous studies, thanks to the use of Deep Learning.

Moreover, the program works efficiently with all types of MRI scanners, either with different manufacturers, or with different protocols, showing a high adaptability to different MRI modes.

All of this is an absolute novelty: the results obtained give us hope for the reliability of a whole system to the particular case study of the Corpus Callosum, analyzing the different relationships between its morphology and other neurological diseases, beyond that Multiple Sclerosis.

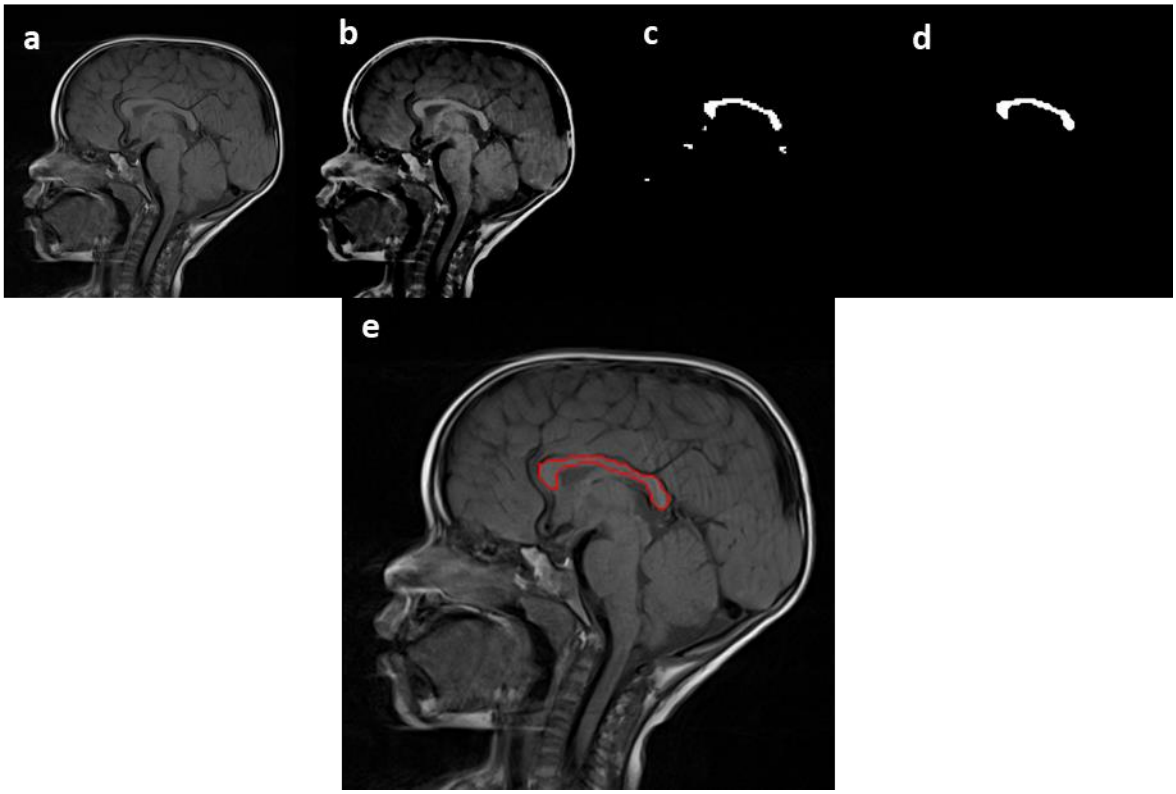


Figure 70. Performance of our proposed method using TL for robust segmentation of the Corpus Callosum tested with T1-w Sagittal Spin-Echo image of a 3 year subject acquired by a 3.0 Tesla Siemens of Bicetre. **a.** Original image. **b.** Normalized image. **c.** Segmented CC as output of the CNN. **d.** Segmented CC after the choice of the largest connected component and the application of closing operator. **e.** Contour of the segmented CC above the original image

We presented three measures allowing the longitudinal comparison of the Corpus Callosum. Even though the perimeter does not give any relevant information, the measurement of the area makes it possible to assess the global evolution of the CC while the calculation of the thickness profile makes it possible to locate the deformations generated by the disease.

The obtained results do not validate the hypothesis of Bicetre's radiologists: they suppose that the Corpus Callosum decreases in size and shape over time in Children who have MS, rather than controls or in those affected by other inflammatory diseases, such as ADEM. However, there are several problems with the images used as shown in Chapter 6, hence it would also be interesting to note that the implementation of a particular protocol for this case study for the acquisition of MRI images is necessary to create consistent results on Corpus Callosum and thus significantly improving performance (reducing the lack of precision).

Finally, the prospects for research and deepening are multiple. A planned change to be made to the system will consist of extensions to other acquisition protocols such as T2-w MRI. Teaching the program how to understand what kind of image it is and in the case of a T2-w convert it to T1-w, there are different algorithms and studies available.

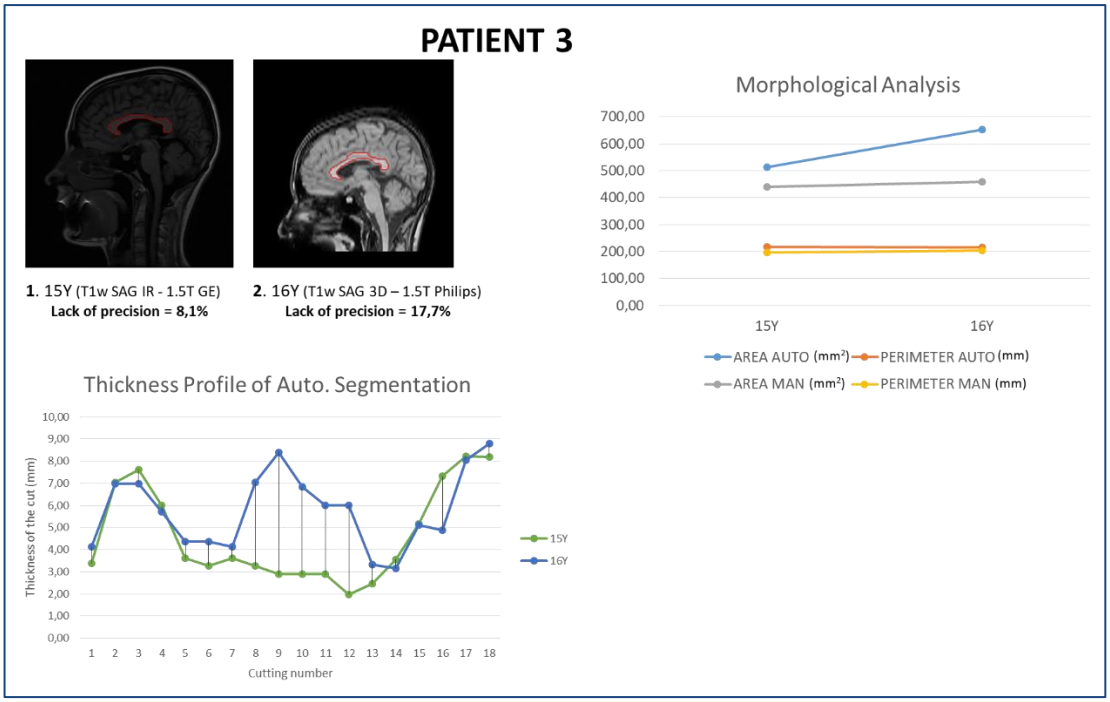
As a further future development of research, it is proposed to do a Statistical Shape Analysis followed by a Principal Component Analysis, because it seems that the CC evolves differently depending on the form that it has. So the goal would be to understand how a healthy subject evolves with that specific form, and if a subject with the same shape does not follow a normal evolution consequently it is a symptom of anomaly.

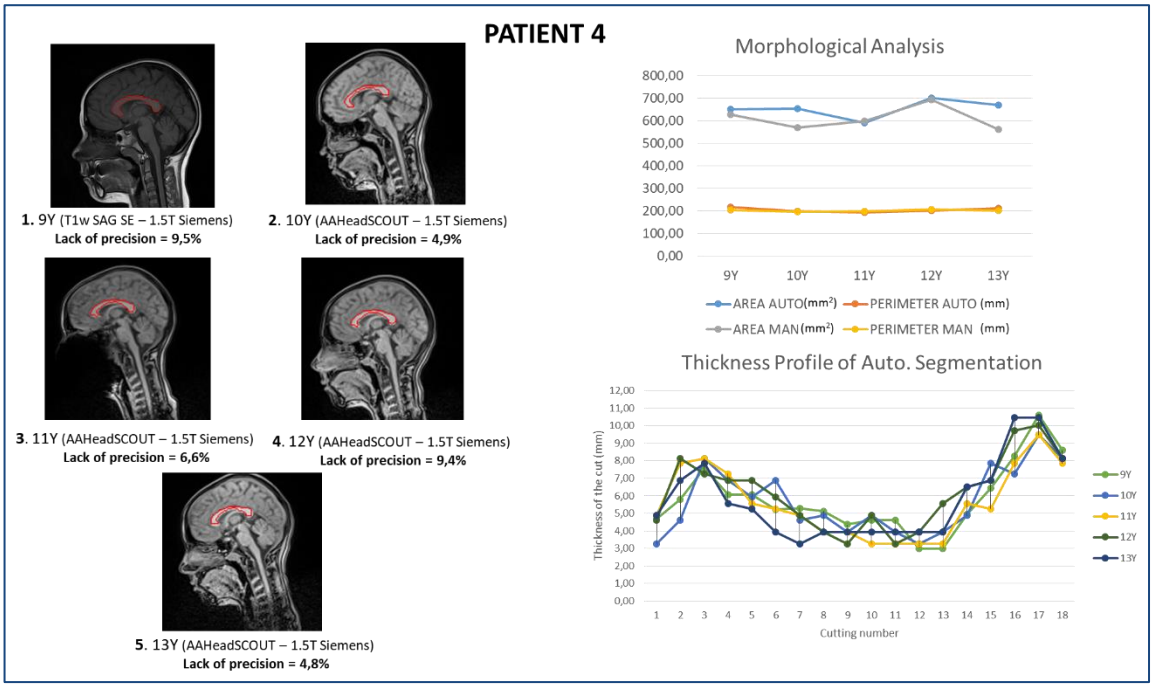
Lastly, the final goal would be the development of a single system capable of telling us right away if a subject presents anomalies (smaller CC area or part of the CC that is more developed than normal) just by giving the program a Sagittal MRI.

Progress in this area seems feasible in the short term. Although new studies to further improve the Convolutional Neural Network could be done, perhaps inserting a particular layers dedicated to shape recognition. However, close collaboration with pediatric radiologists is essential. Exchanges between them and image processing researchers must be made to perfect the methods by helping the technological advancement of the sector and especially for maximizing the benefits for patients.

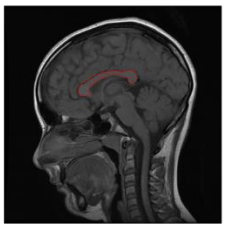
Appendix A

Morphological Analysis of the Segmented Corpus Callosum: Results per Patient

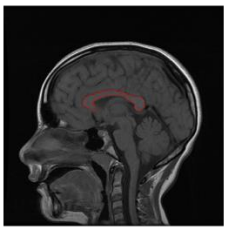




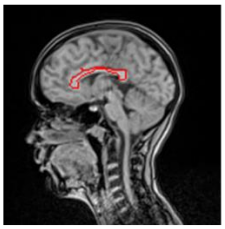
PATIENT 8



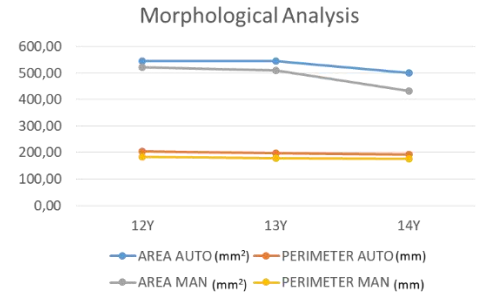
1.12Y (T1w SAG SE – 1.5T GE)
Lack of precision = 2,8%

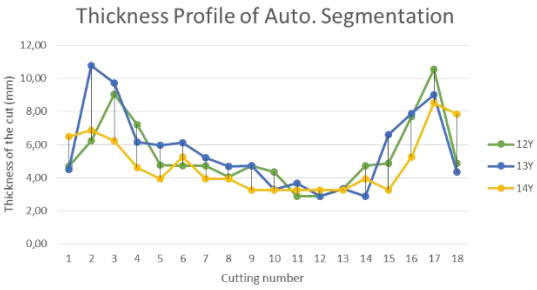


2. 13Y (T1w SAG SE – 1.5T GE)
Lack of precision = 3,9%

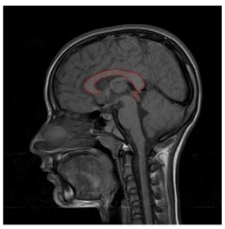


3. 14Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 15,9%

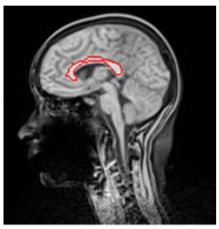




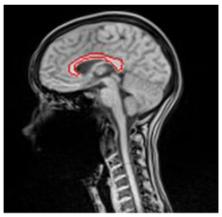
PATIENT 9



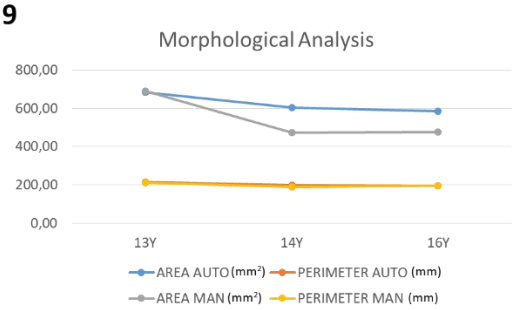
1. 13Y (T1w SAG SE – 1.5T Siemens)
Lack of precision = 6,9%

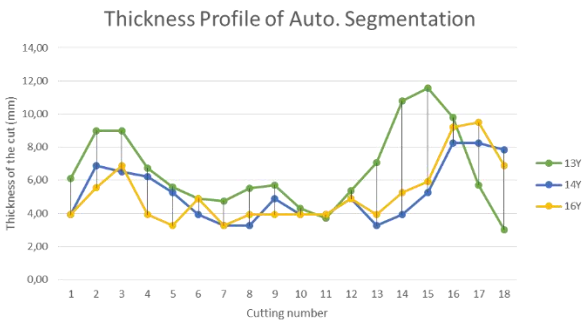


2. 14Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 11,7%

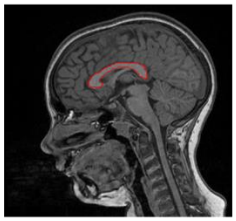


3. 16Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 6,5%

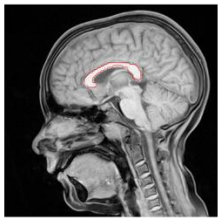




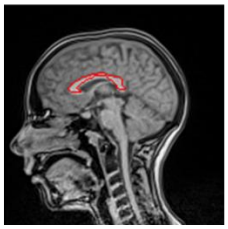
PATIENT 11



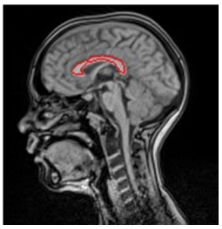
1. 11Y (T1w SAG MPRage – 1.5T Siemens)
Lack of precision = 1,6%



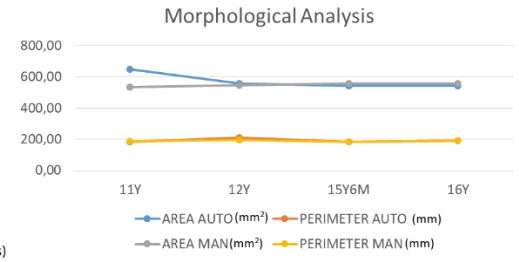
2. 12Y (T1w SAG GradientEcho – 1.5T Siemens)
Lack of precision = 7,5%

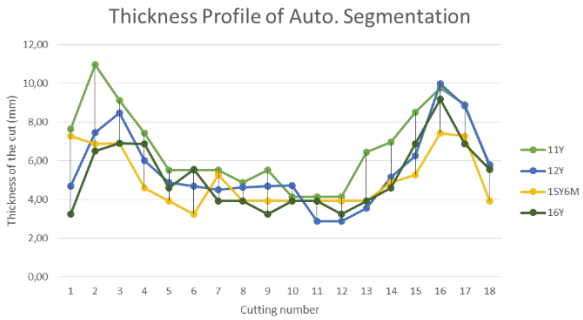


3. 15Y6M (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 15,6%

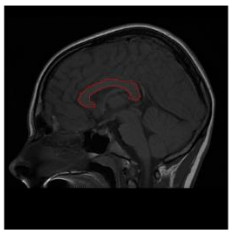


4. 16Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 6,0%

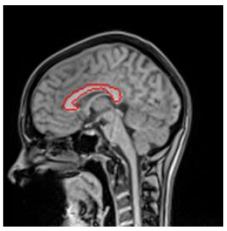




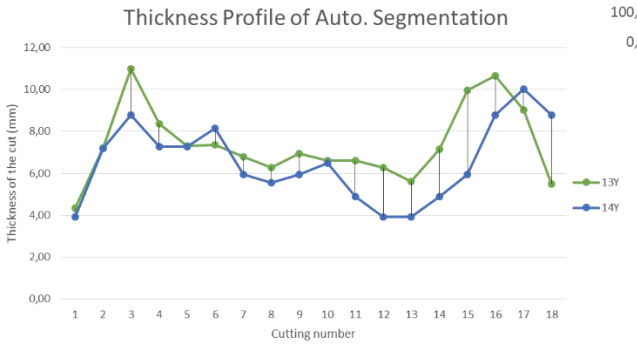
PATIENT 16

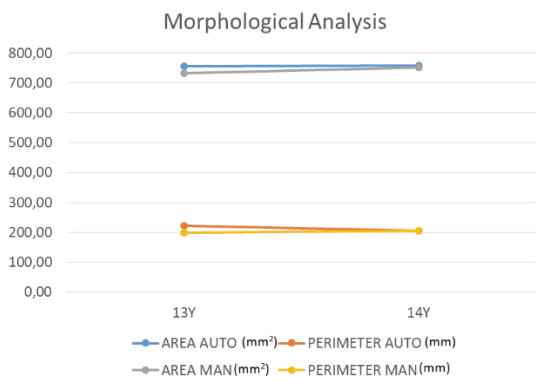


1. 13Y (T1w SAG IR – 1.5T GE)
Lack of precision = 8,5%

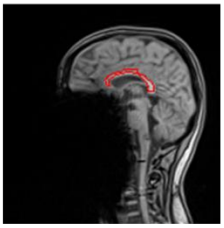


2. 14Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 0,9%

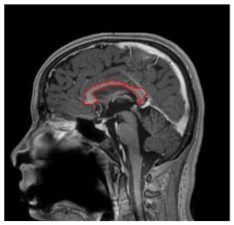




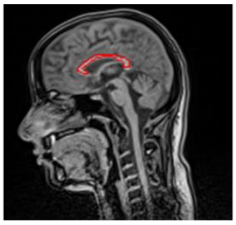
PATIENT 18



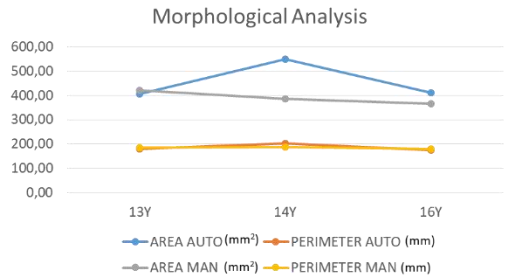
1. 13Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 6,1%

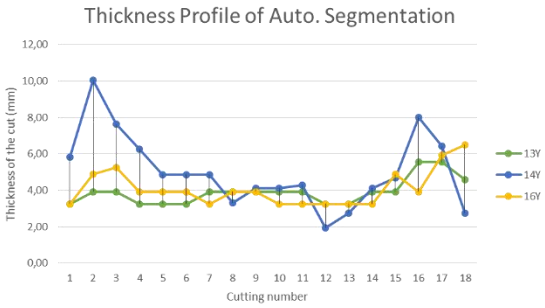


2. 14Y (T1w SAG MPRage – 1.5T Philips)
Lack of precision = 2,1%

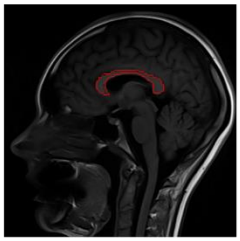


3. 16Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 7,1%

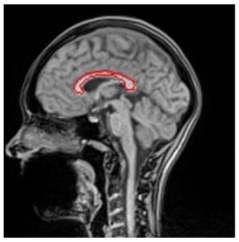




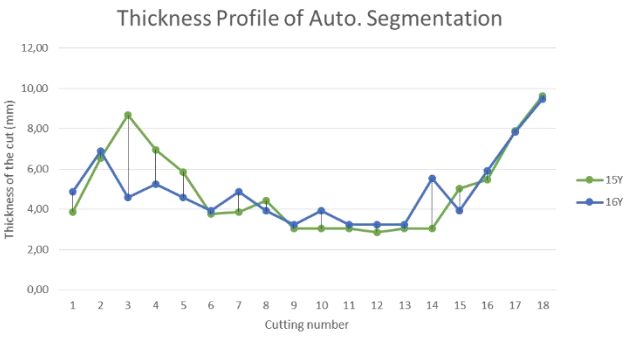
PATIENT 19

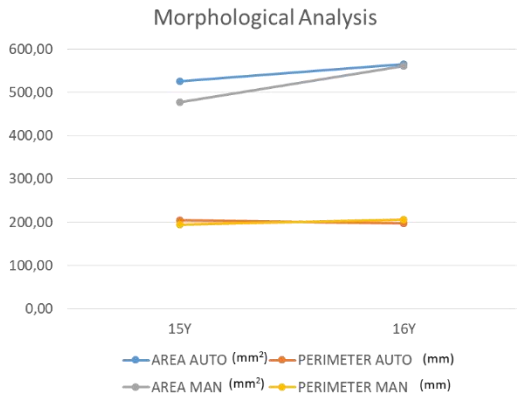


1. 15Y (T1w SAG IR – 3.0T Siemens)
Lack of precision = 4,6%

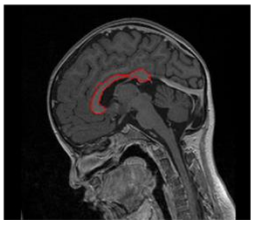


2. 16Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 10,6%

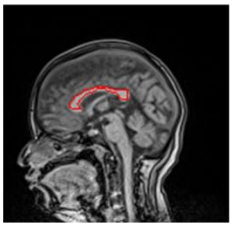




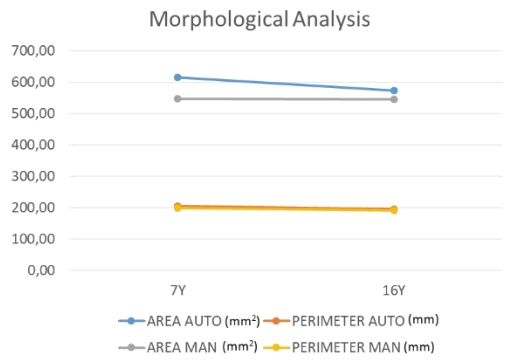
PATIENT 21

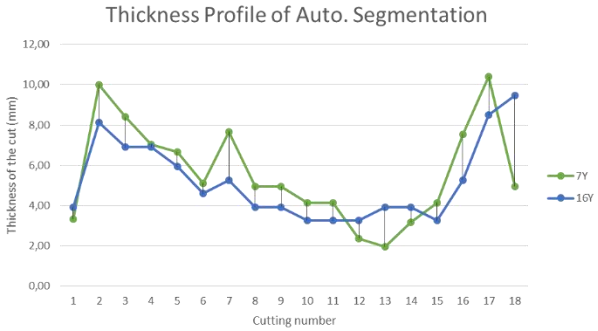


1. 7Y (T1w SAG GADO – 1.5T Siemens)
Lack of precision = 9,7%

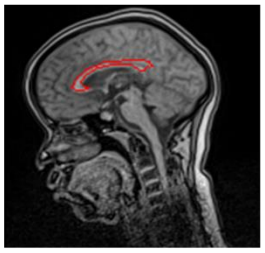


2. 16Y (AAHeadSCOUT -1.5T Siemens)
Lack of precision = 8,0%

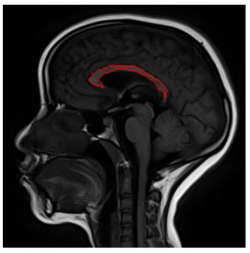




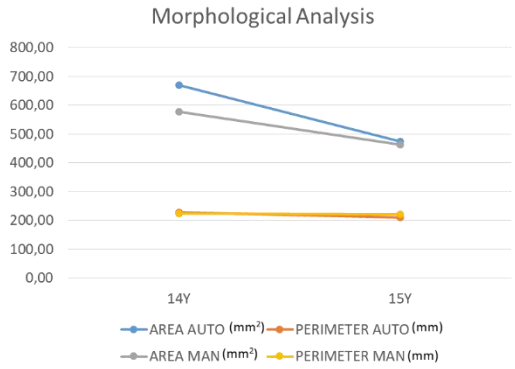
PATIENT 22

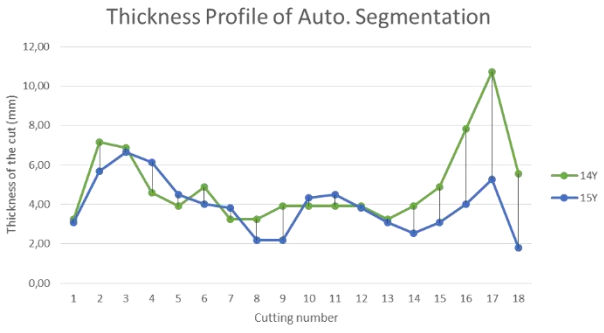


1. 14Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 6,9%

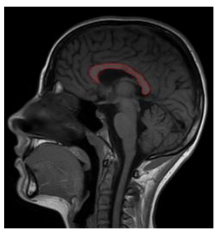


2. 15Y (T1w SAG BLADE – 1.5T Siemens)
Lack of precision = 3,4%

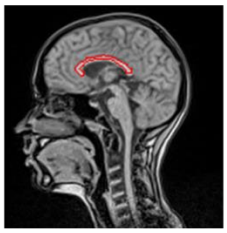




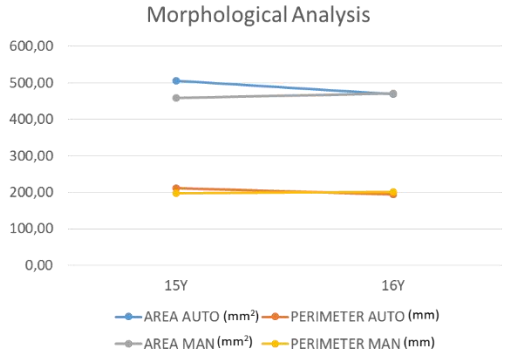
PATIENT 23

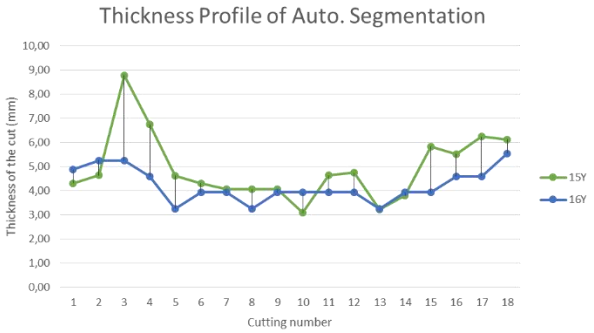


1. 15Y (T1w SAG SE – 1.5T Siemens)
Lack of precision = 4,6%

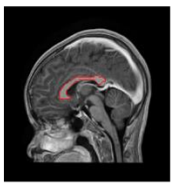


2. 16Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 6,6%

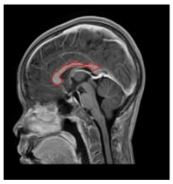




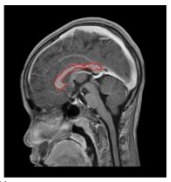
PATIENT 24



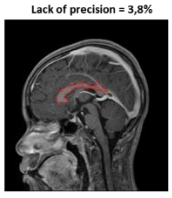
1. 10Y (T1w MPRage GADO – 1.5T Philips)
Lack of precision = 7,9%



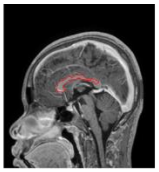
2. 10Y6M (T1w MPRage GADO – 1.5T Philips)
Lack of precision = 3,8%



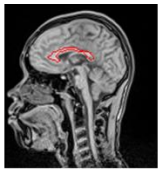
3. 11Y (T1w MPRage GADO – 1.5T Philips)
Lack of precision = 4,2%



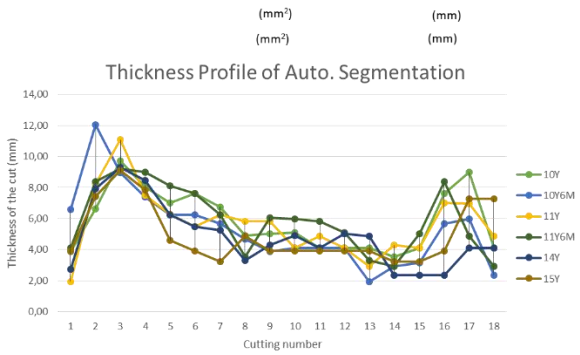
4. 11Y6M (T1w MPRage GADO – 1.5T Philips)
Lack of precision = 3,4%



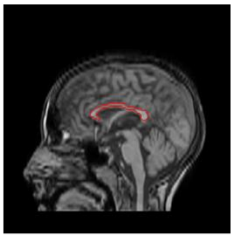
5. 14Y (T1w MPRage GADO – 1.5T Philips)
Lack of precision = 7,3%



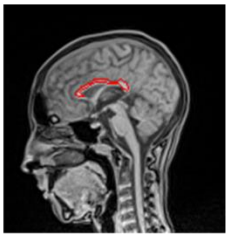
6. 15Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 1,9%



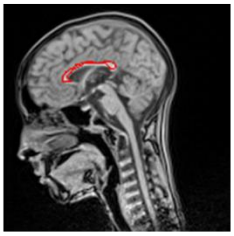
PATIENT 26



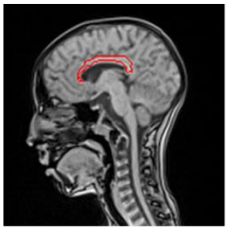
1. 11Y (T1w SAG VE – 1.5T Philips)
Lack of precision = 4,7%



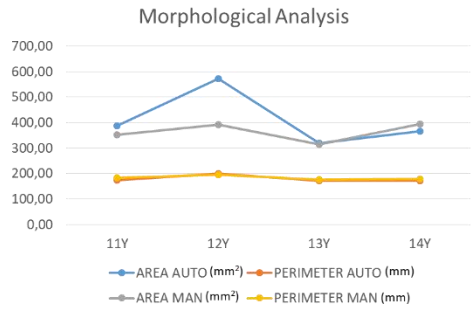
2. 12Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 14,9%

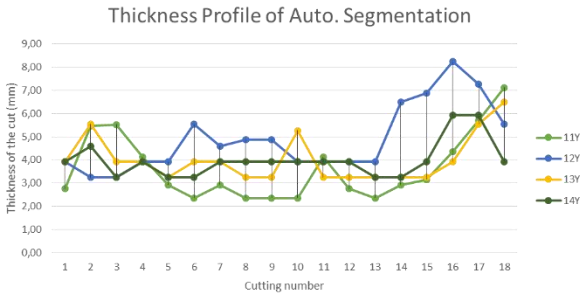


3. 13Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 13,9%

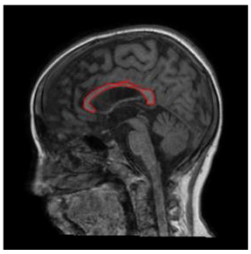


4. 14Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 11,0%

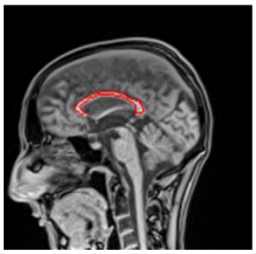




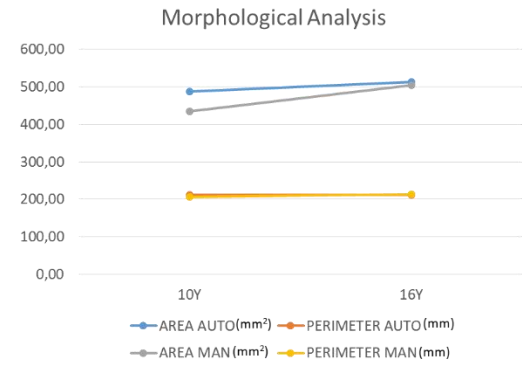
PATIENT 28

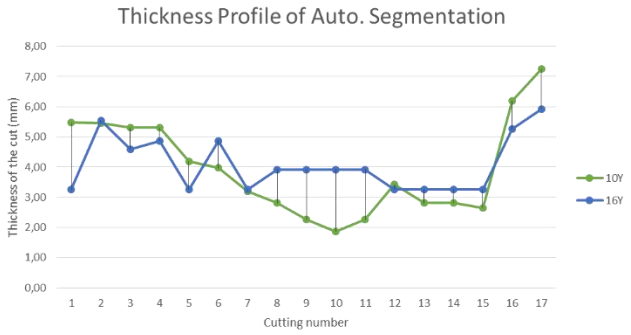


1. 10Y (T1w SAG MPRage – 1.5T Philips)
Lack of precision = 10,4%



2. 16Y (AAHeadSCOUT – 1.5T Siemens)
Lack of precision = 3,0%





References

[1] Vidal CN, Nicolson R, DeVito TJ, et al., «Mapping Corpus Callosum Deficits in Autism: An Index of Aberrant Cortical Connectivity,» *Biol Psychiatry,* vol. 60, p. 218 – 225, 2006.

[2] Harini C, Das RR, Prabhu SP, et al., «Clinical and Neuroimaging Profile of Children with Lesions in the Corpus Callosum,» *J Neuroimaging,* vol. 25, p. 824 – 831, 2015.

[3] Granberg T, Bergendal G, Shams S, et al., «MRI-Defined Corpus Callosum Atrophy in Multiple Sclerosis: A Comparison of Volumetric Measurements, Corpus Callosum Area and Index,» *J Neuroimaging,* vol. 25, p. 996 – 1001, 2015.

[4] Garg N, Reddel SW, Miller DH, et al., «The corpus callosum in the diagnosis of multiple sclerosis and other CNS demyelinating and inflammatory diseases,» *J Neurol Neurosurg Psychiatry,* vol. 86, pp. 1374-1382, 2015.

[5] Vachel C, Yvernault B, Bhatt K, et al., «Automatic Corpus Callosum Segmentation Using a Deformable Active Fourier Contour Model,» in *Proc SPIE Int Soc Opt Eng*, 2012.

[6] Ardekani, BA, « yuki module of the Automatic Registration Toolbox (ART) for corpus callosum,» 14 January 2013. [Online]. Available: http://www.nitrc.org/projects/art.

[7] Gray H, Anatomy of the Human Body, Philadelphia: Lea & Febiger, 1918.

[8] Witelson SF, «Hand and sex differences in the isthmus and genu of the human corpus callosum,» *Brain,* vol. 112, n. 3, pp. 799-835, 1989.

[9] Ishaq O, Hamarneh G, Tam R, et al., «Effects of Mid-Sagittal Plane Perturbation and Image Interpolation on Corpus Callosum Area Calculation,» in *IEEE Internation: Symposium of Signal Processing and Information Technology*, 2006.

[10] Schell-Apacik CC, Wagner K, Bihler M, et al., «Agenesis and dysgenesis of the corpus callosum: clinical, genetic and neuroimaging findings in a series of 41 patients,» *Am J Med Genet A,* n. 146 A, pp. 2501-2511, 2008.

[11] Balboni GC, et al., Corpo calloso, in Anatomia Umana, vol. 3, Ermes s.r.l., 1976.

[12] Preston DC from CaseWesternResverve University, school of medicine, «Magnetic Resonance Imaging (MRI) of the Brain and Spine: Basics,» 2006. [Online]. Available: http://casemed.case.edu/clerkships/neurology/web%20neurorad/mri%20basics.htm.

[13] Kennedy T, «Neuroradiology,» 2018. [Online]. Available: https://sites.google.com/a/wisc.edu/neuroradiology/home.

[14] He Q, Duan Y, Miles J, et al., «A Context-Sensitive Active Contour for 2D Corpus Callosum Segmentation,» *International Journal of Bomedical Imaging,* n. 24826, 2007.

[15] Karsch K, He Q, Duan Y, «A Fast, Semi-Automatic Brain Structure Algorithm for Magnetic Resonance Imaging,» in *IEEE International Conference on Bioinformatics and Biomedicine*, 2009.

[16] Caselles V, Kimmel R, Sapiro G, «Geodesic Active Contours,» *International Journal of Computer Vision,* vol. 22, pp. 61-79, 1997.

[17] Lankton S, Tannenbaum A, «Localizing Region-Baseed Active Contours,» *IEEE Transactions on Image Processing,* vol. 17, pp. 2029-2039, 2008.

[18] Székely G, Kelemen A, Brechbuhler C, et al., «Segmentation of 2-D and 3-D objects from MRI volume data using constrained elastic deformations of flexible Fourier contour and surface models,» *Medical Image Analysis,* vol. 1, pp. 19-34, 1996.

[19] Seixas FL, de Souza AS, dos Santos AA, et al., «Automated Segmentation of The Corpus Callosum Midsagittal Surface Area,» in *XX Brazilian Symposium on Computer Graphics and Image Processing*.

[20] Manasa D, Priya V, «A Hybrid Technique for the Automated Segmentation of Corpus Callosum in Midsagittal Brain Mri,» *Journal of Engineering Research and Applications,* vol. 4, pp. 01-04, 2014.

[21] Li Y, Wang H, Ahmed N, «Automated Corpus Callosum Segmentation In Midsagittal Brain MR Images,» *ICTACT Journal on Image and Video Processing,* vol. 8, pp. 1554-1565, 2017.

[22] Elahi S, Bachman AH, Lee SH, et al., «Corpus Callosum Atrophy Rate in Mild Cognitive Impairment and Prodromal Alzheimer's Disease,» *J Alzheimers Dis.,* vol. 45, n. 3, pp. 921-931, 2015.

[23] Prendergast D, Ardekani B, Ikuta T, et al., «Age and Sex Effects on Corpus Callosum Morphology Across the Lifespan,» *Hum Brain Mapp,* vol. 36, n. 7, pp. 2691-2702, 2015.

[24] Ardekani BA, Guckemus S, Bachman A, et al., «Qunatitative comparison of algorithms for inter-subject registration of 3D volumetric brain MRI scans,» *J Neurosci Methods,* vol. 142, n. 1, pp. 67-76, 2005.

[25] Fisher RA, «The Use of Multiple Measurements in Taxonomic Problems,» *Annals of Eugenics,* vol. 7, n. 2, pp. 179-188, 1936.

[26] «Open Access Series of Imaging Studies (OASIS),» [Online]. Available: https://www.oasis-brains.org/.

[27] «Autism Brain Imaging Data Exchange (ABIDE),» [Online]. Available: http://fcon_1000.projects.nitrc.org/indi/abide/.

[28] da Silva FHS, «Deep Learning for Corpus Callosum Segmentation in Brain Magnetic Resonance Images,» COPPE UFRJ Dissertation of Master's degree presented to the Program of Post-graduation degree in Engineering of Systems and Computation, San Paolo, Brazil, 2018.

[29] Ciresan D, Giusti A, Schmidhuber J, «Deep neural networks segment neuronal membranes in electron microscopy images,» *Adv. Neural Inf. Process. Syst.,* pp. 1-9, 2012.

[30] Ronneberger O, Fischer P, Brox T, «U-Net: Convolutional Network for Biomedical Image Segmentation,» in *MICCAI Medical Image Computing and Computer Assisted Intervention* , 2015.

[31] Fedorov A, Johnson J, Damaraju E, et al., «End-to-end learning of brain tissue segmentation from imperfect labeling,» in *IJCNN International Joint Conference on Neural Networks* , 2017.

[32] Brosch T, Tang LYW, Yoo Y, et al., «Deep 3D Convolutional Encoder Networks With Shortcuts for Multiscale Feature Integration Applied to Multiple Sclerosis Lesion Segmentation,» *IEEE Transactions on Medical Imaging,* vol. 35, pp. 1229-1239, 2016.

[33] Park G, Lee JM, «Segmentation of Corpus Callosum in Midsagittal Palne using Convolutional Neural Networks with Anatomical Information,» in *International Conference IP, Computer Vision and Pattern Recognition* , 2017.

[34] Tang LYM, Borsch T, Liu XT, et al., «Corpus Callosum Segmentation in Brain MRIs via Robust Target-Localization and Joint Supervised Feature Extraction and Prediction,» in *Medical Image Computing and Computer-Assisted Intervetion-MICCAI*, Athens, Greece, 2016.

[35] Chollet F, Deep learning with Python, Manning, 2017.

[36] Mitchell TM, Machine Learning, McGraw-Hill Science/Engineering/Math, 1997.

[37] Rosenblatt, F. , «The perceptron: A probabilistic model for information storage and organization in the brain,» *Psychological Review,* vol. 65, n. 6, pp. 386-408, 1958.

[38] Nielsen MA, Neural Networks and Deep Learning, Determination Press, 2015.

[39] Lee F, «CS231n Convolutional Neural Networks for Visual Recognition,» 2017. [Online]. Available: http://cs231n.github.io/.

[40] Ng A, «Neural Networks and Deep Learning,» 2017. [Online]. Available: https://www.coursera.org/learn/neural-networks-deep-learning.

[41] Wang Z, Bovik AC, Sheikh HR, et al., «Image quality assessment: from error visibility to structural similarity,» *IEEE Transactions on Image Processing,* vol. 13, n. 4, pp. 600-612, 2004.

[42] Zaccone G, Karim MR, Menshawy A, Deep Learning with TensorFlow, Packt Publishing Ltd, 2017.

[43] Yang J, Yu K, Gong Y, et al., «Linear spatial pyramid matching using sparse coding for image classification,» in *IEEE Conference: Computer Vision and Pattern Recognition*, 2009.

[44] Gu J, Wang Z, Kuen J, et al., Recent advances in convolutional neural networks, arXiv preprint arXiv, 2015.

[45] Maas AL, Hannun AY, Ng AY, «Rectifier non-linearities improve neural network acoustic models,» *Proc. ICML,* vol. 30, 2013.

[46] Srivastava N, Hinton G, Krizhevsky A, et al., «Dropout: A Simple Way to Prevent Neural Networks from Overfitting,» *Journal of Machine Learning Research,* vol. 15, pp. 1929-1958, 2014.

[47] Gorgolewski KJ, Auer T, Calhoun VD, et al., «The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments,» Sci Data 3, 2016. [Online]. Available: http://bids.neuroimaging.io/.

[48] Cox RW, Ashburner J, Breman H, et al., «A (Sort of) New Image Data Format Standard: NIfTI-1,» in *OHBM*, 2004.

[49] Tuzikov AV, Colliot O, Bloch I, «Evaluation of the symmetry plane in 3D MR brain images,» *Pattern Recognition Letters,* vol. 24, pp. 2219-2233, 2003.

[50] Kingma DP, Ba J, «Adam: A Method for Stochastic Optimization,» *CoRR,* 2014.

[51] Sudre CH, Li W, Vercauteren T, et al., «Generalised Dice overlap as a deep learning loss function for highly unbalanced segmentations,» in *CoRR*, 2017.

[52] Abadi M, Agarwal A, Barham P, et al., «TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems,» 2015. [Online]. Available: https://www.tensorflow.org/.

[53] Chollet F, «Keras,» 2015. [Online]. Available: https://github.com/keras-team/keras.

[54] Shah M, Xiao Y, Subbanna N, et al., «Evaluating intensity normalization on MRIs of human brain with multiple sclerosis,» *Medical Image Analysis,* vol. 15, pp. 267-282, 2011.

[55] Palmer, S. L., Reddick, W. E., Glass O., et al., «Decline in corpus callosum volume among pediatric patients with medulloblastoma : Longitudinal MR study,» *American Journal of Neuroradiology,* vol. 23, n. 7, pp. 1088-1094, 2002.

[56] Figueira FF, Santos VS, Figueira GM, et al., «Corpus callosum index: a pratical method for long-term follow-up in multiple sclerosis,» *Arquivos de Neuro-Psiquiatria,* vol. 65, 2007.

[57] Hamarneh, G., Abugharbieh, R., and McInerney, T., «Medial Profiles for Modeling Deformation and Statistical Analysis of Shape and their Use in Medical Image Segmentation,» *International Journal of Shape Modeling,* vol. 10, pp. 187-210, 2004.

[58] Freitas P, Rittner L, Appenzeller R, et al., «Watershed-based segmentation of the midsagittal section of the corpus callosum in diffusion MRI,» in *24th SIBGRAPI Conference on Graphics, Patterns and Images*, Alagoas, Maceió, Brazil, 2011.