White Matter, Gray Matter and Cerebrospinal Fluid Segmentation from Brain Magnetic Resonance Imaging using Adaptive U-Net and Local Convolutional Neural Network

Pham The Bao*, Tran Anh Tuan, Tran Anh Tuan (A) , Le Nhi Lam Thuy, Jin Young Kim, João Manuel R. S. Tavares

Pham The Bao is with Department of Computer Science, Sai Gon University, Vietnam

Tran Anh Tuan is with Faculty of Mathematics and Computer Science, University of Science, Vietnam National University, Ho Chi Minh City, Vietnam

Tran Anh Tuan (A) is with Faculty of Mathematics and Computer Science, University of Science, Vietnam National University, Ho Chi Minh City, Vietnam

Le Nhi Lam Thuy is with the Department of Computer Science, Sai Gon University, Vietnam

Jin Young Kim is with Department of Electronic and Computer Engineering, Chonnam National University, South Korea

João Manuel R. S. Tavares is with Instituto de Ciência e Inovação em Engenharia Mecânica e Engenharia Industrial, Departamento de Engenharia Mecânica Faculdade de Engenharia, Universidade do Porto, Portugal

(*Corresponding author. E-mail: ptbao@sgu.edu.vn)

ABSTRACT

According to The World Alzheimer Report 2015, there are 46 million people living with dementia in the world. The diagnosis of diseases helps doctors treating patients better. One of the signs of diseases is related to white matter, grey matter, and cerebrospinal fluid. Therefore, the automatic segmentation of three tissues in brain imaging especially from magnetic resonance imaging (MRI) plays an important role in medical analysis. In this research, we proposed an effective approach to segment automatically these tissues in three-dimensional (3D) brain MRI. First, a deep learning model is used to segment the sure and unsure regions. In the unsure region, another deep learning model is used to classify each pixel. In the experiments, an adaptive U-Net model, is used to segment the sure and unsure regions, and the Local Convolutional Neural Network (CNN) model with multiple inputs is used to classify each pixel only in the unsure region. Our method was evaluated with a real image database, Internet Brain Segmentation Repository database, with 18 persons (IBSR 18) (https://www.nitrc.org/projects/ibsr) and compared with state of art methods being the results very promising.

Keywords:

Medical imaging, image segmentation, deep learning, Brain Magnetic Resonance Imaging segmentation, Convolutional Neural Network, Adaptive U-Net, sure and unsure regions, Local Convolutional Neural Network

1. INTRODUCTION

Nowadays, medical imaging is used for clinical analysis, medical follow-up, and intervention. Many diseases related to the nervous system are commonly diagnosed based on magnetic resonance imaging (MRI) because this imaging modality shows better representations concerning soft tissues than others, mainly in the brain [1-2]. Hence, to assist doctors in diagnosing patients effectively, applications that segment accurately and automatically soft tissues in MRI are highly demanded.

Many brain tissues are important for diagnosing brain diseases such as White Matter (WM), Gray Matter (GM), Cerebrospinal Fluid (CSF), Thalamus, and Amygdala [3]. This article is focused on the segmentation of three of these tissues WM, GM, and CSF [4] because they are related to Alzheimer's and dementia diseases [5-7]. Each MRI has many slices. In Brain MRI T1-weighted, CSF appears dark, WM appears bright, and GM appears gray. Each MRI is 3D and has many 2D slices. The example of three tissues from a slice of MRI T1-weighted is shown in Figure 1.

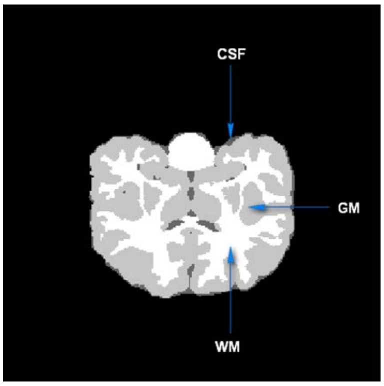

Fig. 1. The example of WM, GM, and CSF tissues in a 2D slice brain MRI where CSF appears dark, WM appears bright, and GM appears gray

Many methods are proposed for WM, GM, and CSF segmentation [8-11]. We categorize current methods into two groups: mathematical and machine learning-based groups. In the mathematical group, there are two well-known tools: Statistical Parametric Mapping (SPM) [12] refers to the analysis of brain imaging data sequences and Functional MRI (fMRI) of the Brain Software Library (FSL) [13] is a comprehensive library of analysis tools for fMRI, MRI, and Diffusion tensor imaging (DTI) brain imaging data. SPM is based on Gaussians, smooth intensity variation, and nonlinear registration with tissue probability maps. On the other hand, FSL segments a 3 dimensional (3D) image of the brain into different tissue types based on a hidden Markov random field model and an associated Expectation-Maximization algorithm. In the machine learning group, in the trends of deep learning, Convolutional Neural Network (CNN) based on a U-shaped network model is used for segmentation [14] or CNN combines with traditional methods to segment three brain tissues [15].

In this article, the segmentation of WM, GM, and CSF from 3D Brain MRI using an Adaptive U-Net model and Local CNN is proposed. The article is organized into four sections. Section 2 presents the proposed method, which is based on a CNN model to segment the sure and unsure regions and after that, uses another CNN model to classify each pixel in the unsure region. Section 3 reports the experiments using the real image database: Internet Brain Segmentation Repository database with 18 persons (IBSR 18). The conclusions are pointed out in the last section.

2. PROPOSED METHOD

The flowchart of our proposed method is shown in Figure 2. The method is composed of two main steps: (1) Using a CNN model to segment the sure and unsure regions; and (2) Classification of each pixel in the unsure region by using another CNN model.

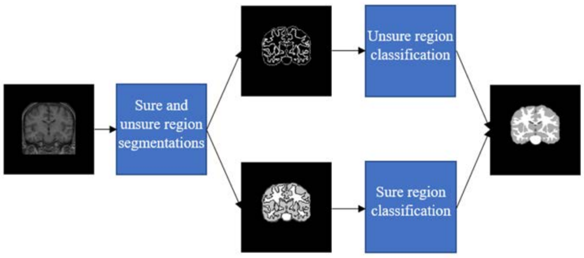
Sure and unsure region segmentations
Unsure region classification
Sure region classification

Fig. 2. The overview of the proposed methodology to segment three brain tissues in 3D MRI

In this study, it is assumed that the sure region is entirely inside the region under analysis. Let I be the original image, T be a region under analysis, then the sure region S is defined as:

$$S = \begin{cases} (S \cap T) \subset T \\ (area(T) - \alpha) \leq area(S) \leq area(T) \end{cases} \quad (1)$$

Where $0 \leq \alpha < area(T)$ and α is a quantity enough to make the $(area(T) - \alpha)$ be the smallest WM area in T region

2.1 Sure and unsure regions segmentation

Deep learning has been used to segment and classify medical images for many years. The most common type of deep learning is CNN [16]. CNN uses layers to transform the input by using filters to get the features for classification automatically. However, recent approaches using CNN for image segmentation show better results than the previous ones [17-18]. Therefore, we propose a module to segment the sure region based on CNN. The developed model contains two important components: the segmentation and the unsure region detection as depicted in Figure 3.

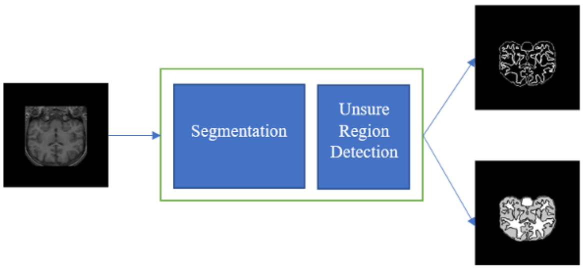

Fig. 3. The proposed method to segment the sure and unsure regions in a brain MRI

There are several methods based on CNN for brain tissue segmentation [14-15, 19-21]. The U-Net [22] is one of the U-shaped network models and it is also a convolutional network architecture for fast and precise segmentation of images. To accomplish the segmentation step in this paper, we developed a model, called Adaptive U-Net, that is based on U-Net and achieves superior segmentation results. The main features of Adaptive U-Net are:

- *The use of multiple kernel size for convolution.*

Let's consider the layer i, and $L = \{K_1(a_1, b_1), \dots, K_n(a_n, b_n)\}$ as the set of output by using n filters K with size *(a, b)*. Hence, the final feature maps are the merge of all feature maps of layer i. For example, the merge as concatenation is $\bigcup_{j=1}^{m} L_j$ where U denote the union of feature maps and m denote the number of feature maps

- *The use of max-pooling and convolution with stride for down-sampling.*

Let $M_i(m, n)$ be the output max-pooling of layer i over non-overlapping rectangular regions of size *(m, n)*, and $K_{i,(m,n)}(a, b)$ be the output feature map of layer i by using convolution kernel with size *(a, b)* and stride *(m, n)*. Then, the total number of feature maps that layer i generates is equal to $M_i \cup K_i$.

In this study, the concept adopted in Adaptive U-Net is illustrated in Figure 4. Two different kernel sizes were used in each convolution. On the other hand, each convolution uses separated down-sampling methods. Finally, the output of the layer is the concatenation of the obtained feature maps which are from each previous convolution.

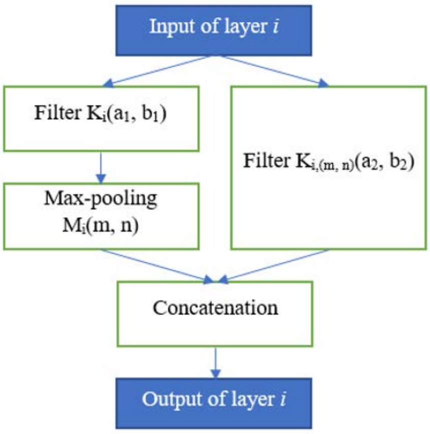

Fig. 4. Combination of using two different kernel sizes in convolution and two down-sampling methods in layer *i*

When using autoencoder as the first component in the unsure region detection, many regions of the resultant segmentation are false because of the down-sampling layers. Despite the merging with the previous layer done, the loss of border pixels and small objects in each convolution still happens. Therefore, an enhancement of the original CNN segmentation was developed to segment the sure and unsure regions, Figure 5.

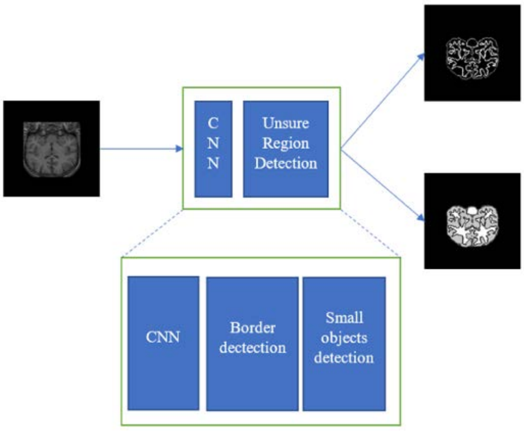

Fig. 5. A proposed method to segment the sure and unsure regions based on the CNN model

With the results of the segmentation from the CNN, $L: \Omega \to \{1 \dots K\}$ the label of each pixel in the image I. Let's consider 2D binary images obtaining each label in L, $B: \Omega \to (0,\ 1)$ is the boundary label of n objects in the images, which is computed based on the surrounding relations among the borders of a binary image [23]. Let $sL: \Omega \to (0,\ 1)$ be the label of n objects in the images:

$$\forall x \in (1 \dots n)\ ,\ sL_x = \begin{cases} 1 \mid \Sigma I_{b,x} > P \\ 0 \mid \Sigma I_{b,x} \leq P \end{cases} \quad (2)$$

where $I_{b,x}$ is the object xth of the binary image I_b obtained from I, and $\Sigma I_{b,x}$ is the sum of all pixel values of the object x^{th} in I_b

The unsure region is the boundary and small objects $(B \cup sL)$, and the sure region is obtained using subtraction operations:

$$\forall (i,j) \in \{\, (x,y) \mid I(x,y) \in I \,\},$$
$$S_{i,j} = \begin{cases} 0 \mid L_{i,j} > 0\ and\ (B \cup sL)_{i,j} > 0 \\ 1 \mid L_{i,j} > 0\ and\ (B \cup sL)_{i,j} = 0 \end{cases} \qquad (3)$$

An example of the sure objects obtained after using the proposed approach is shown in Figure 5, and the proposed sure region segmentation method is given by *Algorithm 1*. With the sure region segmented, the classification of the three desired tissues can be obtained from the CNN which can segment and classify three tissues.

```
Algorithm 1: Segmentation of sure region from a brain 2D MR image
    Input:
I-the 2D MR image
th-contour thickness,default value=1
P-smallest object area, default value=500
K-number of tissues, default value=3
convert2Binary()-Convert to binary
getBorderLabel()-Assign label to border pixel in binary image
getSmallObjects()-Return small object in binary image
    Output:
      S: the sure region
 1.  #using CNN to segment I for K tissues
      $L: \Omega \to \{1,\dots,K\}$
      S: image size(I) with value=0
 2.  for i ∈ {1,…,K} do:
 3.  #convert $i^{th}$ tissues image to binary
      tmp=convert2Binary(L, i)
 4.  #get border $i^{th}$ tissues with thickness=th and assign
      #label=K+1
      B= getBorderLabel(tmp, th, K+1)
 5.  #get small objects from $i^{th}$ tissues with area < P and
      #assign label =K+2
      sL = getSmallObjects (tmp, P, K+2)
 6.  #exclude border and small objects
      tmp [B == (K+1)] = 0
      tmp [sL == (K+2)] = 0
      S[tmp==1]=1
 7.  return S;
```

2.2 Unsure region classification

Deep learning has shown better results in image classification, object detection, and segmentation [24-26]. In this section, we propose using another CNN model to classify each pixel in the unsure region previously obtained. We define a local CNN as follows:

- Input is the window containing pixel prediction in the center
- CNN with multiple windows as input.
- Training only in the unsure region.

In this section, a CNN model with the convolutions is computed between the input and filters, the activation function performs a non-linear transformation, max-pooling layer subsamples the output of the convolutional layer, fully connected layer is the last layer. The output of the last fully connected layer is activated by the soft-max layer to predict k labels. The example of a CNN model for unsure region classification in brain 3D MRI is shown in Figure 6. In Figure 6, four windows with size (69,69) as input are denoted 4@69x69. In the next down-sampling layer, 32 feature maps with size (67,67) are denoted 32@69x69. The last layer has 4 outputs as tissue labels.

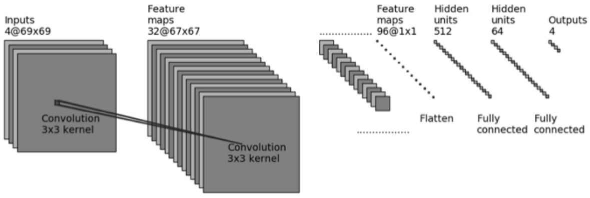

Fig. 6. The CNN model for unsure region classification in brain 3D MRI With 4 windows with size (69,69) as input.

If I_z is the zth image, and U is the mask results of the unsure segmentation from the previous step, the $W_{i,j}$ input windows at position (i, j) is used to predict the label of each pixel as:

$$\forall(i,j) \in \{\,(x,y) \mid U(x,y) > 0\,\},$$
$$W_{i,j} = I[(i-d, j-d);\ (i+d, j+d)] \qquad (4)$$

where d is the half of the sliding window size.

One of the problems related to the input windows W in an input image I is that it does not contain enough information to predict the label of a pixel, especially when a small sliding window is used. Therefore, instead of using one slice as an input image for sliding windows, we proposed using results from the sure region in n previous and next slices of the 3D MRI scan:

$$\forall(i,j) \in \{\,(x,y) \mid U(x,y) > 0\,\},$$

$$W_{i,j} = \begin{bmatrix} U_{z-n}\,[(i-d, j-d);\ (i+d, j+d)] \\ \dots\dots\dots\dots\dots \\ U_z \quad [(i-d, j-d);\ (i+d, j+d)] \\ \dots\dots\dots\dots\dots \\ U_{z+n}\,[(i-d, j-d);\ (i+d, j+d)] \\ I_z\,[(i-d, j-d);\ (i+d, j+d)] \end{bmatrix} \qquad (5)$$

3. EXPERIMENTS AND DISCUSSION

We evaluated the proposed method on the IBSR 18 image database [27], which contains real cases and has been used in many studies regarding tissue brain segmentation in MRI scans. The dataset is composed of 18 T1-w scans with 1.5 mm of slice thickness. Each MRI has a size of 256x128x256. One of the most important characteristics of IBSR 18 is that it provides expert manually segmented results for comparison purposes. The ground truth volumes are divided into four different structures: non-brain, CSF, GM, and WM. Besides, we also compared the accuracy of the proposed method against state-of-the-art methods for brain tissue segmentation [28].

We evaluated the performance of the proposed segmentation method using the Dice Similarity Coefficient [29]:

$$S\,(X, Y) = \frac{2|X \cap Y|}{|X| + |Y|}, \qquad (6)$$

where $|X|$ and $|Y|$ are the number of non-zero voxels in sets X and Y, and $|X \cap Y|$ is the number of non-zero voxels shared by the two sets, respectively.

3.1 Comparison U-Net and Adaptive U-Net

Each brain 3D MRI scan, which is a 3D image of size 256x128x256, was converted to 128 2D images, i.e. slices, with a dimension 256x256. The skull stripping masks were taken from the ground truth included in the dataset. We used Python for algorithm implementation and used the *Keras library* [30] for the two deep neural networks used: U-Net and Adaptive U-Net with input size of *256x256*, *'Adam'* optimizer [31], and *'binary_crossentropy'* loss [32]. Our tests were conducted using a Windows 10 based PC, with 24GB of RAM, and a Geforce GTX980 graphics card. The beginning number of a filter was *32*. On the right side of the model in Figure 8, an upsampling of the feature map which is with a 2x2 convolution concatenate with the correspondingly feature map from the left side. At the final layer, a 1x1 convolution is used with activation '*sigmoid'* to get the number of classes.

In the Adaptive U-Net model, we proposed using multiple kernel sizes to get the local and global features in each layer. We restricted our experiments with two kernel sizes: *(3,3)* and *(5,5)*. The max-pooling was defined with size *(2,2)*. Besides, we used another method for down-sampling layers: convolution with stride *(2,2)*. We run the method with epoch = *100* and use Model Check Point to obtain the best classification model. We run with 3-Fold Validation [33]. In the IBSR 18 dataset, the 18-person scans are numbered from 01 to 18. First, we train 12-persons from number 01 to 12 (1536 2D slices with size 256x256). In the training step, we used validation_split=0.2 of the training sample for validation data. Then, we test 6-persons from 13 to 18 (768 2D slices with size 256x256). Second, we train 12-persons from 01 to 06, 13 to 18 (1536 2D slices with size 256x256). In the training step, we used validation_split=0.2 of the training sample for validation data. Then, we test 6-persons from 07 to 12 (768 2D slices with size 256x256). Third, 12-persons from 07 to 18 are used for training (1536 2D slices with size 256x256). In the training step, we used validation_split=0.2 of the training sample for validation data. Then, we test 6-persons from 01 to 06 are used for testing (768 2D slices with size 256x256) Table 1 shows the summary of the Adaptive U-Net model for segmentation.

Table 1. The summary of the Adaptive U-Net model for segmentation

Layer (type)	Output Shape	Param #	Connected to
input_2 (InputLayer)	[(None, 256, 256, 1)	0	
conv2d (Conv2D)	(None, 256, 256, 32)	320	input_2[0][0]
conv2d_1 (Conv2D)	(None, 256, 256, 32)	9248	conv2d[0][0]
max_pooling2d (MaxPooling2D)	(None, 128, 128, 32)	0	conv2d_1[0][0]
conv2d_2 (Conv2D)	(None, 128, 128, 32)	832	input_2[0][0]
concatenate_4 (Concatenate)	(None, 128, 128, 64)	0	max_pooling2d[0][0] conv2d_2[0][0]
conv2d_3 (Conv2D)	(None, 128, 128, 64)	36928	concatenate_4[0][0]
conv2d_4 (Conv2D)	(None, 128, 128, 64)	36928	conv2d_3[0][0]
max_pooling2d_1 (MaxPooling2D)	(None, 64, 64, 64)	0	conv2d_4[0][0]
conv2d_5 (Conv2D)	(None, 64, 64, 64)	102464	concatenate_4[0][0]
concatenate_5 (Concatenate)	(None, 64, 64, 128)	0	max_pooling2d_1[0][0] conv2d_5[0][0]
conv2d_6 (Conv2D)	(None, 64, 64, 128)	147584	concatenate_5[0][0]
conv2d_7 (Conv2D)	(None, 64, 64, 128)	147584	conv2d_6[0][0]
max_pooling2d_2 (MaxPooling2D)	(None, 32, 32, 128)	0	conv2d_7[0][0]
conv2d_8 (Conv2D)	(None, 32, 32, 128)	409728	concatenate_5[0][0]
concatenate_6 (Concatenate)	(None, 32, 32, 256)	0	max_pooling2d_2[0][0] conv2d_8[0][0]
conv2d_9 (Conv2D)	(None, 32, 32, 256)	590080	concatenate_6[0][0]
conv2d_10 (Conv2D)	(None, 32, 32, 256)	590080	conv2d_9[0][0]
max_pooling2d_3 (MaxPooling2D)	(None, 16, 16, 256)	0	conv2d_10[0][0]
conv2d_11 (Conv2D)	(None, 16, 16, 256)	1638656	concatenate_6[0][0]
concatenate_7 (Concatenate)	(None, 16, 16, 512)	0	max_pooling2d_3[0][0] conv2d_11[0][0]
conv2d_12 (Conv2D)	(None, 16, 16, 512)	2359808	concatenate_7[0][0]
conv2d_13 (Conv2D)	(None, 16, 16, 512)	2359808	conv2d_12[0][0]
conv2d_transpose (Conv2DTranspo	(None, 32, 32, 256)	524544	conv2d_13[0][0]
concatenate_8 (Concatenate)	(None, 32, 32, 512)	0	conv2d_transpose[0][0] conv2d_10[0][0]
conv2d_14 (Conv2D)	(None, 32, 32, 256)	1179904	concatenate_8[0][0]
conv2d_15 (Conv2D)	(None, 32, 32, 256)	590080	conv2d_14[0][0]
conv2d_transpose_1 (Conv2DTrans	(None, 64, 64, 128)	131200	conv2d_15[0][0]
concatenate_9 (Concatenate)	(None, 64, 64, 256)	0	conv2d_transpose_1[0][0] conv2d_7[0][0]

```
conv2d_16 (Conv2D)          (None, 64, 64, 128)  295040   concatenate_9[0][0]

conv2d_17 (Conv2D)          (None, 64, 64, 128)  147584   conv2d_16[0][0]

conv2d_transpose_2
(Conv2DTrans                (None, 128, 128, 64) 32832    conv2d_17[0][0]

concatenate_10                                            conv2d_transpose_2[0][0]
(Concatenate)               (None, 128, 128, 128 0        conv2d_4[0][0]

conv2d_18 (Conv2D)          (None, 128, 128, 64) 73792    concatenate_10[0][0]

conv2d_19 (Conv2D)          (None, 128, 128, 64) 36928    conv2d_18[0][0]

conv2d_transpose_3
(Conv2DTrans                (None, 256, 256, 32) 8224     conv2d_19[0][0]

concatenate_11                                            conv2d_transpose_3[0][0]
(Concatenate)               (None, 256, 256, 64) 0        conv2d_1[0][0]

conv2d_20 (Conv2D)          (None, 256, 256, 32) 18464    concatenate_11[0][0]

conv2d_21 (Conv2D)          (None, 256, 256, 32) 9248     conv2d_20[0][0]

conv2d_22 (Conv2D)          (None, 256, 256, 3)  99       conv2d_21[0][0]
================================================================
Total params: 11,477,987
Trainable params: 11,477,987
Non-trainable params: 0
```

Segmentation results obtained using U-Net and Adaptive U-Net models in terms of the Dice coefficient presented in Table 2 (mean ± standard deviation). In many models, choosing the kernel size (3x3), or (5x5), or (7x7) depends on the problem. Here, we proposed using two popular kernel sizes (3x3) and (5x5). The Concatenation of the output feature maps is the most common one that can give more useful features. In comparison, the Dice coefficient for all tissues segmentation obtained using Adaptive U-Net is better than the one obtained using U-Net. The standard deviation of all tissue segmentation is also better for Adaptive U-Net.

Table 2. Segmentation results obtained using U-Net and Adaptive U-Net models in terms of Dice coefficient (mean ± standard deviation)

Model / Tissue		Train 1 01→12	Train 2 01→06, 13→18	Train 3 07→18	Average
		Test 1 13→18	Test 2 07→12	Test 3 01→06	
U-Net	WM	0.92 ± 0.02	0.93 ± 0.02	0.92 ± 0.01	0.92 ± 0.02
	GM	0.94 ± 0.01	0.88 ± 0.02	0.93 ± 0.03	0.92 ± 0.03
	CSF	**0.80 ± 0.04**	**0.70 ± 0.04**	**0.81 ± 0.06**	**0.77 ± 0.06**
Adaptive U-Net	WM	0.92 ± 0.02	0.93 ± 0.02	0.92 ± 0.01	0.92 ± 0.01
	GM	0.94 ± 0.01	0.88 ± 0.01	0.93 ± 0.02	0.92 ± 0.02
	CSF	**0.81 ± 0.03**	**0.71 ± 0.02**	**0.82 ± 0.05**	**0.78 ± 0.05**

3.2 Comparison of the proposed method with others

After using Adaptive U-Net for three brain tissue segmentation, we applied the proposed method to segment the sure and unsure regions. The unsure region is classified by using CNN for pixel prediction with 17 layers. The input image slices for segmentation were an original image, and the sure segmentation from the original image, previous and next images. The sliding window size was *69*. The optimizer was *'Adam'* and the loss was *'categorical_crossentropy'*. In module *Border Detection*, the border thickness was *1 (one)* and in *Small Object Detection*, the maximum area of small objects was *800* pixels. We run *100* epochs and used model checkpoint to train data. We implement the method on all the pixels in the unsure region and detect that the pixels on the boundary between GM and CSF tissues are classified better

An example of using an original slice combined with the information of two neighbor slices is shown in Figure 7. For each slice, the sure region is segmented first, then the sure region is also calculated in the previous and next slices. The combination of the three sure region segmentation and the original image is used as input for the CNN prediction.

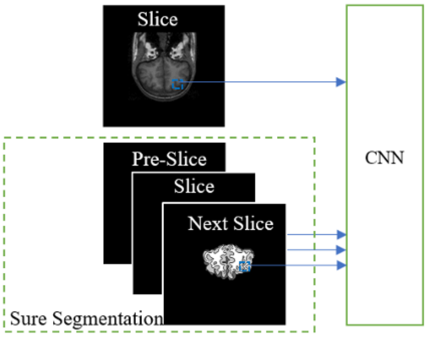

Fig. 7. Sliding windows with four inputs (Original Slice, Previous Slice Sure Segmentation, Slice Sure Segmentation, and Next Slice Sure Segmentation) for pixel classification in the Unsure Region from slice 96 Person 1 dataset IBSR 18.

Table 3 presents the segmentation results obtained using the proposed method on the IBSR 18 dataset. From the segmentation results, one can realize that the Dice coefficient for CSF segmentation and standard deviation for WM segmentation is better because many features from the sure region are extracted from the input. Figure 8 shows the 30th slice when testing IBSR_08, GM region was extracted using U-Net and compared with the ground truth.

Table 3. Segmentation results using the proposed method on the IBSR 18 dataset

Tissue	**Train 1 01→12**	**Train 2 01→06, 13→18**	**Train 3 07→18**	**Average**
	Test 1 13→18	**Test 2 07→12**	**Test 3 01→06**	
WM	0.92 ± 0.01	0.91 ± 0.01	0.92 ± 0.01	0.92 ± 0.01
GM	0.94 ± 0.01	0.90 ± 0.01	0.93 ± 0.01	0.92 ± 0.01
CSF	0.81 ± 0.03	0.77 ± 0.01	0.82 ± 0.04	0.80 ± 0.04

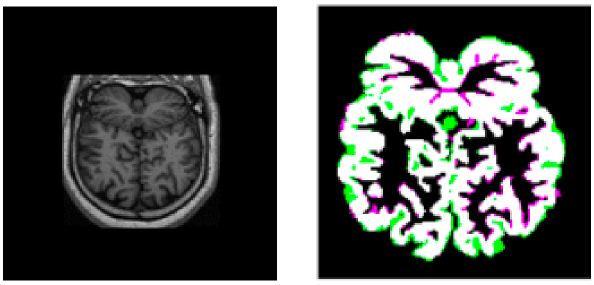

Fig. 8. An example of the results obtained by the proposed method and U-Net for GM segmentation. From the left to right are the 30th slice in testing IBSR_08, the zoom result from our proposed method vs. ground truth. White is the overlapped, blue is the ground truth and purple is the result of our proposed method.

The comparison with the state of art segmentation methods [21] is presented in Table 4. From the comparison results, one can conclude that the deep learning-based approach obtained prominent segmentations in the used dataset.

As to the WM segmentation, the accuracies of the proposed method and U-Net were not better than methods modified directly in the convolution of U-Net such as B-UNet (0.92). With our method, we can classify each pixel in the boundary. However, the feature only extracts from the windows input size (69x69) while with B-UNet, the implementation of Bit-plane in Convolution generates more features than our method. But the drawback of that method is that the feature is useful or not depending on the dataset.

As to the GM segmentation, the Dice coefficient for the proposed method (0.92) was the same as the ones of the other methods because the sliding window did not give more features to classify the wrong voxels in the GM border. CSF segmentation accuracy was better for the proposed method than for the other methods. This finding was because the module *Border Detection and Small Object Detection* detected most of the wrong voxels correctly, which were on the border between GM and CSF tissues. Besides, the CSF region is small and sliding windows size (69x69) can give more useful features. The standard deviation for all tissue segmentation obtained using the proposed methods was also better than the ones obtained by the other methods.

Table 4. Results of the comparison of the proposed method against other methods

	WM	GM	CSF	Average
Proposed method	0.92 ± 0.01	**0.92 ± 0.01**	**0.80 ± 0.04**	0.88
B-UNet[34]	**0.93**	0.92	0.78	0.88
Adaptive U-Net	0.92 ± 0.01	0.92 ± 0.02	0.78 ± 0.05	0.87
U-Net	0.92 ± 0.02	0.92 ± 0.03	0.77 ± 0.06	0.87
CNN-GMM [15]	0.90 ± 0.01	0.91 ± 0.03	0.79 ± 0.03	0.86
State of art [28]:	0.89 ± 0.02	0.91 ± 0.01	0.79 ± 0.08	0.86
- FAST	**0.89 ± 0.02**	0.88 ± 0.01	0.47 ± 0.18	
- SPM5	0.87 ± 0.02	0.89 ± 0.02	**0.79 ± 0.08**	
- SPM8	0.88 ± 0.01	**0.91 ± 0.01**	0.77 ± 0.08	
- GAMIXTURE	0.87 ± 0.02	0.89 ± 0.03	0.52 ± 0.15	
- ANN	0.88 ± 0.03	0.87 ± 0.03	0.52 ± 0.15	
- FCM	0.88 ± 0.03	0.88 ± 0.02	0.52 ± 0.15	
- KNN	0.86 ± 0.03	0.87 ± 0.03	0.46 ± 0.16	
- SVPASEG	0.87 ± 0.02	0.90 ± 0.01	0.57 ± 0.13	
- FANTASM	0.88 ± 0.03	0.88 ± 0.02	0.53 ± 0.15	
- PVC	0.84 ± 0.07	0.83 ± 0.08	0.52 ± 0.15	

4. CONCLUSION

In this article, we proposed an efficient approach to segment brain tissues in 3D MRI. We use an adaptive model based on the U-Net model to segment three tissues from brain images. It is followed by an Unsure Region Detection module to segment the sure and unsure regions. After that, another CNN is used to classify each pixel in the unsure region. The input for this CNN model is an original image, and sure segmentation from the original image, previous and next images. The main contributions are using Adaptive U-Net and using local CNN in unsure region.

The proposed method obtained better results because the original image combined with feature maps from the autoencoder model is used as input for the second CNN model to classify each pixel in the unsure region. In this study, the border detection can obtain incorrect pixels, and the CNN is affected by the size of used sliding windows. However, if we detect the unsure region better by using a suitable threshold in the last layer of U-Net, we can improve the segmentation result.

DATA AVAILABILITY

The data used to support the findings of this study have been deposited in the Internet Brain Segmentation Repository. Please find them in [27] of this paper.

FUNDING

This work was supported by Department of Science and Technology Ho Chi Minh city, Vietnam. The grant number is 43/2019/HD-QPTKHCN

ACKNOWLEDGEMENT

João Manuel R.S. Tavares gratefully acknowledges the funding of Project NORTE-01-0145-FEDER-000022 - SciTech - Science and Technology for Competitive and Sustainable Industries, cofinanced by “Programa Operacional Regional do Norte” (NORTE2020), through “Fundo Europeu de Desenvolvimento Regional” (FEDER).

REFERENCES

1. Luca Saba, (2019) Imaging of the Pelvis, Musculoskeletal System, and Special Applications to CAD, CRC Press, 1st edition, p.

283

2. Y. Lvovsky, P. Jarvis, (2005) Superconducting systems for MRI-present solutions and new trends, IEEE Transactions on Applied Superconductivity, **15**, 1317–1325.
3. Fischl B, Salat DH, Busa E, Albert M, Dieterich M, Haselgrove C, van der Kouwe A, Killiany R, Kennedy D, Klaveness S, Montillo A, Makris N, Rosen B, Dale AM, (2002) Whole brain segmentation: automated labeling of neuroanatomical structures in the human brain, Neuron, **33**, 341-355.
4. Fein G, Di Sclafani V, Tanabe J, Cardenas V, Weiner MW, Jagust WJ, Reed BR, Norman D, Schuff N, Kusdra L, Greenfield T, Chui H, (2000) Hippocampal and cortical atrophy predict dementia in subcortical ischemic vascular disease, Neurology, **55**, 1626-35.
5. Barber R, Scheltens P, Gholkar A, Ballard C, McKeith I, Ince P, Perry R, O'Brien J, (1999) White matter lesions on magnetic resonance imaging in dementia with Lewy bodies, Alzheimer's disease, vascular dementia, and normal aging, *J Neurol Neurosurg Psychiatry*, **67**, 66-72.
6. Gunning-Dixon FM, Raz N, (2000) The cognitive correlates of white matter abnormalities in normal aging: a quantitative review, *Neuropsychology*, **14**, 224-232.
7. Prins ND, van Dijk EJ, den Heijer T, Vermeer SE, Koudstaal PJ, Oudkerk M, Hofman A, Breteler MM, (2004) Cerebral white matter lesions and the risk of dementia, *Arch Neurol*, **61**, 1531-1534
8. Dora L, Agrawal S, Panda R, Abraham A, (2017) State-of-the-Art Methods for Brain Tissue Segmentation: A Review, *IEEE Rev Biomed Eng*, **10**, 235-249.
9. Yu Huang, Lucas C. Parra, (2015) Fully Automated Whole-Head Segmentation with Improved Smoothness and Continuity, with Theory Reviewed, *PLoS ONE*, **10**, e0125477
10. Rashindra Manniesing, Marcel T. H. Oei, Luuk J. Oostveen, Jaime Melendez, Ewoud J. Smit, Bram Platel, Clara I. Sánchez, Frederick J. A. Meijer, Mathias Prokop, Bram van Ginneken, (2017) White Matter and Gray Matter Segmentation in 4D Computed Tomography, *Scientific Reports*, **7**, 119
11. Carlos A. S. J. Gulo, Antonio C. Sementille, João Manuel R. S. Tavares, (2017) Techniques of Medical Image Processing and Analysis accelerated by High-Performance Computing: A Systematic Literature Review, *Journal of Real-Time Image Processing*, **16**, 1–18
12. Karl J Friston, (2007) Statistical Parametric Mapping: The analysis of functional brain images, Amsterdam, Elsevier/Academic Press, p. 81.
13. Smith SM, Jenkinson M, Woolrich MW, Beckmann CF, Behrens TE, Johansen-Berg H, Bannister PR, De Luca M, Drobnjak I, Flitney DE, Niazy RK, Saunders J, Vickers J, Zhang Y, De Stefano N, Brady JM, Matthews PM, (2004) Advances in functional and structural MR image analysis and implementation as FSL, *Neuroimage*, 23, S208-19
14. Christian S. Perone, Evan Calabrese, Julien Cohen-Adad, (2018) “Spinal cord gray matter segmentation using deep dilated convolutions”, *Scientific Reports*, **8**, 5966
15. Duy M. H. Nguyen, Huy T. Vu, Huy Q. Ung, Binh T. Nguyen, (2017) 3D-Brain Segmentation Using Deep Neural Network and Gaussian Mixture Model, *IEEE Winter Conference on Applications of Computer Vision (WACV), USA*
16. Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, (1998) Gradient-based learning applied to document recognition, *Proceedings of the IEEE*, 2278–2324
17. Evan Shelhamer, Jonathan Long, Trevor Darrell, (2017) Fully convolutional networks for semantic segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **39**, 640–651
18. Vijay Badrinarayanan, Alex Kendall, Roberto Cipolla, (2017) SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation, *IEEE Transactions on pattern analysis and machine intelligence*, **39**, 2481-2495
19. Christian S. Perone, Evan Calabrese, Julien Cohen-Adad, (2018) Spinal cord gray matter segmentation using deep dilated convolutions, *Scientific Reports*, **8**, 5966
20. Wang Y, Wang Y, Zhang Z, Xiong Y, Zhang Q, Yuan C, Guo H, (2018) Segmentation of gray matter, white matter, and CSF with fluid and white matter suppression using MP2RAGE, *J Magn Reson Imaging*, **48**, 1540-1550
21. Zeynettin Akkus, Alfiia Galimzianova, Assaf Hoogi, Daniel L. Rubin, Bradley J. Erickson, (2017) Deep Learning for Brain MRI Segmentation: State of the Art and Future Directions, *Journal of Digital Imaging*, **30**, 449–459
22. Ronneberger O., Fischer P., Brox T., (2015) U-Net: Convolutional Networks for Biomedical Image Segmentation, *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 234-241
23. Suzuki S., Abe K., (1985) Topological Structural Analysis of Digitized Binary Images by Border Following, *Computer Vision, Graphics, and Image Processing*, **30**, 32-46
24. Yanming Guo, Yu Liu, Theodoros Georgiou,

Michael S. Lew, (2018) A review of semantic segmentation using deep neural networks, *International Journal of Multimedia Information Retrieval*, **7**, 87-93

25. Rawat W, Wang Z, (2017) Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review, *Neural Comput*, **29**, 2352-2449
26. Litjens G, Kooi T, Bejnordi BE, Setio AAA, Ciompi F, Ghafoorian M, van der Laak JAWM, van Ginneken B, Sánchez CI, (2017) A survey on deep learning in medical image analysis, *Med Image Anal*, **42**, 60-88
27. Frazier, JA, et al., (2007) Internet Brain Segmentation Repository (IBSR) 1.5mm dataset, *Child and Adolescent NeuroDevelopment Initiative*, https://www.nitrc.org/projects/ibsr (accessed 1 December 2020).
28. Valverde S, Oliver A, Cabezas M, Roura E, Lladó X, (2015) Comparison of 10 brain tissue segmentation methods using revisited ibsr annotations, *J Magn Reson Imaging*, **41**, 93-101
29. L. R. Dice, (1945) Measures of the amount of ecologic association between species, *Ecology*, **26**, 297–302
30. Chollet, Francois and others, (2015) Keras. keras.io (accessed 1 December 2020).
31. D.P. Kingma, L.J.Ba, (2015) Adam: a Method for Stochastic Optimization, *International Conference on Learning Representations, San Diego.*
32. Bishop, Christopher M., (2006) Pattern Recognition and Machine Learning, Springer, Verlag New York
33. G. James, D. Witten, T. Hastie, R Tibshirani, (2013) An Introduction to Statistical Learning with Applications in R, Springer, New York, p.181.
34. Tran Anh Tuan, Pham The Bao, Jin Young Kim, João Manuel R. S. Tavares, (2019) White Matter, Gray Matter and Cerebrospinal Fluid Segmentation from Brain 3D MRI Using B-UNET, *VipIMAGE 2019 - Proceedings of the VII ECCOMAS Thematic Conference on Computational Vision and Medical Image Processing*, pp. 188-195