UJA-3DFD: A program to compute the 3D fractal dimension from MRI data

J. Ruiz de Miras a,*, J. Navas b, P. Villoslada c, F.J. Esteban d

a Department of Computer Science, University of Jaén, Campus Las Lagunillas s/n, 23071 Jaén, Spain
b Department of Mathematics, University of Jaén, Jaén, Spain
c Department of Neurology, Hospital Clinic-IDIBAPS, Barcelona, Spain
d Department of Experimental Biology, University of Jaén, Jaén, Spain

ARTICLE INFO

Article history:
Received 20 May 2010
Received in revised form
19 August 2010
Accepted 21 August 2010

Keywords:
Fractal dimension
3D box counting
Magnetic-resonance imaging

ABSTRACT

This work presents a computer program for computing the 3D fractal dimension (3DFD) from magnetic-resonance images of the brain. The program is based on an algorithm that calculates the 3D box counting of the entire volume of the brain, and also of its 3D skeletonization. The validity and accuracy of the software has been confirmed using solids with well-known 3DFD values. The usefulness of the program developed is demonstrated by its successful characterization of several neurodegenerative diseases.

© 2010 Elsevier Ireland Ltd. All rights reserved.

1. Introduction

The characterization and quantification of the morphology of the brain by analyzing the fractal dimension has been an area of growing interest for some years. Most of the work focuses on the 2D analysis of individual images from MRI or SPECT. A review of the main applications of this analysis in 2D can be found in Ref. [1].

The number of works performing 3DFD analysis is much smaller. Almost all use the box-counting technique, although there is some study, as presented in Ref. [2], which is based on mathematical morphology to estimate the amount of area occupied for each white-matter voxel size. A recent and comprehensive review of various existing algorithms for estimating the fractal dimension (1D, 2D, and 3D) and major biomedical studies that use them can be found in Ref. [3].

In Ref. [4] a method for calculating the 3DFD by box counting of a triangular reconstruction of the cerebral cortex is presented. The occupied voxels are determined by triangle-voxel intersections. This technique is useful in the case of the cerebral cortex because it is represented by a surface, instead of the typical representation through a 3D volume that is used for white and gray matter. In Refs. [5,6] a 3DFD on SPECT imaging in Alzheimer's disease is applied. The computation of box counting is made by varying the cut-off value and calculating the number of voxels with radioactivity greater than or equal to this cut-off value. Zhang and his team [7–9], by analyzing the 3DFD, have quantified the cerebral structure of white matter and its degeneration with aging. In this work, the 3DFD is calculated by box counting on the 3D volume comprising the set of MR images segmented to extract the white matter. Additionally, they perform a pseudo3D study on the skeletons calculated for each 2D image. In this case, the pseudo3D box
counting is undertaken by calculating the voxels occupied in each 2D skeleton and making the total sum for all the skeletons.

Another example of a 3DFD study applied in medical imaging is Ref. [10], which performs a three-dimensional reconstruction of neuronal dendrites from electron tomography images in different projections, allowing an estimate of 3DFD by box counting. Also in Refs. [11,12] 3DFD is used to analyze the complexity of the fetal cortical surface. In the first case [11], the 3D version of the box-counting method is implemented on the MatLab 7.1 platform. This same technique has been used in Ref. [13] to demonstrate that the morphological changes of cerebellar structure can be quantified by 3DFD analysis. However, in the second case [12] a variant of the box-counting method is used, called entropy-based information fractal dimension. For the estimation of the value of 3DFD, a triangular reconstruction of the cortical surface is needed.

There are some specific programs to calculate the fractal dimension for certain applications, such as those presented in Refs. [14–16]. These computer programs are valid only for 2D image analysis.

In the rest of the paper, we first describe our algorithm in order to calculate the 3DFD from magnetic-resonance images using the box-counting method. Then we describe the method of skeletonization implemented in our software. Next we show the main modules of our software and the main functionality of its interface. Finally we describe several studies using this software, which has demonstrated the usefulness of 3DFD in the characterization of various neurodegenerative diseases.

2. Computational methods and theory

In this section, we describe how the fractal dimension is calculated in three dimensions from a set of magnetic-resonance images. We first describe how this is done for the entire volume of the brain and later for its skeleton. The method is validated by using solids with well-known fractal dimension value.

2.1. 3D fractal dimension computation from MR images

Our implementation of the algorithm for calculating the 3DFD value is also based on box-counting estimation. Because the boxes are defined in 3D (called voxels), several 2D MR images will be needed to build them. For each case study, all the MR images are stacked to form a 3D matrix. This type of 3D matrix is commonly called a “3D image” or “3D volume” [17]. Each position (x, y, z) of this 3D matrix corresponds to the pixel (x, y) from the MR image z, and stores an integer value between 0 and 255.

For an estimate of the 3DFD value, it is necessary to build several voxelizations from the tissue being studied, varying the voxel size in each voxelization. A voxelization with voxel size l and threshold u (a value between 0 and 255) is constructed as follows:

1. A voxel of size l is considered the cube formed by the l + 1 consecutive pixels in the three directions of the 3D matrix.

Fig. 1 shows an example of constructing two voxels of different sizes within a 3D matrix. This matrix corresponds to 5 images of 5 × 5 pixels each. The highlighted voxel of size l = 4 is classified as GREY because it contains not only pixels with values greater than or equal to the threshold u (blue pixels) but also pixels with values below the threshold u (yellow pixels). The highlighted voxel of size l = 1 is BLACK because all covered pixels have a value greater than or equal to the threshold u.

Fig. 2A shows two voxelizations from a single 3D image consisting of 156 MR images where grey matter has been segmented. Both voxelizations were established for a threshold u = 1, the first with a voxel size l = 2 and the second with a voxel size l = 5. This 3D visualization only shows voxels of the GREY type, like those in contact with the boundary of voxelized tissue. Fig. 2B shows only a row or “slice” of each
Fig. 2 – Voxelizations for voxel sizes 2 and 5. (A) Complete 3D view. (B) Slice view.

voxelization. In this way the classification in BLACK voxels (blue), WHITE voxels (no color) and GREY voxels (green) can be better appreciated.

For each case study and for each selected threshold a set of voxelizations is constructed with voxel sizes from 1 to max_voxel_size (size of the cube enclosing the entire 3D image). For each voxelization of size l, count the number of voxels classified as BLACK, WHITE and GRAY. The set of max_voxel_size × 3 counts for a case study and a particular threshold is its box counting. From these data, the value of 3DFD, for a selected type of voxel (white, grey, black, black + grey), is calculated through a log-log linear regression in which the X-axis represents the inverse of the size of the voxel, l, and the Y-axis represents the box counting for that type of voxel, N(l). The final value for the 3D corresponds to the slope of the linear regression:

$$3DFD = -\frac{\ln N(l)}{\ln l}$$

To adjust the line, the voxel-size range that maximizes the correlation value is selected. Fig. 3 shows an example of calculating 3DFD. For this example, the voxel-size range for adjusting of the line is the set of green points. The red dots correspond to the box-counting voxel sizes that have not been selected for the final calculation. In the example the box counting is the sum of the voxels of types GREY and BLACK.

The validity and accuracy of the software has been contrasted using three solids with well-known 3DFD values: a cube of size 320 × 320 × 320 pixels (dimension 3), a sphere of diameter 320 pixels (dimension 3) and the fractal called “Menger sponge” of size 320 × 320 × 320 pixels (fractal dimension 2.7268). Fig. 4 shows these three solids.

Fig. 3 – Linear regression for 3DFD computation from box-counting data.
Fig. 4 – Solids for validation. Cube, sphere, and “Menger sponge”.

Fig. 5 shows the 3DFD results found with our software. These values are: for the cube 2.9876, 2.9428 for the sphere, and 2.725 for the Menger sponge. The sum of BLACK and GRAY voxels has been used in all cases to compute the box counting. These results are very close to the theoretical values of 3, for the cube and sphere, and 2.7268 for the Menger sponge. This fact demonstrates the validity and accuracy of our software to calculate the 3DFD.

2.2. 3D fractal dimension computation from skeletons

Skeletons [18] are an alternative representation for 3D objects. A skeleton is a 1D representation that captures the topological essence of an object in a simple and very compact way. There are several kinds of algorithms to calculate the 3D skeleton [19]: those based on thinning, those based on distance fields, and those based on the Voronoi diagram. The algorithm implemented in our software is based on thinning, since this technique is particularly suitable when using a voxel representation of the object.

The thinning technique is based on iteratively eliminating those boundary voxels of the object that are considered “simple”, i.e. those voxels that can be eliminated without changing the topology of the object. Such voxels can be characterized locally by inspecting only a small voxel neighborhood through matrix masks which are usually of size $3 \times 3 \times 3$. The process thins the object until no more simple voxels can be removed. There are several implementations of this technique, and in our software we have implemented the algorithm described in Ref. [20].

The skeleton is calculated from the pixels in the 3D matrix of each case study where the threshold is greater than or equal to $u$. Fig. 6 shows the skeleton calculated for 3D matrix in Fig. 2.

The 3DFD calculation for the skeleton is performed using the same algorithm described above but starting from the 3D
matrix defined by the skeleton itself. Fig. 7 shows the result of the 3DFD computation for the skeleton shown in Fig. 6.

3. Program description

In this section, we describe the main modules and functionality of the program. We will also show the main aspects of its graphical user interface.

Fig. 8 shows the main modules and data flow of the application. The input to the program is a file with the set of 2D MR images. This file is selected in a dialog box accessible by pressing the button “Open…” in the application (see Fig. 9). A 3D volume is created from these MR slices as described in Section 2. This 3D volume can be displayed as a 3D voxelization by interactively selecting the voxel size and the threshold value (see panel “Parameters” in Fig. 9). The panel “Info” shows the number of voxels classified as BLACK, WHITE and GREY for that voxelization.

The buttons on the panel “Views” allow us to switch between different views provided by the program:

(a) Voxels view (Fig. 9A): It shows the 3D voxelization.
(b) 2D view (Fig. 9B): It shows a 2D slice of the voxelization in the X (axial), Y (coronal) or Z (sagittal) direction selected.
(c) Skeleton view (Fig. 9C): The 3D skeleton is calculated, as described in Section 3, from the pixels of the 3D volume having a value equal to or greater than the threshold parameter. By interactively varying the voxel size, the 3D skeleton voxelization is shown for that voxel size.
(d) Voxel + skeleton view (Fig. 9D): This view shows overlapping 2D slices for both the 3D volume voxelization as the 3D skeleton voxelization.

In each of these views, we can interact with the parameters of voxel size and threshold, giving immediate visual feedback.

When the button “Fractal Dimension” is pressed, the program calculates the box counting for the voxelization or the skeleton (which at that time is on the display). Then the pro-
ogram displays the regression line that best fits the data in the box counting (see Fig. 10). The initial and final voxel size can be modified interactively, allowing the use to visually choose the most linear range of values (green dots in Fig. 10). In the “3DFD parameters” window, we can also choose the type of voxel for which we calculate the box counting (white, grey, black, grey + black or grey + white). The result of calculation is shown in the panel “3DFD Results”, where the values of 3DFD, the error and correlation interactively vary depending on the type of voxel and the range of values selected.

Other modules of the software allow for the calculation in batch mode, over a collection of case studies, of various 3DFD
values for different ranges of voxel sizes and various thresholds. The results of these processes are stored in text files with a format that allows further analysis with standard statistical packages.

4. Hardware and software specifications

The software has been implemented in ANSI C++ using the OpenGL and GLUI cross-platform libraries for the graphical display. The program has been successfully compiled and run both in Windows and Linux platforms (Windows XP SP3 with an Intel Core2 Duo 2.4 GHz CPU and 1GB RAM; and Ubuntu 8.04 Linux with a four AMD Opteron 2GHz CPU and 4 GB RAM). The computation time of the 3DFD value for a typical volume built with 156 images of 157 × 189 pixels is around 5–7 s, depending on the platform used. However, the calculation of the 3D skeleton for a given volume is a process with a much higher computational cost. This computing time can even be a few minutes. Therefore, one of our current efforts is focused on optimizing the algorithms by using parallel programming techniques in GPU (Graphics Processing Units).

5. Availability of the program

UJA-3DFD is available for research activities upon request from the authors.

6. Applications and usefulness of the program

The initial motivation for developing this software was to try to extend for the 3D case (over the entire brain volume) the results of the application of fractal dimension obtained in 2D for the characterization of the white matter of Multiple Sclerosis (MS) patients in a previous work of the authors [21]. After an extensive search, we found no software that would allow the calculation of the 3DFD for our purposes. Once this 3D software was developed for its initial application in MS, and after getting good results, we proposed its application in other neurodegenerative diseases, with the collaboration of specialists in each pathology.

---

**Fig. 9** – Graphical user interface of UJA-3DFD. (A) 3D voxelization view, (B) 2D voxelization slice, (C) 3D skeleton, and (D) 2D slice overlapping the voxelization and skeleton.
So far, two different studies have been performed using our 3D software. In the first one [22], the software was applied to the gray matter of 52 patients with MS and 20 healthy controls. The results of this work indicate that gray matter morphology is abnormal in patients with MS and that this alteration appears early in the course of the disease. We found that patients with MS had an increase in the 3DFD of the GM compared to controls. Such differences in the 3DFD of the GM were found for either CIS (clinically isolated syndrome) or RRMS (relapsing-remitting MS) compared to controls. Finally, we also found a significant difference in the GM 3DFD between CIS and RRMS patients.

In the second study [23], our software has been applied on a sample of 18 singleton premature infants with a prenatal diagnosis of severe IUGR (intrauterine growth restriction, a major cause of preterm delivery, neonatal morbidities, neonatal death, and stillbirth) diagnosed before 34 weeks of gestation, 15 preterm infants matched one-to-one for gestational age (GA, ±2 weeks) at delivery, and 15 neonates born at term. The results indicated a significant decrease of the 3DFD of the brain GM and WM in the IUGR group when compared to the preterm or at term controls.

7. Conclusions

In this paper, we have presented a computer program for calculating the 3DFD from MR images of the brain. Based on a set of 2D slices, the program builds a 3D volume, which represents the entire brain, from which we can calculate its 3DFD. From this 3D volume, it is also possible to calculate the associated 3D skeleton and its fractal dimension. Other computations provided by the program are the fractal dimension of the individual 2D slices, the 2D skeleton and its fractal dimension, and the pseudo3D calculations from all these data. All this processing is done in a user-friendly way through an intuitive graphical user interface. Alternatively, these same calculations can be performed in batch mode over series of case studies and while using various values of the different parameters of the process.

The usefulness of the program developed is demonstrated by its successful characterization of several neurodegenerative diseases [22,23]. In this way, the computer program UJA-3DFD can be used by different specialists working with MRI for studies over different neurodegenerative pathologies that cause changes in the brain structure or morphology.
Acknowledgments

This work was in part supported by grants: University of Jaén (UJA2009/13/04) and Caja Rural de Jaén to J.R.d.M., J.N.U. and F.E.; Spanish Government (TIN2007-67474-C03-03) to J.R.d.M.; and Junta de Andalucía (BIO-302) to F.E.

REFERENCES


