

# A Survey of Segmentation Algorithms Based on ITK

Zhen-Huan Zhou\*, Xiao-Jun Wen

School of Computer Engineering, Shenzhen Polytechnic, Shenzhen 518055, China  
zhouzhenhuan@szpt.edu.cn, wxjun@szpt.edu.cn



Received 1 March 2019; Revised 8 April 2019; Accepted 7 May 2019

**Abstract.** Segmentation of medical images is a challenging task. A myriad of different methods have been proposed and implemented in recent years. In spite of the huge effort invested in this problem, there is no single approach that can generally solve the problem of segmentation for the large variety of image modalities existing today. In this paper, first we introduce the ITK architecture and installation under Windows operating system and configuration using python language. Secondly, we introduce the main image segmentation algorithms of ITK through experiments, including Region Growing (Connected Threshold, Neighborhood Connected, Confidence Connected, Isolated Connected), Segmentation Based on Watersheds (Watershed Filter, Fast Marching Segmentation, Shape Detection Segmentation, Threshold Level Set Segmentation). Finally, we give the calculation process and experimental results.

**Keywords:** image segmentation, ITK, medical image

## 1 Introduction

Medical image segmentation is a complex and critical step in the field of medical image processing and analysis. Its purpose is to segment the part of medical image with some special meanings and extract relevant features, so as to provide reliable basis for clinical diagnosis and treatment and pathological research and assist doctors to make more accurate diagnosis [1-3]. Due to the complexity of medical image itself, a series of problems such as inhomogeneity and individual difference need to be solved in the process of medical image segmentation, so the general image segmentation method is difficult to be directly applied to medical image segmentation. At present, medical image segmentation is still developing from manual segmentation or semi-automatic segmentation to automatic segmentation [4-5]. The Insight Toolkit provides a basic set of algorithms that can be used to develop and customize a full segmentation application. Some of the most commonly used segmentation components are described in the following Table 1.

**Table 1.** The system organization of segmentation in ITK

ITK (Medical Image Processing)	Registration /Segmentation	Common	System Basic Pipeline Data
		Basic Filters	PixelWise Neighborhood Global IO
		Algorithms	Registration Markov RF Level Sets Connectedness
		Numerics	Optimizers Statistics VNL FEM

\* Corresponding Author

ITK can be downloaded from the following web site: <https://www.itk.org/ITK/resources/software.html>. On the web page, choose the tarball that better fits your system. The options are .zip and .tar.gz files. The first type is better suited for Microsoft-Windows, while the second one is the preferred format for UNIX systems. The data is available from the ITK Web site at the following: <https://www.itk.org/ITK/resources/links.html>. Configuring and Building ITK has been solved through the use of CMake, a cross-platform, open-source build system. CMake can be downloaded from <https://cmake.org/download/>. CMake generates native Makefiles or workspaces to be used with the corresponding development environment of your choice. For example, on UNIX and Cygwin systems, CMake generates Makefiles; under Microsoft Windows CMake generates Visual Studio workspaces. You can download binary versions for Microsoft Windows. Alternatively you can download the source code and build CMake on your system. The minimum version of CMake has been evolving along with the version of ITK. For example, we use the versions of the software as follow:

1. Windows 10 Education (x64);
2. cmake-3.9.6-win64-x64;
3. InsightToolkit-4.13.0;
4. Visual Studio 2017;
5. Python2.7.3.

## 2 Installing ITK on Microsoft Windows System

Running CMake to configure and prepare for compilation a new project initially requires two pieces of information: where the source code directory is located, and where the compiled code is to be produced. These are referred to as the *source directory* and the *binary directory* respectively. We recommend setting the binary directory to be different than the source directory in order to produce an *out-of-source* build.

Installation steps as follows:

- (1) Creating ITK source directory on D disk, extracting InsightToolkit-4.13.0 to D:\ITK;
- (2) Creating the binary directory on D disk: D:\ITK\ITK413bin;
- (3) Installing Visual Studio 2015 (2017).
- (4) Running cmake-3.9.6-win64-x64, source file: D:\ITK\InsightToolkit-4.13.0, target file: D:\ITK\ITK413bin, clicking Configure, adding python language option and BUILD\_EXAMPLE, and then Generating (about 30-40 minutes).
- (5) Running itk.sln, right clicking ALL\_BUILD to generate, and right clicking INSTALL to generate. (about 6-10 hours)

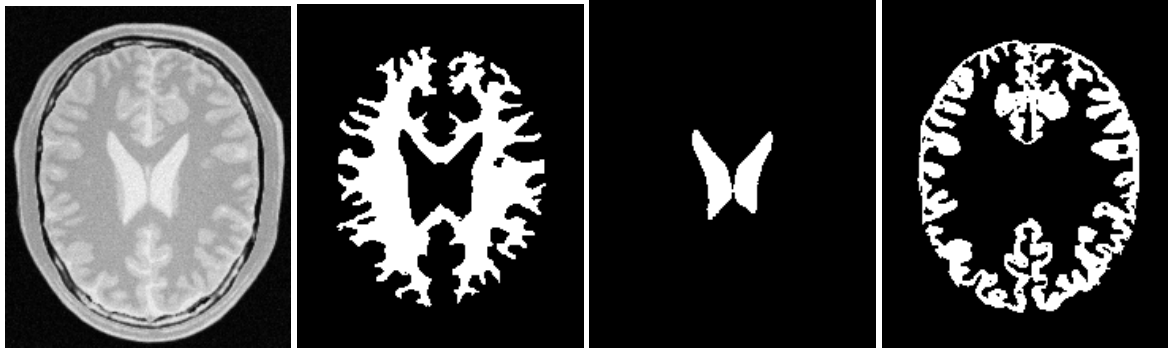
## 3 Region Growing

### 3.1 Connected Threshold

Region growing algorithms have proven to be an effective approach for image segmentation. The basic approach of a region growing algorithm is to start from a seed region (typically one or more pixels) that are considered to be inside the object to be segmented. The pixels neighboring this region are evaluated to determine if they should also be considered part of the object. If so, they are added to the region and the process continues as long as new pixels are added to the region. Region growing algorithms vary depending on the criteria used to decide whether a pixel should be included in the region or not, the type connectivity used to determine neighbors, and the strategy used to visit neighboring pixels [6-8]. We can easily segment the major anatomical structures by providing seeds in the appropriate locations and defining values for the lower and upper thresholds. Fig. 1 illustrates several examples of segmentation. The parameters used are presented in Table 2.

**Table 2.** Parameters used for segmenting some brain structures shown in Fig. 1

Structure	Seed Index	Lower	Upper	Output Image
White matter	(60, 116)	150	180	Second from left in Fig. 1
Ventricle	(81, 112)	210	250	Third from left in Fig. 1
Gray matter	(107, 69)	180	210	Fourth from left in Fig. 1



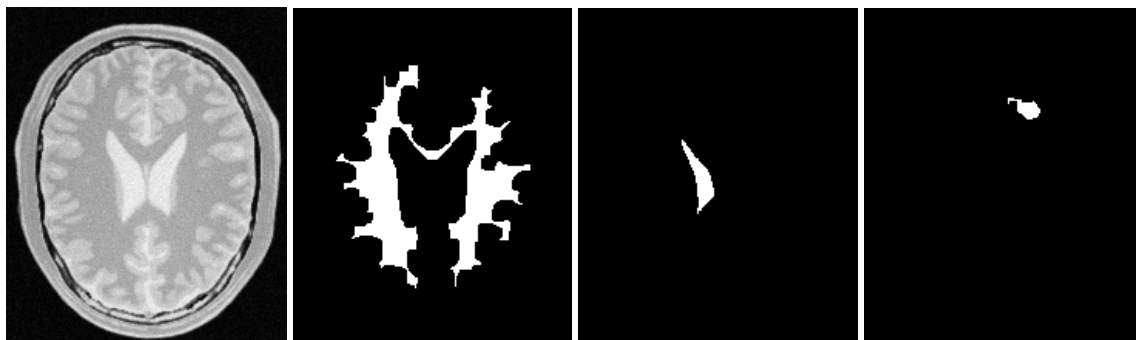
**Fig. 1.** Segmentation results for the Connected Threshold filter for various seed points

### 3.2 Neighborhood Connected

As with the Table 3, several seeds could be provided to the filter by repeatedly calling the neighborhood connected method with different indices. Compare Figures from the Second to the fourth from left in Fig. 2, demonstrating the outputs respectively. It is instructive to adjust the neighborhood radii and observe its effect on the smoothness of segmented object borders, size of the segmented region, and computing time.

**Table 3.** Parameters used for segmenting some brain structures shown in Fig. 2

Structure	Seed Index	Lower	Upper	Output Image
White matter	(60, 116)	150	180	Second from left in Fig. 2
Ventricle	(81, 112)	210	250	Third from left in Fig. 2
Gray matter	(107, 69)	180	210	Fourth from left in Fig. 2



**Fig. 2.** Segmentation results of the NeighborhoodConnectedImageFilter for various seed points

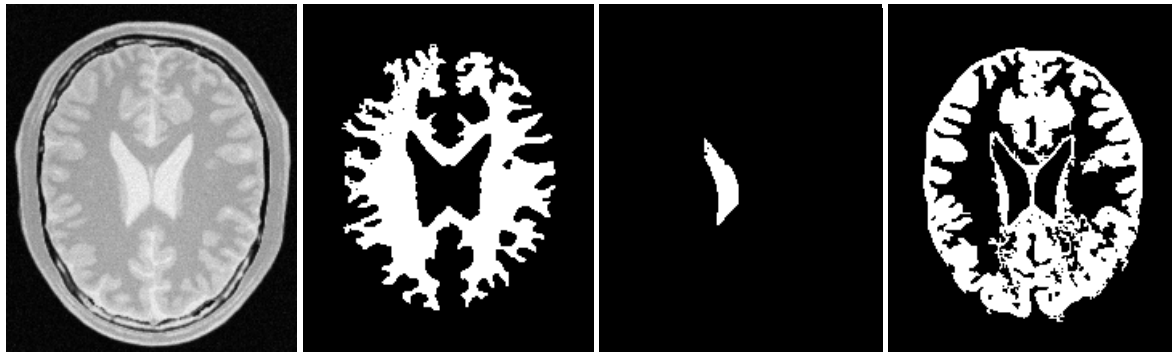
### 3.3 Confidence Connected

The criterion used by the ConfidenceConnectedImageFilter is based on simple statistics of the current region. First, the algorithm computes the mean and standard deviation of intensity values for all the pixels currently included in the region. A user-provided factor is used to multiply the standard deviation and define a range around the mean. Neighbor pixels whose intensity values fall inside the range are accepted and included in the region. When no more neighbor pixels are found that satisfy the criterion, the algorithm is considered to have finished its first iteration. At that point, the mean and standard deviation of the intensity levels are recomputed using all the pixels currently included in the region. This mean and standard deviation defines a new intensity range that is used to visit current region neighbors and evaluate whether their intensity falls inside the range. This iterative process is repeated until no more pixels are added or the maximum number of iterations is reached [9-10].

We can easily segment the major anatomical structures by providing seeds in the appropriate locations in Table 4. Note that the gray matter is not being completely segmented in Fig. 3. This illustrates the vulnerability of the region growing methods when the anatomical structures to be segmented do not have a homogeneous statistical distribution over the image space. You may want to experiment with different numbers of iterations to verify how the accepted region will extend.

**Table 4.** Parameters used for segmenting some brain structures shown in Fig. 3

Structure	Seed Index	Output Image
White matter	(60, 116)	Second from left in Fig. 3
Ventricle	(81, 112)	Third from left in Fig. 3
Gray matter	(107, 69)	Fourth from left in Fig. 3



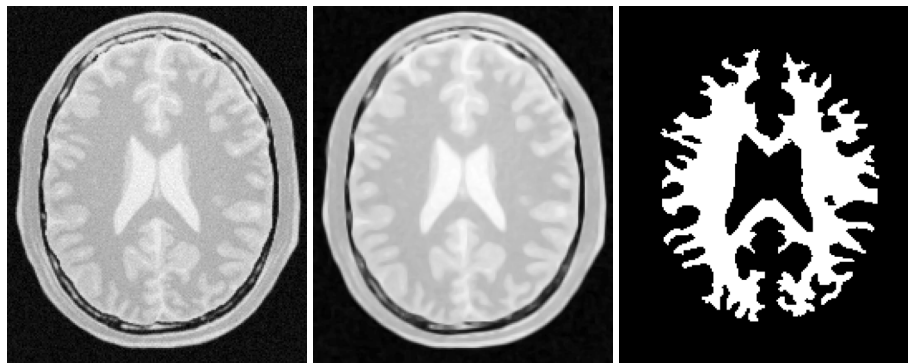
**Fig 3.** Segmentation results for the ConfidenceConnected filter for various seed points

### 3.4 Isolated Connected

This filter is intended to be used in cases where adjacent anatomical structures are difficult to separate. Selecting one seed in one structure and the other seed in the adjacent structure creates the appropriate setup for computing the threshold that will separate both structures. Table 5 presents the parameters used to obtain the images shown in Fig. 4.

**Table 5.** Parameters used for segmenting some brain structures shown in Fig. 4

Adjacent Structures	Seed1	Seed2	Lower	Isolated value found
Gray matter vs White matter	(61, 140)	(63, 43)	150	183.31



**Fig. 4.** Segmentation results of the IsolatedConnectedImageFilter

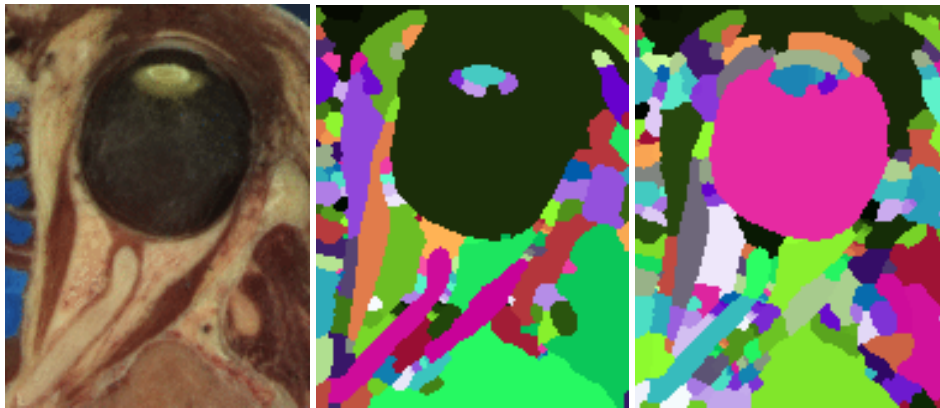
## 4 Segmentation Based on Watersheds

### 4.1 Using the ITK Watershed Filter

Watershed segmentation classifies pixels into regions using gradient descent on image features and analysis of weak points along region boundaries. Imagine water raining onto a landscape topology and flowing with gravity to collect in low basins. The size of those basins will grow with increasing amounts of precipitation until they spill into one another, causing small basins to merge together into larger basins. Regions (catchment basins) are formed by using local geometric structure to associate points in the image domain with local extrema in some feature measurement such as curvature or gradient magnitude. This technique is less sensitive to user-defined thresholds than classic region growing methods, and may be

better suited for fusing different types of features from different data sets. The watersheds technique is also more flexible in that it does not produce a single image segmentation, but rather a hierarchy of segmentations from which a single region or set of regions can be extracted a-priori, using a threshold, or interactively, with the help of a graphical user interface.

In Fig. 5, the input image is taken from the Visible Human female data around the right eye. The images on the right are colorized watershed segmentations with parameters set to capture objects such as the optic nerve and lateral rectus muscles, which can be seen just above and to the left and right of the eyeball. Note that a critical difference between the two segmentations is the mode of the gradient magnitude calculation.

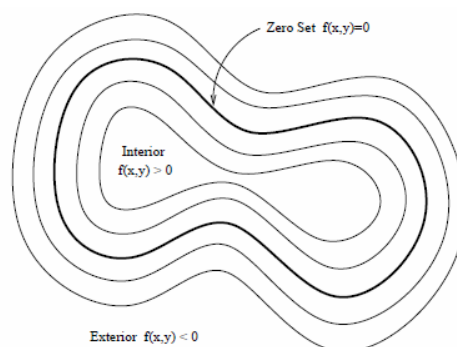


**Fig. 5.** Segmented section of Visible Human female head and neck cryosection data. At left is the original image. The image in the middle was generated with parameters: conductance = 2.0, iterations = 10, threshold = 0.0, level = 0.05, principal components = on. The image on the right was generated with parameters: conductance = 2.0, iterations = 10, threshold = 0.001, level = 0.15, principal components = off

A note on the computational complexity of the watershed algorithm is warranted. Most of the complexity of the ITK implementation lies in generating the hierarchy. Processing times for this stage are non-linear with respect to the number of catchment basins in the initial segmentation. This means that the amount of information contained in an image is more significant than the number of pixels in the image. A very large, but very flat input take less time to segment than a very small, but very detailed input.

#### 4.2 Fast Marching Segmentation

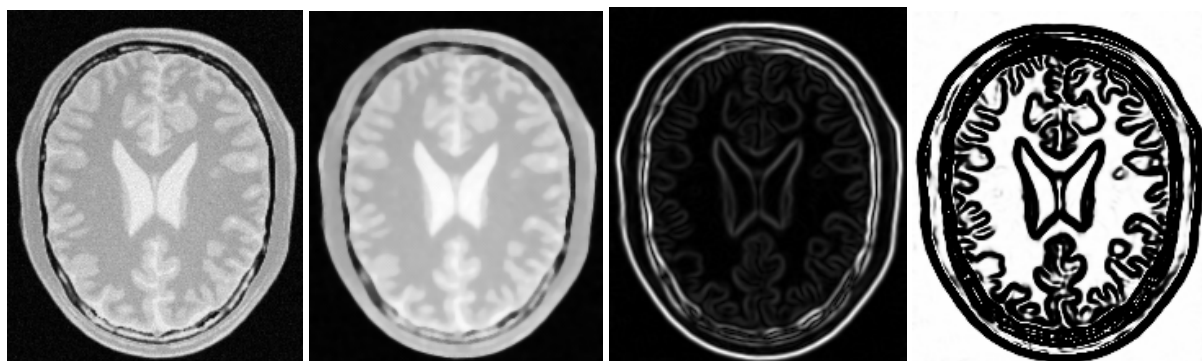
The paradigm of the level set is that it is a numerical method for tracking the evolution of contours and surfaces. Instead of manipulating the contour directly, the contour is embedded as the zero level set of a higher dimensional function called the level-set function,  $\phi(X, t)$ . The level-set function is then evolved under the control of a differential equation. At any time, the evolving contour can be obtained by extracting the zero level-set  $G(X, t) = \{\phi(X, t) = 0\}$  from the output. The main advantages of using level sets is that arbitrarily complex shapes can be modeled and topological changes such as merging and splitting are handled implicitly.



**Fig. 6.** Concept of zero set in a level set

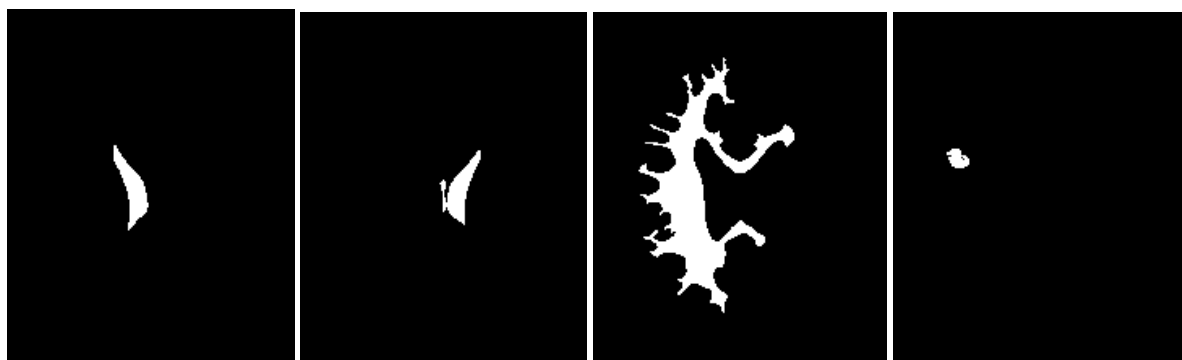
When the differential equation governing the level set evolution has a very simple form, a fast evolution algorithm called fast marching can be used. The `FastMarchingImageFilter` requires the user to provide a seed point from which the contour will expand. The user can actually pass not only one seed point but a set of them. A good set of seed points increases the chances of segmenting a complex object without missing parts. The use of multiple seeds also helps to reduce the amount of time needed by the front to visit a whole object and hence reduces the risk of leaks on the edges of regions visited earlier. For example, when segmenting an elongated object, it is undesirable to place a single seed at one extreme of the object since the front will need a long time to propagate to the other end of the object. Placing several seeds along the axis of the object will probably be the best strategy to ensure that the entire object is captured early in the expansion of the front. One of the important properties of level sets is their natural ability to fuse several fronts implicitly without any extra bookkeeping. The use of multiple seeds takes good advantage of this property.

Fig. 7 presents the intermediate outputs of the pipeline. They are from left to right: the output of the anisotropic diffusion filter, the gradient magnitude of the smoothed image and the sigmoid of the gradient magnitude which is finally used as the speed image for the `FastMarchingImageFilter`.



**Fig. 7.** Images generated by the segmentation process based on the `FastMarchingImageFilter`. From left to right and top to bottom: input image to be segmented, image smoothed with an edge-preserving smoothing filter, gradient magnitude of the smoothed image, sigmoid of the gradient magnitude. This last image, the sigmoid, is used to compute the speed term for the front propagation

Notice that the gray matter is not being completely segmented in Fig. 8. This illustrates the vulnerability of the level set methods when the anatomical structures to be segmented do not occupy extended regions of the image. This is especially true when the width of the structure is comparable to the size of the attenuation bands generated by the gradient filter. A possible workaround for this limitation is to use multiple seeds distributed along the elongated object. However, note that white matter versus gray matter segmentation is not a trivial task, and may require a more elaborate approach than the one used in this basic example.



**Fig. 8.** Images generated by the segmentation process based on the `FastMarchingImageFilter`. From left to right: segmentation of the left ventricle, segmentation of the right ventricle, segmentation of the white matter, attempt of segmentation of the gray matter

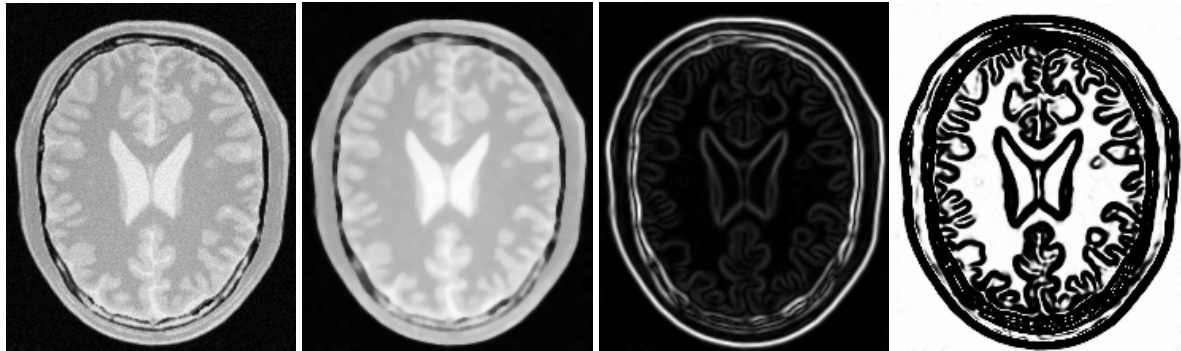
### 4.3 Shape Detection Segmentation

We can easily segment the major anatomical structures by providing seeds in the appropriate locations. Table 6 presents the parameters used for some structures. For all of the examples illustrated in this table, the propagation scaling was set to 1.0, and the curvature scaling set to 0.05.

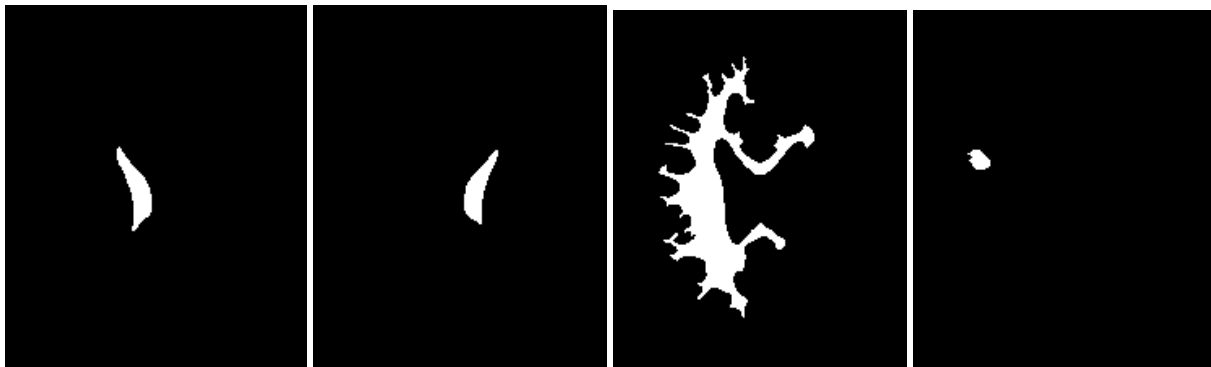
**Table 6.** Parameters used for segmenting some brain structures shown in Fig. 9

Structure	Seed Index	Distance	$\sigma$	$\alpha$	$\beta$	Output Image
Left Ventricle	(81, 114)	5.0	1.0	-0.5	3.0	First in Fig. 10
Right Ventricle	(99, 114)	5.0	1.0	-0.5	3.0	Second in Fig. 10
White matter	(56, 92)	5.0	1.0	-0.3	2.0	Third in Fig. 10
Gray matter	(40, 90)	5.0	0.5	-0.3	2.0	Fourth in Fig. 10

Fig. 9 presents the intermediate outputs of the pipeline. They are from left to right: the output of the anisotropic diffusion filter, the gradient magnitude of the smoothed image and the sigmoid of the gradient magnitude which is finally used as the edge potential for the ShapeDetectionLevelSetImageFilter.



**Fig. 9.** Images generated by the segmentation process based on the ShapeDetectionLevelSetImageFilter. From left to right and top to bottom: input image to be segmented, image smoothed with an edge-preserving smoothing filter, gradient magnitude of the smoothed image, sigmoid of the gradient magnitude. This last image, the sigmoid, is used to compute the speed term for the front propagation

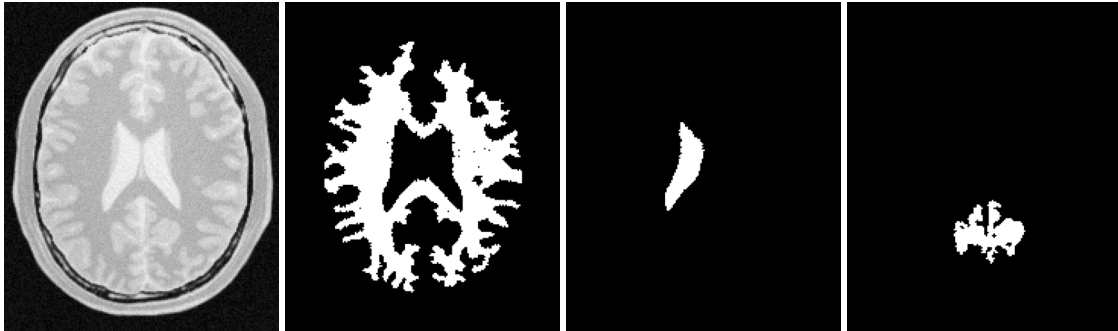


**Fig. 10.** Images generated by the segmentation process based on the FastMarchingImageFilter. From left to right: segmentation of the left ventricle, segmentation of the right ventricle, segmentation of the white matter, attempt of segmentation of the gray matter

### 4.4 Threshold Level Set Segmentation

The Filter is an extension of the threshold connected-component segmentation to the level set framework. The goal is to define a range of intensity values that classify the tissue type of interest and then base the propagation term on the level set equation for that intensity range. Using the level set approach, the smoothness of the evolving surface can be constrained to prevent some of the “leaking” that is common

in connected-component schemes. Compare the results in Fig. 11 with those in Fig. 1. Notice how the smoothness constraint on the surface prevents leakage of the segmentation into both ventricles, but also localizes the segmentation to a smaller portion of the gray matter.



**Fig. 11.** Images generated by the segmentation process based on the `ThresholdSegmentationLevelSetImageFilter`. From left to right: segmentation of the left ventricle, segmentation of the right ventricle, segmentation of the white matter, attempt of segmentation of the gray matter. The parameters used in this segmentations are presented in Table 7

**Table 7.** Segmentation results using the `ThresholdSegmentationLevelSetImageFilter` for various seed points

Structure	Seed Index	Lower	Upper	Output Image
White matter	(60, 116)	150	180	Second from left
Ventricle	(81, 112)	210	250	Third from left
Gray matter	(107, 69)	180	210	Fourth from left

## 5 Conclusions

Medical image is different from general images, it has a special image format and processing methods, in order to pursue speed and real-time, people generally choose C++ programming language, ITK is an open source C++ library of medical image segmentation and registration. But even for a skilled programmer, programming in C++ to process medical images is not an easy task, so ITK provides the python language source code to help beginners get started faster. The Insight Toolkit provides a basic set of algorithms that can be used to develop and customize a full segmentation application. The most effective segmentation algorithms are obtained by carefully customizing combinations of components. The parameters of these components are tuned for the characteristics of the image modality used as input and the features of the anatomical structure to be segmented.

## References

- [1] T.S. Yoo1, Engineering and algorithm design for an image processing API: a technical report on ITK - the insight toolkit, *Studies in Health Technology and Informatics* 85(2002) 586-592.
- [2] V. Chu, G. Hamarneh, MATLAB-ITK Interface for Medical Image Filtering, Segmentation, and Registration, in: *Proc. SPIE* 2006.
- [3] H.J. Johnson, M.M. McCormick, *The ITK Software Guide, Book 2: Design and Functionality Fourth Edition Updated for ITK version 4.13.0*, ITK, New York, 2017.
- [4] M. Saleha, S. Muhammad, M. Afifa, A survey on medical image segmentation, *Current Medical Imaging Reviews* 11(1)(2015) 3-14.



- [5] D.L. Pham, C. Xu, J.L. Prince, Current methods in medical image segmentation, *Annual Review of Biomedical Engineering* 2(2)(2000) 315-337.
- [6] R.T. Whitaker, S.M Pizer, *Geometry-based Image Segmentation Using Anisotropic Diffusion*, Springer Verlag, Heidelberg, 1993.
- [7] J.P. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, Norwell, MA, 1982.
- [8] Y. Zhang, B.J. Matuszewski, L. Shark, Medical image segmentation using new hybrid level-set method, in: *Proc. Fifth International Conference Biomedical Visualization: Information Visualization in Medical and Biomedical Informatics*, 2008.
- [9] R.T. Whitaker, X. Xue, Variable-conductance, level-set curvature for image processing, in: *Proc. International Conference on Image Processing*, 2001.
- [10] J.A. Sethian, *Level Set Methods and Fast Marching Methods*, Cambridge University Press, Cambridge, UK, 1996.