


UNIVERSITAT POLITÈCNICA DE CATALUNYA (UPC) - BarcelonaTech
UNIVERSITAT DE BARCELONA (UB)
UNIVERSITAT ROVIRA I VIRGILI (URV)

Master in Artificial Intelligence

Master of Science Thesis

AUTOMATIC METHODS FOR DETECTION OF MIDLINE BRAIN ABNORMALITIES

Aleix Solanes Font

FACULTAT D'INFORMÀTICA DE BARCELONA (FIB)
FACULTAT DE MATEMÀTIQUES (UB)
ESCOLA TÈCNICA SUPERIOR D'ENGINYERIA (URV)

Supervisor:

Laura Igual Muñoz

Department of analysis and applied Mathematics,
Universitat de Barcelona (UB)

Co-supervisor:

Joaquim Radua

FIDMAG Research Foundation
IoPPN (King's College London)
Karolinska Institutet (Stockholm)

January 20, 2017

ACKNOWLEDGEMENTS

I would like to thank my supervisors Laura Igual and Joaquim Radua for their continuous support in the elaboration of this work, for their patience, motivation and immense knowledge. Their guidance and advice helped me in all the time of development and writing of this thesis.

Besides my supervisors, I would like to thank the rest of my colleagues from FIDMAG, Rai, Erick, Anton… for their insightful comments and encouragement, but also for incentivizing me to widen my approach from various perspectives.

I would also sincerely thank to Carla, for all her love and support in life, and for her patience and empathy in good and bad moments.

Last but not the least, I would like to thank my family: my parents and to my brother for supporting me spiritually throughout the steps of my life to arrive to the elaboration of this thesis.

ABSTRACT

Different studies have demonstrated an augmented prevalence of different midline brain abnormalities in patients with both mood and psychotic disorders. One of these abnormalities is the cavum septum pellucidum (CSP), which occurs when the septum pellucidum fails to fuse.

The study of these abnormalities in Magnetic Resonance Imaging requires a tedious and time-consuming process of manually analyzing all the images in order to detect them. It is also problematic when the same abnormality is analyzed manually by different experts because different criteria can be applied. In this context, in would be useful to develop an automatic method for localizing the abnormality and give the measure of its depth.

In this project, several methods have been developed by using machine learning and deep learning techniques. To test the proposed approaches, data from of 861 subjects from FIDMAG, have been gathered and processed to use with our algorithms. Among the subjects, 639 are patients with mood or psychotic disorders and 223 are healthy controls. This same dataset was previously used in a study where the authors analyzed the prevalence of this abnormality across the subjects manually, the depth of the abnormality previously reported is finally compared with the results obtained from our automatic methods.

The first proposed method implemented is an unsupervised algorithm that segments the abnormality over 2D slices with an accuracy of the 89.72% and a sensitivity of an 84%.

Then, a comparison among different Machine Learning classifiers have been conducted along with different dimensionality reduction methods. Between them, K Nearest Neighbors produced the best overall accuracy, with a 99.1% of accuracy and a sensitivity of a 99% by using HOG descriptors to reduce the dimensionality of the data.

The last group of methods are different Deep Learning architectures. Three different implementations have been designed, and over the best performing architecture, two different input options to take into account contextual information have been proposed. The best Deep Learning method reached an accuracy of 98,8% and a specificity and sensitivity of a 99%.

Finally, an analysis of the similarity with respect to the results with the manually handcrafted ground-truth have been conducted to validate the usability of the automatic method in further analysis of the abnormality.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

Nowadays, the maturation of neuroimaging has led to incredible quantities of digital information about the human brain. To interpret all this information and give a common approach to analyze all this data is still a challenge.

The growth of the data about the brain produced a sheer number of brain imaging research studies being performed to spy on the brain anatomy and function in task or rest. These studies are centered in examining how the brain is build and wired, and what happens in the brain in different situations like when there is a mental illness. Indeed, centers that study the brain continue to collect more study data in a few days than was collected in over an entire year just a decade ago. All this data can bring important information about how our brain works, but we are still in a phase where we try to understand which questions should we formulate to this data, and we are in the first stages on how to proceed with all that information.

Another field of research that has grown in the latest years have been the analysis of large-data, known as "Big Data". This research field is being analyzed through Machine Learning and Deep Learning techniques, and the growth of this new methodologies are being merged with Neuroimaging research to improve prediction and understanding of the huge amounts of data from the brain generated.

In this project, both research fields will be mixed and analyzed in a concrete case study.

This project has been possible thanks to a collaboration between the *University of Barcelona* (UB) and *Fundació per a la Investigació i la Docència Maria Angustias Giménez* (FIDMAG) *Research Foundation*, the clinical partner.

1.1 CLINICAL MOTIVATION

In different studies, it has been demonstrated that patients have an increased prevalence of abnormalities in midline brain structures [1][2][3][4][5][6][7]. A study conducted in FIDMAG [1][2], one of the collaborators to this project, demonstrated an increased prevalence of the cavum septum pellucidum (CSP) in patients with psychotic and mood disorders. The importance of this finding can help to diagnose and study these disorders in terms of neurodevelopmental etiology, in other words, the study of this abnormalities could help to investigate the causation or origination of these pathologies, like schizophrenia or bipolar disorders.

When conducting studies about abnormalities in the brain, one of the most time-consuming steps is to analyze manually all images to detect these abnormalities with the risk that comports the fact that in different studies, different criteria could be used when considering if the abnormality is present or not. The use of automatic techniques should make easier to conduct these studies and to have a more standard and robust method in the detection of these anatomical characteristics in the brain.

In this project, the same data from [1][2] have been used to compare the final performance of the methods proposed.

1.2 OBJECTIVES

In this work, we deal with the detection of abnormalities in the CSP using Magnetic Resonance Images. The aims of this work are to provide an accurate approach to characterize the depth of this area of the brain using computer vision and machine learning state-of-the-art methods; moreover, after computing such information, to check whether this region of interest of the brain is correlated to some pathology.

The objectives could be divided in the following parts:

- Obtain the data and preprocess it
- Develop different methods to detect abnormality in the cavum
    1. A heuristic method making use of anatomical information and tuning the algorithm to find CSP
    2. Test the detection using different common machine learning techniques
    3. Make use of the latest Deep Learning techniques to reach better results
- Evaluation and comparison of the different used methods for cavum detection
- Analyze statistically the results of the cavum depth to find correlations with respect to the original ground-truth in the paper using the best detection result.

1.3 STRUCTURE OF THE REPORT

We start by introducing what neuroimaging is, how MRI data is obtained and the different applications of Magnetic Resonance Imaging (MRI) in research and clinical fields.

In section 3, we will present the context about why we have decided to characterize this zone, where it is found, and previous work about cavum.

After that, in section 4, we will explain the origin of data and its corresponding preprocessing.

In section 5, we present the different methods, a heuristic method to characterize the abnormality. This method is considered as a baseline. In this section we will discuss also about different machine learning and deep learning approaches.

Finally, in Section 6, we will present all results and statistically analyze the correspondence of the results found automatically with the ground-truth found by the researchers from [1][2].

2 MAGNETIC RESONANCE IMAGING (MRI)

Magnetic Resonance Imaging (MRI) is an imaging method that makes use of strong magnetic fields and radio waves to produce detailed images of the inside of the body. It is mainly used in medical analysis in order to help diagnose conditions, plan treatments and assess how effective previous treatment has been. An MRI scan (see Figure 1 for an example of a MRI scanner) can be used to examine almost any part of the body, including the:

- Brain and spinal cord
- Bones and joints
- Breasts
- Heart and blood vessels
- Internal organs, such as the liver, womb or prostate gland

In this project, MRI will be used to analyze brains of different patients.

FIGURE 1 MRI SCANNER (SOURCE: CONEIXELTEUCERVELL.COM)

2.1 FUNDAMENTALS

Human body is mostly composed of water molecules, which consist of hydrogen and oxygen atoms. Located at the center of each hydrogen atom there is a smaller particle called proton. These protons are highly sensitive to magnetic fields.

Due to the fact that under the effect of such powerful scanner magnets, the protons of the body line up in the same direction just like the needle of a compass by the effect of the earth magnetism.

After that, short bursts of radio waves are sent to different areas of the body, so that the protons are knocked out of alignment. In a few seconds, when the radio waves are turned off, those protons recover the original align, and this process sends out radio signals, which are picked up by receivers [8].

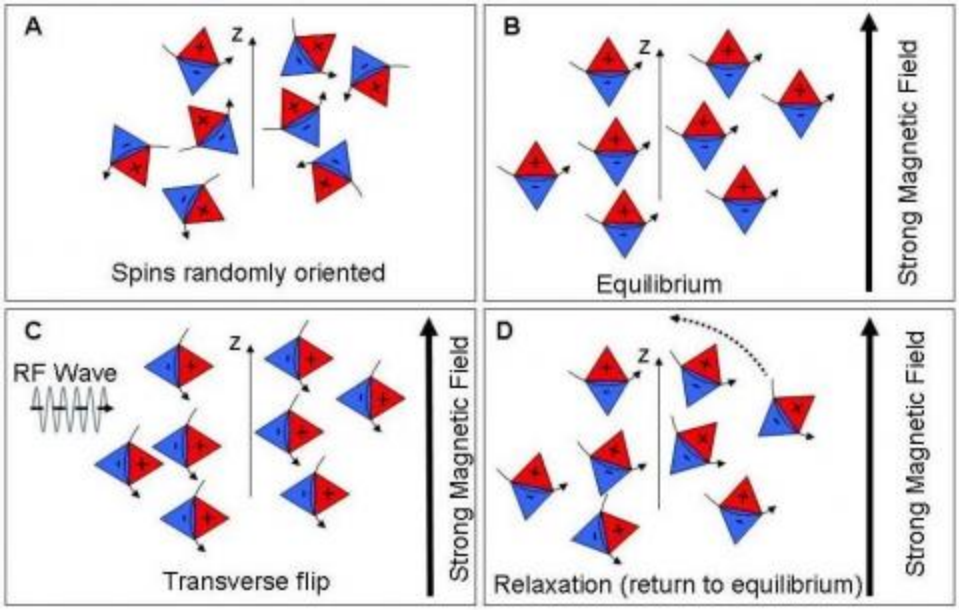

FIGURE 2 EXAMPLE OF ORIENTATION OF ELECTRON SPINS IN CONTACT WITH A MAGNETIC FIELD (SOURCE: HTTP://BLOGS.DICKINSON.EDU/)

All these signals provide information about the exact location of the different protons in the body. The time that a proton lasts to realign and some differences in signals help also to distinguish between the different types of tissue in the body. This is due to the fact that different tissues realign at different speeds. (See Figure 2 for an illustration)

All these properties produce high quality information. There are different processes for analyzing the signal received and convert it to the most appropriate image to the different tasks.

In MRI, image contrast is a function of tissue density, so, different tissues will be in different ranges of density values. In this case, that the source of signal are the protons, the type of density that matters is the hydrogen proton density.

RELAXATION TIME

An important part that contributes to the contrast of the image, in MRI, is the information related to the relaxation of every proton.

The concept 'relaxation' makes reference to the way that protons restore the equilibrium state or the way the protons go back to a low-energy level after excitation.

There are two kind of independent relaxation processes:

- T1 relaxation: reported usually in milliseconds, corresponds to the time when 63% of the longitudinal magnetization has recovered.
- T2 relaxation: reported usually also in milliseconds, corresponds to the time when 63% of the transverse magnetization has decayed.

Repetition Time (TR) is the amount of time between successive pulse sequences applied to the same slice. Time to Echo (TE) is the time between the delivery of the Radio-Frequency (RF) pulse and the receipt of the echo signal. (See Table 1 and Figure 3 Effects of TR and TE on MR signal (Source: Wikipedia.org)Figure 3 for details on TR and TE).

	TR (msec)	TE (msec)
T1-Weighted	500	14
T2-Weighted	4000	90

TABLE 1. MOST COMON MRI SEQUENCES AND THEIR APPROXIMATE TR AND TE TIMES

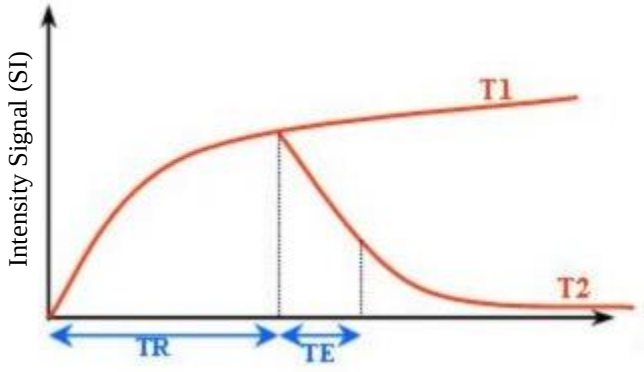

FIGURE 3 EFFECTS OF TR AND TE ON MR SIGNAL (SOURCE: WIKIPEDIA.ORG)

In Figure 4 an example of a T1 image and a T2 image are presented as an example. The left image corresponds to a T1-weighted MRI brain image, and as it can be seen, one of the characteristics of the T1 images is that tissues with high fat content, like for example white matter, appears brighter than compartments filled with water, like cerebrospinal fluid (CSF). This is useful for demonstrating anatomy characteristics of the brain. T2 images, in the right, show CSF bright, and tissues with high fat content, like white matter, darker. This property of the T2 images, are useful for example for demonstrating some pathologies which are associated with an increase of water content.

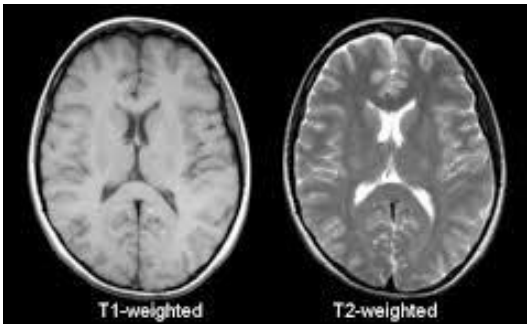

FIGURE 4 EXAMPLES OF T1 AND T2 IMAGES (SOURCE: CASEMED.CASE.EDU)

2.2 IMPORTANCE OF MRI IN RESEARCH AND CLINICAL ENVIRONMENTS

MRI imaging provides a unique view into the interior of the human body and has become an essential tool of modern medical imaging and disease diagnosis. In many cases, MRI provides important diagnostic information that cannot be obtained with other imaging techniques. It is a well-suited technique to analyze soft tissues of the brain.

MRI is particularly useful for:

- Examining the brain, neck and spinal cord
- Identifying bone and joint damage
- Revealing brain abnormalities in people with Alzheimer's Disease and dementia
- Helping with the diagnosis of central nervous system disorders, like multiple sclerosis and strokes
- Detecting breast cancer and damage to soft tissues
- Evaluating blood vessels to detect areas of blockage

Another commonly used technique is computed tomography (CT). This alternative is more extended, but they are used for different tasks. See Table 2. MRI vs CT for a summarized comparison.

MRI	CT
Better for soft tissue	Better for bony tissue
Uses magnetic energy and radio frequency	Uses ionizing radiation (Risk for patients)
More expensive machines (~1 Million $)	Cheaper machines (~100K-300K $)
Slow acquisition (~30-60 minutes)	Fast acquisition (~5 minutes)
Motion-susceptible (due to time of acquisition)	Less motion-susceptible

TABLE 2. MRI VS CT

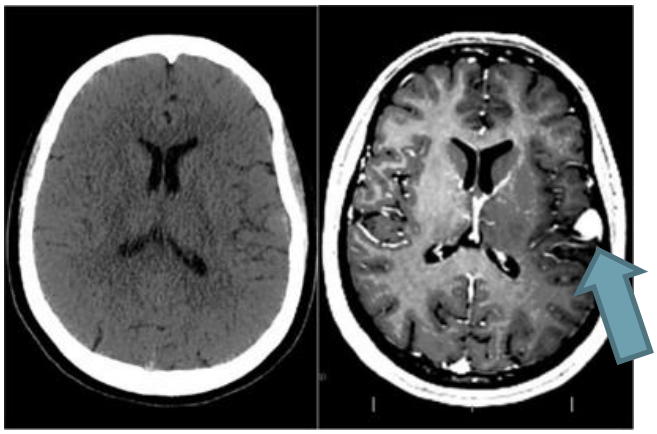

FIGURE 5 CT (LEFT) VS MRI(RIGHT) IN ABNORMALITY DETECTION (SOURCE: PROFESSIONAL.MEDTRONIC.COM/MRI)

In Figure 5, we can see how both imaging techniques perform in a case where an abnormality in the brain is present. So, in MRI the abnormality is more contrasted and can be identified easily.

In Figure 6 and Figure 7 we can see the prevalence of each imaging technique.

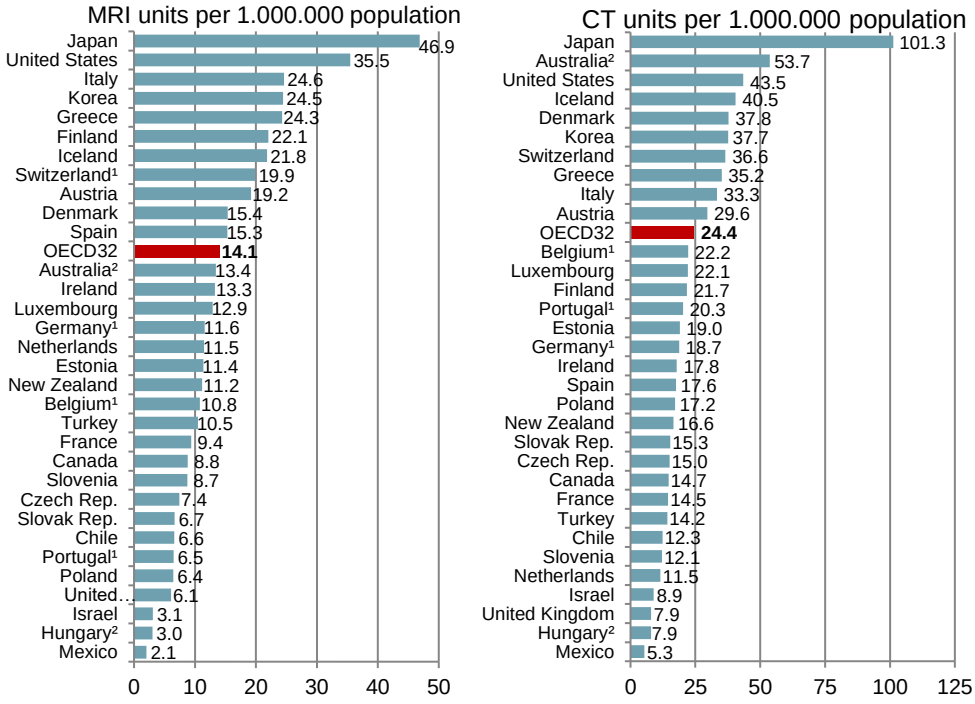

FIGURE 6 NUMBER OF MRI AND CT UNITS PER MILLION POPULATION (SOURCE: OECD OPEN-DATA)

The availability of MRI units and CT scanners has increased rapidly in most OECD countries over the past two decades, and so, the neuroimaging research has grown becoming one of the most innovative fields in neuroscience research.

Both techniques are widely used, but CT, due to the speed and cost of acquisition is more present and more used than MRI.

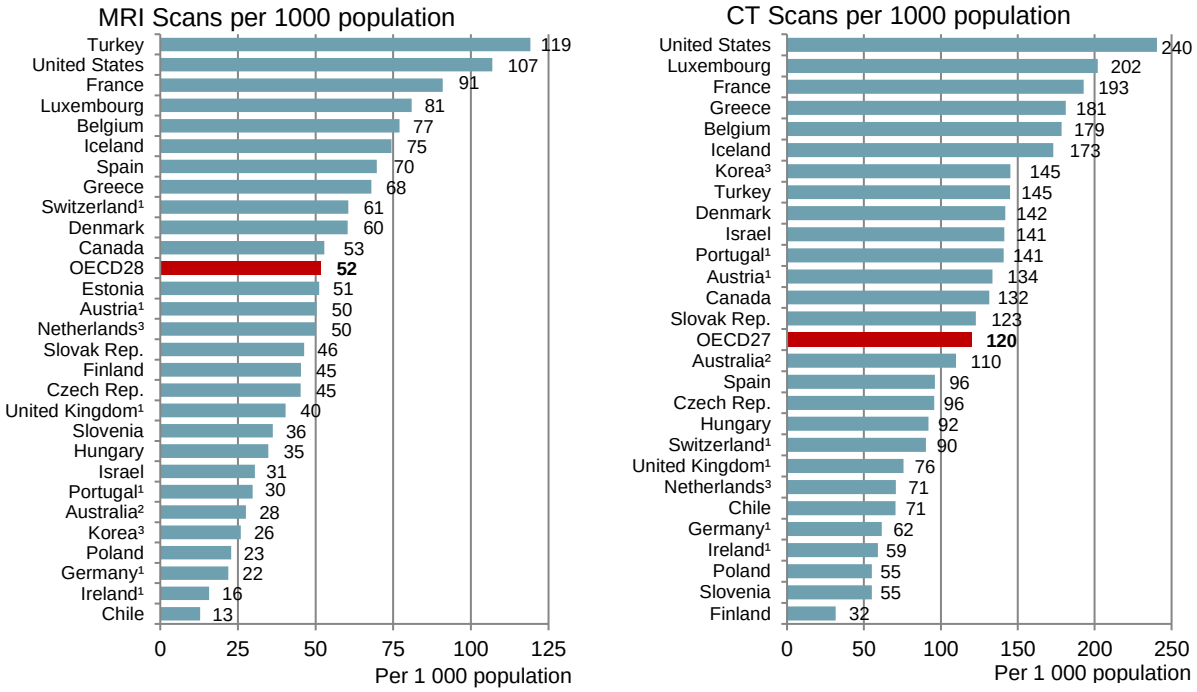

FIGURE 7 NUMBER OF MRI SCANS PER 1000 POPULATION (SOURCE: OECD ILIBRARY)

3 CLINICAL CONTEXT OF MIDLINE BRAIN ABNORMALITIES

In brain anatomy, there are three different anatomical planes, the sagittal, axial and coronal plane. This permits to show the brain from different angles. In Figure 8 all three planes are presented.

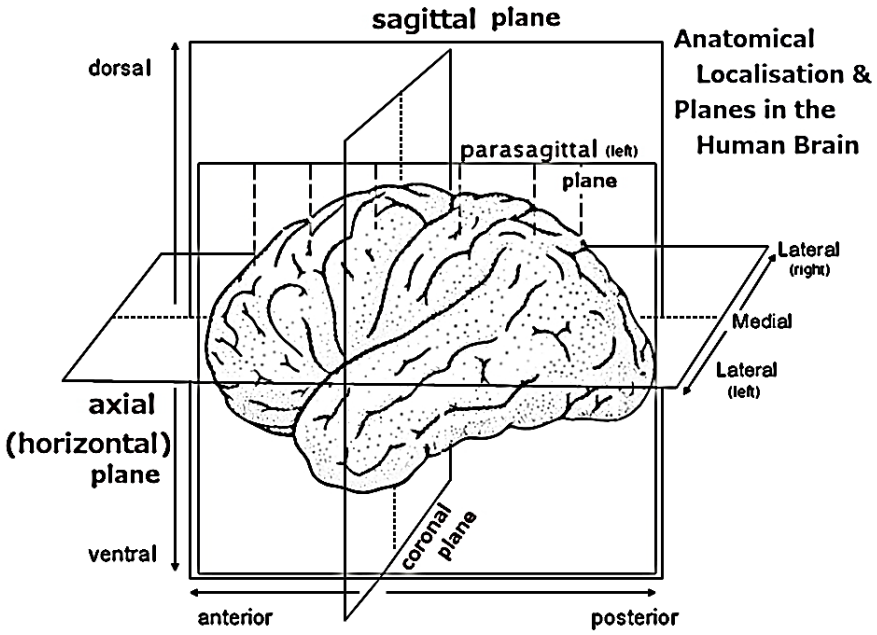

FIGURE 8 PLANES IN HUMAN BRAIN (SOURCE: COMMONS.WIKIMEDIA.ORG)

The coronal plane separates the brain vertically into the anterior and the posterior part. The sagittal plane splits the brain into left and right lateral sides. The latest plane, is the axial plane, which separates the brain between the top and bottom halves of the brain.

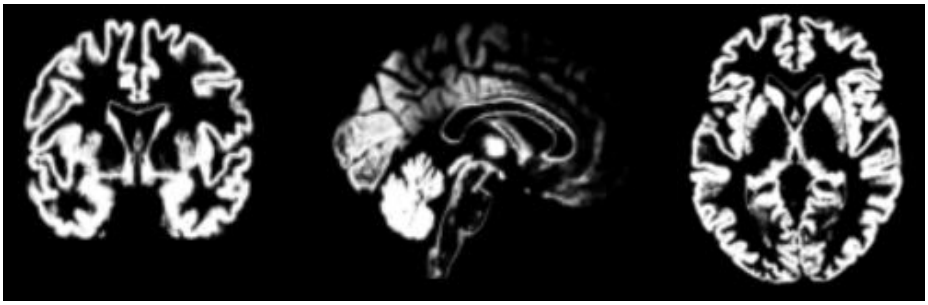

FIGURE 9 THE 3 PLANES OVER A GRAY MATTER IMAGE. LEFT: CORONAL, MIDDLE: SAGITTAL, RIGHT: AXIAL (GENERATED WITH MRICRON)

In MRI imaging, all the software to visualize images are able to show these three planes. In Figure 9 the three planes are presented in a GM segmented image.

This study is focused in the detection of midline brain abnormalities. Concretely, the important part to be analyzed is a normal variant cerebrospinal fluid (CSF) space formed between the leaflets of the septum pellucidum. This zone is called the cavum septum pellucidum (CSP) (see Figure 10). It is sometimes called the fifth ventricle, but this term is not currently in use because CSP has no direct communication with the ventricular system.The septum pellucidum is a component of the limbic system that forms the medial walls of the lateral ventricles. During fetal development both laminae tend to fuse in an anterior to posterior manner between the third and the sixth month of life.

When these laminae do not fuse completely, the space in-between both laminae is the CSP. See Figure 10 for an illustration.

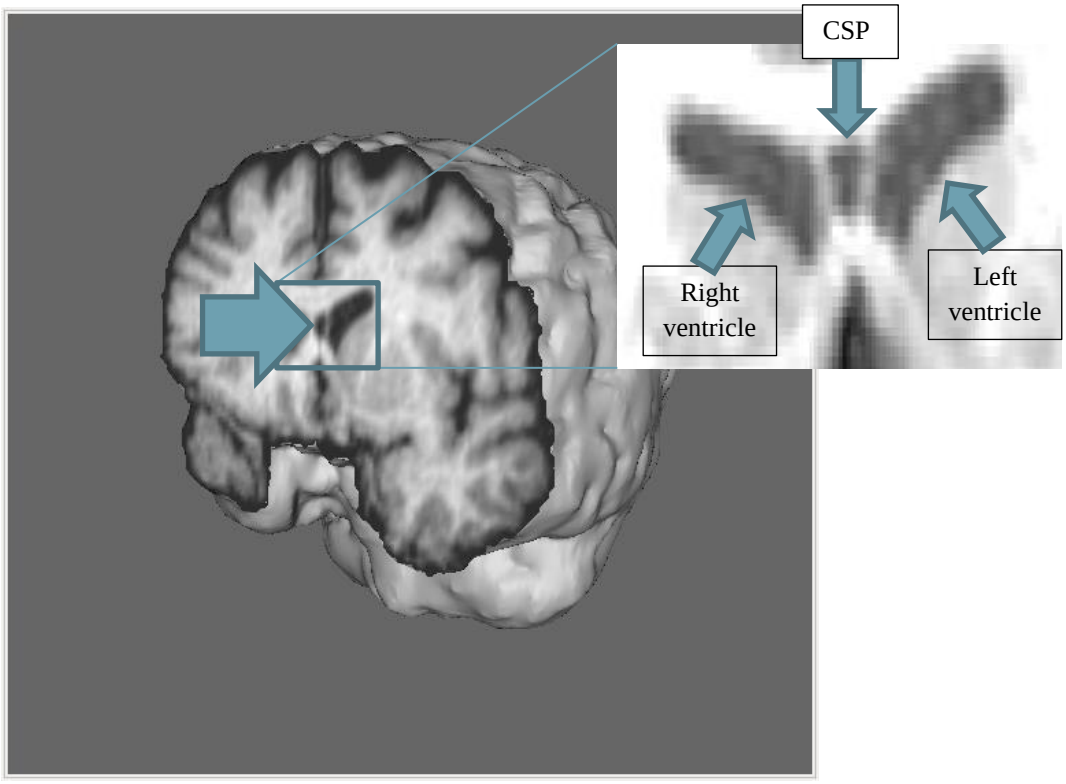

FIGURE 10 BRAIN SLICED TO SHOW THE CAVUM SEPTUM PELLUCIDUM SEEN FROM THE ANTERIOR PART (GENERATED ON FSLVIEW)

The presence of a cavum with a length from anterior to posterior part of the brain of around 1-1.4mm is considered normal anatomy of the brain, with an incidence of 60-80%.[9]

The study of this zone of the brain has special interest because previous studies have demonstrated an augmented prevalence of this abnormality in patients with schizophrenia as well as other mood and psychotic disorders.

The presence of the CSP in schizophrenia has been related to abnormalities in the corpus callosum and limbic structures, such as the thalamus and the hippocampus, during fetal development. During this period, risk factors for

schizophrenia appear and are important to the effect of this disorder, so any abnormality during these months should be subject of research [1][2].

The interest is to analyze the depth of the cavum and correlate it with different pathologies. The process of analyzing manually all these images is costly and slow, and so, this project is centered in automating this task.

4 DATA GATHERING AND PREPROCESSING

In this section, the origin and preprocessing of the data used in this project will be explained.

4.1 ORIGIN AND PREVIOUS WORK

Data for this study are T1-Weighted MRI images. All subjects for this study had been scanned with a 1.5 Tesla GE Signa scanner (General Electric Medical Systems) located at the Sant Joan de Déu Hospital in Barcelona.

The parameters used to obtain a high-resolution structural T1 MRI sequence, are the following:

- Number of axial slices = 180
- Slice thickness = 1mm
- Slice gap = 0mm
- Matrix size = 512x512
- Voxel resolution = 0.5 x 0.5 x 1 mm^3
- Echo time (TE) = 4ms
- Repetition time (TR) = 2000ms
- Flip angle = 15^o

The data used in this study consists in 861 subjects. From which:

- 639 patients with schizophrenia, delusional disorder, schizoaffective disorder, bipolar disorder, major depressive disorder, or a first episode of psychosis, mania or unipolar depression.
- 223 healthy controls

This dataset has been used by the article which is considered as the base for this thesis. In [1], authors demonstrated that patients with psychotic or mood disorders showed an increased prevalence of CSP along with another abnormality called the 'absence of adhesion interthalamica'. For the CSP, which is the concrete abnormality analyzed in this thesis, the author mentioned that they compared healthy controls versus patients with a logistic-regression-derived odds ratio, and they showed an odds ratio (OR) of 2.1 in cavum septum pellucidum being more present in patients with a $P<0.001$. To conduct the study, the authors mentioned that the same expert analyzed all the images blind to any information of the individuals to create the ground-truth with information about the depth of the CSP. Prior to this assessment, one sample of 50 subjects were analyzed by 2 of the researchers in the study in order to avoid discrepancies and reach consensus, after that, two further random subsamples of 100 individuals were conducted until complete inter-rater reliability was achieved.

4.1.1 MANUAL ANNOTATIONS

From the original study we only were able to get the length of the cavum septum pellucidum (CSP) per subject, but not the exact location of each cavum. Then, in order to complete an accurate training for this thesis, and due to the fact that, we created a previous analysis of the data in order to obtain a ground-truth (GT) with the exact location (the number of the slice where the cavum was present). This previous analysis consisted in manually checking as much images as possible to create a decent GT. Concretely, a total amount of 27,398 slices had been categorized between two categories (presence of cavum, no presence of cavum).

This new manually-created ground truth consisted in:

- 888 slices from 213 subjects which contained cavum
- 26510 slices from all subjects which do not contained cavum

As can be seen, the amount of slices containing cavum with respect to the ones that do not have the abnormality is extremely unbalanced; this problem along with others will be assessed in section 5.4.1 Open problems.

4.1.2 MAIN PROBLEMS WHEN DEALING WITH MRI DATA

MRI data is a specific type of imaging data, that has many drawbacks apart from the benefits. Some of them that must be considered before facing an MRI dataset will be explained in the following lines.

The most glaring issue when dealing with MRI data is that all data come from a non-standard intensity range when obtained from different scanners, from the same scanner but with different acquisition protocol, or with different magnet strength (typically 1.5, 3 or 7 Tesla). Even in the same hospital the same patient can have different intensity images.

No reliable shape or intensity priors for brain tumors or lesions exist. Brain pathologies can appear anywhere in the brain, they can have any shape with fuzzy borders, and come with a wide range of intensities.

MRI data comes with a non-negligible amount of white Rician noise that is introduced during the acquisition procedure.

Homogeneous tissues, like gray matter (GM) or white matter (WM), often suffer from spatial intensity variations along each dimension. This is called the MRI bias field effect. This effect, is a smooth low-frequency signal that affect the image intensities.

MRI images may have non-isotopic resolution, such that typically along coronal and sagittal views the resolution is lower.

4.1.3 MACHINE LEARNING CHALLENGES TO PROCESS MRI DATA

Research on machine learning has achieved spectacular results on some types of imaging data, like human or object recognition. The fact is that common photographs are easy, affordable and anyone can create them. By contrast, MRI imaging is extremely expensive. The scanners used to obtain the images cost around 1 M$, and the procedure of obtaining images can be time-consuming, around one hour per image, while in photographs it is a matter of seconds.

In many fields, public datasets are available. An example could be ImageNet, which contains millions of images correctly labeled with the category where it belongs to and for each class 1000 images; or CIFAR-10, which contains 60000 32by32 images from 10 categories (airplanes, automobile, bird, cat, deer, dog, frog, horse, ship and truck). When comparing these huge datasets with some MRI datasets, it is possible to see the difficulties in this imaging field.

Some common datasets in MRI could be BRATS, which contains around 300 subjects with handmade glioma segmentation ground-truth. Another big dataset is the ADHD200 which contains MRI images from around 900 ADHD subjects and controls (with the problem explained before, they come from 8 different scanners around the world) with their related pathologies and clinical information. This dataset is not well-suited for segmentation but for classification. Those two examples are outliers when comparing other MRI datasets, IBSR with 18 subjects, LPBA40 with 40 subjects, OASIS with 38 Alzheimer subjects… all contain few subjects and from many different scanners.

Because of the variability of the data, there is no standard preprocessing procedure, so that for every new scanner, a new procedure should be analyzed and processed.

When dealing with segmentation problems, the ground-truth labelling of the affected zone is heavily unbalanced with respect to the rest of the brain, and many machine learning algorithms assume that classes have a similar size.

4.2 Preprocessing Pipeline

In order to be able to have images ready for further analysis, it is important to follow some common tasks of preprocessing that will prepare the images. The most common preprocessing pipeline for structural images include, among others, the following steps that were reproduced for the preprocessing of the images used.

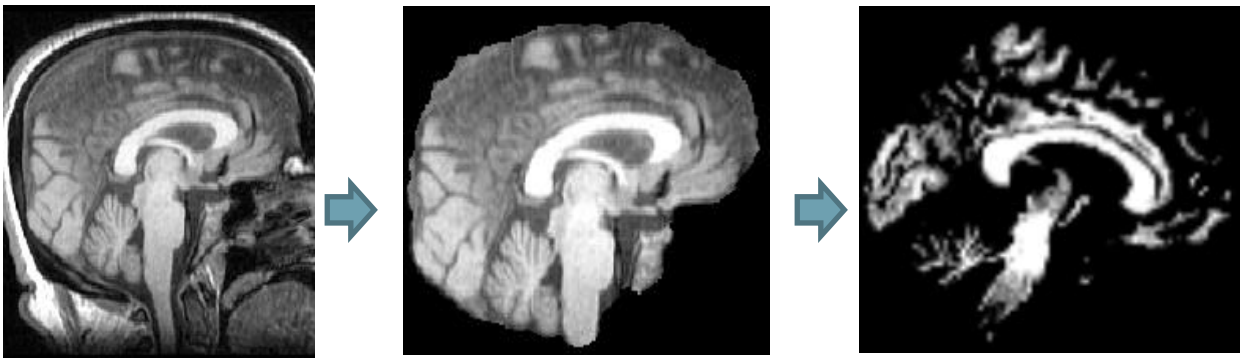

FIGURE 11 PIPELINE OF PREPROCESSING. FROM LEFT TO RIGHT: ORIGINAL IMAGE, SKULL-STRIPPED IMAGE AND WHITE-MATTER (GENERATED WITH MRICRON)

In Figure 11, we can see the three main stages (notice that other intermediate stages exist, but are related to space registration). In the left, we can see the original image, after BET[1] preprocessing, we extract the brain from the image, removing the skull, and after that we segment the image. In the right, we can see the segmented white matter image.

4.2.1 SKULL-STRIPPING

MRI images include some parts that are not of our interest when analyzing the brain. So, it is a common practice to remove the part corresponding to the skull, and leaving only the brain.

The software used to conduct this task was one of the most used methods, FSL-BET (Brain Extraction Tool), with a fractional intensity threshold of 0.4 (-f 0.4)

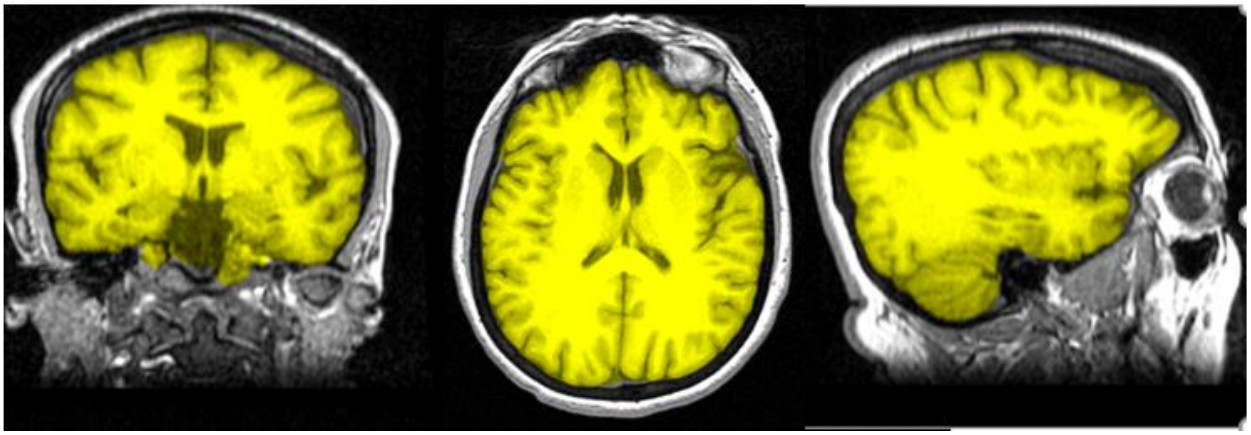

FIGURE 12. SKULL-STRIPPING. IN YELLOW THE STRIPPED BRAIN (GENERATED WITH FSLVIEW)

4.2.2 REGISTRATION

Once we had the images skull-stripped, we estimated the registration matrix to convert from the T1 image with coordinates in native space, to a common space.

There are different common registration spaces; in our case, we used one of the most used called MNI (Montreal Neurological Institute).

In voxel based morphometry analyses, usually the registration is conducted by passing the whole brain to a common space. However, but in this concrete case, assessing the registration by a common MNI registration process was not

[1] Brain Extraction Tool (BET - https://fsl.fmrib.ox.ac.uk/fsl/fslwiki/BET)[39]

recommended because when registering the whole brain to a common space, the cavum did not begin in the same place (the posterior genus was not always at the same voxel coordinate).

Our solution to standardize the beginning of this abnormality was to register the images to MNI but trying to correct the position error.

The first linear registration from T1 to MNI was done using FSL tool FLIRT (FMRIB's Linear Image Registration Tool) which is a fully automated robust and accurate tool for linear (affine) intra- and inter-modal brain image registration.

In order to register the image from native space white matter to MNI, a non-linear registration process was done using FSL FNIRT tool so that local deformations in the cavum zone can be preserved [10].

4.2.3 SEGMENTATION OF TISSUES

In order to find the exact zone of the first slice of the cavum in every single image, it is necessary to find the position of the posterior genus, which it is composed by white matter, and so, to be able to find this zone we should first segment the image in the different tissues (grey matter, white matter, and CSF). After the segmentation is done, we will have an image for each tissue, the one that it is more interesting for this study is the one corresponding to the white matter.

Once we have the white matter image, we can calculate the error of the position of the posterior part, and so, we can apply a transformation in order to obtain the best representation of the data for further analysis.

The images were segmented using FSL software FAST (FMRIB's Automated Segmentation Tool), which segments a 3D image of the brain into different tissue types (Grey Matter, White Matter, CSF, etc.), whilst also correcting for spatial intensity variations.

4.2.4 REGIONS OF INTEREST

To identify the presence of the CSP it is better to not consider the whole image, since this structure is very small compared to the brain size, so what we have done is to extract a customized Region Of Interest (ROI) in the zone where the CSP should be.
In order to obtain a ROI, and following the FSL-VBM recommended pipeline, a FSL tool called fslroi have been used to obtain the desired image [7]. An example of the result of a ROI can be seen in Figure 13.

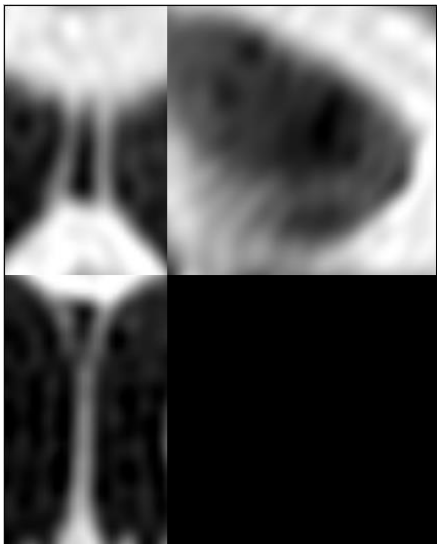

FIGURE 13. EXAMPLE OF THE CAVUM ROI SEEN FROM SAGGITAL, CORONAL AND AXIAL PLANES (GENERATED WITH MRICRON)

4.2.5 CENTERING AND SCALING

Standardization of a dataset is a common requirement for many machine learning estimators. Many estimators expect individual features to look like standard normally distributed data.

Centering and scaling the data is performed independently on each feature by computing the relevant statistics on the samples in the training set.

4.2.6 SLICING

We have opted to work on 2D ROIs of the CSP in coronal slices. These approach, produced 43,050 coronal slices. These slices have a size of 50 by 30 pixels in one single channel.

5 METHODS

5.1 PREVIOUS WORK

One open problem that will be discussed in the following sections, is that it is highly expensive to construct an enough large database in MRI in order to be able to apply Deep Learning approaches, and so, most of machine learning and deep learning previous work centered in MRI images makes use of other modality of images, like resting state MRI (rs-MRI) or functional MRI (fMRI). These modalities of imaging consider, for each subject, a fourth dimension, the time. So, with these imaging techniques, many analyses can be conducted, being with a task to do by the subject, and analyzing the brain activity during this time (fMRI), or without any task (resting) but checking the activity across a certain time (rs-MRI).

Other approaches that are quite used nowadays in Deep Learning and Machine Learning, which are closer to the objective of this thesis, are the approaches to detect brain abnormalities like tumors or lesions like for example in multiple sclerosis (MS).

A quick search at PubMed (PMC; https://www.ncbi.nlm.nih.gov/pubmed) showed that when searching the words "deep learning" and "MRI", most of the articles are based in fMRI and segmentation of brain regions.

As long ago as in year 1996, CNNs were applied to medical image segmentation in a work of Sahiner et al. [11]. In this work, several ROIs (regions of interest) were extracted that contained either biopsy-proven masses or normal tissues. The CNN that was used in this pre-GPU era was a simple CNN with two hidden layers, and authors reported training times "computationally intensive". [12]

Random forests have been quite popular between the methods used in MRI. Reza et al. [13] used a mixture of intensity and texture features to train a random forest for voxel-wise classification. But, the main problem they found with this approach is that the high-dimensional feature space.

In 2014, E. Sweeney et al. [14] performed a comparison of different algorithms in multi-modal MRI data. They used T1, T2 and FLAIR MRI voxel intensities, and used manual segmentations in order to train and validate the different supervised classification algorithms. They concluded that the differences in results were dependent of the feature vectors, rather than the machine learning or classification algorithms. Concretely, they concluded that features that incorporate information about neighboring voxels in the brain were found to improve the performance of the algorithm. They also concluded that for segmentation, simple and fast algorithms like logistic regression, linear discriminant analysis and quadratic discriminant analysis were better, and that more effort on developing methods for feature extraction should improve the performance.

In 2016, Brosh et al. [15] assessed the problem of multiple sclerosis brain lesion segmentation on MRI by the use of a 3D deep convolutional encoder network that combined interconnected convolutional and deconvolutional

pathways. The convolutional pathway learned higher level features and the deconvolutional pathway predicted the voxel level segmentation. They applied their network to two publicly available datasets and one clinical trial data set. In this article, the authors reported that their implementation performed comparably to the best state-of-the-art methods.

5.2 BASELINE METHOD

The first approach to analyze this problem is to produce a simple model to consider as a baseline and check if it is really necessary to opt for more complex approaches like CNNs or Machine Learning techniques.

The steps of the baseline method are the following: segment the image in order to obtain a binary image, and afterwards a flood-fill algorithm could be applied to detect the cavum in the image. In the following sections these first approach will be described.

5.2.1 DENOISING

One of the common steps in image preprocessing is to remove the noise. To achieve this objective, we have used a simple method based on non-local means filter for image denoising. The idea behind this algorithm is to simply replace the color of each pixel with an average of the colors of similar pixels in the analyzed image.[16]

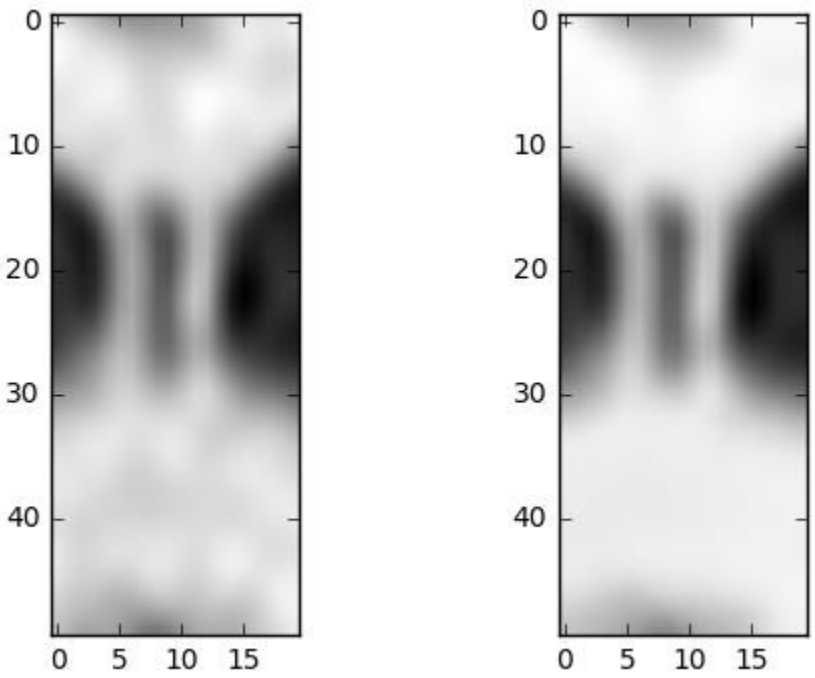

FIGURE 14. LEFT: ORIGINAL IMAGE. RIGHT: DENOISED IMAGE (GENERATED USING LIBRARY MATPLOTLIB)

5.2.2 K-MEANS FOR SEGMENTATION

In the preprocessing step, we obtained a segmented image of the different tissues, but we can not use directly this segmentation because it does not always include important parts for our analysis. In Figure 15, we can see that this segmentation did not return the walls or laminae around cavum.

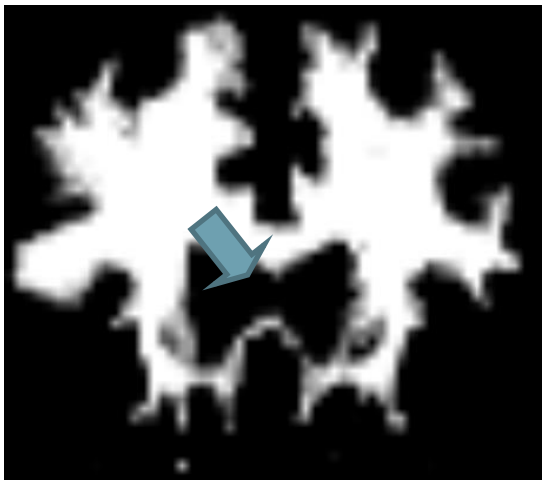

FIGURE 15 WHITE MATTER SEGMENTATION DID NOT DETECT THE LAMINAE SURROUNDING THE CAVUM (GENERATED USING MRICRON)

In order to produce a better segmentation, we will depart from the already preprocessed image that contains only the cavum ROI. The main problem in segmenting this kind of images is the noise, and also the differences of intensities depending on the image, so a simple threshold would not be the best option for a general method.

An approach considered to localize the cavum zone correctly binarized could be the use of clustering techniques, due to the fact that the different tissues that can delimit the zone are in its own range of intensity.

To test this first approach we use K-means clustering, which is based on Least Squares quantization [17]. This method will classify the different pixels of the image in different groups.

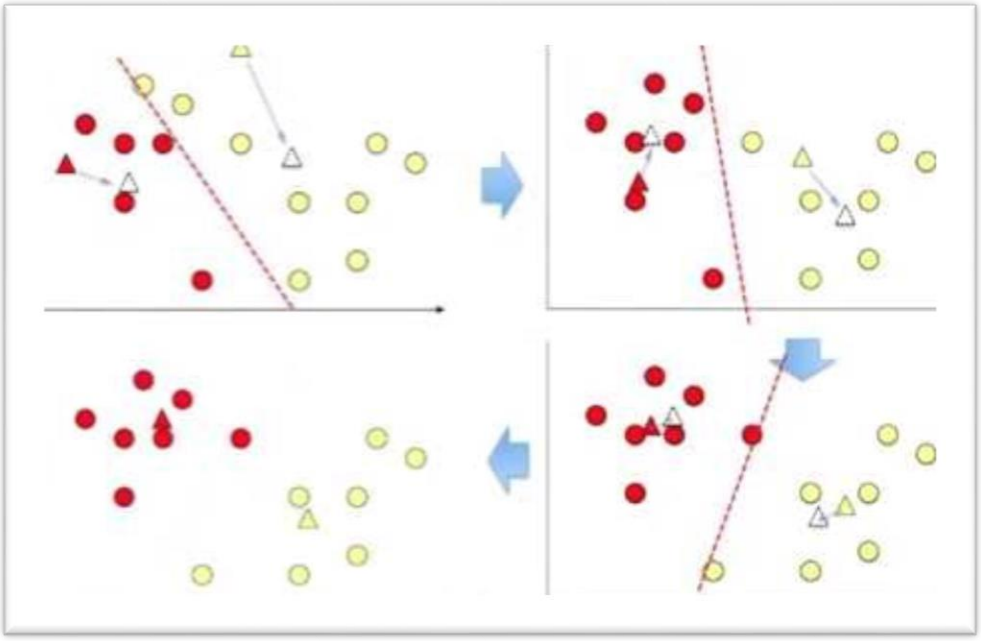

FIGURE 16 K-MEANS EXAMPLE (SOURCE: WWW.CS.US.ES)

In Figure 16 there is an example of the algorithm of K-means. The main steps of the algorithm are shown in Algorithm 1.

1. *Initialize k initial means (in this case k=2, marked as triangles) randomly within the data domain.*
2. *K clusters are created by associating every observation with the nearest mean (in the previous figures, clusters are represented in different colors).*
3. *The centroid of each of the k clusters becomes the new mean.*
4. *Repeat steps 2 and 3 until convergence*

ALGORITHM 1 K-MEANS ALGORITHM

The distances in K-means are calculated by using simple Euclidean distances.

To adapt this algorithm to MRI, we flattened the 2D images into a 1D array, and clustered data with respect to the intensity of the pixels, so that pixels with similar intensities will be in the same cluster.

Different numbers of clusters have been tested to check which one produced the best segmentation of the CSF, which in fact is the objective of this clustering. In Figure 17 and Figure 18 the results of these tests are shown.

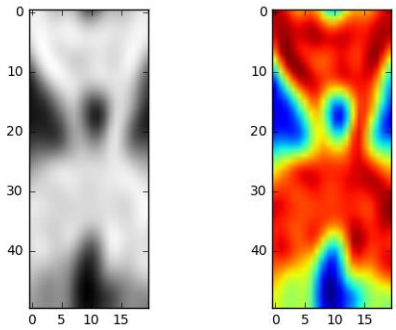

FIGURE 17 ORIGINAL IMAGE (LEFT), INTENSITY IMAGE (RIGHT) (GENERATED WITH MATPLOTLIB)

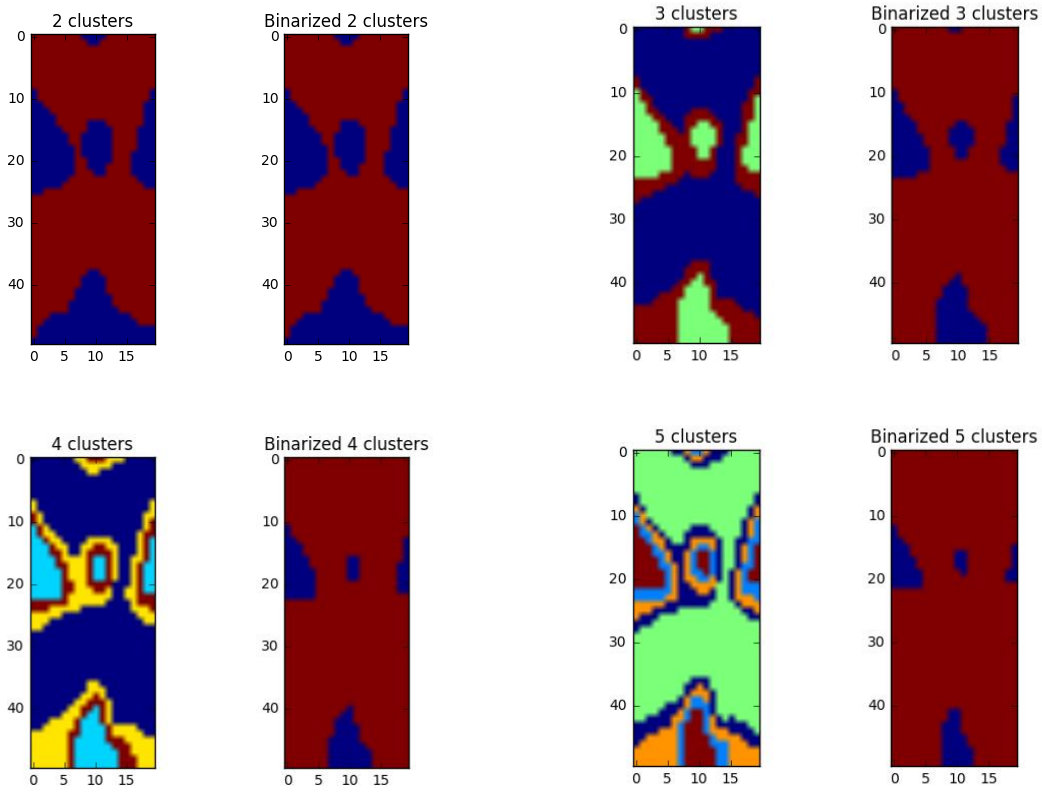

FIGURE 18 K-MEANS RESULTS USING DIFFERENT NUMBER OF CLUSTERS AND ITS BINARIZATION (GENERATED WITH MATPLOTLIB)

In Figure 18, we can see differences depending on the number of clusters (each sample corresponds to a color) together with the corresponding binarized image (in blue CSF, and in red the GM tissue). We can see that in this example even 2 clusters would be enough, and will preserve more information, because in some cases two clusters will remove

important information (Figure 19), but three clusters will be more conservative and preserve more information about the walls.

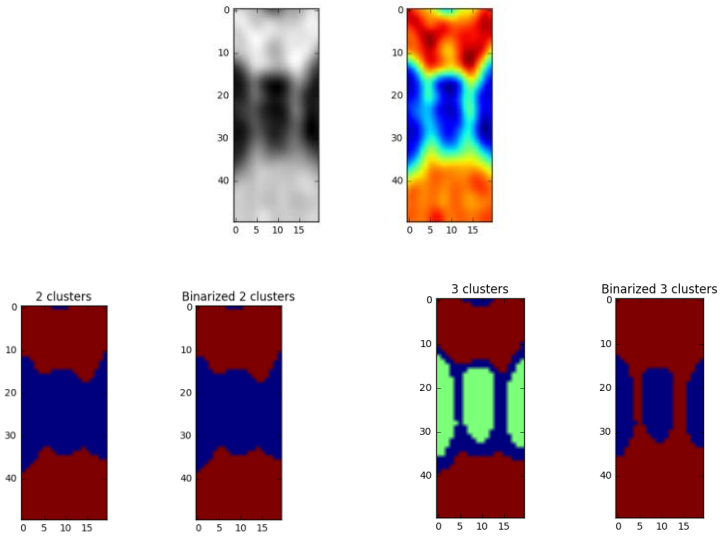

FIGURE 19 EXAMPLE OF WRONG SEGMENTATION (2 VS 3 CLUSTERS). TOP ROW: IN THE LEFT THE ORIGINAL IMAGE IN GRAYSCALE, AND IN THE RIGHT THE INTENSITY BASED IMAGE. BOTTOM ROW: RESULTS OF THE CLUSTERING USING 2 AND 3 CLUSTERS. (GENERATED WITH MATPLOTLIB)

5.2.3 FLOOD-FILL ALGORITHM

The flood-fill algorithm, which is also called seed fill or boundary fill (when applied on an image), is an algorithm that tries to determine the area connected to a given zone. This is done by looking for all nodes in the array that are connected to the start node by a path of the target color and then, it changes them to the replacement color. So, when detecting the cavum, the main idea is that this algorithm will search for a closed hole in the binary image, calculated using K-means segmentation, and fill it. (See Algorithm 2 for details)

1.Flood-fill (node, target-color, replacement-color):
2. If target-color is equal to replacement-color, return.
3. If color of node is not equal to target-color, return.
4. Set Q to the empty queue.
5. Add node to Q.
6. For each element N of Q:
7. Set w and e equal to N.
8. Move w to the west until the color of the node to the west of w no longer matches target-color.
9. Move e to the east until the color of the node to the east of e no longer matches target-color.
10. For each node n between w and e:
11. Set the color of n to replacement-color.
12. If the color of the node to the north of n is target -color, add that node to Q.
13. If the color of the node to the south of n is target-color, add that node to Q.
14. Continue looping until Q Is exhausted

ALGORITHM 2. FLOOD-FILL ALGORITHM

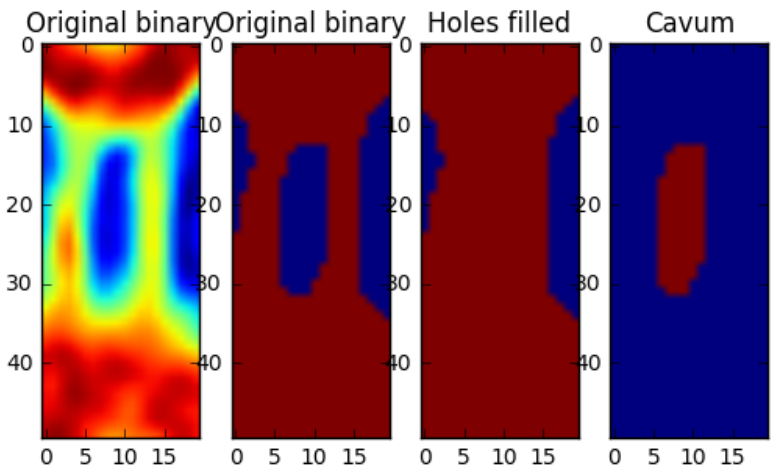

FIGURE 20 SEGMENTATION RESULT

The flood-fill algorithm will fill 'holes', so to obtain the segmented cavum a simple subtraction from the hole-filled image and the binary image is enough [18].

In Figure 20, there are four images corresponding to an original intensity image (left), the k-means binarized image (middle left), the result after flood-fill algorithm (middle right), and finally the segmented cavum (right).

In this approach, to consider an image containing CSP, an area superior to 0 pixels should be found.

5.2.4 Advantages and Drawbacks

This algorithm has the important advantage that it is totally unsupervised and it returns the area of the cavum in a simple manner.

But as it has been categorized, it is only a baseline method, and so, it has some pitfalls; not every image would be correctly classified if the hole is not as clear as the examples in Figure 20, or if the wall of the cavum is so slim that the intensities detected would be clustered to CSF group in K-means classification. (See Figure 21 and Figure 22)

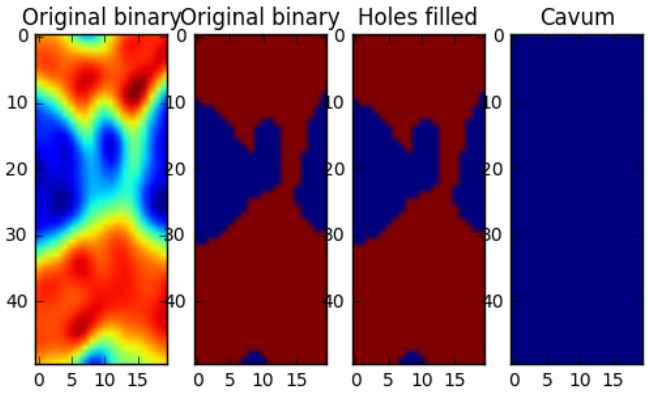

FIGURE 21 EXAMPLE OF WRONG SEGMENTATION (NOT CLOSED HOLE)

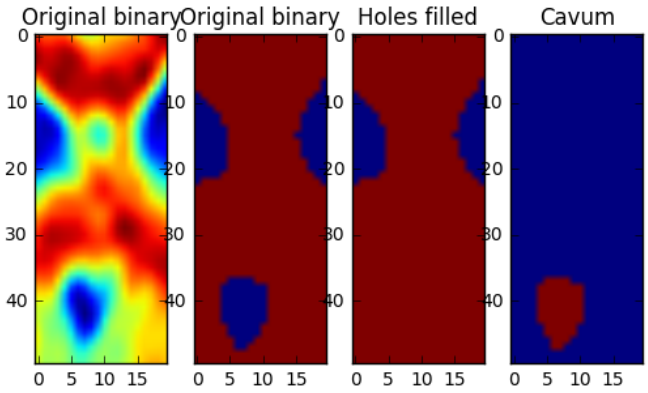

FIGURE 22 EXAMPLE OF WRONG SEGMENTATION (DETECTED ANOTHER HOLE)

5.3 MACHINE LEARNING APPROACHES

To try to improve the results from the baseline method different machine learning methods are considered.

In the following section, the different methods considered are explained and analyzed along with the different considerations taken into account.

We consider now the problem of classifying images as having CSP or not.

5.3.1 DIMENSIONALITY REDUCTION

When considering image analysis, in machine learning, one of the important steps is to define the features that will be used to characterize or classify the data. There are two main ways of proceeding, one is to check for the best feature from the ones available, which is called feature selection, and the other one, is to extract features from the data, called feature extraction.

In the following section, it will be shown the procedure followed to compute feature selection and feature extraction.

5.3.1.1 FEATURE SELECTION

A first approach in machine learning had been to try to characterize the most important features (pixels) of the cavum data images.

Ensemble Method: Extremely Randomized Trees

Extremely randomized trees (extra-trees) is a class of ensemble method, which in general are methods that learn a target function from a set of classifiers, and their individual predictions are combined to classify new examples. Ensembles, generally, improve the generalization performance and robustness over a single estimator.

There are two families of ensemble methods:

- Averaging methods: build several estimators independently and then average their predictions. On average, the combined estimator is usually better than any of the single base estimator because its variance is reduced.
- Boosting methods: base estimators are built sequentially and one tries to reduce the bias of the combined estimator. The motivation is to combine several weak models to produce a powerful ensemble.

Extremely Randomized Trees, does various sub-samples of the dataset and uses an averaging to improve the predictive accuracy and control the over-fitting. So, this algorithm could be considered as a part of the averaging methods family [19].

At each test node, the best split is determined among K random splits, and each one is determined by a random selection of an input (without replacement) and a threshold (See Figure 23).

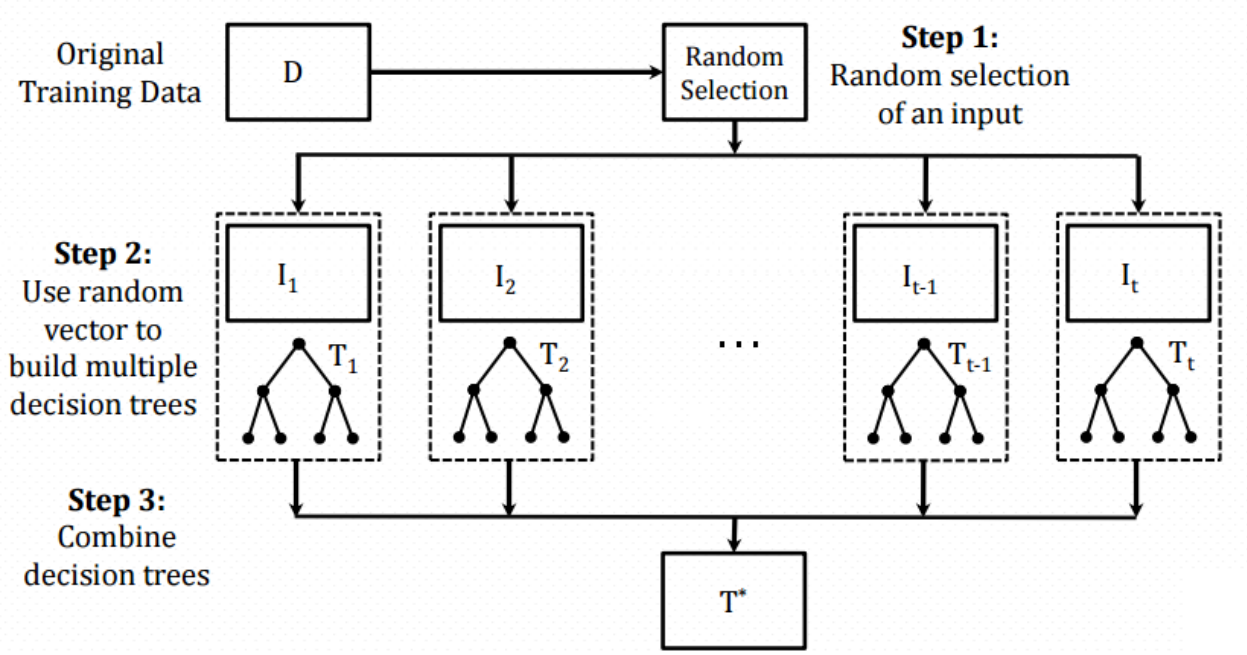

FIGURE 23. EXTREMELY RANDOMIZED TREES (SOURCE: WWW.ND.EDU)

The relative rank of a feature used as a decision node in a tree can be used to assess the relative importance of that feature with respect to the predictability of the target variable. The expected fraction of the samples they contribute to, can thus be used as an estimate of the relative importance of the features. After averaging those expected activity rates over several randomized trees, a reduction of the variance is performed, and this reduction can be used for feature selection.

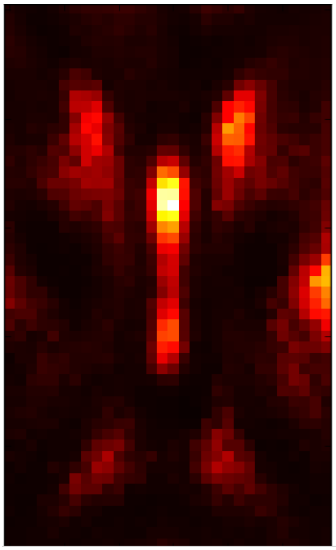

FIGURE 24 FEATURE IMPORTANCES USING RANDOMIZED DECISION TREES (GENERATED WITH MATPLOTLIB LIBRARY)

In Figure 24, we can see that the features learned make sense, the zone with highest importance (brighter) is the one that corresponds to the cavum. As can be seen this method can be useful for classifying images based on the importance of the different pixels.

5.3.1.2 Feature Extraction

5.3.1.2.1 Principal Component Analysis using Randomized SVD (Randomized PCA)

Principal Component Analysis (PCA) is among the most popular tools in machine learning, statistics, and data analysis in general. Computing a PCA of a data set amounts to constructing a singular value decomposition (SVD) that accurately approximates the matrix A containing the data being analyzed, and previously normalized by for example subtracting from each column its mean. A is a m by n matrix, then, the algorithm finds a positive integer k < min(m,n) and construct matrices U, E, and V such that

$$A \approx U * E * V^T$$

Being *U* a *m* x *k* matrix whose columns are orthonormal, *V* being an *n* x *k* matrix whose columns are also orthonormal, and E being a diagonal *k* x *k* matrix whose entries are all non-negative.

In order to reduce the dimensionality of the image,in this project we have used PCA with randomized SVD. The idea behind this variation of PCA, is to project the data to a lower-dimensional space while preserving most of the variance, by dropping the singular vector of components associated with lower singular values. [20]

In our case, images have a shape of 50x30 pixel gray-level values, the dimensionality of the data is 1500. Furthermore, we know that the intrinsic dimensionality of the data should be much lower than 1500 since the different images of cavums look somewhat alike. The use of PCA with randomized singular value decomposition should help in the process of reducing this dimensionality and preserving as much as possible variance for the data analyzed [21].

5.3.1.2.2 HISTOGRAM OF ORIENTED GRADIENTS (HOG)

The Histogram of Oriented Gradients is a feature descriptor that is commonly used for object recognition or detection. This method counts the occurrences of gradient orientations in localized patches of an image.

The idea behind this algorithm is that the appearance of local objects can be summarized by the distribution of intensity gradients or edge directions.

The image is first divided into different small connected regions called cells, and then, for every single cell, a histogram of gradient directions is computed. The concatenation of these histograms will be the descriptor of the image.

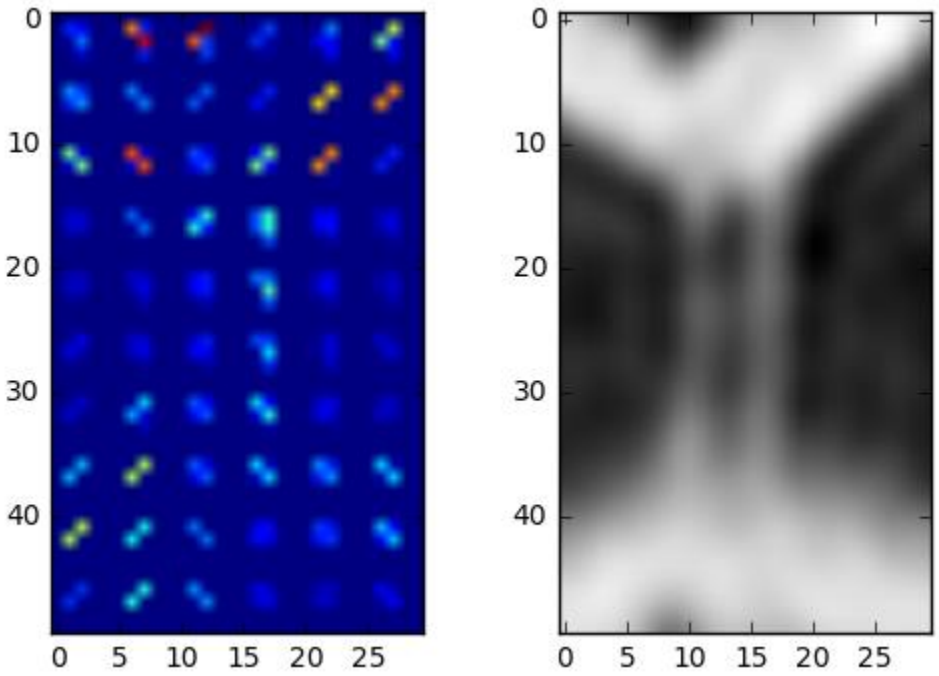

FIGURE 25. LEFT: HOG IMAGE. RIGHT: ORIGINAL IMAGE (GENERATED WITH MATPLOTLIB)

In order to improve the accuracy, the local histograms can be contrast-normalized by calculating a measure of intensity across a larger region of the image, called a block, and this value could be used to normalize all cells in the block. One of the advantages of this method, is that has better invariance to changes in illumination and shadowing. [22]

In many feature detectors, it is required to normalize color and gamma values previous to any calculation, however, this step can be omitted in HOG descriptor computation. The use of a normalization step achieves the same result. So, the only step prior to any classification when using HOG descriptors is the HOG computation itself.

HOG computing has the following steps:

- Gradient computation: Apply 1-D centered, point discrete derivative mask in one or both the horitzontal and vertical directions.

- Orientation binning: Create the cell histograms. Each pixel within the cell casts a weighted vote for an orientation-based histogram channel based on the values found in the gradient computation. These cells, can be rectangular or radial in shape. As stated by Dalal et al. [23], in image characterization, 9 orientation bins in 0^o-180^o give the best summarization of images when applying classification, and so, it is the configuration used to extract features from the images.
- Descriptor blocks: To take into account the changes in illumination and contrast, the different gradient strengths must be locally normalized, which requires grouping the cells together into larger, spatially connected blocks. The HOG descriptor is a concatenated vector of components from the normalized cell histograms from the different blocks of regions. Typically, those blocks overlay, participating more than once in the final descriptor.
- Block normalization: The final step in HOG computation, is the normalization of the blocks. In our case, as stated by [23] the normalization factor has been a L2-norm. Let ||v|| be the k-norm for k=1,2 and *e* be some small constant:

$$\text{L2-norm: } f = \frac{v}{\sqrt{||v||_2^2 + e^2}}$$

5.3.2 CLASSIFICATION

Once the features were selected or extracted, this information had been passed to a classifier. Concretely, different classifiers had been tested, whose results are compared in the section 6. Results. In this section, a few basics of the different algorithms will be explained.

K Nearest Neighbors (KNN)

K-Nearest Neighbors is a non-parametric instance-based supervised algorithm that can be used as a classifier or as a regression predictive problem. In our problem, it has been used as a classifier. The main idea is, given an input image I, search for the K training images that are more related to I, and then select the most frequent class among the selected similar images [24][22].

A commonly used distance metric for continuous variables is the Euclidean distance, which is the distance metric used in our implementation.

KNN is one of the simplest algorithms in machine learning, and has an important drawback, it is sensible to noise and the local structure of the data. So, other more complex ML algorithms has been tested, to compare which is best-suited for our concrete problem.

Linear SVC

Linear Support Vector Classifer (Linear SVC) is a non-probabilistic binary linear classifier based on supervised learning model.

A Support Vector Classifier, is a model that represents the examples as points in space, mapped so that the examples of separate categories are divided by a clear gap that is as wide as possible. When new data is introduced, once the model is trained, this classifier predicts to which category belongs based on which side of the gap they fall.

Applying what is called the kernel trick it is possible to create nonlinear classifiers. The kernel trick applied to maximum-margin hyperplanes, replaces every dot product by a nonlinear kernel function, and so, enables to operate in high-dimensional space without ever computing the coordinates of the data in that space, but rather by simply computing the inner products between the images of all pairs of data in the feature space. This allows the algorithm to fit the maximum-margin hyperplane in a transformed feature space [25].

Random Forests

Random Forest Classifier is the name that receives the ensemble learning method called Random Forests (or decision forests) when applied to classification.

Random Forest Classifier operates by constructing a multitude of decision trees at the stage of training and returning as a result the mode of the class on classification tasks (when used in regression returns the mean of the individual trees [26][27].

The current implementation of the algorithm uses the general technique of bootstrap aggregating, or bagging, to the tree learners. This extension to the original implementation of Random Forests, selects a random sample. with replacement of the training set and fits the trees to these samples. This bootstrapping procedure has the effect of reducing the variance of the model [28].

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is a generalization of Fisher's linear discriminant, which is a method widely used in statistics, pattern recognition and machine learning when dealing with the problem of finding linear combinations of features that characterizes or separates two or more classes of objects or events. This method can be used to reduce dimensionality or a classifier.

The classifier generates a linear decision boundary by fitting the class conditional densities to the data and using Bayes rule. Each class is fitted into a Gaussian density distribution, assuming that all classes share the same covariance matrix.

5.4 Deep Learning Approaches

In the recent years, Deep Learning has been a growing trend in general data analysis. It is an improvement from the well-known artificial neural networks (ANNs). The main concept that differentiates a neural network from a deep learning network is the number of hidden layers, and this is where the term "Deep" comes from.

An increased number of layers permits higher levels of abstraction and could also lead to improved predictions from data.

For computer vision tasks, Convolutional Neural Networks have proven to give outstanding results in classification and detection of objects and other common tasks with respect to classical approaches.

In this project, deep learning has been programmed over the high-level library Keras[2], and speed up with the use of a CUDA-ready GPU.

[2] http://keras.io – A Deep Learning library for Tensorflow (http://tensorflow.org).

5.4.1 OPEN PROBLEMS

When preparing the dataset to use it in a proper way to learn from it, many aspects has to be considered, some of them are concrete for medical imaging. The main open problems in preparing the data for deep learning are analyzed in the following section. In order to preprocess imaging all of them has been considered when preparing the data for the different algorithms used.

5.4.1.1 PREPROCESSING

The grayscale distribution of MR images is dependent on the acquisition protocol and the hardware. One of the principal key points in any statistical or ML problems, is to have the same data distribution from one subject to another. Therefore, a step to bring all subjects to similar distributions is important.

All inputs to the model should be in the same range so that any image has prior advantage over others in deciding the output of the model. Among other methods, the ones used in this study are the following:

- Zero-centering the data by subtracting the mean value of every row.
- Normalize the data by dividing by the max value.

5.4.1.2 BALANCING THE DATASET

A balanced dataset is when a class is overrepresented with respect to others. Unfortunately, in brain imaging, data are rarely balanced due to the small size of the lesion or abnormality compared to the rest of the brain. In our concrete case, the abnormality occupies less than a 1% of the whole brain. Training on an unbalanced dataset can produce very low positive rates since the system gets to be biased towards the one class that is over represented.

We have considered a reduced sample of the zone where this abnormality is located. Once the ROI is created, the new data is unbalanced, because, as stated in the original paper [1] that used the same dataset we focused on, the abnormality appears only in 314 of 639 patients and in 71 of 223 controls. Apart from that, in a single subject, the slices that contain the cavum can range from 0 to 50 (in 1mm resolution) being the most frequent case a 1 or 2 mm cavum (CSP). This situation brought the problem to the following sections, Data Augmentation and adding context information in Global Information.

5.4.1.3 DATA AUGMENTATION

When a class is under-represented, like the cavum class in this project, in order to apply deep learning methods, different processes should be followed so that the final number of samples of each class can be increased. One simple approach could be to apply a simple transformation to some of the images in one of the classes under-represented. In our implementation, the method followed to increase the Cavum class, has been to simply flip the image in both horizontal and vertical axes [29][30][31]. This procedure has been consensuated with clinical experts from FIDMAG.

Other methodologies that have been tested showed to not produce an improvement in this case is to apply warping or small rotations on the images.

The method used should be chosen with respect to how it is the structure of the data. For example, Pereira et al. [32] reported that by only rotating the data in angles multiple of 90º, because their data was strongly dependent from the center pixel of the patch, they increased the data by 4 times and achieved an increase of the effectivity of their methods.

5.4.1.4 GLOBAL INFORMATION

In medical image analysis, adding context information has been always a subject of interest. Anatomical regions in close-up view can appear similar and borders may be diffused in some parts due to lack of contrast of other artifacts, additional context is needed to localize a region of interest (ROI).

In this project, two approaches were made in other to incorporate contextual localization information to the data. The first, already explained, was to create a region of interest where the cavum is, and the second one, when using Convolutional Neural Networks in section 7.2, in the latter method, different scales have been considered. So, three different zoom scales centered at the center of the image, which generally the cavum should be, have been given as input of the model.

This approach let the model to learn simultaneously global and local contextual features.

In [33] H. Ali et al. proposed a system for multi-resolution MRI brain image segmentation based on morphological pyramid. They used a wavelet with different resolutions in order to bring spatial context between pixels. This extra information gave an increase on the prediction power of the system. We have applied this strategy, in our case also let us improve minimally the results.

5.4.2 Convolutional Neural Networks (CNN)

A Convolutional Neural Network (CNN or ConvNet) is a type of feed-forward artificial neural network in which the connectivity pattern between its neurons is neurobiologically motivated on locally sensitive and orientation-selective neurons of the visual cortex of a cat[34][35].

CNNs are similar to ordinary Neural Networks, both share the concept of neuron, that have learnable weights and biases. Each neuron receives some inputs, performs a dot product and optionally follows it with non-linearity. The whole network expresses a single differentiable score function: from the raw image pixels on one end to class scores at the other. CNNs have also a loss function on the last fully-connected layer. The main difference is that CNNs make the explicit assumption that the inputs are images, which allows us to encode certain properties into the architecture. These then make the forward function more efficient to implement and vastly reduce the amount of parameters in the network. See an example in Figure 26.

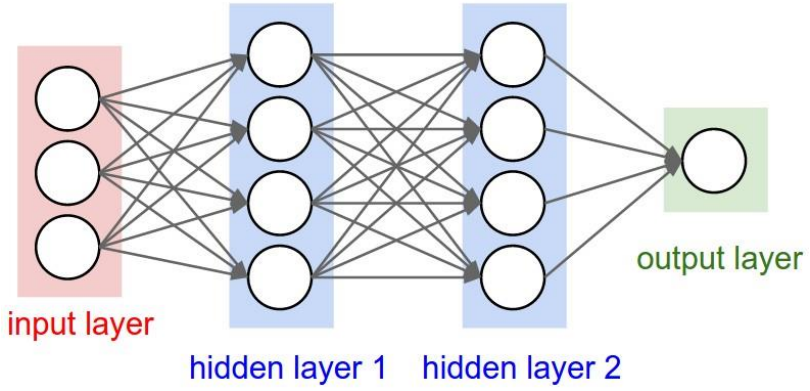

FIGURE 26. REGULAR 3-LAYER NEURAL NETWORK (SOURCE: CS231N.GITHUB.IO)

Regular Neural Network: Neural Networks receive an input (a single vector), and transform it through a series of hidden layers. Each hidden layer is made up of a set of neurons, where each neuron is fully connected to all neurons in the previous layer, and where neurons in a single layer function completely independently and do not share any

connections. The last fully-connected layer is called the “output layer” and in classification settings it represents the class scores.

Regular Neural Networks don’t scale well to full images, so Convolutional Neural Networks include neurons arranged in 3 dimensions: width, height and depth (depth could refer as an example to the number of channels of an image RGB). In CNNs, the neurons in a layer will only be connected to a small region of the layer before it, instead of all of the neurons in a fully-connected manner. Moreover, the final output layer would be a single vector of class scores, arranged along the depth dimension.

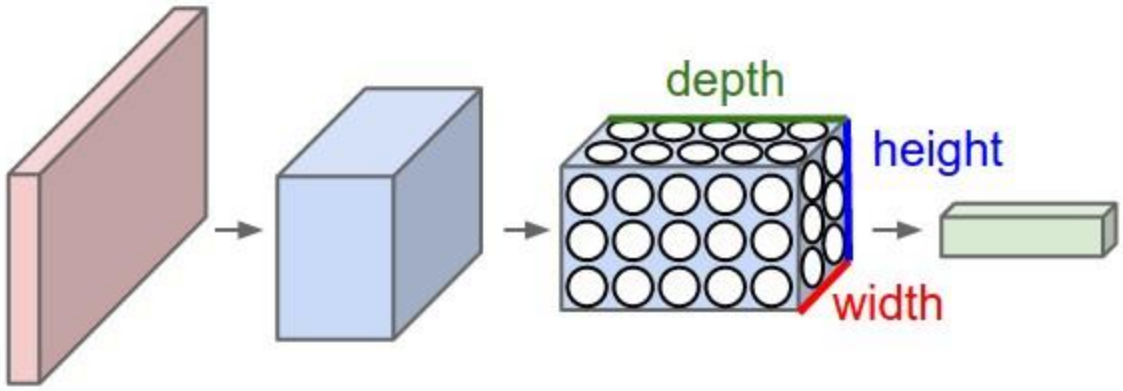

FIGURE 27. CNNS ARRANGE ITS NEURONS TO HAVE 3 DIMENSIONS, AS SEEN IN THE 3RD LAYER (SOURCE: CS231N.GITHUB.IO)

5.4.2.1 Layers in Convolutional Neural Networks

Different types of layers exist that are commonly used in CNNs. In this section the most popular ones, which are also used in our implementation, will be explained.

INPUT Layer

The INPUT layer will handle the input image of the network. It will hold the raw pixels of the data.

CONV Layer

CONV or Convolutional layer will compute the output of neurons that are connected to local regions in the input, each computing a dot product between their weights and a small region they are connected to in the input volume.

RELU Layer

ReLU or Rectified Linear Units layer will perform an element-wise activation function. For example, this layer could perform a thresholding to zero such as $f = \max(0, x)$. It increases the nonlinear properties of the decision function and of the overall network without affecting the receptive fields of the convolution layer.

POOL Layer

POOL or Pooling layers will perform a non-linear down-sampling operation along spatial dimensions. The most common is a max pooling. It partitions the input image into a set of non-overlapping rectangles and, for each sub-region, outputs the maximum. By this operation, it progressively reduces the spatial size of the representation to reduce the amount of parameters and computation in the network, and this will help to control overfitting.

FC Layer

Fully-Connected layers will compute the scores for each class. As the name suggests, each neuron in this layer will be connected to all the numbers from the previous layer.

Loss Layer

This layer specifies how the network training penalizes the deviation between the predicted and true labels. It is usually the last layer in the network. There are three kind of layers:

- Softmax: used to predict a single class of K mutually exclusive classes
- Sigmoid cross-entropy: used for predicting K independent probability values in [0,1]
- Euclidean: used for regressing to real-valued labels [-∞, ∞]

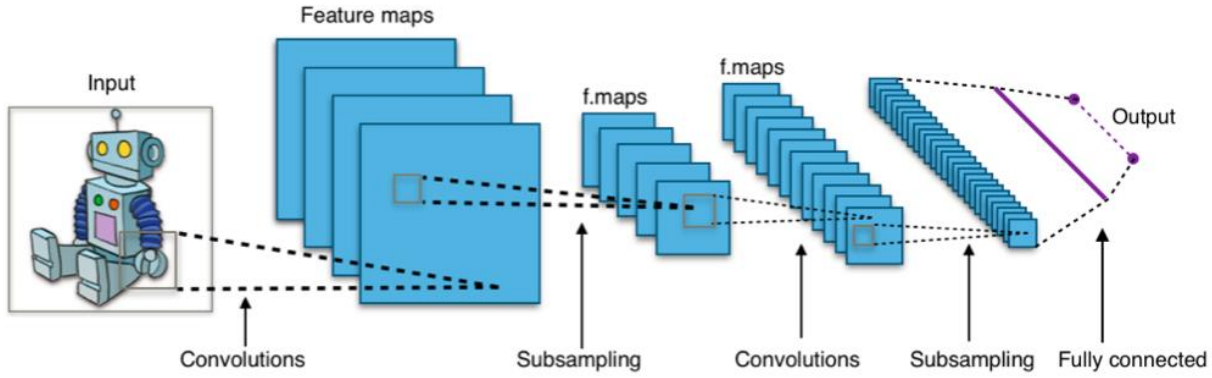

FIGURE 28. TYPICAL CNN ARCHITECTURE. (SOURCE: WIKIPEDIA.ORG)

In Figure 28 a common architecture of a CNN is shown. Between the input and the output there are different layers that perform different operations on data to give as a result an output.

5.4.3 REGULARIZATION

Fully-connected layers occupies most of the parameters, and so, it is prone to overfitting. To avoid this problem, a method called "dropout" is introduced.

5.4.3.1 DROPOUT

At each training stage, every individual node is either removed (dropped out) of the net with probability $1 - p$ or kept with p probability. This is used to reduce the network. The incoming and outgoing edges to a dropped-out node are also removed. At this stage, only the reduced network is trained on the data. Those nodes that were previously removed, are then reinserted into the network with their original weights. So, the key idea is to randomly drop units along with their connections during training and dropout samples from an exponential number of different "thinned" networks (see Figure 29).

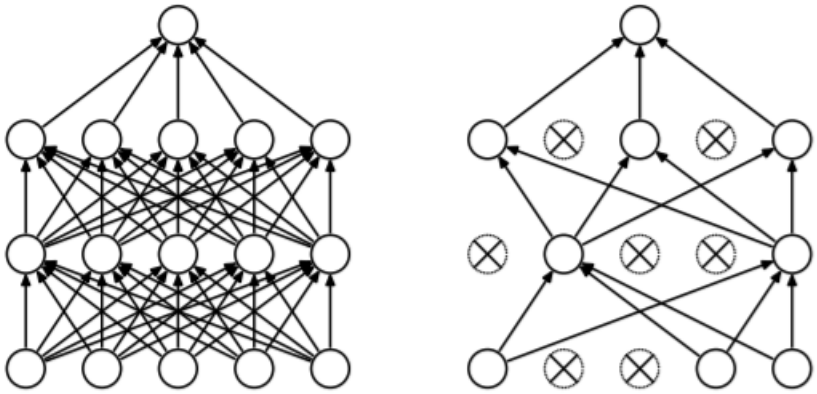

FIGURE 29. LEFT: STANDARD NEURAL NET WITH 2 HIDDEN LAYERS. RIGHT: AN EXAMPLE OF THINNED NET PRODUCED BY APPLYING DROPOUT, CROSSED UNITS HAVE BEEN DROPPED. (SOURCE: [36]**)**

By not training all nodes on all training data, dropout decreases overfitting in neural networks, and also decreases the time required for training as the complexity is reduced[36].

5.4.4 PROPOSED MODELS

Three different models of CNN have been designed based on different proposals to use Keras to classify the MNIST handwritten digit database from the playground documentation examples in http://keras.io. The different models are ordered from simplest to the most complex one, considering simple to have less layers than complex.

5.4.4.1 MODEL 1

The first model is the simpler in terms of number of hyperparameters, it has 1,234,658 parameters. The main idea is to stack Convolution layers and Activation layers which performs ReLu activations at the beginning of the network, and dropout layers and a softmax activation layer in the end. The structure is presented in Figure 30.

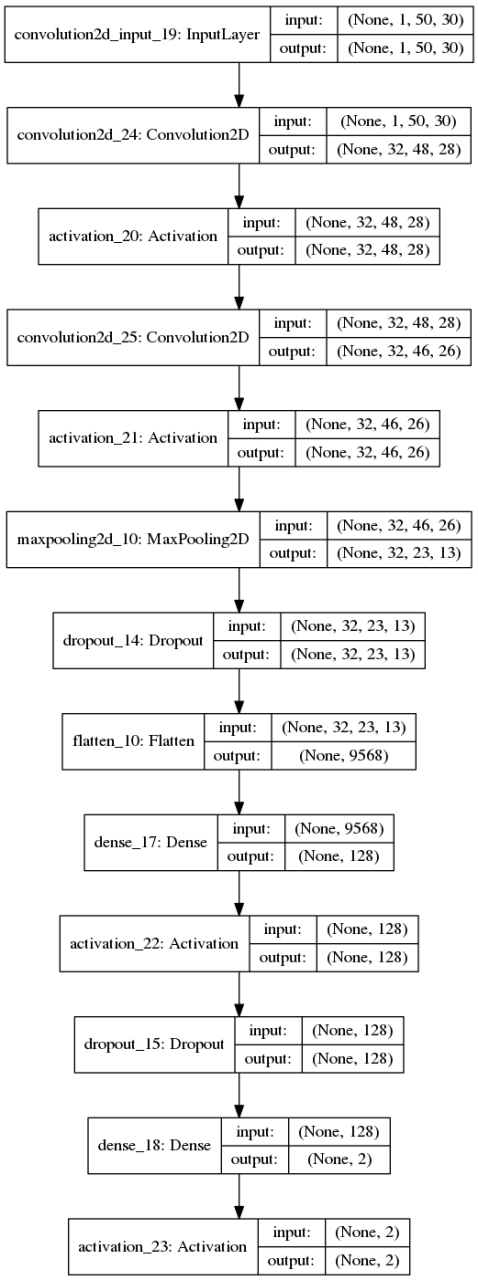

FIGURE 30. CNN MODEL 1: ARCHITECTURE

5.4.4.2 MODEL 2

This design is based on a more compact model, which has a single convolutional layer, followed by a max pooling layer, a dropout for regularization layer, reduces the outputs until a binary classifier is obtained. The objective is to have a simple neural network to compare. This structure is presented in Figure 31.

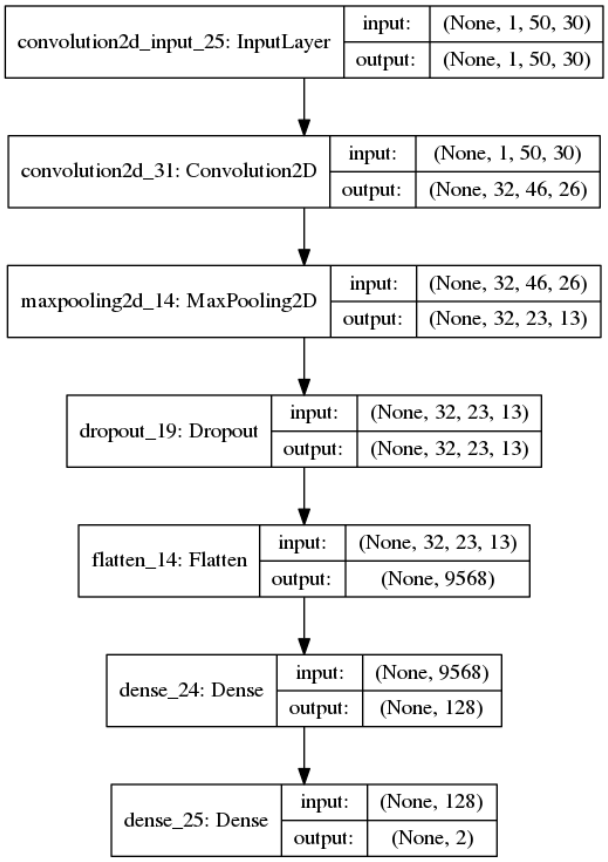

FIGURE 31. MODEL 2: ARCHITECTURE

5.4.4.3 MODEL 3

A third implementation have been designed to check the performance when different convolution layers are stacked separated by Max Pooling after every one. It is a variation of the first layer without ReLu layers. The structure is presented in Figure 32.

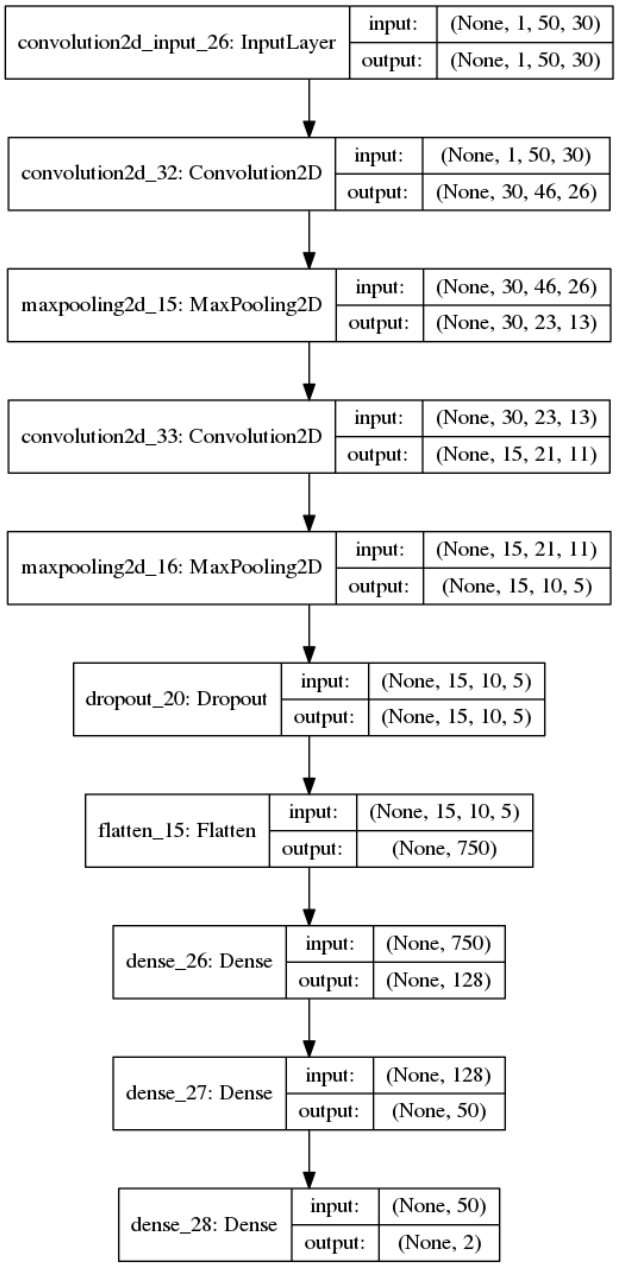

FIGURE 32. MODEL 3: ARCHITECTURE

5.4.4.4 MODEL 2: 3 INPUT CHANNELS

Two extra implementations have been designed in order to give location information to the network.

In this version, and using Model 2 design as a base (see Figure 31), we have modified the architecture to receive as an input, images with three channels. One channel contains the original image, the second channel contains a zoomed version of the image centered at the center of the image, and the third one an image a little more zoomed. The structure of this network is presented in Figure 33.

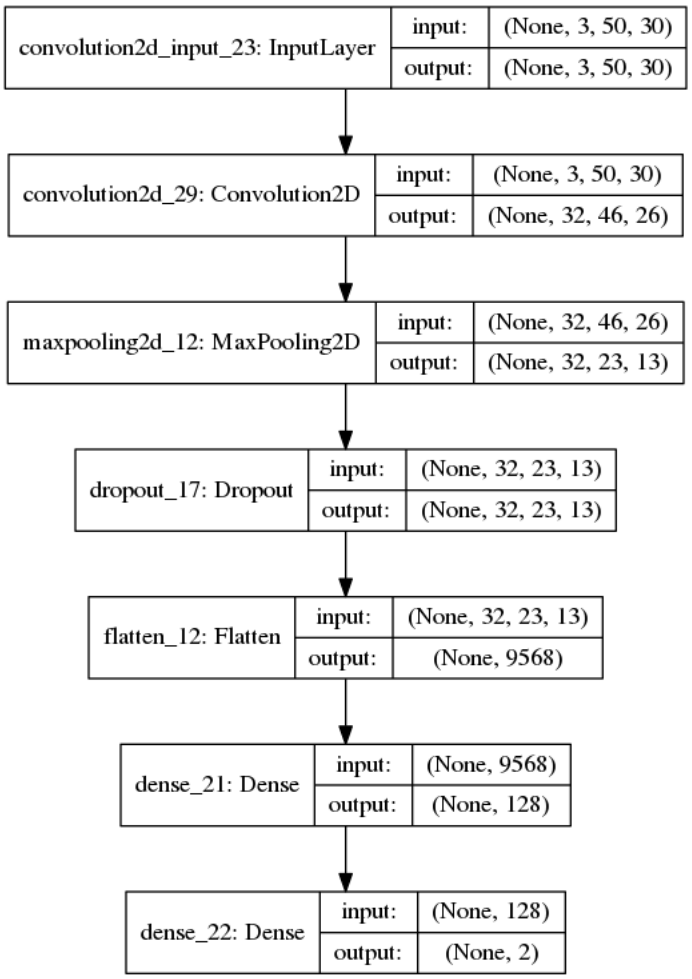

FIGURE 33. MODEL 2 WITH 3 INPUT CHANNELS

5.4.4.5 MODEL 2: 3 CONNECTED LAYERS

This version of Model 2 CNN, receives location information in three different input layers that performs a convolution individually to each zoomed image, and then these layers are merged in a Merge layer which prepares data to be received by a Max pooling layer. Then, all the layers are the same as in Model 2. In Figure 34 the structure of this network is presented.

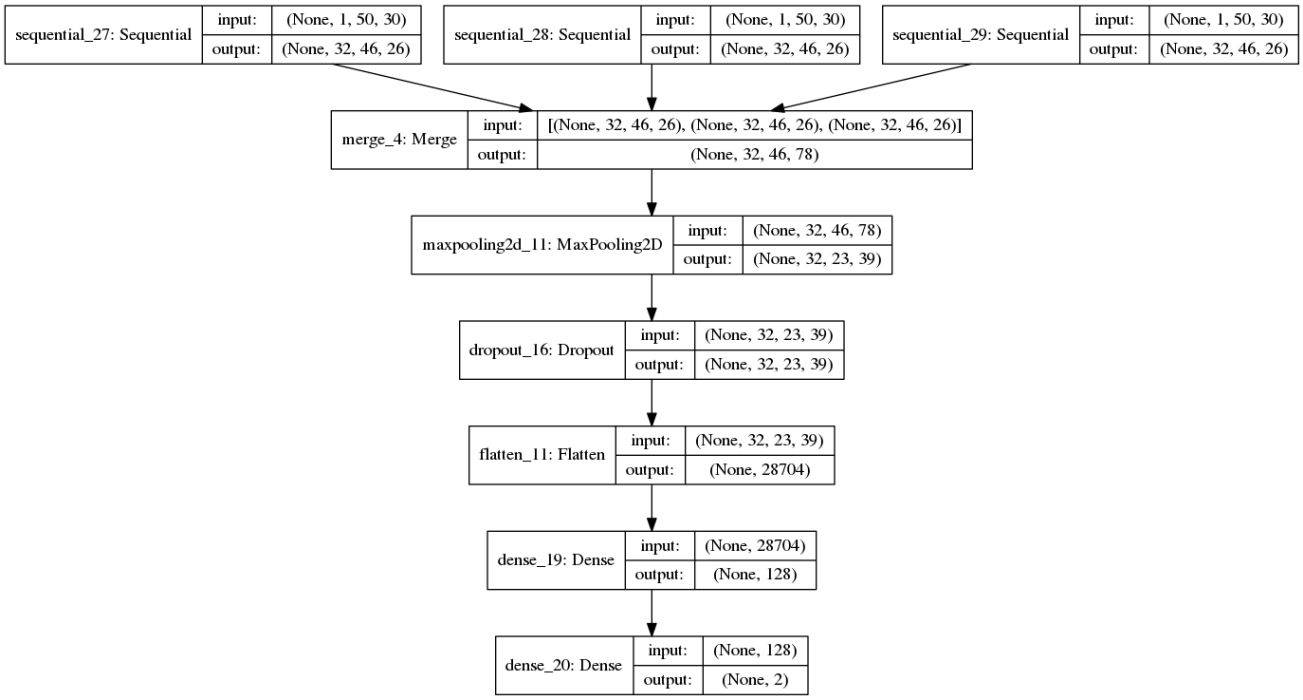

FIGURE 34. MODEL 2 WITH 3 MERGED CONVOLUTIONAL LAYERS AS INPUT

5.5 SOFTWARE

Here, we detail the software we have used during the development of this project. The preprocessing is based on the FSLVBM pipeline, which uses the following main packages from the toolkit FSL:

- FSLMATHS: Perform operations over neuroimaging data files.
- BET: Skull-stripping software.
- FLIRT: for linear registrations
- FNIRT: non-linear registrations
- FAST: tissue segmentation into CSF, GM and WM.
- FSLROI: extract defined ROI

Visualization software:

- MRICron: Software for visualization of MRI images
- FSLView: Software used to produce 3D images

Programming environment:

- Python: The programming language for version 3.5
- Anaconda: An open-source data science platform powered by Python. Some of the more important modules used in this project are:
    - Scikit-learn
    - Scipy
- Nibabel: Read/write neuroimaging file formats in python
- OpenCV: to process and manipulate on images

Deep Learning

- Keras: High-level neural networks library, written in Python and capable of running on Tensorflow.
- Tensorflow: Open-source software library for numerical computation using data flow graphs.

6 RESULTS

6.1 QUANTITATIVE RESULTS

6.1.1 USED METRICS

In order to compare the different methods, the same metrics have been considered for all of them.

Accuracy

The accuracy, also known as Rand Index or Rand accuracy, is the rate of true results (both true positives and true negatives) among the total number of cases examined.

$$Accuracy = \frac{True\ Positives + True\ Negatives}{True\ Positives + True\ Negatives + False\ Positives + False\ Negatives}$$

Sensitivity

The sensitivity, also known as True Positive Rate (TPR) or Recall, refers to the probability of detection. In other words, it measures the proportion of positives that are correctly identified as such.

$$Sensitivity = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

Specificity

The specificity, also known as True Negative Rate, measures the proportion of negative rates that are correctly identified as such.

$$Specificity = \frac{True\ Negatives}{True\ Negatives + False\ Postives}$$

Confusion matrix

A confusion matrix, also known as error matrix, is a table that allows to see the performance of an algorithm where each column represents the instances in a predicted class and each row refers to the instances in an actual class.

	Positive Predicted Condition	Negative Predicted Condition
Positive True Condition	**True Positive (TP)**	**False Negative (FN)**
Negative True Condition	**False Positive (FP)**	**True Negative (TN)**

TABLE 3. CONFUSION MATRIX

In Table 3. Confusion Matrix it is explained a general Confusion Matrix. In this project, next to each value, it is included the ratio for simplicity.

ROC Curve

Receiver Operating Characteristic (ROC) curve is a graphical plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied. It is the relation of the false positive rate (FPR, y axis) and the true positive rate (TPR, x axis).

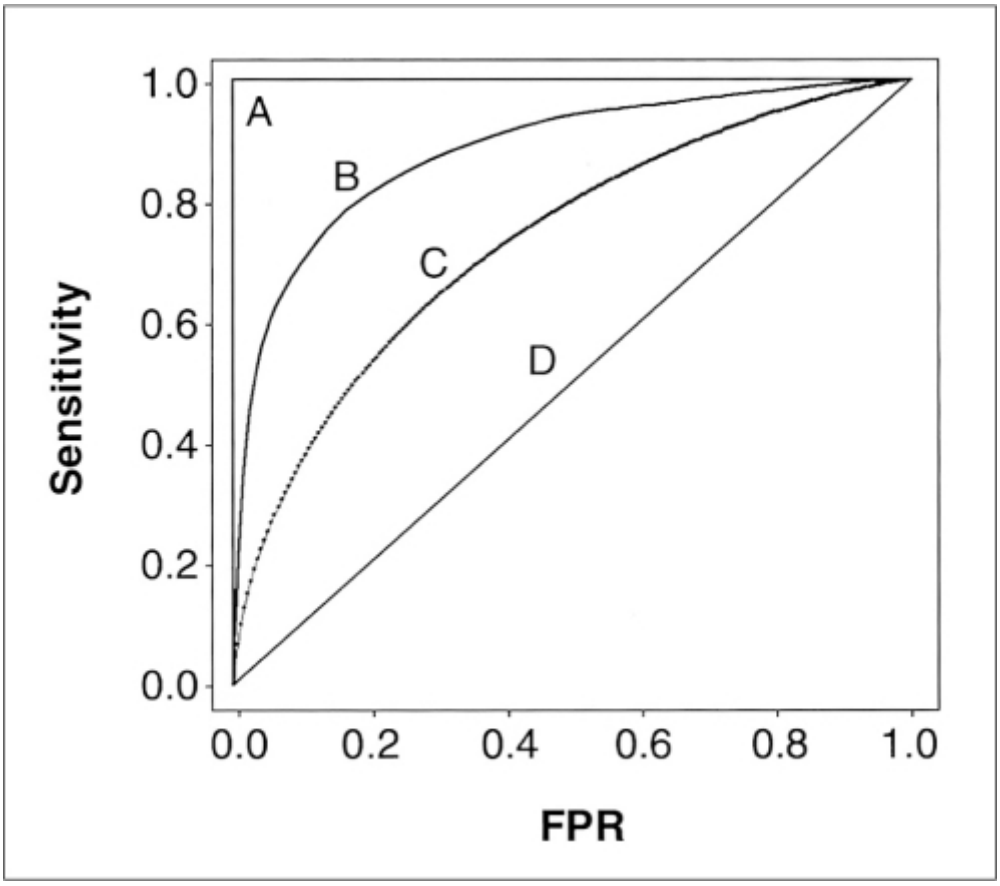

FIGURE 35. EXAMPLES OF FOUR DIFFERENT ROC CURVES (SOURCE: OPEN.NLM.NIH.GOV)

In Figure 35, an example of four different ROC curves are shown. The best possible test is A, with an area under the ROC curve of 1. The chance diagonal, which corresponds to the same predictive power as a random classifier, is D, and has a ROC area under the curve of 0.5. B and C are two different areas between A and D, and between them B has a better overall diagnostic performance than test C.

This plot is created by changing the threshold of the classifier, in our case incrementing the size of the threshold to consider if it can be considered as CSP or not. So, in the case of CSP size, this graphic would be created by first looking at a threshold, like considering CSP when size > 1mm, in both results (GT and predicted values) and plotting the result considering the sensitivity in axe Y and the specificity in axe X, then increase that threshold and plot sensitivity over sensibility, and so on.

In our case, this type of graphic will let us compare between the GT and the predicted value of the different algorithms, this will let an analysis of the kind: 'In our classifier, when it finds a large CSP of more than 5mm depth, it has a sensitivity of X and a specificity of Y.' From that reasoning, it will be possible to know the predictive power depending on the result obtained.

6.1.2 BASELINE METHOD RESULTS

In Table 4 the confusion matrix for the baseline is showed. The sample dataset to test this algorithm has been provided with more negative samples in order to let the algorithm learn better features.

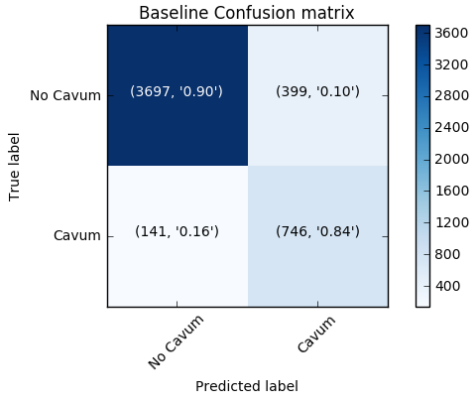

TABLE 4. BASELINE CONFUSION MATRIX

6.1.3 MACHINE LEARNING RESULTS

All methods, except Extra-Trees, in Machine Learning and Deep Learning techniques have been tested on the same data. Data Augmentation was performed to obtain a decent number of CSP detected in order to improve results. Extra-Trees was tested across the same data sample but with more slices labeled as No Cavum, in same conditions of balancing it performed on low rates of accuracy.

6.1.3.1 FEATURE SELECTION: EXTRA-TREES

In Table 5, the results for the Feature Selection method, Extremely Randomized Trees (Extra-Trees) are presented.

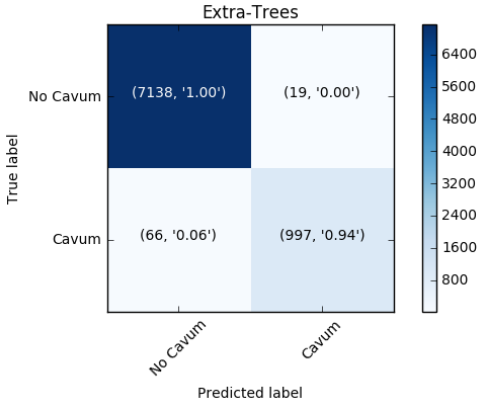

TABLE 5. EXTRA-TREES CONFUSION MATRIX

6.1.3.2 FEATURE EXTRACTION

Next, we present the results for the different strategies used for Feature Extraction.

6.1.3.2.1 PCA USING RANDOMIZED SVD

The first dimensionality reduction method used, is the Principal Component Analysis. In Figure 36 we can see the variance ratio explained on the first 25 components, which reaches more than a 95% of variance explained. In the different algorithms, these 25 components are the features used to learn.

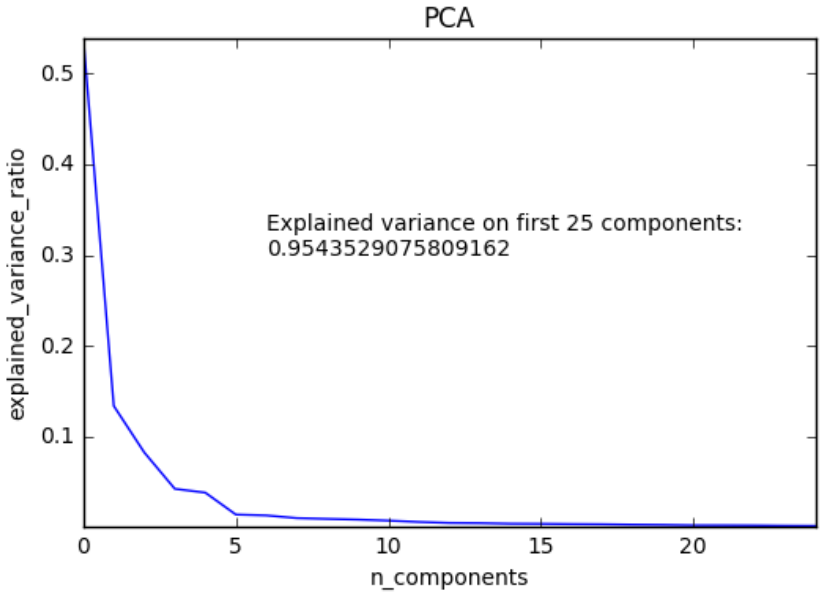

FIGURE 36. EXPLAINED VARIANCE RATIO ON FIRST 25 COMPONENTS

LDA (with PCA)

In Table 6 we present the results for Linear Discriminant Analysis. The features are extracted using PCA.

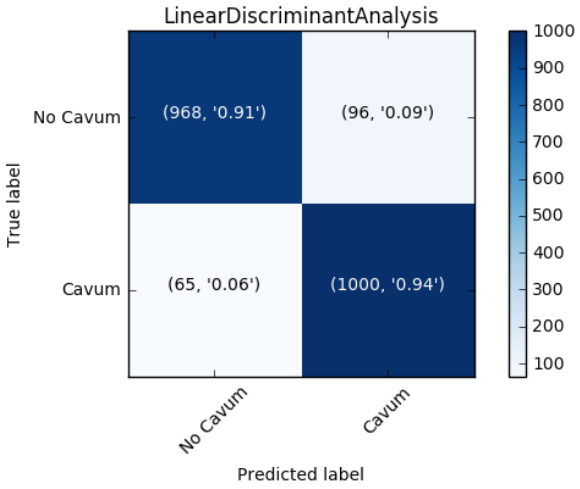

TABLE 6. LDA CONFUSION MATRIX

KNN (with PCA)

In Table 7 the results for the K Neighbors Classifier with k=3 are presented. The features are extracted using PCA.

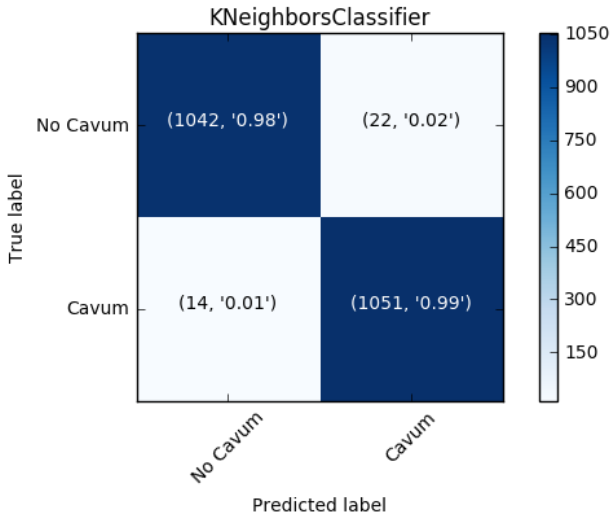

TABLE 7. KNN CONFUSION MATRIX

Linear SVC (with PCA)

In Table 8, we present the results for a Support Vector Classifier with Linear kernel. The features are extracted using PCA.

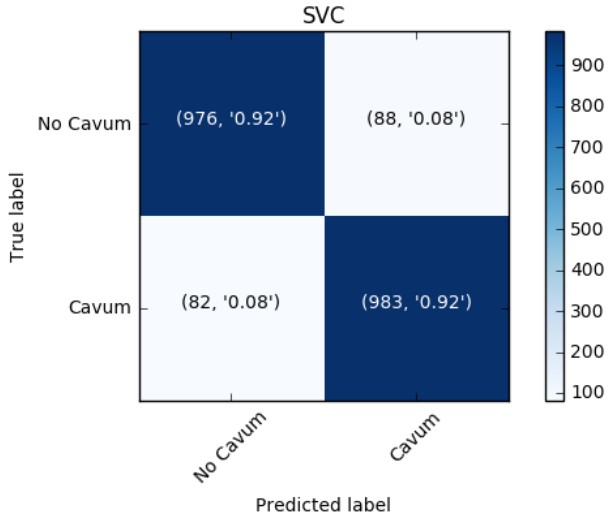

TABLE 8. SVC CONFUSION MATRIX

AdaBoost (with PCA)

In Table 9 the results of an AdaBoost Classifier are presented. The features are extracted using PCA.

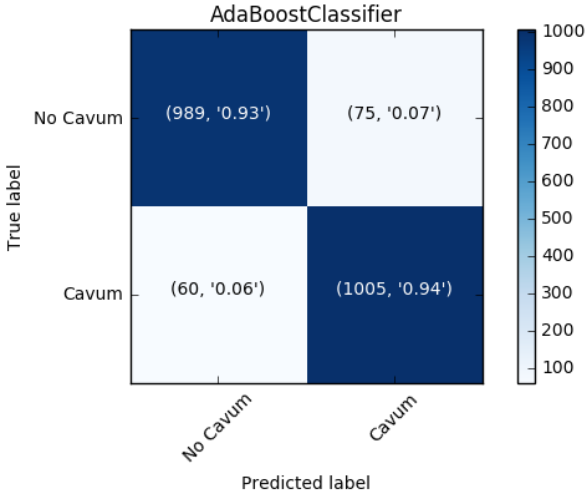

TABLE 9. ADABOOST CONFUSION MATRIX

Random Forests (with PCA)

In Table 10, the results for the Random Forests Classifier are presented. The features were extracted using PCA.

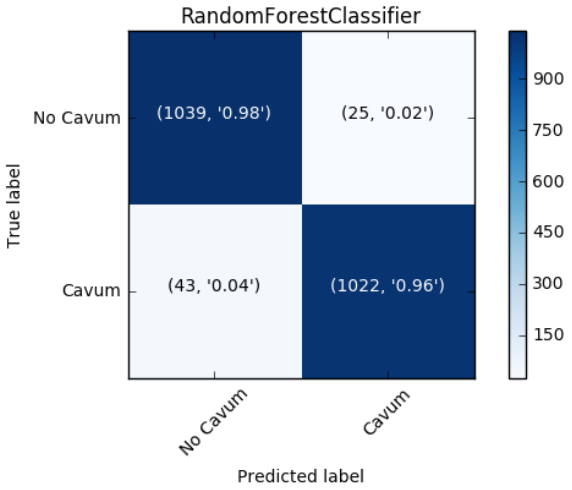

TABLE 10. RANDOM FOREST CONFUSION MATRIX

6.1.3.2.2 HOG DESCRIPTORS

In this section, we present the results of the different classifiers using HOG Descriptors.

Linear Discriminant Analysis (with HOG Descriptors)

In Table 11, we present the results of Linear Discriminant Analysis. Features are the HOG Descriptors.

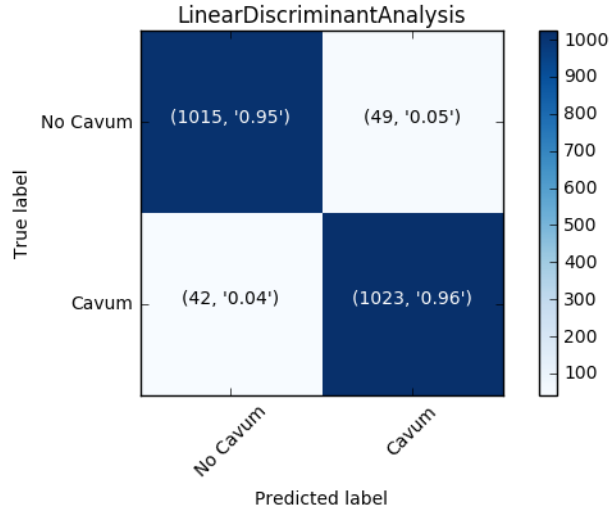

TABLE 11. LDA WITH HOG DESCRIPTORS CONFUSION MATRIX

KNN (with HOG Descriptors)

In Table 12, the results for the classifier K Nearest Neighbors are presented. Features are the HOG Descriptors.

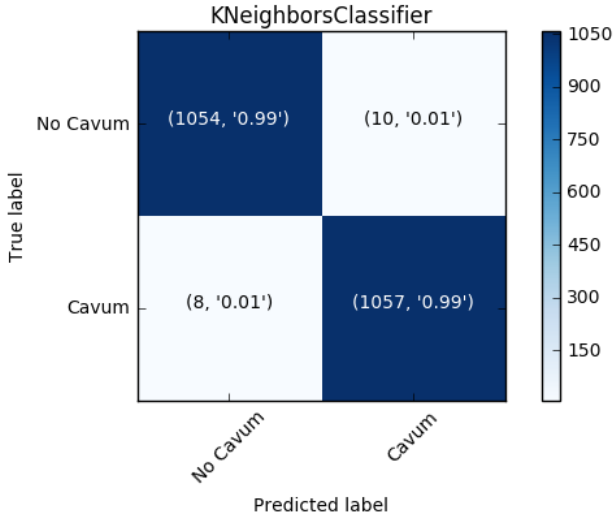

TABLE 12. KNN WITH HOG DESCRIPTORS CONFUSION MATRIX

Linear SVC (with HOG Descriptors)

In Table 13, the results for the Support Vector Classifier are shown. Features are the HOG Descriptors.

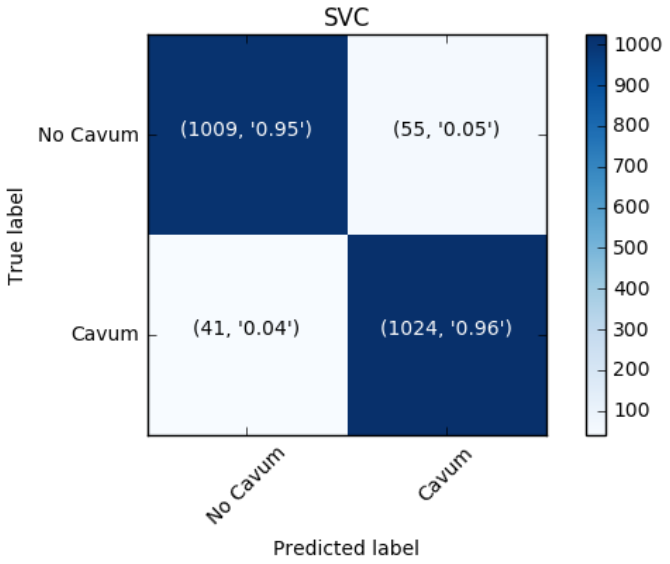

TABLE 13. SVC WITH HOG DESCRIPTORS CONFUSION MATRIX

AdaBoost (with HOG Descriptors)

In Table 14, the results for the AdaBoost Classifier are presented. The features are HOG Descriptors.

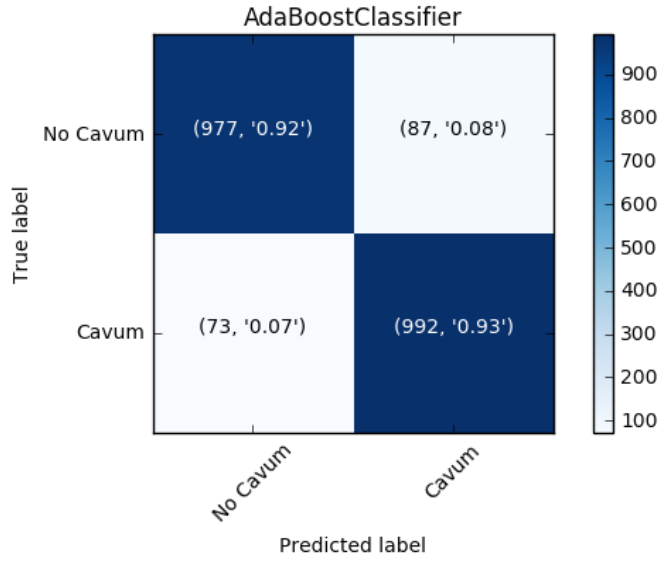

TABLE 14. ADABOOST WITH HOG DESCRIPTORS CONFUSION MATRIX

Random Forests (with HOG Descriptors)

In Table 15, the results for the Random Forests Classifier are shown. The features are HOG Descriptors.

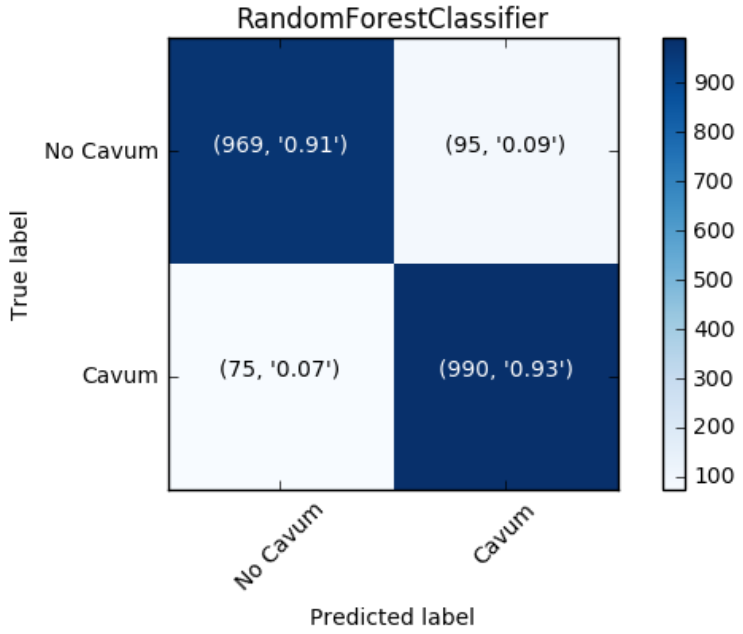

TABLE 15. RANDOM FOREST WITH HOG DESCRIPTORS CONFUSION MATRIX

6.1.4 Deep Learning methods

In this section, the results for the different designed methods of Deep Learning are presented.

CNN model 1

In Table 16, the results for the CNN Model 1 are shown.

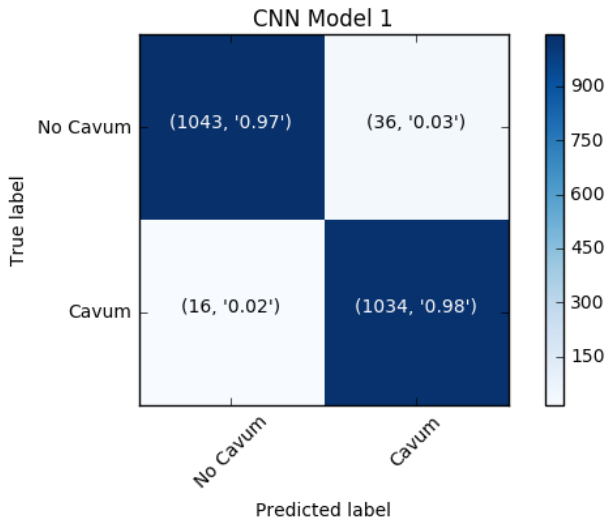

TABLE 16. MODEL 1 CNN CONFUSION MATRIX

CNN model 2

Table 17, the results for the CNN Model 2 are presented.

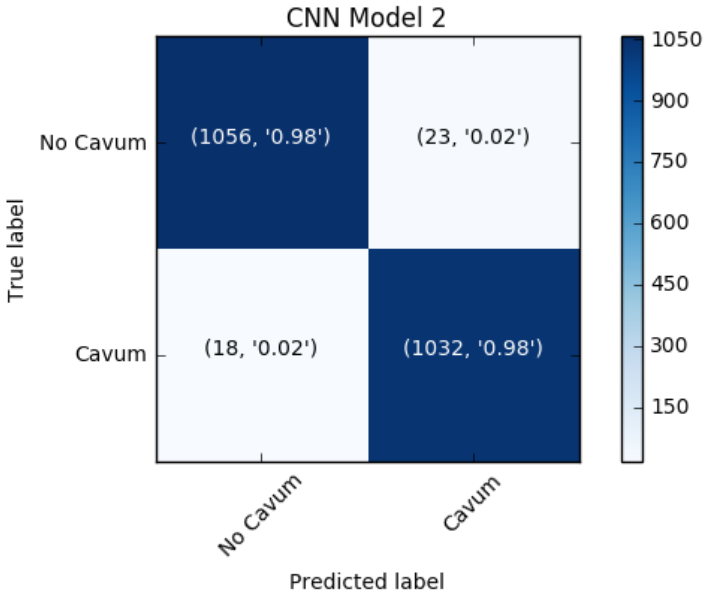

TABLE 17. MODEL 2 CNN CONFUSION MATRIX

CNN Model 3

In Table 18, the results for the CNN Model 3 are presented.

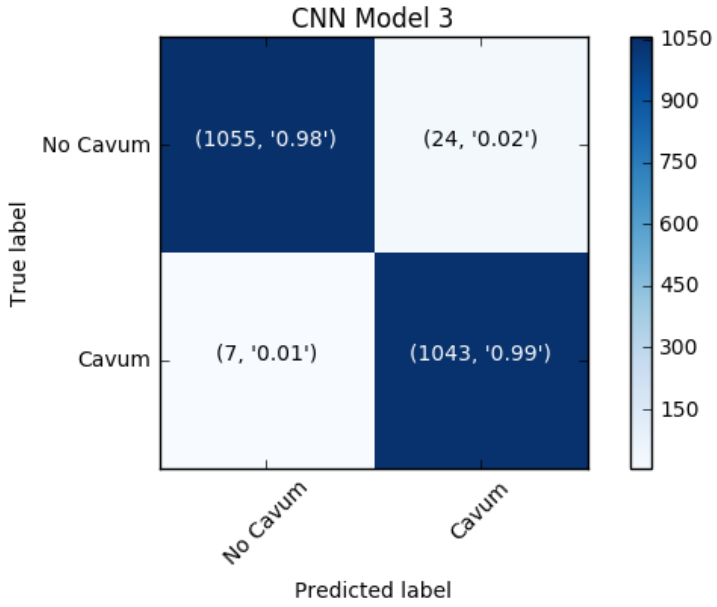

TABLE 18. MODEL 3 CNN CONFUSION MATRIX

CNN Model 2: 3 Different Channels

In Table 19, the results for the CNN Model 2 with 3 different channels as inputs are shown.

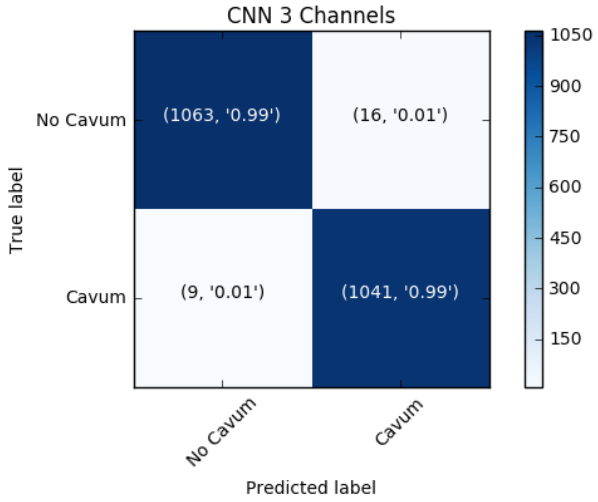

TABLE 19. CNN 2 WITH 3 INPUT CHANNELS CONFUSION MATRIX

CNN Model 2: 3 Connected Layers

In Table 20, the results for the CNN Model 2, a model with 3 connected input layers with contextual information, are presented.

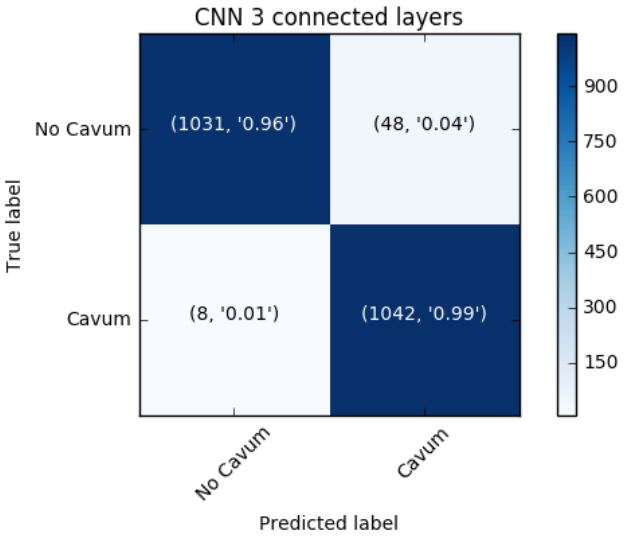

TABLE 20. CNN WITH 3 CONNECTED LAYERS CONFUSION MATRIX

6.1.4.1 COMPARISON OF THE THREE MODELS

Once the three models were developed, a test between the three were conducted in order to look for the best design. The architecture with best performance was chosen to perform new experiments, explained in "5.4.4.4. Model 2: 3 Input Channels" and "5.4.4.5. Model 2: 3 Connected Layers". To evaluate the performance of the three methods, two charts were generated, the evolution of the loss and the evolution of the accuracy over the epochs on the training stage (see Figure 37. Loss over epochs and Figure 38. Accuracy over epochs).

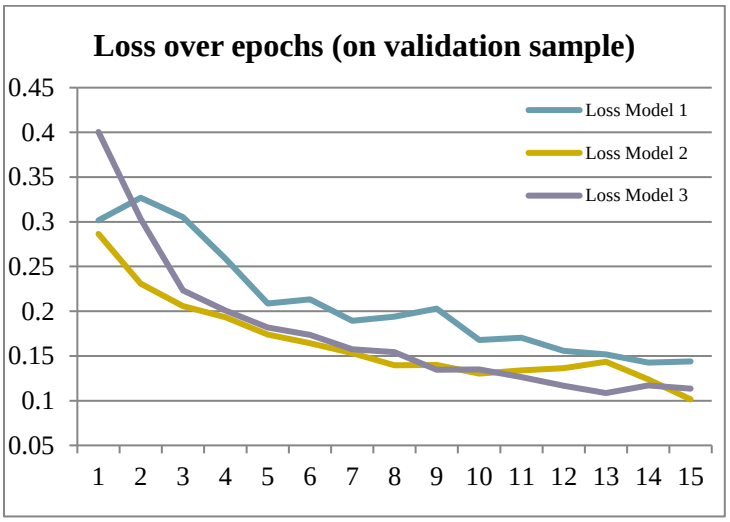

FIGURE 37. LOSS OVER EPOCHS

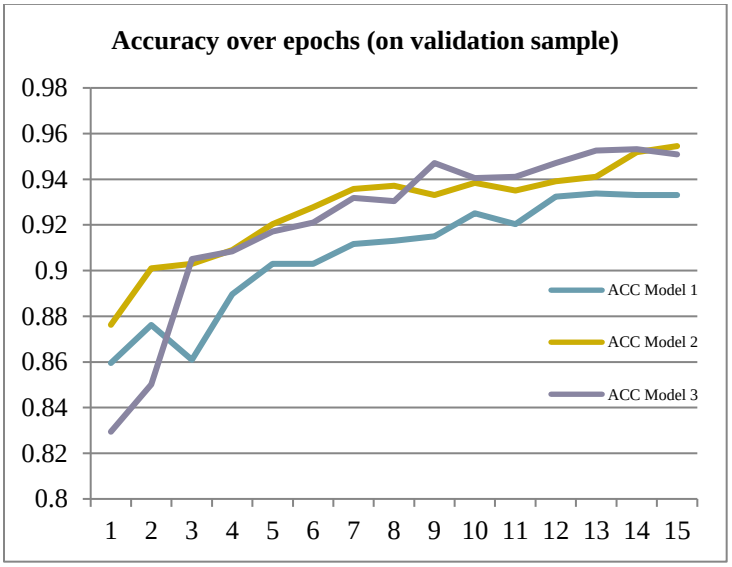

FIGURE 38. ACCURACY OVER EPOCHS

In both graphs the best method seems to be Model 2 (see 5.4.4.2. Model 2). This will be the method chosen for the following experiments.

6.1.5 DEPTH COMPARISON WITH HANDMADE GROUND-TRUTH

All the training and testing performed when comparing the performance of the different methods have been computed using an own manually handmade ground-truth where each 2D image is classified between having CSP or not.

The authors from [1][2] provided a ground-truth of the depth in number of slices for every subject, but not which concrete slices had CSP. In this section, we will compare the CSP depth computed by our approach (using our training set), to the manually ground-truth proportioned by the authors.

Once the model with the best performance was found, an analysis to compare the best method in deep learning method (Model 3 with 3 input channels) and the best machine learning approach (KNN with 3 neighbors) with respect to the manual ground-truth defined [1]. To conduct this analysis, a comparison using ROC curves between both values have been conducted (see Figure 39 ,Figure 40, Figure 41 and Figure 42). The performance of both methods when comparing to the ground-truth of the study was conducted by increasing the number of samples of no CSP to improve the overall detection. Two plots per method are presented. The first one showing the detection of CSP of any size (number of slices > 0). The second one will be the detection of a CSP considered as Large by the authors of [1] and [2], a size of CSP>5mm. The first plot will show the overall performance of the classifier, and the second one will give information about the cases where Large CSP is found. A small CSP is considered as normal part of the anatomy, so this second plot will show the performance of the classifier when an abnormal size of the CSP is found.

KNN

As it can be seen in Figure 39 a comparison using ROC curves to detect CSP of any size.

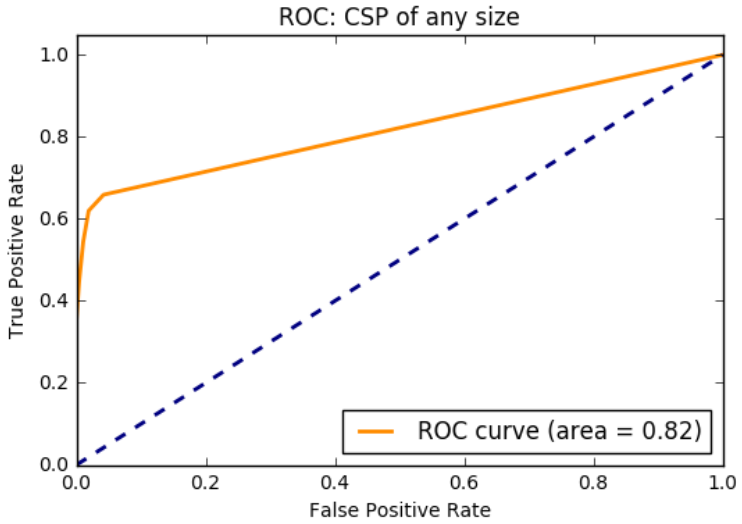

FIGURE 39. KNN VS GROUND. ROC CURVE CSP OF ANY SIZE

In Figure 40, the comparison using ROC curves when finding large CSP (>5mm) is presented.

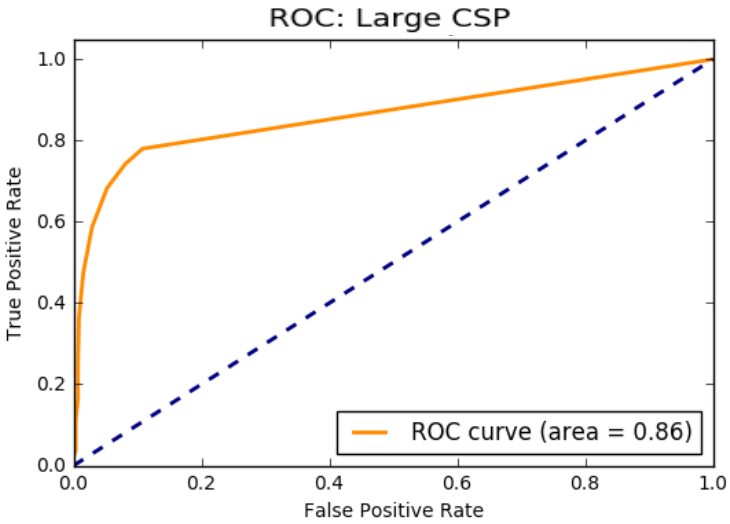

FIGURE 40. KNN VS GROUND. ROC CURVE FOR LARGE CSP

Model 3

In Figure 41, a comparison using ROC curve is presented when finding any CSP size.

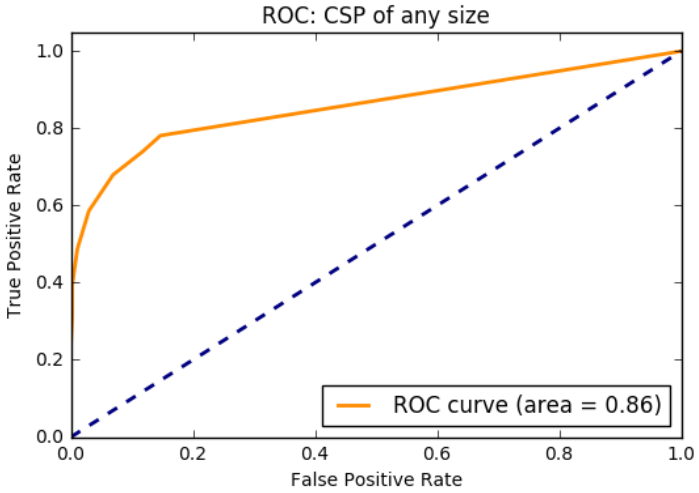

FIGURE 41. MODEL 3 VS GROUND. ROC CURVE CSP OF ANY SIZE

In Figure 42, a comparison using ROC curve is presented when finding large CSP (>5mm).

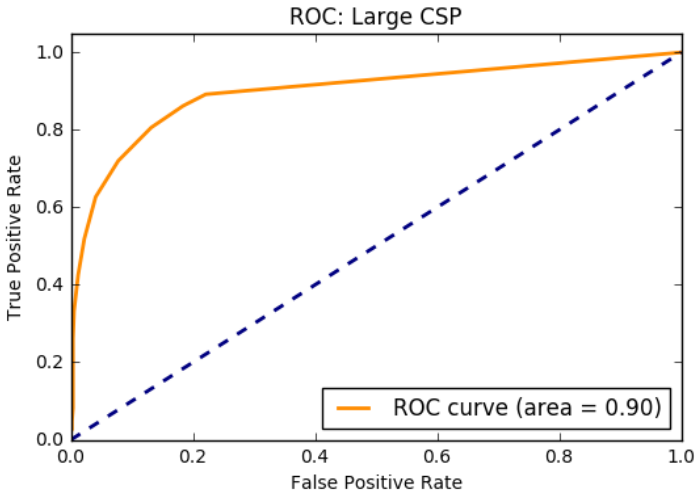

FIGURE 42. MODEL 3 CNN 3 CHANNELS VS GROUND. ROC CURVE OF LARGE CAVUM

As seen in Figure 41 and Figure 42, this is the best approach when comparing with the original ground [3]. These results can be interpreted as follows, when thinking in applying to a software to automatically apply it:

- If the predicted result is 0 slices of depth: it will be considered as not having CSP (22% of False Negatives).
- Between 1 and 3 slices: To check manually (6% of total subjects to be checked in this data).
- More than 3 slices: considered as CSP (3% of False Positives)

[3] The ground as found in [1]. The major difference between ground and deep learning results is due to the fact that the training ground have been created especially for this project, and the original ground-truth did not include which slices were considered as CSP or not, only the depth of it.

6.2 SUMMARY OF THE QUANTITATIVE RESULTS

In Table 21 the results for all the methods are summarized in one single table. The best methods with similar results are both KNN with HOG Descriptors and CNN Model 2 with 3 Channels as input with different zooms.

METHOD	ACCURACY	SENSITIVITY	SPECIFICITY
BASELINE *	0.892*	0.84*	0.9*
EXTRA-TREES *	0.989*	0.94*	0.99*
LDA (PCA)	0.924	0.94	0.91
LDA (HOG)	0.957	0.96	0.95
KNN (PCA)	0.983	0.99	0.98
KNN (HOG)	**0.991**	**0.99**	**0.99**
SVC (PCA)	0.920	0.92	0.92
SVC (HOG)	0.955	0.96	0.95
ADABOOST (PCA)	0.937	0.94	0.93
ADABOOST (HOG)	0.925	0.93	0.92
RANDOM FORESTS (PCA)	0.968	0.96	0.98
RANDOM FORESTS (HOG)	0.920	0.93	0.91
CNN 1	0.976	0.98	0.97
CNN 2	0.981	0.98	0.98
CNN 3	0.980	0.99	0.98
CNN 2: 3 CHANNELS	**0.988**	**0.99**	**0.99**
CNN 2: 3 LAYERS	0.974	0.99	0.96

TABLE 21. SUMMARY OF THE RESULTS IN ALL THE METHODS. *METHODS WITH INCREASED NEGATIVE SAMPLES

6.3 Qualitative Results

In this section, we will see some examples of correct and incorrect classification results from the best methods (KNN and CNN Model 3 with 3 channel inputs). In Table 22 and Table 23 some examples of classification are shown.

QUALITATIVE KNN CLASSIFICATION EXAMPLES

True Positives	True Negatives	False Negatives	False Positives

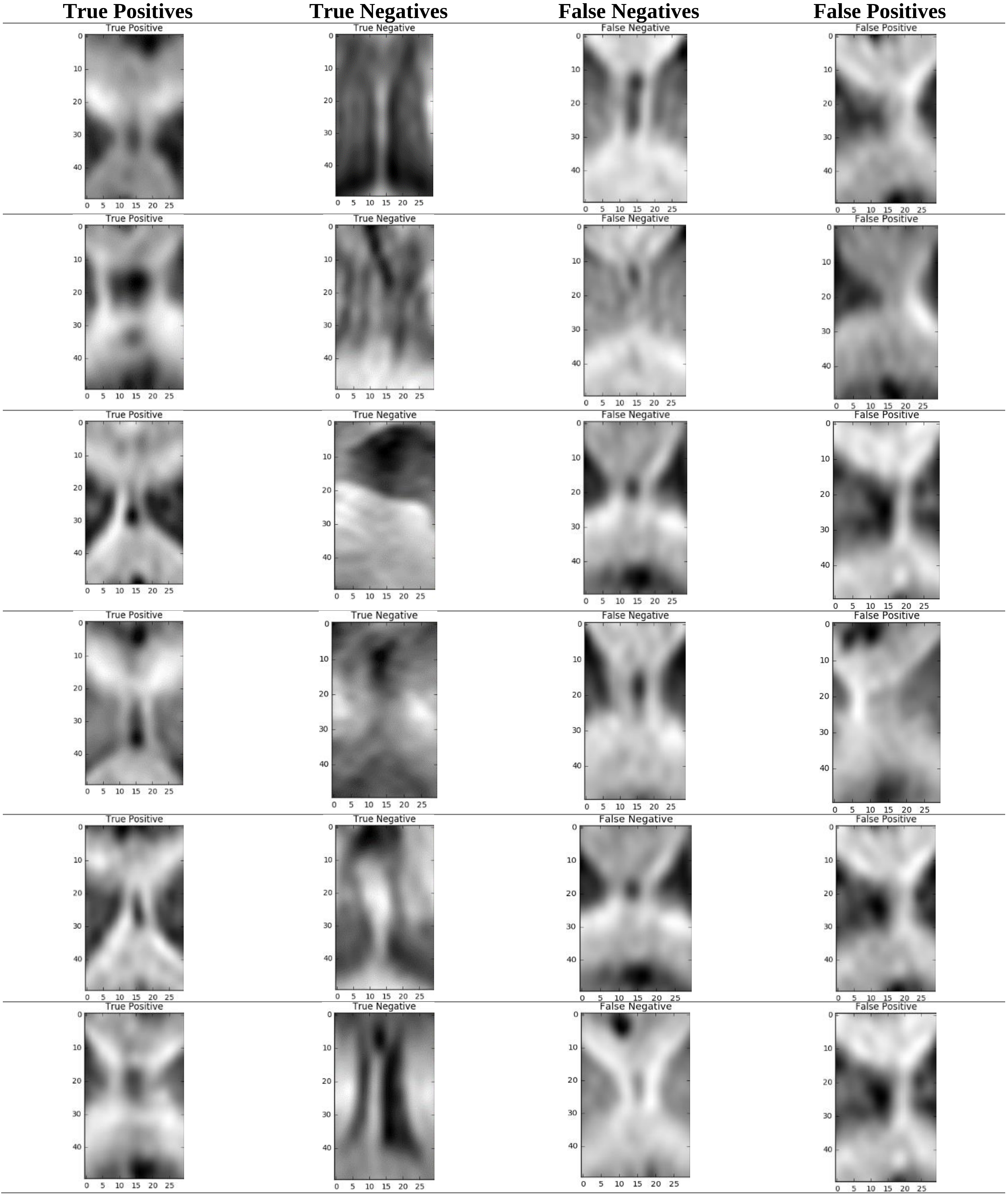

TABLE 22. QUALITATIVE EXAMPLES OF THE KNN CLASSIFICATION ALGORITHM

QUALITATIVE CNN MODEL 3 WITH 3 INPUT CHANNELS CLASSIFICATION EXAMPLES			
True Positives	**True Negatives**	**False Negatives**	**False Positives**

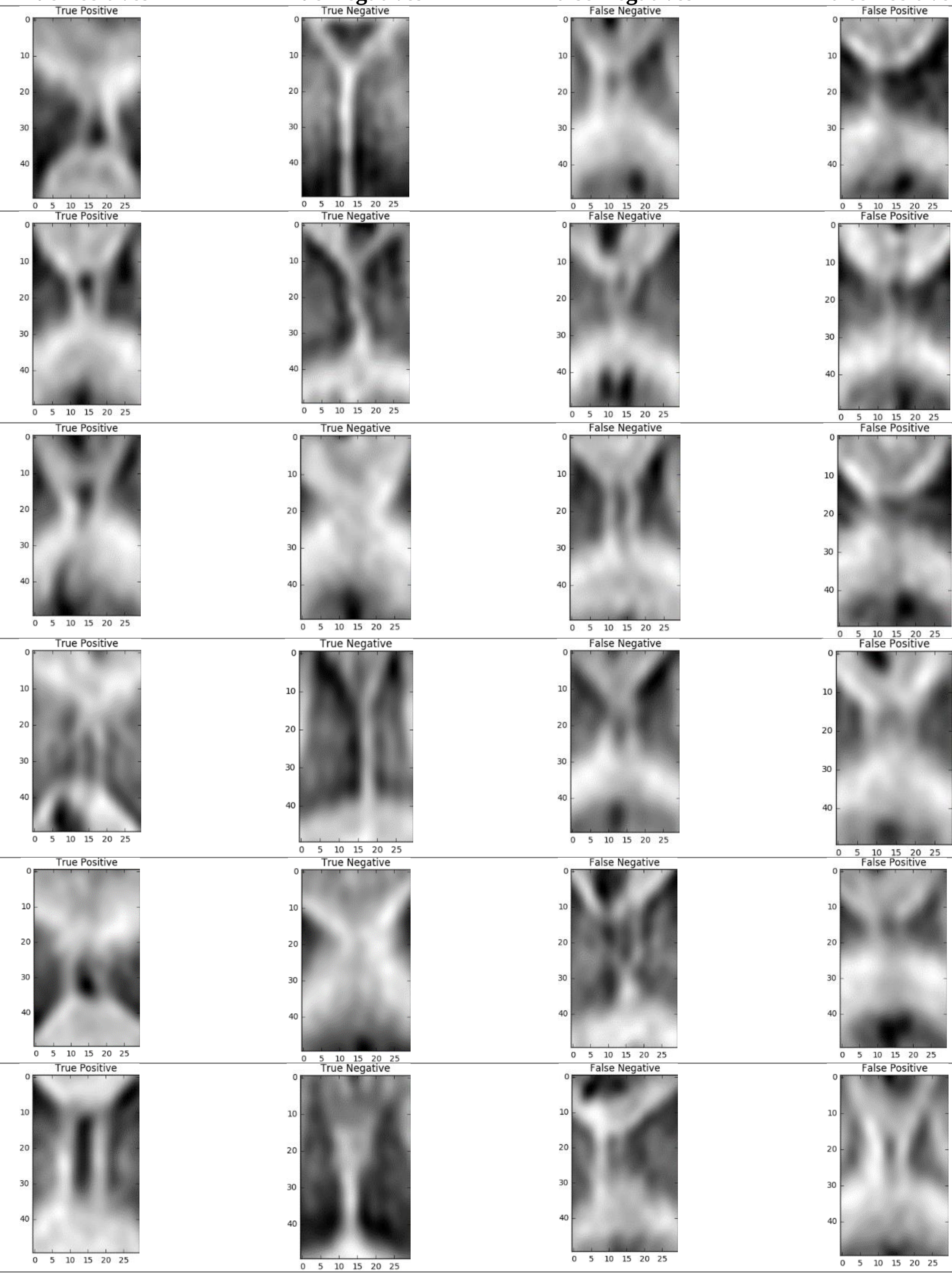

TABLE 23. QUALITATIVE EXAMPLES OF THE CNN MODEL 3 WITH 3 CHANNELS CLASSIFICATION ALGORITHM

As can be seen in Table 22 and Table 23, False Positive and False Negative are images that are in some cases difficult to classify even manually due to the fact that they belong to the first or the last slices of the CSP. Even some images could be considered correct depending the expert that analyzes them.

In Figure 43, the different 32 5x5 filters learned at the first layer in CNN model 3 are presented. Convolutional layers produce different filters for the image, and these filters, when more smooth and without noisy patterns, the better.

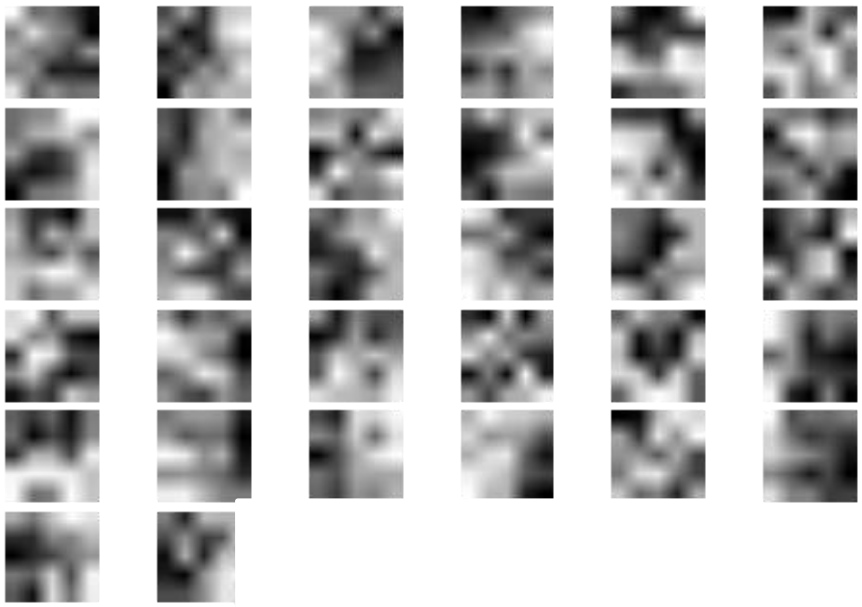

FIGURE 43. EXAMPLE OF THE 32 5X5 FILTERS IN THE FIRST LAYER OF CNN MODEL 3

7 CONCLUSIONS

In this project, we have proposed different approaches to automatically detect a midline brain abnormality, the cavum septum pellucidum (CSP). This abnormality is a space that appears in between the leaflets from the septum pellucidm when these walls fail to fuse.

In the first baseline method proposed, the detection of the CSP have been performed by using an unsupervised approach, which tries to find a correct binarization of the image, and after that a flood-fill algorithm is performed to identify the zone where the CSP appears. Despite not producing such reliable detection results as the other methods, this approach has the main advantage that it is unsupervised and also can obtain the area of the CSP.

In the second group of methods, we performed a comparison between different common machine learning classifiers using different dimensionality reduction techniques.

Three different models of CNNs have been proposed in order to test the latest techniques in object recognition/detection in deep learning. Once the best of the three methods was identified, having this architecture as a base, we proposed an improvement of the design by inputting additional contextual information about the location of this abnormality by passing to the first layer of the network different zoomed images.

Finally, using the results found with the best machine learning method and the best deep learning design, we have compared the predicted values with the manual ground-truth [1].

The manual detection and characterization of this zone is a time-consuming task.

In this project, we have shown that the proposed automatic methods can be useful and used by researchers to obtain information about this abnormality and investigate about the midline brain abnormalities in an easier manner.

8 FUTURE WORK

After seeing the good results obtained when testing the proposed methods, the next step to be performed would be to let other researchers take profit of this approach to detect abnormalities. The first step would be to create an easy-to-use software that any clinical expert could use to create a personalized approach to their data.

The first method proposed, which is the baseline unsupervised approach, returns the area of the CSP, and thus the volume of this abnormality for each subject would be trivial to calculate. The use of this method to create a segmentation training dataset of the CSP, and passing this information to a modified approach to the best CNN model would let the deep learning method to be used to segment instead of binary classification. This would produce new information for each subject that, as far as we looked for, any study has analyzed.

The cavum septum pellucidum (CSP) structure is always found in sequential slices of the same subject. Adding a last step in the different methods like a Hidden Markov Model could add this contextual information to the calculated model, and should improve the results, being able to obtain better results even with fewer slices.

Regarding the deep learning approaches, we could try to improve the results in two different ways. First adding contextual information. The CSP structure is always found in sequential slices of the same subject. Then, adding a last step in the different models, like a Hidden Markov Model, could learn that neighbor slices should have a higher probability of having the same label. Second, we could try new promising models as Generative Adversarial Networks, which are producing great results in different image classification datasets.

.

REFERENCES

[1] R. Landin-Romero *et al.*, "Midline Brain Abnormalities Across Psychotic and Mood Disorders," *Schizophr. Bull.*, vol. 42, no. 1, pp. 229–238, 2016.

[2] R. Landin-Romero *et al.*, "Prevalence of cavum vergae in psychosis and mood spectrum disorders," *J. Affect. Disord.*, vol. 186, pp. 53–57, 2015.

[3] T. Takahashi *et al.*, "Midline brain abnormalities in established bipolar affective disorder," *J. Affect. Disord.*, vol. 122, no. 3, pp. 301–305, 2010.

[4] S. Nuñez, M. T. Mantilla, and S. Bermúdez, "Midline Congenital Malformations of the Brain and Skull," *Neuroimaging Clinics of North America*, vol. 21, no. 3. pp. 429–482, 2011.

[5] W. S. Gharaibeh, F. J. Rohlf, D. E. Slice, and L. E. DeLisi, "A geometric morphometric assessment of change in midline brain structural shape following a first episode of schizophrenia," *Biol. Psychiatry*, vol. 48, no. 5, pp. 398–405, 2000.

[6] N. C. Andreasen *et al.*, "Abnormalities in midline attentional circuitry in schizophrenia: evidence from magnetic resonance and positron emission tomography," *Eur. Neuropsychopharmacol.*, vol. 5, no. SUPPL. 1, pp. 37–41, 1995.

[7] G. Douaud *et al.*, "Anatomically related grey and white matter abnormalities in adolescent-onset schizophrenia," *Brain*, vol. 130, no. 9, pp. 2375–2386, Sep. 2007.

[8] Z.-P. Liang and P. C. Lauterbur, *Principles of Magnetic Resonance Imaging*, vol. 177, no. 4. 2000.

[9] P. Nopoulos, V. Swayze, M. Flaum, J. C. Ehrhardt, W. T. C. Yuh, and N. C. Andreasen, "Cavum septi pellucidi in normals and patients with schizophrenia as detected by magnetic resonance imaging," *Biol. Psychiatry*, vol. 41, no. 11, pp. 1102–1108, 1997.

[10] J. L. R. Andersson, M. Jenkinson, and S. Smith, "Non-linear registration aka Spatial normalisation FMRIB Technial Report TR07JA2," *In Pract.*, no. June, p. 22, 2007.

[11] B. Sahiner *et al.*, "Classification of mass and normal breast tissue: A convolution neural network classifier with spatial domain and texture images," *IEEE Trans. Med. Imaging*, vol. 15, no. 5, pp. 598–610, 1996.

[12] H. Greenspan, B. van Ginneken, and R. M. Summers, "Guest Editorial Deep Learning in Medical Imaging: Overview and Future Promise of an Exciting New Technique," *IEEE Trans. Med. Imaging*, vol. 35, no. 5, pp. 1153–1159, 2016.

[13] S. K. M. Reza and K. M. Iftekharuddin, “Multi-class Abnormal Brain Tissue Segmentation Using Texture Features,” vol. 2013, pp. 38–42, 2013.

[14] E. M. Sweeney *et al.*, “A comparison of supervised machine learning algorithms and feature vectors for MS lesion segmentation using multimodal structural MRI,” *PLoS One*, vol. 9, no. 4, 2014.

[15] T. Brosch, L. Y. W. Tang, Y. Yoo, D. K. B. Li, A. Traboulsee, and R. Tam, “Deep 3D Convolutional Encoder Networks With Shortcuts for Multiscale Feature Integration Applied to Multiple Sclerosis Lesion Segmentation,” *IEEE Trans. Med. Imaging*, vol. 35, no. 5, pp. 1229–1239, 2016.

[16] A. Buades, B. Coll, and J.-M. Morel, “Non-Local Means Denoising,” *Image Process. Line*, vol. 1, pp. 490–530, 2011.

[17] S. P. Lloyd, “Least Squares Quantization in PCM,” *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[18] H. M. Zhang, L. S. Peh, and Y. H. Wang, “Study on Flood-Fill Algorithm Used in Micromouse Solving Maze,” *Appl. Mech. Mater.*, vol. 599–601, pp. 1981–1984, Aug. 2014.

[19] P. Geurts, D. Ernst, and L. Wehenkel, “Extremely randomized trees,” *Mach. Learn.*, vol. 63, no. 1, pp. 3–42, 2006.

[20] P. G. Martinsson, V. Rokhlin, and M. Tygert, “A randomized algorithm for the decomposition of matrices,” *Appl. Comput. Harmon. Anal.*, vol. 30, no. 1, pp. 47–68, 2011.

[21] N. Halko, P.-G. Martinsson, Y. Shkolnisky, and M. Tygert, “An algorithm for the principal component analysis of large data sets,” *SIAM J. Sci. Comput.*, vol. 33, no. 5, 2011.

[22] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[23] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, 2005, vol. I, pp. 886–893.

[24] N. S. Altman, “An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression,” *Am. Stat.*, vol. 46, no. 3, pp. 175–185, Aug. 1992.

[25] C. Cortes and V. Vapnik, “Support-Vector Networks,” *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[26] Tin Kam Ho, “Random decision forests,” *Proc. 3rd Int. Conf. Doc. Anal. Recognit.*, vol. 1, pp. 278–282, 1995.

[27] T. K. Ho, “The random subspace method for constructing decision forests,” *IEEE Trans. Pattern Anal. Mach.*

Intell., vol. 20, no. 8, pp. 832–844, 1998.

[28] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[29] A. Krizhevsky, I. Sutskever, and H. Geoffrey E., "ImageNet Classification with Deep Convolutional Neural Networks," *Adv. Neural Inf. Process. Syst. 25*, pp. 1–9, 2012.

[30] D. C. Cireşan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Mitosis detection in breast cancer histology images with deep neural networks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2013, vol. 8150 LNCS, no. PART 2, pp. 411–418.

[31] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Med. Image Comput. Comput. Interv. -- MICCAI 2015*, pp. 234–241, 2015.

[32] S. Pereira, A. Pinto, V. Alves, and C. A. Silva, "Deep convolutional neural networks for the segmentation of gliomas in multi-sequence MRI," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, vol. 9556, pp. 131–143.

[33] H. Ali, M. Elmogy, E. El-Daydamony, and A. Atwan, "Multi-resolution MRI Brain Image Segmentation Based on Morphological Pyramid and Fuzzy C-mean Clustering," *Arab. J. Sci. Eng.*, vol. 40, no. 11, pp. 3173–3185, Nov. 2015.

[34] D. H. Hubel and T. N. Wiesel, "Receptive fields of single neurones in the cat's striate cortex.," *J. Physiol.*, vol. 148, pp. 574–591, 1959.

[35] D. H. HUBEL and T. N. WIESEL, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex.," *J. Physiol.*, vol. 160, no. 1, pp. 106–54, Jan. 1962.

[36] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.

[37] I. Goodfellow, J. Pouget-Abadie, and M. Mirza, "Generative Adversarial Networks," *arXiv Prepr. arXiv ...*, pp. 1–9, 2014.

[38] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *arXiv*, pp. 1–15, 2015.

[39] S. M. Smith, "Fast robust automated brain extraction," *Hum. Brain Mapp.*, vol. 17, no. 3, pp. 143–155, 2002.