IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 7, 2022, accepted March 7, 2022, date of publication March 10, 2022, date of current version March 18, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3158342

Multi-Model Medical Image Segmentation Using Multi-Stage Generative Adversarial Networks

AFIFA KHALED[1], JIAN-JUN HAN[1], AND TAHER A. GHALEB[2]
[1]School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China
[2]School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, ON K1N 6N5, Canada

Corresponding author: Jian-Jun Han (jasonhan@hust.edu.cn)

This work was supported in part by the National Science Foundation of China (NSFC) under Award 61872411 and Award 61472150.

ABSTRACT Image segmentation is a challenging problem in medical applications. Medical imaging has become an integral part of machine learning research, as it enables inspecting interior human body with no surgical intervention. Much research has been conducted to study brain segmentation. However, prior studies usually employ one-stage models to segment brain tissues, which could lead to a significant information loss. In this paper, we propose a multi-stage Generative Adversarial Network (*GAN*) model to resolve existing issues of one-stage models. To do this, we apply a *coarse-to-fine* method to improve brain segmentation using a multi-stage *GAN*. In the first stage, our model generates a *coarse* outline for both the background and brain tissues. Then, in the second stage, the model generates a *refine* outline for the white matter (*WM*), gray matter (*GM*), and cerebrospinal fluid (*CSF*). We perform a fusion of the *coarse* and *refine* outlines to achieve high results. Despite using very limited data, we obtain an improved Dice Coefficient (DC) accuracy of up to 5% compared to one-stage models. We conclude that our model is more efficient and accurate in practice for brain segmentation of both infants and adults. In addition, we observe that our multi-stage model is $2.69-13.93$ minutes faster than prior models. Moreover, our multi-stage model achieves higher performance with only a few-shot learning, in which only limited labeled data is available. Therefore, for medical images, our solution is applicable to a wide range of image segmentation applications for which convolution neural networks and one-stage methods have failed. This helps to advance the process of analyzing brain images, thus providing many advantages to the healthcare system, especially in critical health situations where urgent intervention is needed.

INDEX TERMS Brain segmentation, coarse-to-fine, generative adversarial network, semi-supervised learning, multi-stage method.

I. INTRODUCTION

Magnetic resonance imaging (*MRI*) employs a magnetic field to generate detailed images of tissues without using harmful radiations [1], [2]. However, these images tend to be segmented manually, a process that is considered time-consuming and clinically expensive [3]. Hence, automated segmentation of infant and adult brain images has received a substantial research attention [4], [5]. However, training deep learning models requires large sets of labeled images [6]. Due to the limited sets of data in medical applications [7], [8], semi-supervised learning techniques has been used to address this issue by means of unlabeled image [9], [10]. Segmentation results can be improved by adopting unlabeled images [11] or images with weak annotation, such as image level tags [12].

For object detection, a one-stage method is normally used to predict the class probability and position information [13], [14]. With the recent success of two-stage method, many models took advantage of that for semantic segmentation. Recently, Xiaohao *et al.* [1] proposed a two-stage image segmentation method using a convex variant of the Mumford–Shah model and thresholding. In computer vision, two-stage methods are used for generating global information in the first stage and local information in the second stage [15], [16]. Good results can be achieved by fusing the global information and local information together [17], [18]. In addition, the adoption of multi-stage Generative Adversarial Networks (*GAN*) in medical imaging remains unexplored.

The associate editor coordinating the review of this manuscript and approving it for publication was Cristian A. Linte.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

In this paper, we propose a coarse-to-fine method to improve brain segmentation using a multi-stage GAN with three generators, referred to as G, as follows:

- In the first generator, our model generates a coarse outline for both background and brain tissues. The main role of the first G is to generate coarse segmentation information to guide the third G.
- In the second generator, two inputs, image x and a random vector z, are taken to encourage generating as many different values for each x as those of z.
- In the third generator, an encoder and decoder are used along with a dense skip connection to combine features from different scales. This generator generates an outline for (i) white matter (WM), (ii) gray matter (GM), and (iii) cerebrospinal fluid (CSF). This process is similar to that of human learning in a clinical practice. Specifically, the role of the third G is to generate more detailed results using the coarse segmentation from the first G.

We evaluate our proposed multi-stage generative adversarial model on two datasets of brain tissues, including infant and adult brain. Our model achieves higher results compared to the state-of-the-art models. In particular, despite using very limited data, we obtain an improved Dice Coefficient (DC) accuracy of up to 5% compared to one-stage models. In addition, we observe that our multi-stage model is $2.69 - 13.93$ minutes faster than prior models. Therefore, for medical images, our solution is applicable to a wide range of image segmentation applications for which convolution neural networks and one-stage methods have failed. This helps to advance the process of analyzing brain images, thus providing many advantages to the healthcare system, especially in critical health situations where urgent intervention is needed.

The rest of this paper is organized as follows. Section II presents the prior studies and techniques related to brain segmentation. Section III presents the design of multi-stage model. Section IV presents our experimental design and evaluation. Section V presents our results and discussion. Section VI discusses the validity threats to our results. Finally, Section VII concludes the paper and discusses directions for future work.

II. BACKGROUND AND RELATED WORK

This section presents the prior studies and the techniques related to brain segmentation. First, we describe in detail semi-supervised learning. Second, we describe generative adversarial networks (GAN). Finally, we show how loss functions are used to improve the stability of training GAN models.

A. SEMI-SUPERVISED LEARNING

Training a deep model using a small datasets may cause overfitting [11], [19]. To prevent overfitting, large amounts of unlabeled data with a small amount of labeled data should be used [20], [21]. Training deep models using both labeled and unlabeled data encourages neural networks to have a similar distribution [22], [23]. In particular, semantic segmentation works by taking an image as an input and generating a segmentation map as an output [16], [24], [25]. Figure 1 and Figure 2 show the semantic segmentation labels and semantic segmentation classes, respectively.

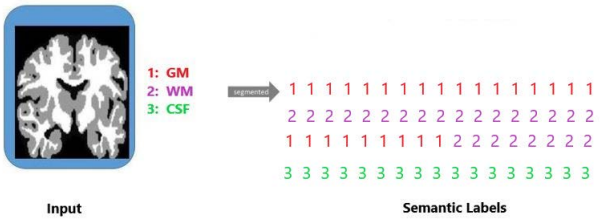


FIGURE 1. The illustration of semantic labels.

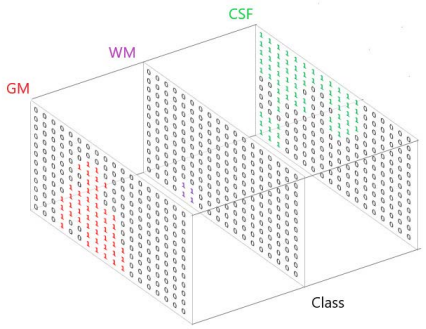


FIGURE 2. The illustration of semantic classes.

Much research has applied semantic segmentation for brain images, in particular, images for brain tissues. Bdair *et al.* [26] proposed ROAM, a random layer mixup that allows neural networks to be less confident for interpolated data points on any selected space. Gillmann *et al.* [27] proposed two architectures for brain tumor segmentation. Their results have been evaluated using the *pinnacle BraTS confront2017* datasets. Similarly, Majib ett al. [28] proposed a rethinking atrous convolution model for semantic images. Differently from the above models, rethinking atrous convolution model targets long range contexts, as it does not require convolution layers. Instead, it utilizes an atrous convolution with up-sampled filters to extract dense feature maps. The model was evaluated on the *PASCAL VOC 2012* semantic image segmentation benchmark, consisting of 3,475 finely annotated images and extra 20,000 coarsely annotated images. Their experimental results of the sentiment task show that atrous convolution is necessary when building more blocks cascadedly. The results also show that the more blocks are added, the better the performance.

TABLE 1. Example GAN models applied in medical applications.

Publication	Method
Han et al. [32]	WGAN
Bowles et al. [33]	PGGAN
Andrew et al. [34]	LAPGAN
Mondal et al. [35]	DCGAN
Kang et al. [36]	CycleGAN
Chuquicusma et al. [37]	DCGAN
Frid-Adar et al. [38]	DCGAN/ACGAN

B. GENERATIVE ADVERSARIAL NETWORK (GAN)

GANs have demonstrated promising results for medical image diagnostics [29] and brain image segmentation [21], [25]. Figure 3 shows an overview of how GANs work.

Many researchers have applied generative adversarial network for brain segmentation. Cirillo *et al.* [30] proposed a 3D volume-to-volume (*GAN*) for segmenting brain tumors. Their model achieved a good result when the generator loss was weighted five times higher than the discriminator loss. The proposed model was evaluated on the *BraTS 2018* datasets. Their model outperformed previous models with an overall accuracy of 0.66%. Delannoy *et al.* [31] proposed a super resolution and segmentation framework using generative adversarial networks to neonatal brain *MRI* images. The framework consists of (a) a training of a generating network that estimates the corresponding HR image for a given input image and (b) a discriminator network D to distinguish real HR and segmentation images. In Table 1, we provide example GAN models applied in medical applications.

C. LOSS FUNCTIONS

Loss functions have been developed to improve the training stability of *GAN* models [39], [40]. In this section, we describe five loss functions that are used for *GAN*s.

1) MINIMAX *GAN* LOSS

Minimax *GAN* loss function consists of two components: a generator and a discriminator. The generator attempts to minimize the loss function, whereas the discriminator attempts to maximize. Their formulas are given below.

Generator loss function [41]:

$$l_{\mathrm{D}}^{GAN} = -E_{\mathrm{z}\sim p_d}[logD(x)] - E_{\mathrm{x}\sim p_g}[log(1-d(x))] \quad (1)$$

Discriminator loss function [41]:

$$l_{\mathrm{G}}^{GAN} = -E_{\mathrm{z}\sim pg}[log(1-D(X))] \quad (2)$$

In the discriminator loss function:

- $D(x)$ denotes the discriminator's estimate of the probability that real data x is real.
- $E(x)$ denotes the expected value over all real data.
- $G(z)$ denotes the generator's output for a given noise z.
- $G(z)$ denotes the generator's output for a given noise z.
- $D(G(z))$ denotes the discriminator's estimate of the probability that fake data is real.
- $E(z)$ denotes the expected value over all generated fake data $G(z)$.

2) NON-SATURATING LOSS (*NSGAN*)

Non-saturating loss is used to solve the saturation problem.

Generator loss function [21]:

$$l_{\mathrm{D}}^{NSGAN} = -E_{\mathrm{z}\sim p_d}[logD(x)] - E_{\mathrm{x}\sim p_g}[log(1-D(x))] \quad (3)$$

Discriminator loss function [21]:

$$l_{\mathrm{G}}^{NSGAN} = -E_{\mathrm{z}\sim p_g}[logD(x)] \quad (4)$$

3) WASSERSSTEIN LOSS (*WGAN*)

*GAN*s are commonly used in the area of computer vision [42], [43], but the main problem is with training instability [28]. Many loss functions have been developed toward providing a stable training of *GAN*s [35]. Wassersstein (*WGAN*) achieves a good progress for training stability of *GAN*, but still suffers from poor results. It has been argued that Wassersstein's poor result is due to the use of weight clipping. To address tha, Adler and Lunz [44] proposed a better approach for clipping weights. This resulting model is a modification of the standard *GAN*. The discriminator training tries to make the output bigger for real data than for fake data. The output of the discriminator is a number, which does not have to be between 0 and 1. More details can be found in [44].

Generator loss function [44]:

$$l_{\mathrm{G}}^{NSGAN} = -E_{\mathrm{z}\sim p_g}[D(x)] \quad (5)$$

Discriminator loss function [44]:

$$l_{\mathrm{D}}^{WGAN} = -E_{\mathrm{z}\sim p_d}[D(x)] - E_{\mathrm{x}\sim p_g}[D(x)] \quad (6)$$

In these functions:

- $D(x)$ denotes the discriminator's output for real data.
- $G(z)$ denotes the generator's output for a given noise z.
- $D(G(z))$ denotes the discriminator's output for fake data.

4) LEAST-SQUARES LOSS (*LSGAN*)

This model proposed $a-b$ coding scheme for the discriminator where a, b denote to the labels of fake and real data.

Generator loss function [21]:

$$l_{\mathrm{G}}^{LSGAN} = -E_{\mathrm{z}\sim p_g}[D(x-1)^2] \quad (7)$$

Discriminator loss function [21]:

$$l_{\mathrm{D}}^{LSGAN} = -E_{\mathrm{z}\sim p_d}[D(x)-1]^2 - E_{\mathrm{x}\sim p_g}[D(x^2)] \quad (8)$$

5) WASSERSSTEIN GRADIENT PENALTY LOSS (AC-GAN)

AC-GAN uses a noise z and a sample with class label $c \sim p$. This model is a modification of the standard *GAN*. In the standard GAN, $X_{\mathrm{fake}} = G(Z)$, whereas in $AC-GAN$, $X_{\mathrm{fake}} = G(c, z)$. IIn addition, the output of standard *GAN* is a probability distribution $P(s, x) = D(x)$, whereas

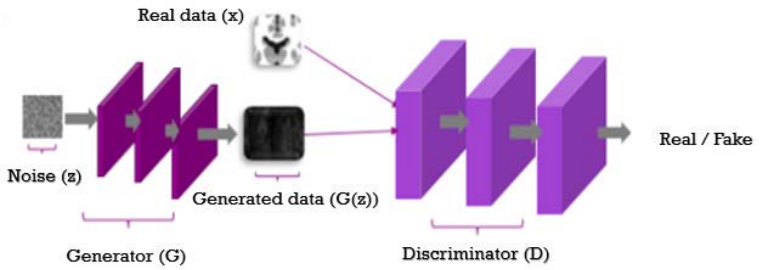


FIGURE 3. The illustration of Generative Adversarial Networks (*GAN*).

in $AC-GAN$, the output is two probability distributions. $P(s, x), P(c, x) = D(x)$.

Mondal *et al.* [35] proposed a model that uses a *GAN* for brain segmentation. The authors used a dataset of 43 subjects, where they generate fake images using a generator, followed by labeled, unlabeled, and fake data to train the discriminator to distinguish between generated data and true data. Besides, an encoder was used to compute the predicted noise mean and log-variance. However, their approach only supports one-stage, whereas our model supports multi-stage modeling.

Unlike previous work, we aim in this paper to solve the problem of information loss suffered by one-stage modeling. To do this, our first generator generates a coarse outline to be used by a third generator. Then, the encoder and decoder generate a fine outline. Moreover, we use a dense skip connection to combine the features from different scales. To validate our multi-stage model, we use the Dice coefficient metric.

III. METHODOLOGY

In this section, we present the design of our proposed multi-stage *GAN* model. We first give a more detailed description of the *GAN* model that we used. Then, we give a more detailed description of the loss functions (discriminator and generator) we used. Table 2 shows a list of the symbols defined in this paper.

A. GENERATIVE ADVERSARIAL NETWORK (GAN)

Generative adversarial network (*GAN*) refers to a network composed of two networks: a *generator* G, which is used to generate a fake images from a noise vector, and a *discriminator* D, which is used to distinguish between generated data and true data. In particular, G is trained to map a noise vector $z \in R$ to a fake image, whereas D is trained to differentiate between true data x and generated data $G(z)$. The core idea behind *GAN*s is to play a two player *min*/*max* game: $min_G max_D E_{x\sim pdata}[logD(x)] + E_{z\sim noise}[1 - D(G(z))]$

Figure 4 shows an overview of our proposed *GAN* network, which mainly consists of the 3-stage generator network and the discriminator network. The discriminator is used to distinguish between true and generated data. The first generator is mainly used to generate an outline for the background and brain tissues from the input images. The second generator takes two inputs: an image x and a random vector z. This encourages the generator to generate as many different values for each x as those of z. Specifically, training a network with a random vector z and an image x encourages the network to give better output. The third generator is used to generate an outline for (i) white matter (WM), (ii) gray matter (GM), and (iii) cerebrospinal fluid (CSF). The main role of the first G is to generate a coarse segmentation that can be used to guide the third G. The main role of the third G is to generate more detailed results using the coarse segmentation from the first G. The third G consists of an encoder and a decoder. The encoder and decoder use a dense skip connection to combine the features from different scales. Figure 5 shows the network architecture of the encoder, decoder, and dense skip connection.

TABLE 2. List of symbols defined in this paper.

Symbol	Definition
G	Generator
D	Discriminator
z	Noise
$G(z)$	Generated data
x	Real data
WM	White matter
GM	Gray matter
CSF	Cerebrospinal fluid
$Conv$	Convolutional
$LeReLU$	Activation function
GAN	Generative adversarial network
E	Expected value
DC	Dice Coefficient
MRI	Magnetic resonance imaging
$T1$	subject-1-to-subject-10
T	subject-11-to-subject-23
$NSGAN$	Non-saturating loss
$WGAN$	Wassersstein loss
$lSGAN$	Least-squares loss
$ACGAN$	Wassersstein Gradient Penalty loss
V_auto	Automated segmentation
V_ref	Reference segmentation

We used the generator proposed by Dai *et al.* [41] and change it as follows:

1- K classes are changed to $(K + 1)$ classes.
2- The segmentation network is changed to be fully-convolutional.

We used the discriminator network proposed by Çiçek *et al.* [45] and change it as follows:

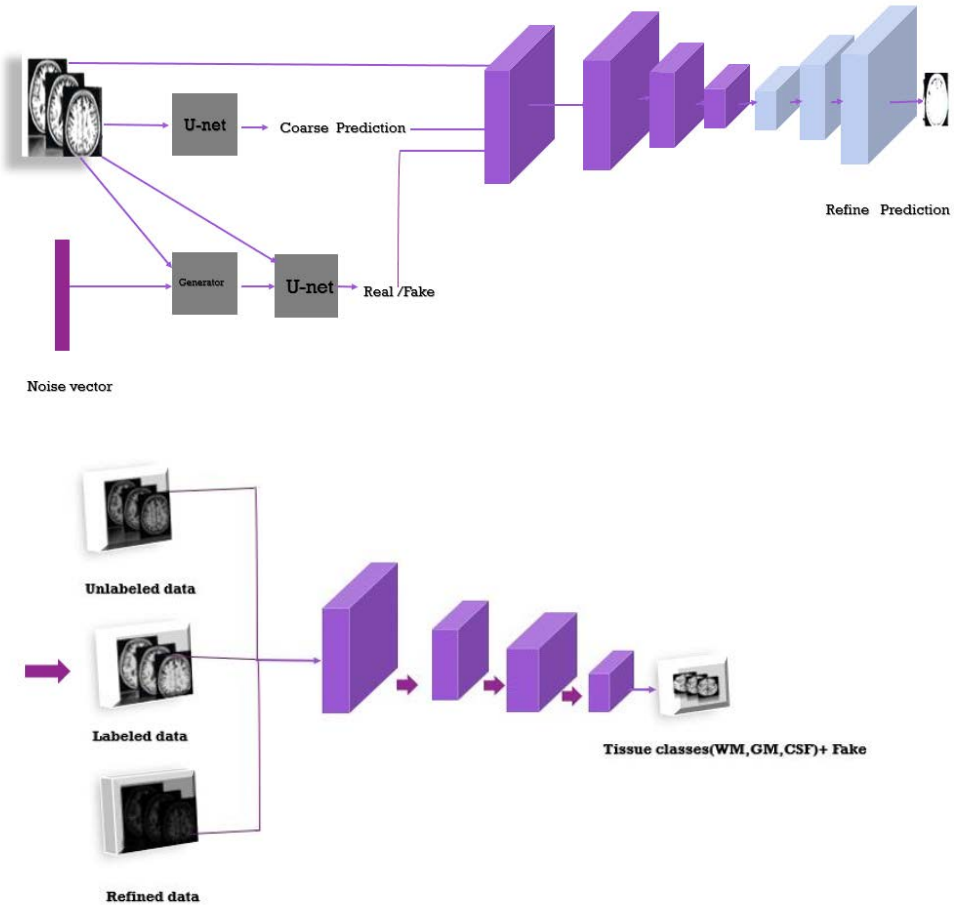


FIGURE 4. Our proposed multi-stage *GAN* model.

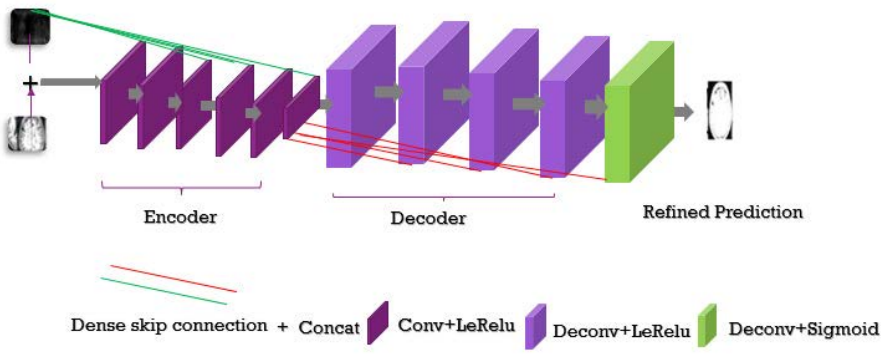


FIGURE 5. The illustration of our encoder/decoder network.

1- ReLUs are changed to leaky ReLUs.
2- Max pooling was changed to average pooling.

To implement our encoder and decoder. For the encoder, we use four blocks, as follows:

The 1st block: consists of conv, LeReLU and concatenation.

The 2nd block: consists of conv, LeReLU and concatenation.

The 3rd block: consists of conv, LeReLU and concatenation.

The 4th block: consists of conv, LeReLU and concatenation.

For the decoder, we use four blocks, as follows:

The 1st block: consists of deconv, conv, LeReLU and concatenation.

The 2nd block: consists of deconv, conv, LeReLU and concatenation.

The 3rd block: consists of deconv, conv, LeReLU and concatenation.

The 4th block: consists of deconv, conv, LeReLU and concatenation.

Moreover, we use the dense skip connection to combine the features from different layers.

B. LOSS FUNCTION

1) DISCRIMINATOR LOSS FUNCTION

The discriminator in our model has an unlabeled data loss, labeled data loss, and refined prediction loss. The overall loss function is computed as follows:

$$l_{\text{discriminator}} = \lambda_{\text{labeled}} l_{\text{labeled}} + \lambda_{\text{unlabeled}} l_{\text{unlabeled}} + \lambda_{\text{fake}} l_{\text{fake}} \tag{9}$$

where λ_{labeled}, $\lambda_{\text{unlabeled}}$, and λ_{fake} are hyper-parameters. We set the hyper-parameters in Equation 9 to $\lambda_{\text{labeled}} = 1.0$, $\lambda_{\text{unlabeled}} = 1.0 and \lambda_{\text{fake}} = 2.0$.

For labeled data, we use the same loss function in the standard segmentation network. Mondal *et al.* [35] showed that using $l_{i,k+1}$ as a subtracted function, the softmax function is changed as follows:

$$l_{\text{labeled}} = -E_{\text{x,y}\sim\text{pdata(x,y)}} \sum_{\text{i=1}}^{H\times W\times D} \log(Pmodel(y_{\text{i}}|X) \tag{10}$$

$$l_{\text{unlabeled}} = -E_{\text{x}\sim\text{pdata(x)}} \sum_{\text{i=1}}^{H\times W\times D} \log((Z_{\text{i}}(x)/Z_{\text{i}}(x)) + 1) \tag{11}$$

$$l_{\text{fake}} = -E_{\text{z}\sim\text{noise}} \sum_{\text{i=1}}^{H\times W\times D} \log[((1/Z_{\text{i}}(G_{\Theta G}(z) + 1)] \tag{12}$$

$$Z_{\text{i}} = \sum_{\text{k=1}}^{K} exp[l_{\text{i,k}}(x)], \tag{13}$$

The idea is to introduce unlabeled loss and fake loss, which are analogous to the two components of the discriminator loss in the standard *GAN*, whereas labeled loss represents the cross-entropy. More details can be found in [35].

2) GENERATOR LOSS FUNCTION

We proposed a novel generated loss to encourage G to generate real data. Let x and z denote to the real data and noise, respectively.

$$C = E_{\text{x}\sim\text{pdata(x)}} f(x) - log(1 - D(G(z))) \tag{14}$$

In our paper, $f(x)$ contains the activation of the last layer.

$$L(G) = ||C - x||_2^2 \tag{15}$$

By minimizing this loss, we force the generator to generate real data to match our data and the corresponding K classes of real data, which is defined as $Classes = 1, \ldots, K$.

IV. EXPERIMENTS

This section presents our experimental design and evaluation. First, we give a more detailed description of the datasets used in our experiments. Then, we show our experimental setup. Finally, we explain the Dice coefficient of the segmentation evaluation.

A. DATASETS

1) DATASETS

In our experiments, we use two different datasets of brain images: the *MICCAI iSEG* dataset and *MRBrainS* dataset. The MICCAI iSEG-2017 dataset contains data of 6-month infants, whereas the MRBrainS-2013 dataset contains adult data. We should note that there are significant differences between the two datasets in term of image data characteristics, such as voxel spacing and the number of available modalities. However, these two datasets were both used to evaluate the state-of-the-art models in this context [46], [47]. We describe each of these datasets in the following.

2) MICCAI iSEG-2017 DATASET

The aim of the evaluation framework 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 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 Figure 6. On the other hand, Table 3 shows the parameters used to generate $T1$ and $T2$. The dataset has two different times: longitudinal relaxation time and transverse relaxation time, which are used to generate $T1$ and $T2$. The dataset has been interpolated, registered, and skull-removed by the *MICCAI iSEG* organizers. We present the evaluation equations in subsection IV-B.

3) MRBrainS-2013 DATASET

The *MRBrainS* 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) brain stem on $T1$, $T2$, and FLAIR. Five subjects, 2 male and 3 female, are provided to the training set and 15 subjects are provided for the testing set. To evaluate the segmentation, these structures were merged into gray matter ($a - b$), white matter ($c - d$), and cerebrospinal fluid ($e - f$). The cerebellum and brainstem were excluded from the evaluation.

FIGURE 6. An example of the *MICCAI iSEG* dataset (*T*1,*T*2, manual reference contour).

TABLE 3. Parameters used to generate *T*1 and *T*2.

Parameter	TR/TE	Flip angle	Resolution
$T1$	1,900/4.38 ms	7	1×1×1
$T2$	7,380/119 ms	150	1.25×1.25×1.25

4) EXPERIMENTAL SETUP

We implement our proposed model using Python on a computer with a *NVIDIAGPU* and Ubuntu 16.04. Training our model took 30 hours in total, whereas testing took 5 minutes for.

B. SEGMENTATION EVALUATION

1) DICE COEFFICIENT (*DC*)

To better demonstrate the significance of our model, we use the Dice Coefficient (*DC*) metric for evaluation. Dice Coefficient (*DC*) has been considered as a baseline (benchmark) in the literature to compare brain segmentation models. We use V_{ref} for the reference segmentation and V_{auto} for the automated segmentation. The *DC* is given by the following equation [41]:

$$DC(V_{\text{ref}}, V_{\text{auto}}) = \frac{2V_{\text{ref}} \bigcap V_{\text{auto}}|}{|V_{\text{ref}}| + |V_{\text{auto}}|} \tag{16}$$

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

C. EVALUATING THE HYPER-PARAMETERS MULTI-STAGE

To evaluate the effectiveness of our model, we evaluate different hyper parameters: epochs, learning rate, and batch size. Table 4, Table 5, and Table 6 show training epochs, learning rate, and batch size, respectively. We observe that a batch size of 30 is 95%, 94%, and 92% for *CSF*, *GM* and *WM*, respectively. Large training epochs can cause overfitting, whereas and small training epochs can cause underfitting. To mitigate these issues, we validate whether the training epochs will be significantly impacted the network performance. To do this, we use training epochs of 20, 40, 60, 80 epoch. In the 80 epochs, we observe that the network performance was the best. We followed a similar approach to select the best learning rate values. A large learning rate can make the parameters of network updated quickly, whereas a small learning rate can make the parameters updated slowly. To address this, we first randomly start with a learning rate of 1×10^{-1}. Then, we use multiple runs while changing the learning rate. Experimental results showed that our multi-stage model achieves a higher result for the learning rate of 1×10^{-4}.

TABLE 4. Experiments on Training epoch obtained on the *MRBrainS* dataset. The best performance for each tissue class is highlighted in bold.

Training epoch	Dice Coefficient (DC) Accuracy		
	CSF	GM	WM
20	67%	69%	60%
40	86%	84%	82%
60	87%	88%	85%
80	**95%**	**94%**	**92%**

TABLE 5. Experiments on Learning Rate obtained on the *MRBrainS* dataset. The best performance for each tissue class is highlighted in bold.

Learning Rate	Dice Coefficient (DC) Accuracy		
	CSF	GM	WM
$1 \times 10^{-}1$	80%	80%	82%
$1 \times 10^{-}2$	84%	83%	82.4%
$1 \times 10^{-}3$	87%	88%	87%
$1 \times 10^{-}4$	**95%**	**94%**	**92%**

TABLE 6. Experiments on batch size obtained on the *MRBrainS* dataset. The best performance for each tissue class is highlighted in bold.

Batch size	Dice Coefficient (DC) Accuracy		
	CSF	GM	WM
10	77%	76%	76%
20	82%	83.2%	82.4%
30	**95%**	**94%**	**92%**
40	89%	88%	87%

V. RESULTS AND DISCUSSION

To better demonstrate the significance of our model, we train and test our multi-stage *GAN* model on two datasets of different ages: infants and adults, as follows:

- **MICCAI iSEG-2017 dataset**
  - For the 13 unlabeled images, that are actually part of the testing set, we use them to train our *GAN* model
  - For the 10 labeled images, we use two for training, one for validation, and seven for testing
- **MRBrainS-2013 dataset**
  - For the 15 unlabeled images, that are actually part of the testing set, we use them to train our multi-stage *GAN* model
  - For the five labeled images, we use one for training, one for validation, and three for testing

The main goal of our multi-stage *GAN* model is to improve the performance with a few-shot learning case. Table 7 presents the results of our model to segment *CSF*,*GM*, and *WM* using the *MICCAI iSEG* dataset, in comparison with the state-of-the-art models. Our model achieves *DC* values of 95% in *CSF* segmentation. The *DC* values obtained from segmenting *CSF* by the state-of-the-art models ranged between 86% and 91%. In addition, our model achieves a *DC* values of 94% and 92% in segmenting *GM* and *WM*, respectively. The state-of-the-art models, on the other hand, obtain *DC* values in the ranges of 80%- 93% for *GM*

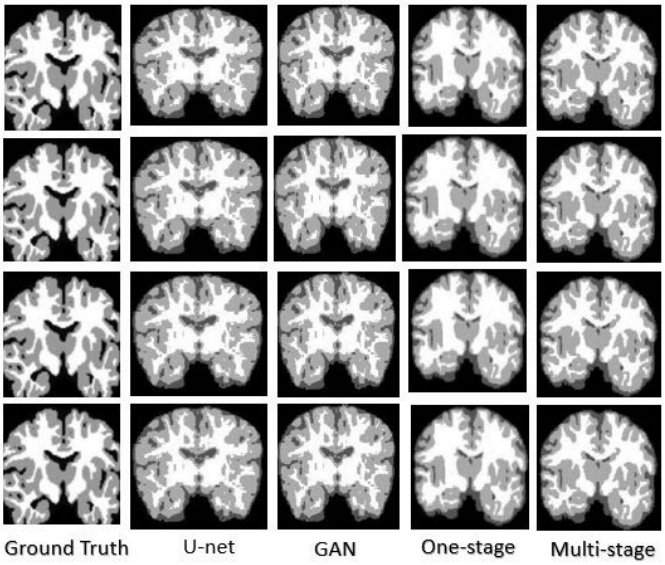


FIGURE 7. Visualization results on *MRBrainS* dataset.

TABLE 7. Segmentation performance in Dice Coefficient (*DC*) obtained on the *MICCAI iSEG* dataset. The best performance for each tissue class is highlighted in bold.

Model	Dice Coefficient (DC) Accuracy		
	CSF	GM	WM
U-net	86.2%	80.1%	81.1%
Standard-GAN	87.5%	89.2%	82.4%
One-stage	91.3%	93.8%	90.3%
Multi-stage	**95%**	**94%**	**92%**

TABLE 8. Segmentation performance in Dice Coefficient (*DC*) obtained on the *MRBrainS* dataset. The best performance for each tissue class is highlighted in bold.

Model	Dice Coefficient (DC) Accuracy		
	CSF	GM	WM
U-net	83%	80%	80%
Standard-GAN	86%	86%	81%
One-stage	88%	88%	84%
Multi-stage	**93%**	**93%**	**88%**

segmentation and 81%- 90% for *WM* segmentation. Such results highlight the remarkable efficiency gained by using multi-stage *GAN*.

Table 8 presents the results achieved by our model using the *MRBrainS* dataset, in comparison with the state-of-the-art models. We observe that our model achieves a *DC* value of 93% on *CSF* segmentation, 93% on *GM* segmentation, and 88% on *WM* segmentation. Such results surpass the results achieved by the state-of-the-art models. Therefore, we argue that our model can perform better in a few-shot learning case.

Table 9 shows the execution time (in minutes) for our multi-stage *GAN* model, in comparison with the state-of-the-art models. We observe that the execution of our proposed model is faster than the state-of-the-art models. Such results

TABLE 9. Average execution time (in minutes) and standard deviation (*SD*) in the *MRBrainS* dataset.

Model	Time (SD)
U-net	36.54(0.12)
Standard-GAN	30.52(0.31)
One-stage	25.30(0.16)
Multi-stage	**22.61(0.21)**

indicate that our model is more efficient and, hence, more practical to be used in real-time systems.

Figure 7 visualizes the results of our model on the images used as a validation set. We observe that the segmentation results achieved by our multi-stage model are fairly close to the manual reference contour, i.e., ground truth, provided by the *MICCAI iSEG* organizers.

VI. THREATS TO VALIDITY

A. EXTERNAL VALIDITY

Threats to external validity are related to the generalizability of our results. In this paper, we use two datasets that belong to two organizers. The total number of the subjects in the two datasets are 43 subjects. One could argue that the 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. Our model obtains a higher performance than the state-of-the-art models. We believe that our model performs as similar as human learning in clinical practice. Moreover, while we only targeted three tissues, our proposed model can be easily extended to segment more tissues as it does not require more labeled data. The intuition behind our multi-stage model is that it improves the performance in a few-shot learning case where only a few labeled data are available for training.

B. 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. 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 deficient all the implemented models. Nevertheless, our model obtains higher performance than previous models.

VII. CONCLUSION

In this paper, we proposed a multi-stage generative adversarial network (*GAN*) model for brain segmentation that generates a coarse outline for both background and brain tissues. Then, our model generates an outline for white matter (*WM*), gray matter (*GM*), and cerebrospinal fluid (*CSF*). We evaluated our results using both infant and adult datasets, in comparison with three baseline state-of-the-art models. We found that our segmentation results are fairly close to the manual reference. In addition, we observe that our model surpasses the state-of-the-art models by achieving a performance improvement of up to 5%. In particular, we obtain Dice coefficients (*DC*) ranging between 88% and 95%. Such results indicate that the adoption of our multi-stage *GAN* model 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. We aim in the future to consider more datasets in our study. Moreover, we intend to expand the evaluation of our multi-stage model to investigate its performance on segmenting more brain tissues. Finally, we aim to investigate whether our multi-stage model achieves a higher performance for pathological brain images, such as with tumor or edema.

VIII. DECLARATIONS

A. COMPETING INTERESTS

The authors declare that they have no known competing financial interests.

B. ETHICS APPROVAL AND CONSENT TO PARTICIPATE

Huazhong University of Science and Technology the ethics committee that approved our study and the committee's reference number is 430074.

C. CONSENT FOR PUBLICATION

Not applicable.

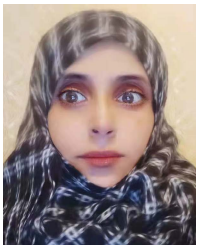
D. AVAILABILITY OF DATA AND MATERIALS

The data that support the findings of this study are available from *MICCAI* grand challenge on 6-month infant brain *MRI* segmentation [48] and *MRBrainS* and are publicly available.

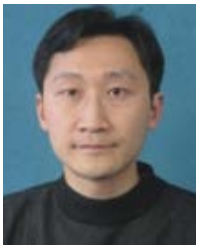
REFERENCES

[1] X. Cai, R. Chan, and T. Zeng, "A two-stage image segmentation method using a convex variant of the Mumford–Shah model and thresholding," *SIAM J. Imag. Sci.*, vol. 6, no. 1, pp. 368–390, Aug. 2013.
[2] L. Sun, J. Wang, Y. Huang, X. Ding, H. Greenspan, and J. Paisley, "An adversarial learning approach to medical image synthesis for lesion detection," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 8, pp. 2303–2314, Aug. 2020.
[3] X. Yi, E. Walia, and P. Babyn, "Generative adversarial network in medical imaging: A review," *Med. Image Anal.*, vol. 58, Dec. 2019, Art. no. 101552.
[4] H. A. Helaly, M. Badawy, and A. Y. Haikal, "Toward deep MRI segmentation for Alzheimer's disease detection," *Neural Comput. Appl.*, vol. 34, pp. 1047–1063, Aug. 2021.
[5] S. Kazeminia, C. Baur, A. Kuijper, B. van Ginneken, N. Navab, S. Albarqouni, and A. Mukhopadhyay, "GANs for medical image analysis," *Artif. Intell. Med.*, vol. 109, Sep. 2020, Art. no. 101938.
[6] A. Khaled, C.-M. Own, W. Tao, and T. A. Ghaleb, "Improved brain segmentation using pixel separation and additional segmentation features," in *Proc. Asia–Pacific Web (APWeb) Web-Age Inf. Manage. (WAIM) Joint Int. Conf. Web Big Data*. China: Springer, 2020, pp. 85–100.
[7] T. Chen, X. Song, and C. Wang, "Preserving-texture generative adversarial networks for fast multi-weighted MRI," *IEEE Access*, vol. 6, pp. 71048–71059, 2018.
[8] T. Iqbal and H. Ali, "Generative adversarial network for medical images (MI-GAN)," *J. Med. Syst.*, vol. 42, no. 11, pp. 1–11, Nov. 2018.
[9] A. A. Adegun, S. Viriri, and R. O. Ogundokun, "Deep learning approach for medical image analysis," *Comput. Intell. Neurosci.*, vol. 2021, May 2021, Art. no. 6215281.
[10] S. Dara, P. Tumma, N. R. Eluri, and G. R. Kancharla, "Feature extraction in medical images by using deep learning approach," *Int. J. Pure Appl. Math.*, vol. 120, no. 6, pp. 305–312, 2018.
[11] C. Liang and S. Xin, "Research status and prospects of deep learning in medical images," in *Proc. Int. Conf. Commun., Inf. Syst. Comput. Eng. (CISCE)*, Jul. 2020, pp. 380–382.
[12] Y. Gu, Y. Peng, and H. Li, "AIDS brain MRIs synthesis via generative adversarial networks based on attention-encoder," in *Proc. IEEE 6th Int. Conf. Comput. Commun. (ICCC)*, Dec. 2020, pp. 629–633.
[13] M. Chen, X. Shi, Y. Zhang, D. Wu, and M. Guizani, "Deep feature learning for medical image analysis with convolutional autoencoder neural network," *IEEE Trans. Big Data*, vol. 7, no. 4, pp. 750–758, Oct. 2021.
[14] N. Yamanakkanavar and B. Lee, "Using a patch-wise M-net convolutional neural network for tissue segmentation in brain MRI images," *IEEE Access*, vol. 8, pp. 120946–120958, 2020.
[15] L. Zhang, X. Wang, D. Yang, T. Sanford, S. Harmon, B. Turkbey, B. J. Wood, H. Roth, A. Myronenko, D. Xu, and Z. Xu, "Generalizing deep learning for medical image segmentation to unseen domains via deep stacked transformation," *IEEE Trans. Med. Imag.*, vol. 39, no. 7, pp. 2531–2540, Jul. 2020.
[16] M. Ozbey and T. Cukur, "T1-weighted contrast-enhanced synthesis for multi-contrast MRI segmentation," in *Proc. 28th Signal Process. Commun. Appl. Conf. (SIU)*, Oct. 2020, pp. 1–4.
[17] J. M. H. Noothout, B. D. De Vos, J. M. Wolterink, E. M. Postma, P. A. M. Smeets, R. A. P. Takx, T. Leiner, M. A. Viergever, and I. Isgum, "Deep learning-based regression and classification for automatic landmark localization in medical images," *IEEE Trans. Med. Imag.*, vol. 39, no. 12, pp. 4011–4022, Dec. 2020.
[18] J. Andén, E. Katsevich, and A. Singer, "Covariance estimation using conjugate gradient for 3D classification in CRYO-EM," in *Proc. IEEE 12th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2015, pp. 200–204.
[19] M. Geng, X. Meng, J. Yu, L. Zhu, L. Jin, Z. Jiang, B. Qiu, H. Li, H. Kong, J. Yuan, K. Yang, H. Shan, H. Han, Z. Yang, Q. Ren, and Y. Lu, "Content-noise complementary learning for medical image denoising," *IEEE Trans. Med. Imag.*, vol. 41, no. 2, pp. 407–419, Feb. 2022.
[20] X. Chen, C. Xu, X. Yang, L. Song, and D. Tao, "Gated-GAN: Adversarial gated networks for multi-collection style transfer," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 546–560, Feb. 2019.
[21] U. Niyaz, A. S. Sambyal, and Devanand, "Advances in deep learning techniques for medical image analysis," in *Proc. 5th Int. Conf. Parallel, Distrib. Grid Comput. (PDGC)*, Dec. 2018, pp. 271–277.
[22] S. Singh and N. Singh, "Object classification to analyze medical imaging data using deep learning," in *Proc. Int. Conf. Innov. Inf., Embedded Commun. Syst. (ICIIECS)*, Mar. 2017, pp. 1–4.

[23] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.
[24] Y. Wang, A. K. Katsaggelos, X. Wang, and T. B. Parrish, "A deep symmetry convnet for stroke lesion segmentation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 111–115.
[25] Y. Ding, F. Chen, Y. Zhao, Z. Wu, C. Zhang, and D. Wu, "A stacked multi-connection simple reducing net for brain tumor segmentation," *IEEE Access*, vol. 7, pp. 104011–104024, 2019.
[26] T. Bdair, B. Wiestler, N. Navab, and S. Albarqouni, "ROAM: Random layer mixup for semi-supervised learning in medical imaging," 2020, *arXiv:2003.09439*.
[27] C. Gillmann, L. Peter, C. Schmidt, D. Saur, G. Scheuermann, and M. Potel, "Visualizing multimodal deep learning for lesion prediction," *IEEE Comput. Graph. Appl.*, vol. 41, no. 5, pp. 90–98, Sep. 2021.
[28] M. S. Majib, M. M. Rahman, T. M. S. Sazzad, N. I. Khan, and S. K. Dey, "VGG-SCNet: A VGG net-based deep learning framework for brain tumor detection on MRI images," *IEEE Access*, vol. 9, pp. 116942–116952, 2021.
[29] Y. Sun, C. Zhou, Y. Fu, and X. Xue, "Parasitic GAN for semi-supervised brain tumor segmentation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 1535–1539.
[30] M. D. Cirillo, D. Abramian, and A. Eklund, "Vox2Vox: 3D-GAN for brain tumour segmentation," in *Proc. Int. MICCAI Brainlesion Workshop*. Peru: Springer, 2020, pp. 274–284.
[31] Q. Delannoy, C.-H. Pham, C. Cazorla, C. Tor-Díez, G. Dollé, H. Meunier, N. Bednarek, R. Fablet, N. Passat, and F. Rousseau, "SegSRGAN: Super-resolution and segmentation using generative adversarial networks—Application to neonatal brain MRI," *Comput. Biol. Med.*, vol. 120, May 2020, Art. no. 103755.
[32] C. Han, H. Hayashi, L. Rundo, R. Araki, W. Shimoda, S. Muramatsu, Y. Furukawa, G. Mauri, and H. Nakayama, "GAN-based synthetic brain MR image generation," in *Proc. IEEE 15th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2018, pp. 734–738.
[33] C. Bowles, L. Chen, R. Guerrero, P. Bentley, R. Gunn, A. Hammers, D. Alexander Dickie, M. Valdés Hernández, J. Wardlaw, and D. Rueckert, "GAN augmentation: Augmenting training data using generative adversarial networks," 2018, *arXiv:1810.10863*.
[34] A. Beers, J. Brown, K. Chang, J. Peter Campbell, S. Ostmo, M. F. Chiang, and J. Kalpathy-Cramer, "High-resolution medical image synthesis using progressively grown generative adversarial networks," 2018, *arXiv:1805.03144*.
[35] A. K. Mondal, J. Dolz, and C. Desrosiers, "Few-shot 3D multi-modal medical image segmentation using generative adversarial learning," 2018, *arXiv:1810.12241*.
[36] E. Kang, H. J. Koo, D. H. Yang, J. B. Seo, and J. C. Ye, "Cycle-consistent adversarial denoising network for multiphase coronary CT angiography," *Med. Phys.*, vol. 46, no. 2, pp. 550–562, Feb. 2018.
[37] M. J. M. Chuquicusma, S. Hussein, J. Burt, and U. Bagci, "How to fool radiologists with generative adversarial networks? A visual Turing test for lung cancer diagnosis," in *Proc. IEEE 15th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2018, pp. 240–244.
[38] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification," *Neurocomputing*, vol. 321, pp. 321–331, Dec. 2018.
[39] M. Jafari, D. Auer, S. Francis, J. Garibaldi, and X. Chen, "DRU-Net: An efficient deep convolutional neural network for medical image segmentation," in *Proc. IEEE 17th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2020, pp. 1144–1148.
[40] G. Madhupriya, N. M. Guru, S. Praveen, and B. Nivetha, "Brain tumor segmentation with deep learning technique," in *Proc. 3rd Int. Conf. Trends Electron. Informat. (ICOEI)*, Apr. 2019, pp. 758–763.
[41] Z. Dai, Z. Yang, F. Yang, W. W. Cohen, and R. R. Salakhutdinov, "Good semi-supervised learning that requires a bad GAN," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.
[42] B. Li, C. Wu, J. Chi, X. Yu, and G. Wang, "A deeply supervised convolutional neural network for brain tumor segmentation," in *Proc. 39th Chin. Control Conf. (CCC)*, Jul. 2020, pp. 6262–6267.
[43] T. Li, F. Zhou, Z. Zhu, H. Shu, and H. Zhu, "A label-fusion-aided convolutional neural network for isointense infant brain tissue segmentation," in *Proc. IEEE 15th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2018, pp. 692–695.
[44] J. Adler and S. Lunz, "Banach Wasserstein GAN," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 1–10.
[45] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: Learning dense volumetric segmentation from sparse annotation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Greece: Springer, 2016, pp. 424–432.
[46] G. Patel and J. Dolz, "Weakly supervised segmentation with cross-modality equivariant constraints," *Med. Image Anal.*, vol. 77, Apr. 2022, Art. no. 102374.
[47] J. Dolz, K. Gopinath, J. Yuan, H. Lombaert, C. Desrosiers, and I. B. Ayed, "HyperDense-Net: A hyper-densely connected CNN for multi-modal image segmentation," *IEEE Trans. Med. Imag.*, vol. 38, no. 5, pp. 1116–1126, May 2019.
[48] L. Wang *et al.*, "Benchmark on automatic six-month-old infant brain segmentation algorithms: The iSeg-2017 challenge," *IEEE Trans. Med. Imag.*, vol. 38, no. 9, pp. 2219–2230, Sep. 2019.



AFIFA KHALED received the M.Sc. degree in software engineering from Tianjin University, in 2020. She is currently pursuing the Ph.D. degree in computer science with the Huazhong University of Science and Technology (HUST), Wuhan, China. Her research interests include machine and deep learning methods for medical image segmentation.



JIAN-JUN HAN received the Ph.D. degree in computer science and engineering from the Huazhong University of Science and Technology (HUST), in 2005. He is currently a Professor with the School of Computer Science and Technology, HUST. He worked with the University of California at Irvine, Irvine, as a Visiting Scholar, between 2008 and 2009, and Seoul National University, between 2009 and 2010. His research interests include AI algorithm, real-time systems, and parallel computing.

TAHER A. GHALEB received the Ph.D. degree in computing from Queen's University, Canada, in 2021. He is currently a Postdoctoral Research Fellow with the School of EECS, University of Ottawa, Canada. During his Ph.D., he held an Ontario Trillium Scholarship, a Highly Prestigious Award for Ph.D. students. His research interests include continuous integration, software testing, mining software repositories, applied machine learning, program analysis, and empirical software engineering.