Automated Brain Segmentation Algorithm for 3D Magnetic Resonance Brain Images

Jong Geun Park, Taeuk Jeong and Chulhee Lee
Dept. Electrical and Electronic Eng., Yonsei University
134 Shinchon-Dong, Seodaemun-Ku, Seoul 120-749, KOREA
TEL: 82-2-2123-2779 FAX: 82-2-312-4584
Email: chulhee@yonsei.ac.kr

Abstract—In this paper, we propose a new brain segmentation method for 3D magnetic resonance (MR) brain images. The proposed method consists of four steps: background rejection, image normalization, initial slice segmentation, and brain segmentation. In the image normalization step, intensity non-uniformity is removed. In the brain segmentation step, we use mathematical morphological operators and masking. The proposed algorithm was tested with twenty 3D MR normal brain image sets. Experiment results showed the proposed algorithm is fast and provides robust and satisfactory results.

I. INTRODUCTION

Image processing techniques make it possible to extract meaningful information from medical images [1]. Brain segmentation, which segments the brain region from medical brain images, is a part of medical image processing. Since MR images can provide detailed information for various tissues (e.g., white matter, grey matter, skull, meninges, other organs, etc) [2-3], numerous segmentation algorithms have been proposed for brain segmentation of MR brain images. Typical approaches include methods using mathematical morphology [3-7], edge-based methods using active contour model such as the snake and level set [2, 8], histogram-based methods [9], fuzzy-based methods [10], statistic-based methods using HMM (Hidden Markov model) [11], and connectivity-based threshold segmentation methods [1, 12].

Although many brain segmentation algorithms have been proposed, brain segmentation is still a challenging work due to the complexity and large variations of anatomical structures, partial volume effects, intensities non-uniformity, etc [1-4, 13]. In order to overcome these problems and provide robust performance on various T1-weighted 3D MR brain images, we propose a new automatic brain segmentation algorithm using mathematical morphology and masking. Experimental results on 20 normal MR brain images show that the proposed method is reliable and produces consistent results.

II. METHODS

The proposed segmentation algorithm is based on mathematical morphological operators and masking. It is performed on 2D coronal sequential slices. Fig. 2a shows a coronal slice.

In the first step, the ROI (region of interest) is extracted by applying thresholding. The second step is to reduce the variation of voxel intensities to improve robustness. Finally, an initial slice is chosen from a middle coronal image. After applying thresholding, we apply the brain segmentation algorithm and then remove leakage regions caused by similar intensities between the brain and non-brain regions.

A. Background Rejection

Fig. 1 shows a histogram of a MR brain image. It can be seen that the intensities of background voxels are smaller than 10% of the maximum gray level. Therefore, we use the following equation to reject background and extract ROI:

\[
R(i,j,k) = \begin{cases} 
1, & \text{if } f(i,j,k) > 0.1L \\
0, & \text{otherwise} 
\end{cases}
\]

(1)

where \( f(i,j,k) \) is the gray level at position \((j, k)\) in the \(i\)-th 2D coronal slice. \( f(i,j,k) \) can take \(L\) gray level values \([0, L-1]\).

B. Image Normalization

Ideally, the intensity distribution of voxels of the same tissue (white matter, gray matter) would be identical. However, in real MRI acquisition systems, intensity non-uniformity is other factors [2-4]. Fig. 2(a) shows such an image. Therefore, image normalization is desirable for consistent brain segmentation.
First, we compute mean and standard deviation of each slice. The mean \( \mu_i \) and standard deviation \( \sigma_i \) of i-th slice are defined as

\[
\mu_i = \frac{\sum_{j=1}^{J} \sum_{k=1}^{K} f(i,j,k)R(i,j,k)}{\sum_{j=1}^{J} \sum_{k=1}^{K} R(i,j,k)},
\]

\[
\sigma_i = \sqrt{\frac{\sum_{j=1}^{J} \sum_{k=1}^{K} (f(i,j,k) - \mu_i)^2 R(i,j,k)}{\sum_{j=1}^{J} \sum_{k=1}^{K} R(i,j,k)}}.
\]

where \( J \) is the number of slice, \( I \) and \( K \) are the number of column and row of a slice respectively.

Using the mean and standard deviation of (2) and (3), we can apply a linear transform to each slice so that it will have a zero mean and a unit standard variation. After this linear transformation, the intensity distribution of each slice would be similar. However, since the intensities may have low contrast, we apply the normalization process using the global mean \( \mu_g \) and standard deviation \( \sigma_g \). Finally, the normalized image \( f_g(i,j,k) \) is obtained as follow:

\[
f_g(i,j,k) = \frac{\sigma_g}{\sigma_i} (f(i,j,k) - \mu_i) + \mu_g,
\]

where

\[
\mu_g = \frac{\sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} f(i,j,k)R(i,j,k)}{\sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} R(i,j,k)},
\]

\[
\sigma_g = \sqrt{\frac{\sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} (f(i,j,k) - \mu_g)^2 R(i,j,k)}{\sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} R(i,j,k)}}.
\]

Since the intensity non-uniformity may exist in all directions (coronal, axial, and sagittal), the normalization procedure is repeated for all directions. Fig. 2(b) shows the result of image normalization.

C. Initial Slice Segmentation

The proposed algorithm uses mathematical morphological operators and masking. The mask image is the segmented result of the previous slice, and the algorithm is iterated from the initial slice, which is that segmentation is firstly performed among all the 2D slices, toward both sides. We use a middle coronal slice as an initial slice.

In order to segment the initial slice, we perform the procedure from (7) to (10). Firstly, for values between 0.1L and 0.3L, thresholding is applied as follows.

\[
r_i(j,k) = \begin{cases} 1, & \text{if } f_g(i_j, j, k) \geq t \text{ and } R(i, j, k) = 1 \\ 0, & \text{otherwise} \end{cases},
\]

where \( t \) is an integer between 0.1L and 0.3L and \( i_j \) is the index of the initial slice.

Let the set of voxels satisfying \( r_i(j,k) = 1 \) be \( X_i \). After thresholding, the brain region may be still connected to non-brain regions. These erroneous links can be eliminated by the following binary erosion operator:

\[
X_{E,t} = X_i \ominus B,
\]

where \( B \) is a set of the size 2 rhombus structuring element. The eroded set \( \{X_{E,t}\} \) may include several connected components. We assume that the largest component \( \{X_{B,t}\} \) among the connected components is the brain region. Since \( |X_{B,t}| \) is smaller than \( |X_i| \) due to the erosion operator, where \(|.|\) indicates the number of voxels, we restore \( X_{B,t} \) using the following dilation operator:

\[
X_{D,t} = X_{B,t} \oplus B.
\]

Then we define a function \( D(t) \) as follows:

\[
D(t) = |X_i| - |X_{D,t}|.
\]

In general MRI, the brain region has higher intensity than
cerebrospinal fluid (CSF). Therefore, as \( t \) increases, erroneous links between the brain region and non-brain regions are reduced because more non-brain and CSF voxels are removed. Consequently, non-brain regions can be effectively removed by erosion operation. Figs. 3 and 4 show such examples. As can be seen in Fig. 3, when \( t \) is small, most non-brain regions exist. However, as \( t \) increases, non-brain regions are removed at a certain value of \( t \) (Fig. 4) and \( D(t) \) increases also suddenly (Fig. 5). We examine discontinuities and the value of the last discontinuity is used as an optimum threshold value (\( t_o \)). We segment the initial slice using \( t_o \) and (7)-(9). The segmented initial slice is used as a mask image for the adjacent slices.

**D. Segmentation of the Other Slices**

In order to segment the remaining slices, we apply the following thresholding to the entire MR brain images:

\[
f_{b}(i,j,k) = \begin{cases} 1, & \text{if } T_{low} < f_{b}(i,j,k) < T_{high} \\ 0, & \text{otherwise} \end{cases},
\]

where \( T_{low} \) and \( T_{high} \) are defined as follow:

\[
T_{low} = \min_i \{ l | H(l) \geq 0.01H(L-1) \},
\]

\[
T_{high} = \min_i \{ l | H(l) \geq 0.99H(L-1) \},
\]

\[
H(l) = \sum_{m=t_i}^{l} h(m),
\]

where \( h(m) \) is the histogram of the entire MR brain volume. However, there is no threshold value which can perfectly separate the brain region from its neighboring tissues. In order to improve segmentation results, we apply the following procedure to the binary images \( f_{b}(i,j,k) \):

1. **Step 1.** Set \( i = i_1 + 1 \), where \( i_1 \) is the index of the initial slice.
2. **Step 2.** The previous segmented slice (index: \( i-1 \)) is used as a mask image (\( M \)) for the segmentation of the \( i \)-th slice.
3. **Step 3.** Let \( X \) be the \( i \)-th binary image. Apply erosion operator to \( X \), producing \( E_{X} \). After applying AND operation between the mask image and \( E_{X} \), we choose the largest region as the brain region. Then we apply dilation operator to restore the eroded area of the brain region, producing \( D_{X} \).
4. **Step 4.** Since the brain regions of successive two slices are similar, we check whether there are leakage regions in \( D_{X} \). We define the leakage ratio \( (LR) \) as follows:

\[
LR = \frac{|M \cap X_D|}{(|M \cap X_D| + |M^C \cap X_D|)}. \tag{15}
\]

If \( LR \) is close to unity, leakage regions are small. In this paper, we consider two cases. If \( LR<0.85 \), we use case 1. Otherwise, we use case 2.

**Case 1:** \( X_D \) is determined as the brain region of the \( i \)-th slice and used as a mask image for the segmentation of the next slice. Set \( i = i+1 \) and go to step 3.

**Case 2:** \( M^C \cap X_D \) represents the leakage regions. Set \( X_D = X_D \cdot SR \left( M^C \cap X_D \right) \), where \( SR(A) \) is the operator selecting the largest component among connected components in a set \( A \). This operator eliminates the largest leakage region in \( X_D \). Iterate Step 4.

**Fig. 3.** An example of the initial segmentation when \( t \) value is small; (a) \( X_{98} \); (b) \( X_{E,98} \); (c) \( X_{D,98} \).

**Fig. 4.** An example of the initial segmentation when \( t \) value is large; (a) \( X_{190} \); (b) \( X_{E,190} \); (c) \( X_{D,190} \).
By changing \( i = i_j + 1 \) to \( i = i_j - 1 \) in Step 1 and \( i = i + 1 \) to \( i = i - 1 \) in Step 4, we can process the other half slices.

### III. EXPERIMENTAL RESULTS

In order to assess the performance of the proposed algorithm, we tested the algorithm using the 20 normal MR data brain sets provided by the Center for Morphometric Analysis (CMA) at Massachusetts General Hospital [14]. Objective performance assessment is possible with the CMA data sets because manually segmented data sets (ground truth) are also provided. The data sets consist of T1-weighted 2D sequential coronal slices with \( 256 \times 256 \) dimensions. The number of slices of each data set ranges from 60 to 65.

We used Jaccard similarity [15] to measure similarity between the brain regions found by the proposed algorithm \( (S_1) \) and the ground truth \( (S_2) \). Jaccard similarity is computed as follow:

\[
J(S_1, S_2) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}
\]

Also, we used Dice similarity [16] which is defined as

\[
\kappa(S_1, S_2) = \frac{2|S_1 \cap S_2|}{|S_1| + |S_2|}
\]

Using the Jaccard and Dice similarity measures, we assessed brain segmentation performance of the proposed algorithm on each data set. We also compared the proposed algorithm with the brain surface extractor (BSE) which has been widely referred [3]. BSE was tested with its default parameters (diffusion iterations: 3, diffusion constant: 25, edge constant: 0.62, erosion size: 1). However, since the segmented brain regions of data sets 7_8 and 8_4 were not good with the default parameters, data sets 7_8 and 8_4 were tested with parameters (3, 5, 0.75, 1). Table 1 shows that the performance of the proposed algorithm is slightly better than that of BSE. It is noted that there exist significant differences in several data sets. For instance, the proposed algorithm provides inferior performance on several data sets (1_24, 4_8, 7_8, 8_4) which have relatively large intensity variations. We found that parts of the brain region are truncated by excessive thresholding. On the other hand, the proposed method provides noticeable improvement with some data sets (6_10, 13_3, 111_2).

The processing time of the proposed algorithm is only 0.073 seconds per a slice (2.8GHz CPU and 1 Gbytes memory) since the algorithm is based on mathematical morphological operators which are fast and simple. The algorithm was implemented using C language. Furthermore, the algorithm, which does not need any adjustment for a give data set, can be more robust on various data sets. Fig. 6 presents some examples of segmentation results by the proposed algorithm.

**IV. CONCLUSIONS**

We proposed a new brain segmentation algorithm based on mathematical morphological operations and masking for T1-weighted MR brain images. The test on the 20 normal MR data brain sets provided by CMA showed that the algorithm is reliable and efficient. The proposed algorithm is fast and robust. However, it still has a problem of losing parts of the brain with a data set which has large intensity variances. Since the lost brain regions were caused by excessive thresholding, future researches on adaptive thresholding techniques would be required.

![Fig. 6. The brain segmentation results by proposed algorithm; (a) Original coronal slice; (b) Segmented coronal slice; (c) Original axial slice; (d) Segmented axial slice; (e) Original sagittal slice; (f) Segmented sagittal slice.](image)
### TABLE I

The brain segmentation performance of proposed algorithm and other algorithms.

<table>
<thead>
<tr>
<th>Data set</th>
<th>Proposed</th>
<th>BSE [3]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Jaccard</td>
<td>Dice</td>
</tr>
<tr>
<td>1_24</td>
<td>0.804</td>
<td>0.892</td>
</tr>
<tr>
<td>2_4</td>
<td>0.901</td>
<td>0.949</td>
</tr>
<tr>
<td>4_8</td>
<td>0.771</td>
<td>0.871</td>
</tr>
<tr>
<td>5_8</td>
<td>0.722</td>
<td>0.839</td>
</tr>
<tr>
<td>6_10</td>
<td>0.900</td>
<td>0.948</td>
</tr>
<tr>
<td>7_8</td>
<td>0.728</td>
<td>0.842</td>
</tr>
<tr>
<td>8_4</td>
<td>0.799</td>
<td>0.888</td>
</tr>
<tr>
<td>11_3</td>
<td>0.835</td>
<td>0.910</td>
</tr>
<tr>
<td>12_3</td>
<td>0.896</td>
<td>0.945</td>
</tr>
<tr>
<td>13_3</td>
<td>0.888</td>
<td>0.941</td>
</tr>
<tr>
<td>15_3</td>
<td>0.834</td>
<td>0.910</td>
</tr>
<tr>
<td>16_3</td>
<td>0.855</td>
<td>0.922</td>
</tr>
<tr>
<td>17_3</td>
<td>0.877</td>
<td>0.934</td>
</tr>
<tr>
<td>100_23</td>
<td>0.937</td>
<td>0.967</td>
</tr>
<tr>
<td>110_3</td>
<td>0.917</td>
<td>0.957</td>
</tr>
<tr>
<td>111_2</td>
<td>0.907</td>
<td>0.951</td>
</tr>
<tr>
<td>112_2</td>
<td>0.927</td>
<td>0.962</td>
</tr>
<tr>
<td>191_3</td>
<td>0.926</td>
<td>0.962</td>
</tr>
<tr>
<td>202_3</td>
<td>0.910</td>
<td>0.953</td>
</tr>
<tr>
<td>205_3</td>
<td>0.929</td>
<td>0.963</td>
</tr>
<tr>
<td>Avg.</td>
<td>0.863</td>
<td>0.925</td>
</tr>
<tr>
<td>Std.</td>
<td>0.067</td>
<td>0.04</td>
</tr>
</tbody>
</table>

### REFERENCES


