3D INTERACTIVE SEGMENTATION OF BRAIN MRI

Konstantin Levinski, Alexei Sourin and Vitali Zagorodnov

School of Computer Engineering
Nanyang Technological University
Singapore
E-mail: konstantin.levinski@gmail.com, assourin|zvitali@ntu.edu.sg

ABSTRACT

Automatic segmentation of brain MRI data usually leaves some segmentation errors behind that are to be subsequently removed interactively, using computer graphics tools. This interactive removal is normally performed by operating on individual 2D slices. It is very tedious and still leaves some segmentation errors which are not visible on the slices. We have proposed to perform a novel 3D interactive correction of brain segmentation errors introduced by the fully automatic segmentation algorithms. We have developed the tool which is based on 3D semi-automatic propagation algorithm. The paper describes the implementation principles of the proposed tool and illustrates its application.

Keywords: computer graphics, MRI segmentation, image processing

1. INTRODUCTION

Magnetic Resonance Imaging (MRI) is mainly used to visualize the structure and function of the body. Each point on an MRI scan corresponds to a certain point in the body being scanned. Determining which part of MRI scan corresponds to what organ can be problematic. The process of establishing relations between MRI data and their meaning is called *segmentation.*

All segmentation approaches can be classified into two groups: *automatic* and *interactive*.

Automatic segmentation is a well attended area of research. For example, segmentation with a generic brain model is used in [1], with the toolkit presented in [2]. Statistical properties of different areas of the brain are proposed to be used to determine which voxels belong to it in [3]. Graph-cut algorithm, as described in [4], represents MRI as a graph and uses a minimum flow partitioning for segmentation.

Interactive segmentation involves direct guidance by the user during the segmentation process. In [5] and [6] the user controls the segmentation process interactively to obtain correct results.

To detect the border of a certain segment, it is common to define energy related to this surface and minimize that energy [7]. Initial configuration is usually defined interactively by the user, with interactive minimization resulting in operations similar to Adobe Photoshop lasso tool, as it was implemented in [8] and [9]. A complete extension to 3D using surfaces was described in [10], where the interactively defined original surface evolves to the energy minimum.

Interactive methods do not assume any pre-existing segmentation. Hence, they are not suitable for correction of segmentations done by the automatic algorithms. The automatic brain segmentation algorithms, however, are quite robust, and even when they do produce an incorrect segmentation, it is usually easily fixable. Therefore, the most efficient way to segment a large amount of data is to apply an automatic algorithm to the bulk of MRI data and then check and correct the result.

In this paper we propose a 3D visualization method designed for efficient interactive segmentation error discovery and correction. Even though the 2D sections convey all the information without any ambiguity, some artifacts can only be seen in 3D since they do not contribute significantly to each individual 2D slice. We also describe a method to perform such corrections.

2. METHOD

In this section we introduce our visualization method for interactive segmentation. Interactive segmentation places important restrictions on what kind of visualization techniques are required. For example, if interactive segmentation requires the user to have information on the extent of currently segmented area, it is important to provide a comprehensive feedback from the process so that the user does not have to switch between different views to get a complete picture. Hints on where to look for erroneously segmented areas are also important and have to be properly detected and visualized. The focus of the visualization process is on conveying 3D information relevant to the segmentation needs while filtering out unneeded parts.

Automatic segmentation algorithms are quite advanced and usually produce correct results. Even when they do fail, it is often a small problem which could be corrected interactively.

The task of interactive correction of the automatic segmentation results is naturally classified in two sections:

error localization and *error correction*.

Error localization is important as most of the segmentations are correct, and one has to find which ones need to be edited. Current automatic segmentation methods do not provide the users with any clue on where to look for errors.

The proposed method is based on the error estimation of a particular segmented area, using both values from MRI scan and automatically generated 3D surface. The estimation is then used to provide a 3D view of the segmentation so that the user is provided with clues on possible segmentation problems, as shown in Fig. 1. The 3D view also uncovers defects which are difficult to spot using only 2D sections. The error hinting method is also used by the error correction algorithm which does not require a precise input from the user, i.e. the user just has to initiate and monitor the automatic detection process in a potential problematic error area.

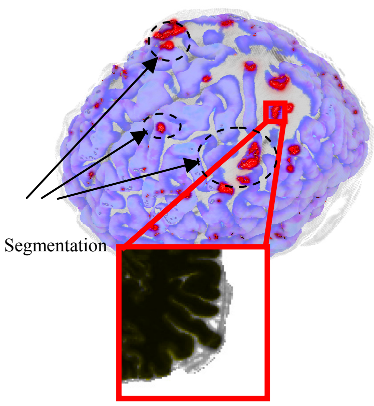

Fig. 1. 3D MRI region and 2D plane section. Erroneous regions are highlighted.

Error correction can be still tedious, and correction of wrong segmentations is different from doing segmentation from scratch. Automatic segmentation algorithms use different criteria for determining how each point of the volume should be classified. Whenever automatic algorithms fail, it means that the chosen criteria were insufficient to distinguish between the brain and non-brain tissues. Therefore, there can be defined additional distinguishing criteria, which, when combined and aided with user interaction, provide us with the correct segmentation.

The probability estimation is based on several *error criteria* each of which deals with a specific aspect of correctness estimation. The criteria are combined by weighted average to produce the resulting estimation.

2.1 Correction criteria

To calculate an error criterion, one has to examine common artifacts produced by the fully automatic algorithm. Usually, incorrect segmentations have similar image intensities. Then, they are unlikely to be located far from the automatically generated surface of the brain. Finally they may be dependent on the intensity changes.

The *Depth* criterion assigns smaller error probability to deeper voxels, as they are less likely to be erroneously segmented.

The *Topology* criterion checks that there are no unconnected parts in the segmentation. There are automatic algorithms which can mark small chunks of dura matter as belonging to the brain. The topology criterion is designed to mark such chunks as erroneous by analyzing the length of the line containing the point.

The *Intensity* criterion uses the user input and the intensity information. It exploits the fact that the most erroneous areas are of similar intensity, as they are usually from the same tissue, e.g., skull, eye, etc.

To allow a user to guide the correction process, it is required to provide efficient feedback mechanisms, in our case visualization methods tailored to displaying and highlighting segmentation errors.

2.2 Visualization

All automatic segmentation errors in skull stripping happen on the generated surface of the brain. There is no point to overwhelm the user by displaying internals of the segmented region. We just take outer voxels and color them according to the respective error criteria, so that the user could determine which part is most likely problematic. If available, white matter surface with segmentation error hints is visualized behind transparent brain surface, as shown in Fig. 1.

In general, it is not always possible to calculate every criterion until the user selects a seed point. It is also not possible to set a seed point until all criteria are known, as there is no information to base the decision upon. We have solved this problem by providing the user with preliminary information, which can help the initial judgment by the user. As the user only sees the surface of the segmented area, it is impractical to use direct intensity information of the surface voxels, as the surface is usually of a uniform intensity. Volume rendering would be redundant, as we only need information on several layers deep. To provide an idea on internal structure without resorting to unnecessary volume rendering and without requesting an input from the user, we propose to color each surface voxel with an average intensity of the surrounding segmented voxels. If there are abnormalities beyond the surface of the segmented area, they will be immediately noticeable as a surface intensity pattern.

Should such averaging be not sufficient, it is also possible to visualize layers of voxels below the surface. By interactively changing the layer, the user can get valuable insights on the structure of the upper layers of the brain. Fig. 2 shows results of the visualization by progressive layer removal.

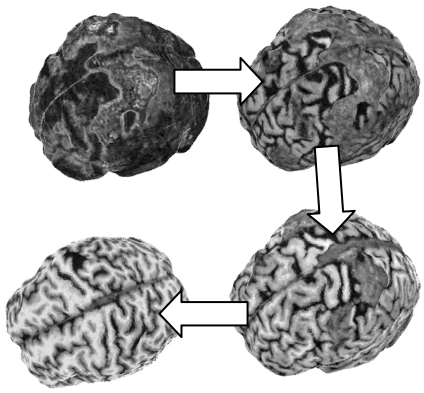

Fig. 2. Progressive interactive layer removal provides information on the outer voxels layout.

Another promising approach for generating hints for incorrectly segmented locations is to use white matter surface and analyze MRI values along the normals shown in Fig. 3. From our experience, the users who have tried this feature claimed it to be very useful and generally better and more efficient than scanning every slice for possible defects.

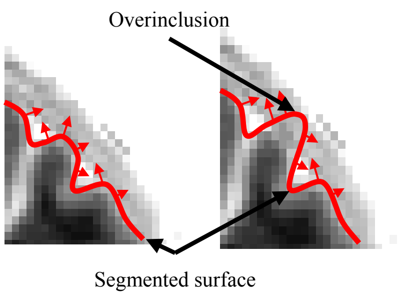

Fig. 3. Segmentation error hint generation by analyzing normals.

While we avoid volume rendering, the seeds placed by the user can be located beneath the surface of the segmented area. Therefore, it is necessary to provide an ability to make the surface display transparent. Once the user suspects a region to be erroneously segmented, it is required that there must be an easy access to the original 2D MRI data slices for verification.

3. APPLICATION

To construct an application one has to define input data, consider how to arrange the software components and, finally, define how the software would fit into a general workflow.

3.1 Workflow

The interactive segmentation process starts with the result of a fully automatic processing. The correctness is evaluated by an expert. Automatic skull stripping algorithms are tuned to avoid classification of voxels belonging to brain as non-brain, i.e. to avoid false negatives. Therefore, all segmentation errors are essentially non-brain tissues classified as brain.

The segmentation consists of 2 steps: model examination and model correction. A pure 3D display is still insufficient for the conclusive assessment of the segmentation since we only display the surface and the selected voxels. To help the users navigate through the volume, a 2D section display is also provided as shown in Fig. 4. The sections are continuously updated while the cursor is being moved across the volume, so that the user can better understand the internal structure of the volume to apply the interactive operations to it.

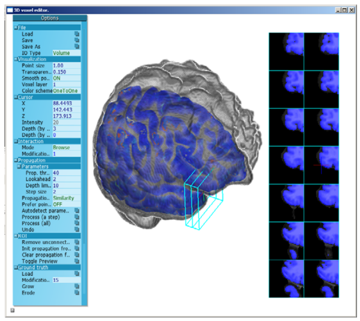

Fig. 4. Application interface

Automatic skull stripping requires a lot of processing power and it runs without supervision. It produces several hundreds images, which should be checked for correctness. The improved workflow of the interactive checking and correcting skull-stripped volumes is organized into the following steps, repeated for every MRI scan produced:

An MRI scan is loaded into the application and the user can see the 3D outer surface of the object colored according to the average intensity of the voxels located close to the surface.

3D Surface generated by the automatic approach is loaded and analyzed to highlight the most probable problem areas.

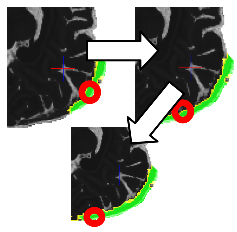

Fig. 5. Interactive 3D control over segmentation process. Circle shows where interactive focus point is located.

The user examines the pattern and scans suspicious areas with the 2D section tool. If the area indeed contains a segmentation error, the user places a seed point there, using either the 3D or the 2D section view.

Once one or the several seed points are selected, the user starts the propagation process, which automatically attempts to select points similar to the initial ones. The automatically selected points are prominently displayed with a different color. The user monitors the process using the 2D section view or a 3D transparent view, and constantly checks that only invalid voxels are selected. At any moment, the propagation can be smoothly reverted. Fig. 5 shows how propagation direction can be interactively controlled by the user.

The automatic process completes when any further propagations select only the valid voxels. The user then removes all the automatically selected voxels and scans for more segmentation errors to correct. If the user realizes that some valid voxels are removed, they can be recovered with the multilevel undo function.

3.2 Performance

Let's consider an average defect spanning about 50 slices. Each slice takes around 10-15 sec to correct, which amounts to around 10-15 minutes per scan. Given 50 erroneous images per batch, it would take more than 10 hours to correct one batch. Our approach requires from the operator on average 2 minutes to locate and remove a similar defect.

Therefore, it provides estimated 5-fold productivity increase for the correction phase. Extending the software to handle different segmentation tasks would save even more time.

In some cases, initial automatic segmentation of white matter has only slight defects which are easier to correct than the mask itself. While we can correct such minor voxel misclassification, it is still necessary to remove non-brain voxels from the mask in order to run WM surface estimation algorithms reliably. We can replace the interactive mask correction process with the correction of the WM+GM segmentation, and then use the segmentation to obtain the mask for the second automatic segmentation run.

4. CONCLUSION

Novel visualization algorithms, developed specifically for segmentation purposes, have been proposed along with a method for 3D interactive correction of brain segmentation errors introduced by the fully automatic segmentation algorithms. 3D visualization of the misclassification hints allows the user to focus attention on the problematic areas and avoid working with separate slices where it is not necessary. We have developed the tool which is based on 3D semi-automatic propagation algorithm. The proposed semi-automatic method uses controlled propagation and allows for efficient correction of the segmentation errors. We have also proposed an efficient method for hinting the user where an error might be. This is done by averaging several layers of the image closest to the surface. This method is simple to implement and provides satisfactory results but it has high failure ratio and has to be replaced with a more robust approach. A video with the demo of the developed tool can be seen at: *http://intune.ntu.edu.sg/SCE/courses/Alexei/Video/segmentation.wmv*.

5. ACKNOWLEDGEMENTS

This project is supported by SBIC Innovative Grant RP C-012/2006 "Improving Measurement Accuracy of Magnetic Resonance Brain Images to Support Change Detection in Large Cohort Studies"

6. REFERENCES

[1] Rohlfing, T. and J.C.R. Maurer, *Multi-classifier framework for atlas-based image segmentation.* Pattern Recognition Letters, 2005. **26**(13): p. 2070-2079.

[2] Bazin, P.-L., et al., *Free software tools for atlas-based volumetric neuroimage analysis.* Medical Imaging 2005: Image Processing, 2005. **5747**: p. 1824-1833.

[3] Ibrahim, M., et al., *Hidden Markov models-based 3D MRI brain segmentation.* Image and Vision Computing, 2006. **24**(10): p. 1065-1079.

[4] Boykov, Y.Y. and M.P. Jolly, *Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images.* 2001. **1**: p. 105-112 vol.1.

[5] Hahn, H.K. and H.-O. Peitgen, *IWT-interactive watershed transform: a hierarchical method for efficient interactive and automated segmentation of multidimensional gray-scale images.* Medical Imaging 2003: Image Processing, 2003. **5032**: p. 643-653.

[6] Armstrong, C.J., B.L. Price, and W.A. Barrett, *Interactive segmentation of image volumes with Live Surface.* Computers & Graphics, 2007. **31**(2): p. 212-229.

[7] Giraldi, G., E. Strauss, and A. Oliveira, *Dual-T-Snakes model for medical imaging segmentation.* Pattern Recognition Letters, 2003. **24**(7): p. 993-1003.

[8] de Bruin, P.W., et al., *Interactive 3D segmentation using connected orthogonal contours.* Computers in Biology and Medicine, 2005. **35**(4): p. 329-346.

[9] Falcao, A.X. and J.K. Udupa, *A 3D generalization of user-steered live-wire segmentation.* Medical Image Analysis, 2000. **4**(4): p. 389-402.

[10] Yushkevich, P.A., et al., *User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability.* NeuroImage, 2006. **31**(3): p. 1116-1128.

[11] Kang, Y., K. Engelke, and W.A. Kalender, *Interactive 3D editing tools for image segmentation.* Medical Image Analysis, 2004. **8**(1): p. 35-46.