Automated image processing: Brain Volume determination method for Alzheimer’s disease

Bruno Santos¹, Sandra Rua², João Manuel R. S. Tavares¹

Abstract— Alzheimer’s disease (AD) is a prevalent neurodegenerative disease whose incidence tends to augment with the increase in life expectancy seen in recent years. Its symptomatic phase only manifests in medium to advanced stages which can hinder and difficult diagnosis. Previous studies have shown that there is a strong correlation between the whole-brain volume (WBV) and the incidence of this disease, which may allow for earlier detection and treatment. In this sense, and to serve as a foundation for the thesis work regarding this topic, this work aims to develop an image processing tool that is capable of calculating the WBV of healthy MR brain scans in an automated manner. This was achieved by the combine usage of two programming languages (MATLAB and Java) that allowed the construction of an organized image processing pipeline and respective GUI. The results obtained were then validated against the measured image volume controls in the software ITK-Snap and scrutinized. These results proved to be very consistent and maintained a high identity ratio to the control measurements (82%) which portrays success to the developed application.

Index Terms— Alzheimer’s disease, Whole-Brain Volume, Magnetic Resonance, Image processing, Segmentation, Automated analysis.

¹Faculdade de Engenharia da Universidade do Porto, Rua Dr. Roberto Frias s/n 4200-465 Porto, Portugal
²Escola Superior de Saúde, Rua Dr. António Bernardino de Almeida 400, 4200-072 Porto, Portugal

I. INTRODUCTION

The word dementia has been around for centuries and was broadly used to describe any kind of mental illness, disability or psychological incongruence. It was only in the beginning of the 20th century that this clinical term was narrowed to address the symptomatic cognitive decay to brain deterioration and/or injury [1]. During the same period, a specific disease was first documented, a disease that became the most prevalent form of dementia in the 21st century. AD is a neurodegenerative syndrome that causes neuron death and shrinkage resulting in the gradual atrophy of the brain. Although it is commonly associated with the loss of memory by the non-medical community, there is a plethora of other symptoms that may vary according to the patient and the AD stage, some of them include: agnosia, apraxia, speech impairment, short-term memory loss and mood swings [2].

While the specific triggering mechanism of AD is still unknown, past studies have shown a strong correlation between the alteration of the brain’s anatomy and the presence of the disease. From this information, medical imaging of the brain becomes a very strong contender in aiding AD diagnosis. With the loss of brain cells and the consequential atrophy of the brain, one of the most important aspects to consider is the brain’s volume, usually referred as WBV [3]. This volume is abnormally reduced on AD patients and is easily noticeable on the border of the brain, where it causes the expansion of the indentations called sulci and thinning of the areas between the sulci called the gyri. Furthermore, the expansion of the ventricles and the severe shrinkage of the entorhinal cortex where the hippocampus is located also contribute to the screening of this disease [4][5]. However, this volumetric evaluation process is very time consuming and usually done by manually passing the images through enhancement, segmentation and assembly softwares.

State-of-the-art work done in mid-2016 by R.Zeinali et al. portrayed a automated morphological estimation of different brain component volumes (gray matter, white matter and cerebrospinal fluid) using stereology and dilation approaches which showed promising results in estimating the volume of these individual components [6].

In this sense, the aim of this work was to develop an automated image processing tool capable of reading, filtering, segmenting and giving volumetric information of the WBV for healthy controls using a similar stereological approach. The development of this work will hopefully serve as a foundation for the posterior thesis theme regarding the detection of AD by computer-based algorithms. This was achieved using a combination of programming languages and softwares such as MATLAB, Java and ITK-Snap.

II. METHODOLOGY

A. Dataset characterization

For this study, healthy MR brain images under the Metaimage format were collected and made available by the CASILab at the University of North Carolina at Chapel Hill and were distributed by the MIDAS Data Server at Kitware, Inc. Every image was obtained using a 3 tesla MRI unit and the sequences used were T1 FLASH (Fast low angle shot magnetic resonance imaging). Furthermore, these images were acquired...
at $1 \times 1 \times 1 \text{ mm}^3$ and had a digital resolution of $176 \times 256 \times 176$ pixels in three-dimensional space. A total of 20 images were retrieved and used in this study.

B. Image Pre-processing

The reading of the dataset gathered was done in MATLAB using a toolbox by Dirk-Jan Kroon which allowed the stereological reading which is the sequential reading of 2D image slices according to a third defined axis. In this work, the axial view was used throughout.

By inspecting the images histogram, I was able to manually adjust the several grey tones, having obtained a much better-defined image for processing (Fig. 1). To this process it was added a simple sharpen filter to further refine the edges.

![Fig. 1- Contrast treatment differences. On the left is the original axial slice and, on the right, the enhanced contrast treatment](image)

This process narrows down greatly the color spectrum on the image by bringing all the pixel values closer to their borderline values which removes ambiguity and improves filter accuracy when applied to these images.

C. Image filtering

Since the main interest of this work was to determine the WBV, it is of utmost importance that the brain boundaries are well defined, for this, a filtering process that would be able to make the accurate threshold of the brain was prioritized. In this sense, the approach chosen to meet this requirement was the implementation of a Canny edge detector filter.

This method was developed in 1986 by John Canny hence its given name. It was a multi-stage process that could accurately discern structural information from an image while greatly reducing the weight of the non-edge elements, making it a very light and objective method for edge detection [7].

This process can be broken down into five main stages: Gaussian filter application, Intensity gradient determination, Non-maximum suppression (NMS), double thresholding and finally, hysteresis-based edge tracking.

1) Gaussian filter application

The gaussian filter is a widely used filtering technique that follows the well-known gaussian distribution. In this specific application, the gaussian filter was applied to each slice of the brain image individually, hence the usage of the following function:

$$g(x,y) = \frac{1}{2\pi \sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}}$$  \hspace{1cm} (1)

This function illustrates a 2D approach to a gaussian filter where the $x$ and $y$ variable represent the distance from the origin on the horizontal and vertical axis, and the $\sigma$ represents the standard deviation subjacent to the gaussian distribution. Functionally, this process transforms pixel values according to the weighted averaged of neighbor pixels resulting in the visual smoothing and blurring of the image [8].

2) Finding the intensity gradient of the image

After the application of the gaussian filter, the canny method utilizes four different filters for edge detection in multiple orientations. This is achieved by the creation of a 2D mask by a kernel-based convolution process that uses derivatives as approximation parameters. The following equations represent an example described by John Canny of a 2D mask [7]:

$$G = \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$  \hspace{1cm} (2)

$$G_n = \frac{\partial g}{\partial n} = n \cdot \nabla G.$$  \hspace{1cm} (3)

Where the $G$ represents the gaussian function, the $x$ and $y$ the distances to the origin in their respective axis, $\sigma$ is once more the standard deviation and finally $n$ is the direction selected for the detection.

3) NMS

After the aforementioned process the edges traced are still quite ambiguous as the resulting blur from the gaussian filter tends to overestimate the edges thickness. In this sense, using NMS can improve this estimate. The NMS role is to suppress the gradient pixels that do not share the maximum intensity value. In practical terms, this is essentially an edge thinning process [7].

4) Double thresholding

This process is a continuation of the NMS process in the sense that its role is to suppress pixels to improve edge sharpness and accuracy. In this step, this is done by establishing low and high threshold values. These thresholds aid in the further filtering of the detected edges, as if a pixel gradient is above the high threshold, it will be incorporated and if a pixel is below the low threshold, it will be suppressed instead. These thresholds can be tinkered with to alter the edge’s roughness and overall thickness depending on the desired application [7].
5) **Hysteresis-based edge tracking**

The last step of the Canny method encompasses a hysteresis-based edge tracking. This process addresses pixels that fall between both threshold values as they produce uncertainty as whether they should be considered or not. Edge tracking by hysteresis helps clear those ambiguities by checking their connections with the already well defined strong edges. If this connection is verified, these edges are reincluded in the final image [7].

![Fig. 2 - Image filtering by Canny's Method](image.png)

This process was then applied to the 2D slices, resulting in images such as the example demonstrated in Fig. 4. To guarantee the continuity of the brain threshold in all the slices, and additional bridge filter was applied that would connect edges of the same value.

**D. Segmentation**

After the implementation of the canny filter, there are only 2 pixel intensity values present on the image generated: 0 representing the color black and 250 representing the white edges. With this now simplistic information on the image, segmentation can be done by incrementing all the black pixels within the brain mass. To achieve this, a region growth segmentation algorithm developed by Dirk-Jan Kroon was used. Region-growth algorithms require manually inputted points to act as reference values for the segmentation. These points which are user selected pixels are often referred as seeds. From the seed point, the segmentation will spread to all the neighboring pixels of the same value and consequently these neighboring pixels to their own neighborhood. This will create a wave effect which encompasses all the pixels of the same value within a limited area, in this case within the brain boundaries [9]. To establish the best seeding coordinates, a three-dimensional approach visualization was conducted in the software ITK-Snap to determine the most homogeneous region of the brain and the best position to cover the total brain’s length. Furthermore, the seeding location was specifically chosen in order to comprehend only the brain region so that volumetric analysis can be performed. The seeding coordinates chosen based on these parameters were (126, 112) in the digital space which corresponds to the midpoint of the left hemisphere of the brain (Fig. 5), next to the left ventricle’s location since the seeding over the ventricle would yield undesired segmentations. An example of the segmentation method used can be visualized in Fig. 5 being the white region the segmented region of interest.

![Fig. 3- Segmented brain slice by Region Growth method and the respective seed point (in red)](image.png)

**E. WBV determination**

Upon the completion of all the previous steps, the developed script stores all the slices and their pixel information regarding the segmented brain. To calculate the WBV using this information, an incremental algorithm was developed in MATLAB that operates by checking every pixel value in each slice and counting the ones that have an intensity value corresponding to the color white. The summation of all the white pixels from all the slices in the segmentation would hopefully correspond to the brain volume in pixels.

To convert these digital values to real world measurements, the image resolution scale covered in the dataset characterization was used. Since the images have a 1x1x1 mm\(^3\) resolution, each slice area will be equal to its pixel count times 1 mm\(^2\). Similarly, the WBV will be equal to the sum of all the slice’s areas times 1 mm. The resulting value from this operation was then converted to cm\(^3\) for standardized reading display. This procedure can be mathematically described by the following equations:

\[
V(p) = \sum_{p=155}^{p=55} A(p) \cdot S
\]

\[
A(p) = p \cdot S^2
\]

Where \(p\) is the number of segmented pixels in a given slice, \(A\) is the area according to the segmented pixels, \(S\) is the voxel spacing and \(V\) is the volume according to \(p\).

For comparison purposes, the same images were segmented in the software ITK-snap following a standardized pipeline and the respective WBV readings were stored as controls for the experiment.
F. Application

During the course of this work, several programming languages and applications were used in order to meet the objectives of the subject. Each language and application served a different unique purpose: Image processing, Interfacing and Result validation which will be respectively addressed.

1) MATLAB

MATLAB was the software and language that served as the foundation for this project. All image processing related methods previously described were executed in this platform (reading data, pre-processing, filtering, segmenting and calculating volume). Side operations such as image visualization and histogram analysis were also performed with this tool. Fig. 6 demonstrates the pipeline process of the MATLAB algorithm developed.

2) Java

In Java, an interactive and comprehensible GUI was developed that was directly connected to MATLAB so that the user could run the image processing script from Java by specifying the desired parameters in an intelligible manner. This interface consisted of a small window with access to a file selector which would prompt a navigation window so that the user could easily select the file’s path and load it into the program. The interface also included information on the currently loaded file and two checkboxes that performed different tasks (one would output a slice example of a filtered image and the other would output a slice example of a segmentation and filtered image plus the WBV of the selected file). Furthermore, the output button would initiate the MATLAB processing script and return the variable “Volume” that was then displayed on the lower left corner of the window in mm$^3$. This application can be visualized in Annex 1.

The graphical development of the interface was done using Swing, a GUI widget toolkit designed for Java development.

The connection between Java and MATLAB was done using an API named matlabcontrol.

3) ITK-Snap

Result validation would only be possible against a standardized control. In order to achieve that, ITK-Snap, an ITK image processing derived application was used to measure all the tested brain images. The region of interest was defined according to the pipeline used in the project method described and the segmentation method was similarly by region growth as well. An example of the described method can be seen in Annex 2.

III. RESULTS AND DISCUSSION

Upon the development of the proposed application and its testing, the WBV values obtained were registered in Microsoft Excel and compared against the standardized values measured in ITK-Snap. Statistical analysis of the results followed this procedure and the following table was obtained.

From the calculated WBV, the obtained values oscillated between 836 cm$^3$ and 1187 cm$^3$ while from the standardized control the limits ranged from 1015 cm$^3$ to 1378 cm$^3$ (Table 1). When comparing both these ranges to the mean demographic statistics that are documented for this age and ethnicity sample group (1130 to 1260 cm$^3$ depending on sex [10]), the control
values relate more closely to what the demographic tendency is. There is also a clear trend between all the calculated values as they are always lower than their measure control counterparts. Despite the slight offset of measurements, the identity percentages of between both calculated and control show a consistent positive ratio being the lowest percentage a 72% as seen on subject 18.

Being these results satisfactory and coherent, the value offset between the calculated and control volumes is still very significant (Table 2) as the average difference value is 226 cm³ which possibly suggests the deficient performance of the Canny edge detector filter applied. This can be inferred by the analysis of the filtered slice (Fig. 4). There are multiple small islands on the 2D slice that are considered as edges and therefore become isolated despite belonging to the brain mass. This can cause the loss of small segmentation regions which will not contribute to the volumetric analysis. The occurrence of this phenomenon on several slices adds up the deficit of volume measured resulting in the significant offset verified in the calculated WBV.

It is also important to note that WBV can differ significantly between individuals of the same sex, age group and ethnicity as the brain is a very plastic organ that can be morphologically altered by several pathological and/or innate factors [11]. In this sense, a later approach to the normalization by cranial volume is needed to provide further validation of these results. Yet, considering the verified offset and the brain nuances, the high percentage of identity to the controls (Table 2) suggests that this method was successful in determining the WBV of the retrieved dataset.

### IV. Conclusion

The application developed during the course of this work proved to be successful and consistent in determining the WBV of the acquired dataset. This was built in an intelligible and user-friendly interface in which the user can quickly load brain images and determine their volume. Furthermore, several image processing concepts were consolidated, their programming counterparts integrated and learned throughout using two very strong programming languages: MATLAB and Java.

Future works should aim to correct the offset manifested in this project results by further refining the edge detector filter parameters or by experimentation of a different filtering technique. Normalization by cranial volume should also be a priority to ensure the accuracy and trustworthiness of the WBV calculations. Although this work presented a solid foundation for the thesis theme regarding AD and the WBV as a potential diagnosis parameter, there is still much work to be done in regard to this indicator. One of the aspects being the analysis in 3D space as a whole and not by stereological analysis as this may result in loss of information.

Overall, this presented a very enriching experience where multiple interdisciplinary fields were able to meet in a biomedical application.

### V. References


VI. ANNEXES

[Annex 1] Java interface, developed using Swing and matlabcontrol. The user can freely select an image file which will be stored in memory, apply a filter, segment and obtained the respective volume.

![Image of Java interface]

Volume: 889547.0 mm^3

[Annex 2] ITK-Snap application with a segmented example used in the validation method.