Improved Brain Segmentation Using Pixel Separation and Additional Segmentation Features

Afifa Khaled[1], Chung-Ming Own[1], Wenyuan Tao[1(✉)], and Taher Ahmed Ghaleb[2]

[1] College of Intelligence and Computing, Tianjin University, Tianjin, China
{afifakhaied,chungming.own,taowenyuan}@tju.edu.cn
[2] College of Computing, Queen University, Kingston, Canada
taher.ghaleb@queensu.ca

Abstract. Brain segmentation is key to brain structure evaluation for disease diagnosis and treatment. Much research has been invested to study brain segmentation. However, prior research has not considered separating actual brain pixels from the background of brain images. Not performing such separation may (a) distort brain segmentation models and (b) introduce overhead to the modeling performance. In this paper, we improve the performance of brain segmentation using $3D$, fully Convolutional Neural Network (CNN) models. We use (i) infant and adult datasets, (ii) a multi-instance loss method to separate actual brain pixels from the background and (iii) Gabor filter banks and K-means clustering to provide additional segmentation features. Our model obtains dice coefficients of 87.4%–94.1% (i.e., an improvement of up to 11% to the results of five state-of-the-art models). Unlike prior studies, we consult experts in medical imaging to evaluate our segmentation results. We observe that our results are fairly close to the manual reference. Moreover, we observe that our model is $1.2x$–$2.6x$ faster than prior models. We conclude that our model is more efficient and accurate in practice for both infant and adult brain segmentation.

Keywords: Brain segmentation · Multi-instance loss (MIL) · Gabor filter banks · Convolutional Neural Network (CNN)

1 Introduction

Brain tissues grow rapidly at early stages of human's life. Over the past two decades, brain segmentation has relied on manual segmentation, which is extremely expensive and time consuming [1]. For example, 15–20 images of infant's brain may require 9–11 h to segment. Obtaining accurate tissue segmentation of infant's brain into white matter (WM), gray matter (GM) and cerebrospinal fluid (CSF) is important to (a) measure abnormal early brain development, (b) monitor their progression and (c) evaluate treatment outcomes [2].

X. Wang et al. (Eds.): APWeb-WAIM 2020, LNCS 12318, pp. 85–100, 2020.
https://doi.org/10.1007/978-3-030-60290-1_7

However, due to low contrast and unclear boundaries between WM and GM, it could be difficult to obtain accurate segmentation. Moreover, different experts may produce different segmentation results.

Much research has been invested to perform brain segmentation using automated models, including atlas-based, statistical, and deep learning models. Deep learning models, in particular Convolutional Neural Networks ($CNNs$), have recently been used to perform automated segmentation of infant brain [5]. Previous models have achieved acceptable segmentation performance. However, prior studies have not considered separating actual brain pixels from the background of brain images. Not performing such separation may (a) distort brain segmentation models and (b) introduce overhead to the modeling performance. Therefore, it is important to develop robust models to segment brain regions to improve pathology detection and diseases diagnosis.

In this paper, we improve the performance of brain segmentation using fully CNN models. To this end, we employ (i) a multi-instance loss method to separate actual brain pixels from background and (ii) Gabor filter banks and K-means clustering to provide additional segmentation information to support the machine-learned features. To overcame the lack of medical imaging applications [6], we use full images as input to our model and apply max pooling and mean pooling to process the data. To evaluate our model, we use both infant and adult datasets and measure the performance of our model using dice coefficients. Unlike prior studies, our results are evaluated by the `MICCAI iSEG` organizers (experts in medical imaging) [4]. Our model obtain dice coefficients ranging between 87.4% and 94.1% (i.e., an improvement of up to 11% to the results obtained by five state-of-the-art models). Moreover, our model is $1.2x$–$2.6x$ faster than prior models. Such results indicate that our model is more efficient and accurate in practice for both infant and adult brain segmentation.

The rest of this paper is organized as follows. Section 2 presents prior studies related to brain segmentation. Section 3 presents the methodologies used in our paper. Section 4 presents our experimental results. Section 5 discusses threats to the validity of our results. Finally, Sect. 6 concludes the conclusion and discusses directions for future work.

2 Related Work

This section presents prior studies related to brain segmentation. The main objective of a brain segmentation model is to solve the problem of having low contrast and unclear boundaries between the white matter and the gray matter in brain images. Some prior models for brain segmentation targeted infantile stages [10] (e.g., using multiple modalities [11]), whereas some other targeted early adult (<12 months). Images used by prior models are either $T1$, or $T2$ MRI images.

Dolz et al. [6] proposed $3D$ and fully CNN for subcortical brain structure segmentation. Later on, Bao and Chung [2] have improved the model proposed by Dolz et al. by using a multi-scale structured CNN with label consistency.

Badrinarayanan et al. [7] have also proposed $CNNs$ models with the use of residual connections to segment white matter hyperintensity from $T1$ and flair images. Their models outperformed previous models with an overall dice coefficients of 0.75% on $H95$ and 27.26% on an average surface distance. Fechter et al. [8] also used fully $CNNs$ for brain segmentation. Using five datasets, they obtained dice coefficient rangin between 0.82 and 0.91 for each dataset. Visser et al. [1] proposed CNN models for brain segmentation using a multimodal method and subcortical segmentation. de Brebisson and Montana [9] proposed a random walker approach driven by a $3D$ fully CNN to different tissue classes. Their model was able to segment the esophagus using CT images.

Despite the research invested on brain segmentation, we observe that previous models were trained using images that contain actual brain pixels intermixed with the image background. Therefore, in our work, we propose to separate brain pixels from background to improve the overall performance of brain segmentation. Then, we use fully CNN model and supply it with additional machine-learned features. In summary, our proposed method:

- speeds up model training;
- produces more accurate segmentation results;
- improves information and gradients flow throughout the entire network; and
- reduces the risk of overfitting.

Furthermore, what distinguishes our work from prior work is that our results are evaluated by the `MICCAI iSEG` organizers.

3 Methodology

This section presents the methods that we use to process brain images, extract addition features, and construct brain segmentation models. Figure 1 shows an overview of our proposed network.

3.1 The Proposed CNN Model

In our proposed model, we use two paths where each path has six groups of layers, as follows:

- **The 1^{st} group of layers:** consists of two layers, each of which containing 90 filters. Each filter in a layer is applied to the input images. The outcome of this process is known as a feature map. Feature maps are fed into the second group of layers.
- **The 2^{nd} group of layers:** consists of two layers, each of which contains 120 filters. Our kernel size is $3 \times 3 \times 3$, which allows the network to learn more complex features with a reduced risk of overfitting. Feature maps from the second convolutional layers were up-sampled through a deconvolution layer.
- **The 3^{rd} group of layers:** consists of two convolutional layers, each with 120 filters.

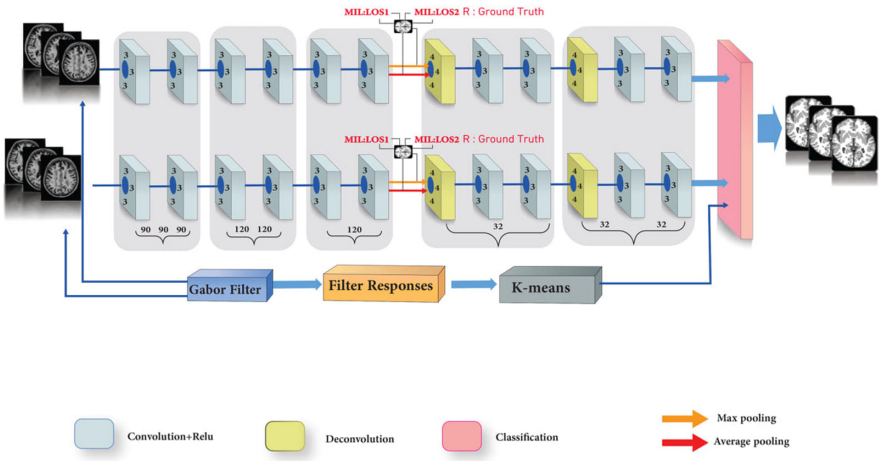
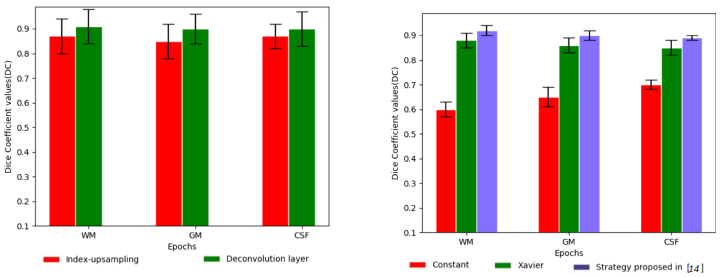


Fig. 1. The proposed fully CNN on multi-instance loss and Gabor filter bank

– **The 4^{th} and 5^{th} groups of layers:** consists of deconvolution layers. Since we employ four classes (i.e., WM, GM, CSF, and background), the last deconvolution layer has four filters (i.e., one filer per class). Convolution layers are used after each deconvolution operation.
– The last layer performs classification with **softmax** units.



(a) (b)

Fig. 2. (a) Two upsampling strategies, (b) Three initialization strategies.

Overfitting is a major problem in deep neural networks. Jonathan et al. [24] reported that deconvolution layers perform upsampling by learning to deconvolve the input feature map. Badrinarayanan et al. [7] reported that index-upsampling uses max pooling indices to upsample feature maps (without learning) and convolves with a bank of trainable filters. We experiment both upsampling strategies

using our data and observe, in Fig. 2 (a), that deconvolution layer performs better than index-upsampling. Therefore, we choose to use the deconvolution layer to upsample the input feature map to higher spatial space. After each convolution layer, we use `PReLU` [13] as an activation function, which (a) introduces a smaller number of extra parameters (equal to the number of channels) and (b) prevents overfitting.

As is the case with deep models, the weights were initialized by random weights. He et al. [14] included *Restricted Boltzmann* machines and showed that the equivalence between RBM's and infinite directed nets with tied weights suggests an efficient learning algorithm for multi-layer networks in which the weights are not tied. Besides, He et al. [14] reported that the deep models can have difficulties to converge and proposed a weights initialization strategy to improve the accuracy of deep neural networks. Figure 2 (b) shows that the initialization strategy proposed by He et al. performs better than two other strategies. Therefore, in our model, we use the initialization strategy proposed by He et al., which employs variant responses in each layer.

A careful selection of a learning rate value can lead to better performance results. However, increasing learning rate makes model training slower due to local optimizations used to update the parameters. To this end, we experiment different learning rates to investigate what suits our data and topology. We start with a learning rate that is taken from a group of comparable models. First, we use multiple runs by changing the learning rate value by alternating factors of 3 or 10 (i.e., 0.01, 0.003, 0.001, 0.0003, and so on). When obtain an acceptable estimate of the sweet spot, where the final digit is tweaked to reach an optimum value. Second, we increase the initial learning rate by a factor of 10 until the model fails to converge to an optimum value. Similarly, we perform experiments to identify the lowest number of epochs needed to train our model. Finally, we initially set the learning rate to 0.01 and then reduce it by a factor of 10 after every 10 epochs.

Dropout and normalization techniques are also used to reduce overfitting in a neural network models and other gradient-related problems [14]. During forward propagation in neural network models, activations are passed from one layer to another. Such activations may not fit a single distribution. In addition, in model training, each layer has to learn a new distribution every time, which slows down the training process (i.e., internal covariate shift). Hence, fixing the distribution of layer inputs eliminates the internal covariate shift and offers faster and better model training. Therefore, in our model, we compute batch mean and batch variance to normalize the inputs/outputs of each layer. In batch normalization, layer outputs are normalized to a fit a single distribution by maintaining a standard deviation of 1 and a mean of 0. Dropout randomly sets the activations of a certain number of neurons (i.e., dropout rate) to 0. That allows neurons to survive and participate in the learning at the next layer. In our model, we applied batch normalization according to the strategy proposed in [13]. We note that we do not preprocess the $T1$ and $T2$ input images.

3.2 Loss Methods

In our proposed model stochastic gradient descent was used with two loss methods:

In our proposed model, we use stochastic gradient descent with two loss methods: (i) Multi-instance loss at the intermediate stages and (ii) Cross-entropy loss at the final stage.

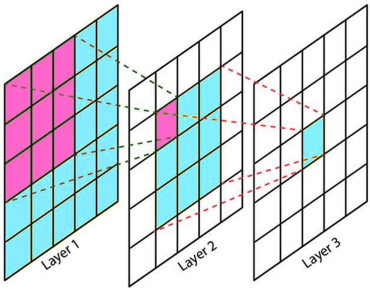


Fig. 3. Pink area marks the receptive field of one pixel in Layer 2. Blue area marks the receptive field of one pixel in Layer 3. (Color figure online)

3.2.1 Multi-Instance Loss

Multi-instance learning is used to describe learning examples in a diverse array. Each learning example contains a bag of instances instead of a single feature vector. Each bag is associated with a label. In the training examples, a single example object has feature vectors (instance). Only one of those feature vectors is responsible for the classification of the object [15]. In traditional supervised learning, the aim was to find a model that predicts the value of a target variable, $y \in \{0, 1\}$, for a given instance, $x \in R^D$). In multi-instance learning, there is a bag of instances instead of a single instance $X = \{x1, ..., x^K\}$ and there is a single label Y associated with the bag, $y^k \in \{0, 1\}$. During the training set, there is no access to the labels as they remain unknown.

Multi-instance loss was inspired by multi-instance learning assumptions. The assumption is that (a) if at least one instance in the bag is positive, then the bag is positive and (b) if all instances in the bag are negative, then the bag is negative. In our model, the third and forth layers can be considered as a multi-instance problem. We use two loss functions: one for positive pixels (i.e., inside the MRI image) and another for negative pixels (i.e., outside the MRI image). The *I-loss* function (inside the MRI image) is given by the following equation:

$$LOS1 = \sum_{\mathrm{i,j}, R_{\mathrm{i,j}}=1} \log(1 + \exp(-R_{\mathrm{i,j}} * H^m_{\mathrm{i,j}})), \tag{1}$$

where $R_{i,j}$ is the ground truth provided by the dataset organizers. $R_{i,j} = 1$ if pixel (i, j) is inside the MRI image. The novelty here is that pixel (i, j) is in $R_{i,j}$ and such pixel has a receptive field. The receptive field refers to a certain part of an image. If the receptive field has at least one positive pixel (inside MRI image), then (i, j) should be positive. Otherwise, (i, j) is negative. Figure 3 depicts the receptive field where pink area marks the receptive field of one pixel in Layer 2, whereas blue area marks the receptive field of one pixel in Layer 3. A bag-level predictive map $H^m_{i,j}$ represents max pooling from feature maps. *Ou-loss* functions (outside the MRI image) is given by the following equation:

$$LOS2 = \sum_{i,j,R_{i,j}=-1} \log(1 + \exp(-R_{i,j} * H^a_{i,j})), \tag{2}$$

where $R_{i,j} = -1$ if pixel (i, j) is outside the MRI image. A bag-level predictive map $H^a_{i,j}$ represents average pooling from feature maps. The total multi-instance loss function is given by the following equation:

$$MIL = LOS1 + \parallel W \parallel_2 LOS2, \tag{3}$$

where the MIL ensures a proper differentiation between actual brain pixels and background. $\parallel W \parallel_2$ presents the weights in the neural network, which is given by the following equation:

$$\parallel W \parallel_2 = \sqrt{W_1^2 + W_2^2 + ... + W_n^2}. \tag{4}$$

3.2.2 Cross-Entropy Loss

Loss functions are crucial in machine learning pipelines. However, knowing which one loss function to use can be challenging. Cross-entropy loss is commonly used as a cost function when training classifiers. Cross-entropy loss is also used to measure the performance of a classification model. In our model, we use the `softmax` function to convert the output of the classification layer into normalized probability values.

3.3 Gabor Filters

Due to low contrast and lack of clear boundaries between WM and GM, features are not sufficient for accurate segmentation. Gabor filter is a strong tool for the description of textures in images. Figure 4 shows the process of obtaining Gabor filters. Gabor filter can be obtained by convolving the image and applied to our model as human-designed features to improve the segmentation results [23]. The equation is given by:

$$G(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp(-((x'^2 + y^2 y'^2)/2\sigma 2)) \exp(i(2\pi x'/y + \psi)), \tag{5}$$

where σ is the standard deviation of Gaussian envelope, ψ is the phase shift, λ is the wavelength of the sinusoid, θ is the spatial orientation of the filter and

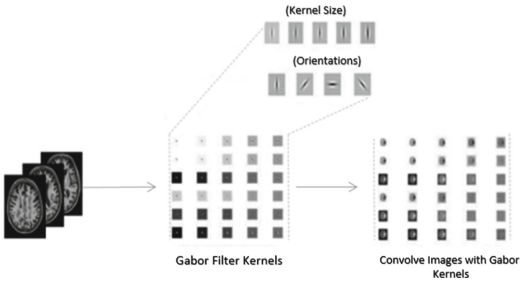


Fig. 4. Gabor filter bank.

γ is the spatial aspect ratio. The terms x' and y' are given by the following equations:

$$x' = xcon(\theta) + ysin(\theta), \tag{6}$$

$$y' = ycos(\theta) - xsin(\theta), \tag{7}$$

where filter sizes are from 0.3 to 1.5 and the wavelength of sinusoid coefficients was 0.8, 1.0, 1.2 and 1.5.

3.4 K-means

K-means is a technique used to split a dataset into k groups. In our model, the filter responses are merged together and apply the K-means clustering algorithm to cluster the pixels with similar features.

4 Experiments

This section presents our experimental design and evaluation.

4.1 Datasets

In our work, we use two different datasets of brain images: the `MICCAI iSEG` dataset and `MRBrains` dataset. We describe each of these datasets in the following.

4.1.1 `MICCAI iSEG` Dataset

The aim of the evaluation framework[1] introduced by the `MICCAI iSEG` organizers is to compare segmentation models of WM, GM and CSF on $T1$ and $T2$. The `MICCAI iSEG` dataset contains 10 images, named subject-1 through

[1] http://iseg2017.web.unc.edu.

subject-10, subject $T1$: $T1$ -weighted image, subject $T2$: $T2$-weighted, and a 'manual segmentation' label used as a training set. The dataset also contains 13 images, named subject-11 through subject-23, used as a testing set. An example of the `MICCAI iSEG` dataset ($T1$, $T2$, and manual reference contour) is shown in Fig. 5. The dataset has two different times (i.e., longitudinal relaxation time and transverse relaxation time), which are used to generate $T1$ and $T2$ (Table 1). The dataset has been interpolated, registered, and skull-removed by the `MICCAI iSEG` organizers. We present the evaluation equations in Subsect. 4.2.

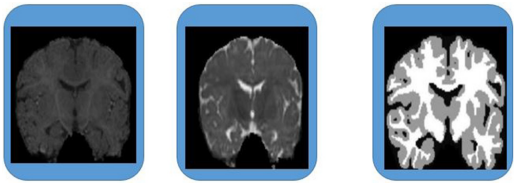


Fig. 5. Example of *MICCAIiSEG* dataset ($T1$, $T2$, manual reference contour)

Table 1. Shows the parameters used to generate $T1$ and $T2$.

Parameters	TR/TE	Flip angle	Resolution
$T1$	1900/4.38 ms	7	$1 \times 1 \times 1$
$T2$	7380/119 ms	150	$1.25 \times 1.25 \times 1.25$

4.1.2 `MRBrains` Dataset

The `MRBrains`[2] dataset contains 20 subjects for adults for segmentation of (a) cortical gray matter, (b) basal ganglia, (c) white matter, (d) white matter lesions, (e) peripheral cerebrospinal fluid, (f) lateral ventricles, (g) cerebellum, and (h) brainstem on $T1$, $T2$, and FLAIR. Five (i.e., 2 male and 3 female) subjects are provided to the training set and 15 subjects are provided for the testing set. On the evaluation of the segmentation, these structures merged into gray matter (a–b), white matter (c–d), and cerebrospinal fluid (e–f). The cerebellum and brainstem were excluded from the evaluation.

4.2 Segmentation Evaluation

To better demonstrate the significance of our model, we submitted our results to be evaluated by the `MICCAI iSEG` organizers [4]. The `MICCAI iSEG` organizers have used three metrics to evaluate our model.

[2] https://mrbrains13.isi.uu.nl/results.php.

4.2.1 Dice Coefficient (*DC*)

V_{ref} is the reference. V_{auto} is automatic segmentations. The DC is given by the following equation:

$$DC(V_{\text{ref}}, V_{\text{auto}}) = \frac{2|V_{\text{ref}} \bigcap V_{\text{auto}}|}{|V_{\text{ref}}| + |V_{\text{auto}}|}, \tag{8}$$

DC values are given in this range [0, 1]. 1 corresponding to the perfect overlap and 0 indicating the total mismatch.

4.2.2 Modified Hausdorff Distance (*MHD*)

P_{ref} denote to sets of voxels within the reference and P_{auto} indicate automatic segmentation boundary. MHD is given by the following equation:

$$MHD\left(P_{\text{ref}}, P_{\text{auto}}\right) = \max\left\{\max_{q \in P_{\text{ref}}} d(q, P_{\text{auto}}), \max_{q \in P_{\text{auto}}} d(q, P_{\text{ref}})\right\}, \tag{9}$$

$d(q, p_{\text{auto}})$ denote to the point-to-set distance. The equation is given by:

$$d\left(q, p_{\text{auto}}\right) = \min_{p \in P} \|q - p_{\text{auto}}\|, \tag{10}$$

$||.||$ indicate the Euclidean distance.

4.2.3 Average Surface Distance (*ASD*)

The ASD is given by the following equation:

$$ASD\left(P_{\text{ref}}, P_{\text{auto}}\right) = \frac{1}{|(P_{\text{ref}}|} \sum_{P \in P_{\text{ref}}} d(p, P_{\text{auto}}), \tag{11}$$

$|.|$ denote to the cardinality of a set. Due to the limited number of pages, we refer readers to read about the three metrics on [4].

4.2.4 Comparing Our Results with Prior Work

To demonstrate the significance of our model, we compare our obtained results with the results of five state-of-the-art models. We choose these five models because (a) they have been considered as a baseline (i.e., benchmark) to compare segmentation models in the literature to compare brain segmentation models [4,17,19–21] and (b) all implementation details of such models are publicly available.

4.3 Result and Discussion

We train and test our model on two datasets of different ages (i.e., infants and adults). Table 2 presents the results of our model to segment CSF, GM, and

WM using the `MICCAI iSEG` dataset. Our model obtains a DC values of 94.1% in CSF segmentation. The DC values obtained from segmenting CSF by state-of-the-art models range between 83.5% and 91.5%. The results indicate that our proposed model improves CSF segmentation by 2.6%–10.6%. In addition, our model obtains a DC values of 90.2% and 89.7% in segmenting GM and WM, respectively. The state-of-the-art models, on the other hand, obtain DC values in the ranges of 85.2%–88.6% for GM segmentation and 80.6%–88.7% for WM segmentation. According to the results obtained, we observe that our model achieves a significant improvement of 1.5%–9.6% on segmenting GM and WM. Such results highlight the remarkable efficiency gained by separating actual brain pixels from background and the additional features used in our model.

Table 2. Results for our model provided by $MICCAIiSEG$ organizers [4]. (a) CSF evaluation, (b) WM evaluation, (c) GM evaluation.

(a)				(b)				(c)			
ID	DC	MHD	ASD	ID	DC	MHD	ASD	ID	DC	MHD	ASD
11	0.89	10.8	0.44	11	0.89	7.3	0.44	11	0.89	8.5	0.42
12	0.93	10.2	0.49	12	0.87	6.4	0.49	12	0.87	5.0	0.43
13	0.93	9.5	0.47	13	0.89	7.3	0.47	13	0.89	10.7	0.44
14	0.89	10.9	0.48	14	0.89	5.8	0.48	14	0.88	6.7	0.48
15	0.93	12.2	0.47	15	0.89	6.0	0.47	15	0.89	11.0	0.42
16	0.92	11.8	0.49	16	0.89	8.1	0.49	16	0.89	7.1	0.45
17	0.92	9.00	0.38	17	0.90	10.6	0.38	17	0.90	10.7	0.37
18	0.93	10.8	0.44	18	0.89	7.4	0.44	18	0.89	8.6	0.40
19	0.93	11.1	0.48	19	0.89	8.1	0.48	19	0.89	6.7	0.42
20	0.90	13.2	0.69	20	0.83	8.5	0.69	20	0.83	7.8	0.58
21	0.92	10.8	0.52	21	0.87	8.1	0.52	21	0.87	8.1	0.45
22	0.89	10.1	0.65	22	0.85	8.6	0.65	22	0.85	6.7	0.58
23	0.92	10.6	0.56	23	0.86	7.7	0.56	23	0.86	7.6	0.47
Mean	0.92	10.9	0.2	Mean	0.88	7.69	0.50	Mean	0.89	8.09	0.45
Std	0.01	1.1	0.03	Std	0.02	1.26	0.08	Std	0.01	1.80	0.06

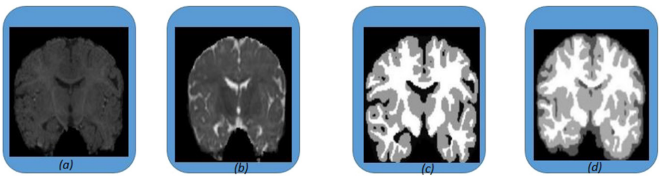


Fig. 6. (a) $T1$, (b) $T2$, (c) manual reference contour, and (d) our model result on the subject used for validation.

Table 3. Segmentation performance in Dice Coefficient (DC) obtained on the $MICCAIiSEG$ dataset. The best performance for each tissue class is highlighted in bold.

Model	Dice Coefficient (DC) Accuracy		
	CSF	GM	WM
[19]	91.2%	86.1%	84.1%
[20]	83.5%	85.2%	86.4%
[17]	90.3%	86.8%	84.3%
[21]	85.5%	87.3%	88.7%
3D, FCN	85.2%	87.6%	80.6%
3D, FCN + MIL	91.5%	88.6%	87.6%
3D, FCN + MIL+G+K	**94.1%**	**90.2%**	**89.7%**

Table 3 compares the results obtained using the `MRBrains` dataset. We observe that our model achieves a DC value of 87.4% on CSF segmentation, 90.6% on GM segmentation, and 90.1% on WM segmentation. Such results are superior to the results obtained using the state-of-the-art models. Therefore, we argue that our mode can perform better on segmenting both infant or adult brain structures.

Figure 6 shows a sample of the results of our model on the subject used for validation set. We observe that our model performs well for brain segmentation, especially for the brain tissues. In Fig. 6, we show an example of our segmentation results and how our result compares to the ground truth. We observe that segmentation results obtained by our model fairly close to the manual reference contour provided by the `MICCAI iSEG` organizers [4]. As expected, much of the improvement by our model is gained at the brain boundary (i.e., between GM and WM). Moreover, we observe that the use of multi-instance loss and Gabor filter banks enabled our model to better handle thin regions than using original brain images. Table 5 presents the execution time (in minutes) for the state-of-art-models and our proposed model. We observe that the execution of our proposed model is faster than the state-of-the-art models. Such results indicate that our model is more efficient and practical to be used in real time systems.

5 Threats to Validity

This section discusses the validity threats of our results and how we address them in our study.

5.1 External Validity

Threats to external validity are related to the generalizability of our results. One could argue that our datasets do not have enough samples. We mitigate

such threat by using two datasets that (a) contain both infant and adult brain data and (b) were previously used by prior studies. In addition, we compare our model with five prior models using the same datasets. Furthermore, we use the small-size kernels, deconvolution layer (to upsample the input), `PReLU`, dropout and normalization methods to reduce the risk of overfitting. Hence, any potential deficiency in the data should affect all the implemented models. Nevertheless, our model obtains higher performance than prior models (Table 4 and Fig. 7).

Table 4. Segmentation performance in Dice Coefficient (*DC*) obtained on the *MRBRAINS* datasets. The best performance for each tissue class is highlighted in bold.

Model	Dice Coefficient (DC) Accuracy		
	CSF	GM	WM
[4]	83.9%	88.9%	89.4%
[20]	83.5%	85.4%	88.9%
[21]	82.8%	84.8%	88.5%
[22]	83.7%	84.8%	88.3%
3D, FCN	81.4%	86.1%	85.2%
3D, FCN + MIL	84.3%	87.6%	89.4%
3D, FCN + MIL+G+K	**87.4%**	**90.6%**	**90.1%**

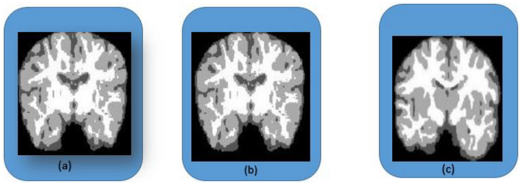


Fig. 7. A sample of our model result on the subject employed for validation. (a) 10 epochs, (b) 20 epochs, (c) 30 epochs.

5.2 Internal Validity

Threats to internal validity are related to experimental errors and bias. Our model is constructed using data extracted from medical images in which contracts might be low. To mitigate such threat, we use the multi-instance loss method to reduce any potential noise in the data by separating actual brain pixels from background. Such method has improved the efficiency and accuracy of our model as well as the accuracy. In addition, our results have been evaluated by the same medical experts (i.e., the organizers of the `MICCAI iSEG` dataset).

Table 5. Average execution time (in minutes) and standard deviation (SD) in the `MRBrains` dataset

Model	Time (SD)
[19]	15.40 (0.16)
[20]	19.23 (0.20)
[17]	17.6 (0.18)
[21]	18.4 (0.15)
3D, FCNN	7.2 (0.12)
3D, FCNN + MIL	**5.9 (0.11)**
3D, FCNN + MIL+G+K	**5.9 (0.11)**

6 Conclusion and Future Work

In this study, we propose an improved fully Convolutional Neural Network (CNN) model for brain segmentation supported by (i) separating brain pixels from background using the multi-instance loss method and (ii) adding additional features using Gabor filter bank and K-means clustering. Our results have been evaluated by the `MICCAI iSEG` organizers and found to be fairly close to the manual reference. In addition, we compare our model with five baseline state-of-the-art models and observe that our model achieves an improvement of up to 11%. In particular, we obtain dice coefficients that range between 87.4% and 94.1%. Such results indicates that the adoption of the multi-instance loss method and Gabor filter banks has significantly improved segmentation results. We argue that our model is more efficient and accurate in practice for both infant and adult brain segmentation.

Despite the promising results obtained from our proposed model, we believe that further improvements can be achieved in the future. For example, conditional random fields (i.e., statistical modeling methods) can be used to predict sequences in pattern recognition and machine learning. We plan to supply a conditional random field to brain segmentation models to investigate whether it is possible to gain better segmentation performance.

References

1. Visser, E., et al.: Automatic segmentation of the striatum and globus pallidus using mist: multimodal image segmentation tool. NeuroImage **125**, 479–497 (2016)
2. Bao, S., Chung, A.C.: Multi-scale structured CNN with label consistency for brain MR image segmentation. Comput. Methods Biomech. Biomed. Eng. Imaging Vis. **6**(1), 113–117 (2018)
3. Chen, H., Dou, Q., Yu, L., Qin, J., Heng, P.A.: VoxResNet: deep voxelwise residual networks for brain segmentation from 3D MR images. NeuroImage **170**, 446–455 (2018)

4. Wang, L., et al.: Benchmark on automatic 6-month-old infant brain segmentation algorithms: the ISEG-challenge. IEEE Trans. Med. Imaging. **38**(9), 2219–2230 (2019)
5. Bernal, J., et al.: Deep convolutional neural networks for brain image analysis on magnetic resonance imaging. Artif. Intell. Med. **95**, 64–81 (2017)
6. Dolz, J., Desrosiers, C., Ayed, I.B.: 3D fully convolutional networks for subcortical segmentation in MRI: a large-scale study. NeuroImage **170**, 456–470 (2018)
7. Badrinarayanan, V., Kendall, A., Segnet, R.: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39**(12), 2481–2495 (2017)
8. Fechter, T., Adebahr, S., Baltas, D., Ayed, I.B., Desrosiers, C., Dolz, J.: Esophagus segmentation in CT via 3D fully convolutional neural network and random walk. Med. Phys. **44**(12), 6341–6352 (2017)
9. de Brebisson, A., Montana, G.: Deep neural networks for anatomical brain segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 20–28 (2015)
10. Moeskops, P., Viergever, M.A., Mendrik, A.M., de Vries, L.S., Benders, M.J., Isgum, I.: Automatic segmentation of MR brain images with a convolutional neural network. IEEE Trans. Med. Imaging **35**(5), 1252–1261 (2016)
11. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: The Thirteenth International Conference on Artificial Intelligence and Statistics, pp. 249–256 (2010)
12. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv. 1502.03167 (2017)
13. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In: IEEE International Conference on Computer Vision, pp. 1026–1034 (2015)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
15. Foulds, J., Frank, E.: A review of multi-instance learning assumptions. Knowl. Eng. Rev. **25**(1), 1–25 (2010)
16. Deng, X., Feng, S., Guo, P., Yin, Q.: Fast image recognition with Gabor filter and pseudoinverse learning autoencoders. In: Cheng, L., Leung, A.C.S., Ozawa, S. (eds.) ICONIP 2018. LNCS, vol. 11306, pp. 501–511. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-04224-0_43
17. Kamnitsas, K., et al.: Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. Med. Image Anal. **36**, 61–78 (2017)
18. Zhang, W., et al.: Deep convolutional neural networks for multi-modality isointense infant brain image segmentation. NeuroImage **108**, 214–224 (2015)
19. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3D U-Net: learning dense volumetric segmentation from sparse annotation. In: Ourselin, S., Joskowicz, L., Sabuncu, M.R., Unal, G., Wells, W. (eds.) MICCAI 2016. LNCS, vol. 9901, pp. 424–432. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46723-8_49
20. Nie, D., Wang, L., Gao, Y., Shen, D.: Fully convolutional networks for multi-modality isointense infant brain image segmentation. In: IEEE 13th International Symposium on Biomedical Imaging (ISBI), pp. 1342–1345 (2016)
21. Mahbod, A., Chowdhury, M., Smedby, O., Wang, C.: Automatic brain segmentation using artificial neural networks with shape context. Pattern Recogn. Lett. **101**, 74–79 (2018)

22. Stollenga, M.F., Byeon, W., Liwicki, M., Schmidhuber, J.: Parallel multi-dimensional with application to fast biomedical volumetric image segmentation. CoRR abs/1506.07452. arXiv:1506.07452 (2018)
23. Hitesh, A., Gupta, A., Raju, A.: Non-linear dimension reduction of Gabor features for noise-robust ASR. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1715–1719 (2014)
24. Jonathan, L., Evan, S., Trevor, D.: Fully convolutional networks for semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
25. Xavier, G., Yoshua, B.: Understanding the difficulty of training deep feedforward neural networks. In: The Thirteenth International Conference on Artificial Intelligence and Statistics, pp. 249–256 (2010)