IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received June 5, 2019, accepted June 22, 2019, date of publication July 2, 2019, date of current version August 13, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2926448

A Stacked Multi-Connection Simple Reducing Net for Brain Tumor Segmentation

YI DING[1,2], FUJUAN CHEN[1], YANG ZHAO[1], ZHIXING WU[1], CHAO ZHANG[1], AND DONGYUAN WU[1]

[1]School of Information and Software Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu 610054, China
[2]Institute of Electronic and Information Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu 610054, China

Corresponding author: Yang Zhao (zhaoyang@uestc.edu.cn)

This work was supported in part by the Natural Science Foundation of Guangdong Province under Grant 2018A030313354, in part by the Neijiang Intelligent Showmanship Service Platform Project under Grant 180589, in part by the Sichuan Science-Technology Support Plan Program under Grant 2019YJ0636, Grant 2018GZ0236, and Grant 18ZDYF2558, and in part by the National Science Foundation of China–Guangdong Joint Foundation under Grant U1401257.

ABSTRACT It is well known that the Unet has been widely used in the area of medical image segmentation because of the cascade connection in the up-sampling process. However, it does not perform well in dealing with complex medical images, such as brain MRI. In order to achieve better segmentation performance by adopting the Unet, many researchers have paid more attention to stacking the Unet. However, the stacking process leads to a large increase in the number of parameters. This is not a good choice when considering the tradeoff between precision and efficiency. Another problem is that as the depth of the network increases, the excessive loss of information is also a tricky problem. To address those problems, in this paper, we are trying to improve the network structure of Unet to make it more suitable for brain tumor segmentation. We propose a novel framework called Stack Multi-Connection Simple Reducing_Net(SMCSRNet) that are stacked by our basic blocks called Simple Reducing_Net(SRNet). The basic block SRNet is improved from the original Unet, which consists of four downsampling and upsampling operations during the encoding and decoding. Only one convolution operation is performed before each downsampling process. The operation of copy and crop is preserved between encoding and decoding. The main advantage of the SRNet is that the amount of parameters is reduced by 4/5 by comparing with the original Unet. Except for the problem of parameters number, we also proposed a series of bridge connections among the stacked cascade network to improve the loss of information. More specifically, some bridge connections will be adopted before the pooling operation in each layer during the downsampling process. It means that each layer in one basic block has a bridge connection with the same feature size from the previous basic block before pooling, and it is worth noting that the training time of the proposed framework is much less than the original stacked Unet. Moreover, the performance of the proposed method is also improved compared to the stacked Unet. When further comparing with other state-of-the-art segmentation networks, it can be found that the performance is as good as the most popular DenseNet or ResNet. Overall, by evaluating the proposed framework on the BRAT2015, it can be proven that the proposed segmentation network has the ability to accurately extract the brain tumor boundary so as to obtain higher recognition quality with high efficiency.

INDEX TERMS Semantic segmentation, Unet; stacked Unet, feature fusion, medical image segmentation, brain tumor segmentation, deep convolutional neural networks.

I. INTRODUCTION

In recent years, with the increase of unhealthy diets and environmental pollution, there are more and more patients with brain tumors. However, the human segmentation method requires a large amount of learning and relevant work experiences. It is also a time-consuming and labor intensive process. Therefore, the automatic brain tumor segmentation method has emerged, and is mainly used to support the diagnostic process. Among different brain tumors, glioma is one of the most common tumors for the central nervous system, accounting for about half of the primary intracranial tumors.

The associate editor coordinating the review of this manuscript and approving it for publication was Kim-Kwang Raymond Choo.

In clinical practice, how to accurately segment brain tumors for further diagnosis and treatment planning becomes a key step.

Brain tumor images are more complicated than natural images. There are two main reasons. The first one is that the resolution of different medical images is different, especially the gray value greatly changes among different images. The second reason is about the brain tumor. There are too many changes for the tumor morphology in different images, and there are also too many noises in medical images, but many of them are not well distinguished from morphology and gray scale. In recent years, many researchers have made many efforts on the traditional machine learning method [1] and the deep learning method [2]. Traditional machine learning algorithms mainly include principal component analysis proposed by Kaya *et al.* [3], fuzzy c-means Hsieh and region growing algorithm [4], multilevel fuzzy c-means [5], and Gabor filter [6]. However, due to the limitations of traditional machine learning in the field of computer vision, its performance is not satisfactory. At present, the deep learning algorithms have been widely used in various industries [26]–[29], for example, it has been widely used to solve problems in computer vision, and has achieved great success in image recognition. Therefore, adopting the convolution neural network for medical image analysis has attracted wide attention. Relevant works mainly include: the patch-based brain tumor segmentation proposed by LLA [7], the patch-based multi-scale CNN [8] proposed by Havaei, the patch-based DCNN [9] proposed by Kamnitsas, and the FCN, full convolution-based CNN [10] proposed by Zhao.

According to research [16], the increase of network depth can better capture semantic features. In order to achieve better segmentation performance by adopting the Unet, many researchers have made many efforts to stack the Unet. Stacked Unet increase the depth of the network, but they also bring many problems. There are two main problems. At first, the stacking process lead to a large increase in the amount of parameters. The second, as the stacking level increases, the loss of information is also a tricky problem. These problems have a great influence on the segmentation of brain tumors. When the brain tumor is segmented, the information propagation in the existing deep learning algorithm can be further improved. The low level features are lost with continuous convolution operations. The final prediction, which relies mainly on the top, that will lose the opportunity to gather more different information from the bottom and middle. Specifically, combining low-level features with top-level features is a challenging task because there are long paths between them.

In this paper, we propose a novel framework called Stack Multi-Connection Simple Reducing_Net (SMCSRNet). This network consists of many basic blocks called Simple Reducing_Net (SRNet). Compared to the original Unet, the number of parameters in SRNet are reduced by 4/5. Because of fewer parameters in the SRNet, it is a better choice when stacking the cascade network. What is more, the performance of our model far exceeds of the Stacked Unet. When further comparing with other state-of-the-art segmentation networks, it can be found that the performance is as good as the most popular DenseNet or ResNet. Therefore, our proposed model can accurately locate the tumor boundary of brain tumors to obtain higher recognition quality. The network structure proposed in this paper fully integrated features from the low level to the high level. It can be proven that the proposed framework achieves a well performance on the BRATS2015 dataset.

Our primary contributions include:

i) A basic block called Simple Reducing_Net (SRNet) has been proposed to solve the problem of too many parameters in the stacked Unet. Compared to the original Unet, the number of parameters in SRNet are reduced by 4/5.

ii) Some bridge connections have been proposed to build a series of bridges inside a cascade network. For each layer in one basic block, the proposed connection is used to connect the feature in the same layer from the previous block during the downsampling process. These ''bridges'' are mainly used to make full use of the information and to reserve the useful information as far as possible.

iii) An improved end-to-end framework, which is inspired by the stack Unet, has been investigated and developed. This framework called a Stacked Multi-Connection Simple Reducing_Net (SMCSRNet) was proposed to combine the advantages of both the SRNet and bridge connection. By evaluating the proposed framework on the dataset, it can be proven that it can achieve a better segmentation performance for brain tumors with high efficiency by comparing with the stacked Unet and other state-of-the-art counterpart methods.

II. RELATED WORK

Many models [10]–[14] have boosted the performance of semantic segmentation networks. There are two main frameworks for medical image segmentation, one based on CNN and the other based on FCN.

The first-class frameworks are based on CNN framework, and the idea is also simple: to classify each pixel of the image, take a patch at each pixel, and use it as an image to enter the neural network for training. the convolutional neural networks (CNN) not only achieve good image classification, but also make great progress in the segmentation problem. Initially, image patches classification is a commonly used deep learning method, in which each pixel is separately divided into corresponding categories by using image blocks around each pixel.

The second-class frameworks are based on the FCN framework: in the field of medical image processing, it can be seen that it extends the original CNN structure to enable intensive prediction without a fully connected layer, and the input and output are images. The shallower high-resolution layer is used to solve the problem of pixel positioning, and the deeper layer is used to solve the problem of pixel classification. Almost all recent research on semantic segmentation adopted this structure.

This paper is mainly inspired by the encoder-decoder form. We are trying to improve Unet by the way of stacking cascade to make it more suitable for brain tumor segmentation. The concept of cascade network exists in a large number of computer vision tasks. However, the information transmitted between cascaded sub networks is usually chosen differently and sometimes implicit in the structure of problem being solved.

In [15], a cascade of two Unet is applied to the segmentation of liver and lesion in CT images as the backbone of model, followed by a 3D Conditional Random Field(CRF). Subsequently, the lesion is a small area in the liver. The cascade application is as follows: the first Unet block is used to segment the liver, and then its local ROI is transmitted to the second Unet block. Experiments show that compared with single Unet, the Dice score can be increased by 20%.

A learning-based stacked Unet called DocUNet firstly proposed by Vision Technology Face++ [16]. They found in the experiment that the output of a single Unet is not satisfactory and should be optimized, so another U-Net is stacked on the output of the first Unet as a refiner. DocUNet can smooth and restore distorted document images. And it fills a technical gap in the field of deep learning. Due to the effectiveness and efficiency of flattened document images, DocUNet can significantly reduce the difficulty of text recognition, optimize the development of OCR technology. What is more, DocUNet can promote text recognition and retrieval capabilities in real world, network and other scenarios.

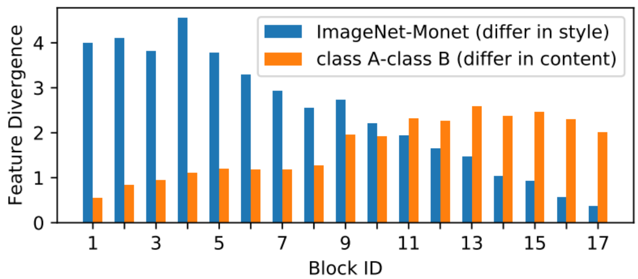


FIGURE 1. Feature divergence calculated from image sets with appearance difference (blue) and content difference (orange).

According to research [17], the increase of network depth can better capture semantic features. This paper [17] shows feature divergence calculated via symmetric KL divergence. As shown in Fig.1, the x-axis denotes the depth of a network and the y-axis shows feature divergence calculated via symmetric KL divergence, the divergence decreases as layer depth increases, manifesting the appearance difference mainly lies in shallow layers. On the contrary, compared with two disjoint ImageNet splits (orange bar), the object level difference attributes to majorly higher layer divergence and partially low layer ones. Based on these observations, they introduce IN layers to CNNs following two rules. Firstly, they want to reduce feature variance caused by appearance in shallow layers while not interfering the content discrimination in deep layers. Secondly, they also want to preserve image content information in shallow layers. These are proved in the experiment, and IBN-Net proposed by them significantly improves performance across domains.

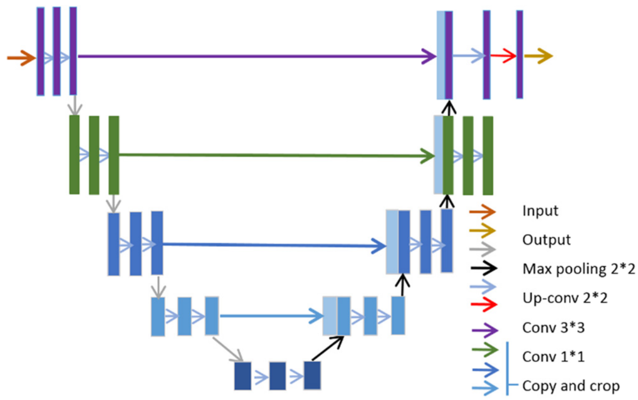


FIGURE 2. The architecture of Unet.

Unet is proposed by Ronneberger *et al.* [18] originally for medical image segmentation. This network structure is based on multi-scale and won the ISBI cell tracking challenge 2015 with a large margin. As shown in Figure 2, Unet [18] consists of two parts, the first part is feature extraction, which is similar to VGG [19]. In the feature extraction part, there is a scale for each passing through a pooling layer, there are totally five scales including the original image scale. The second part is the upsampling part. In the upsampling part, each time the up-sampling is sampled, then integrate it with what is from the same scale of the channel corresponding to the feature extraction part. But it is cropped before the fusion. The fusion here is also stitching. Among them, the operation of copy and crop combines low-resolution information (providing object-based recognition) with high-resolution information (providing accurate segmentation and positioning), making it suitable for medical image segmentation.

III. METHOD

We propose a novel framework called Stack Multi-Connection Simple Reducing_Net(SMCSRNet) which are stacked by some basic blocks called Simple Reducing_Net(SRNet). We want to further modify Unet, to make it more suitable for stacking to brain tumor segmentation. Our basic block is shown in Fig.3, which have four downsampling/upsampling operations were performed during the encoding/decoding. Only one convolution operation is performed before each downsampling. The operations copy and crop are preserved between encoding and decoding, which maintains Unet's multi-scale feature fusion. The purpose of designing the basic block in this way is to simplify its network structure and further reduce parameters. It is worth noting that on a stacked cascade network, the spending on training time of our model called SMCSRNet is far less than stacked Unet. And our accuracy is improved compared to stacked Unet.

A. THE STRUCTURE OF OUR BASIC BLOCK

The basic block we proposed is shown in Figure 3 (The rectangle marked with a purple dashed line is a basic block, which actually used for stacking), which is obtained by

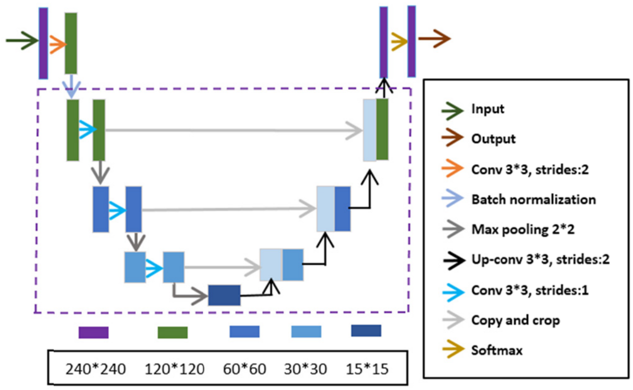


FIGURE 3. The architecture of basic block.

modifying Unet. The left-to-up process is downsampling, and the right-to-bottom process is upsampling. Compared with the Unet structure, the convolution operation is reduced before and after the pooling. In more detail, our model called SMCSRNet performs a convolution only once after each downsampling. A total of 4 downsamplings were performed during the encoding. After downsampling to 15*15, it is sent to the decoding for upsampling. Compared to the original Unet, our basic block SRNet parameters are reduced by 4/5.

And this basic block shows better performance on the stack cascade than the original Unet. We simplify Unet operations, not only reducing the number of parameters, but also significantly reducing training time. This structure is more suitable for stacking and has been proven in experiments.

B. THE NETWORK STRUCTURE OF OUR MODEL CALLED STACKED MULTI-CONNECTION SIMPLE REDUCING_NET (SMCSRNet)

The previous section implemented the operation of building a simple basic block, which facilitated the stacking of the network to a deeper level. But it also brought some problems. For example, as the stacking level increases, the gradient disappears or information is lost. In order to solve this thorny problem, we want to build a series of bridges inside the stacking network. Them can provide rich information for the back-end network during forward training, and minimize the risk of gradient disappearance when the gradient is transmitted back.

To build such a series of bridges, the first step is to determine where to build. The study found that it is easy to lose small object information after pooling in the codec structure, so we choose to build these bridges before the pooling operation. The previous basic block provides supplementary information for the latter base block, enriching each level before pooling. Enter the feature information. The second step is how suitable is the number of bridges? The number of these internal bridges is not as good as possible, when considering the tradeoff between precision and efficiency. If they are too much, it will increase the load on the network, and provide a lot of redundant information to interfere with the final segmentation result.

The model structure proposed in this paper is shown in Figure 4. The rectangle marked by the purple dashed line in this figure represents a basic block, and each basic block is the same in cascade network proposed by us. All the basic blocks except the last one, end with 32 feature maps, which are stacked with the input image by long skip connections (shown light-grey in the figure, the plus sign in the figure represents the contact operation.). The latter provide original information to the basic block, so that it refines the previous features by directly accessing information from the input image. The long skip connections connect all the basic blocks. We proposed adding some bridge connections before pooling of each layer (indicated by the orange dotted arrows in the figure). Each basic block is used to enhance the output features of the previous block. During the downsampling of every basic block, each layer is only connected to the same level feature from the previous basic block before pooling. We chose to fuse features from the previous. We chose to fuse features from the previous basic block rather than the several basic blocks or all the basic block before this basic block. This will be verified in the experiment. We also compare our models with the stacked Unet. Our model called SMCSRNet not only greatly reduces the training time, but also improves the accuracy of brain tumor segmentation.

C. LOSS FUNCTION

As a loss function, we define it as $l(A, B)$:

$$l(A,B) = -\log d(A,B), where: d(A,B) = \frac{2\sum_{i,j} a_{ij}b_{ij}}{\sum_{i,j} a_{ij}^2 + \sum_{i,j} b_{ij}^2} \quad (1)$$

where $A = (a_{ij})_{i=1,j=1}^{H,W}$ is a predicted output map, containing probabilities that each pixel belongs to the foreground, and $B = (b_{ij})_{i=1,j=1}^{H,W}$ is a correct binary output map. d(A, B) is a real-valued extension of Dice score for binary images $Dice(A,B) = \frac{2(|A|\cap|B|)}{|A|\cup|B|}$. where A and B are defined as above.

IV. EXPERIMENT AND RESULT

In this section, we first introduce the database which is provided by MICCA. And then, we will provide detailed network parameters and training details. Finally, a comparison of the segmentation results of the model with other methods.

A. DATASET

The proposed method has been evaluated on the BRAS2015 dataset. In BRATS2015, each MRI image has four different modules: T1, t1c, T2, and Flair, which are regarded as four inputs of the network. RMSProp optimization method was employed and its attributes are set as follows: decay = 0.95, momentum = 0.9, epsilon = 1e-8. The size of the input is 240*240*4 and the batch size is 10 in our experiments. The BRATS2015 datasets have been further divided into two categories: low-grade gliomas (LGG) and high-level gliomas (HGG), depending on the tumor cell's pathological malignancy.

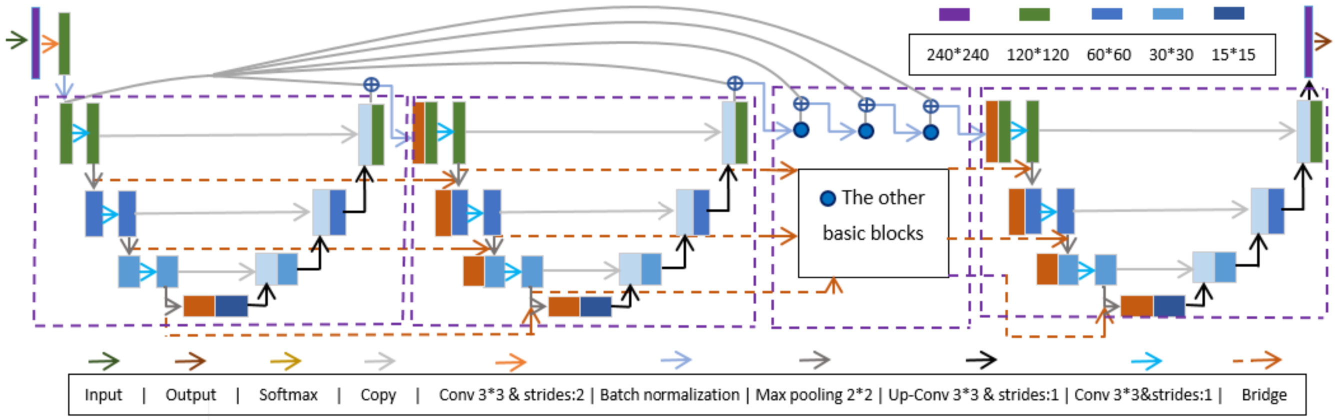


FIGURE 4. The architecture of our model called SMCSRNet.

B. EVALUATION

Three metrics are employed as the evaluation criteria: DSC (Dice Similarity Coefficient), PPV (Positive Predictive Value) and SCS (Sensitivity Coefficient Sensitivity).

The DSC measures the overlap area between the manual and automatic segmentation results. It is defined as:

$$\text{Dice} = 2\text{TP}/(\text{FP} + 2\text{TP} + \text{FN}) \tag{2}$$

where TP represents a positive sample that is correctly segmented to the positive sample point, FP is a negative sample that is wrongly segmented to the positive sample point. FN is a positive sample that is wrongly segmented to the negative sample point.

The PPV represents the proportion of the true positive sample in the positive sets indicated by the experimental the experimental result. The calculation formula is as follows:

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{3}$$

The sensitivity coefficient represents the proportion of the positive case that is correctly determined to account for the total positive case, and reflects the case of the segmented positive sample. The calculation formula is as follows:

$$\text{Sensitivity} = \text{TP}/(\text{TP} + \text{FN}) \tag{4}$$

These evaluation metrics will work together to represent the accuracy and error rate of the proposed method for image segmentation.

C. NETWORK STRUCTURE

The network structure of our basic block(SRNet) have the same basic block id as shown in Table 1. According to the experiment, the number of parameters for our basic block is only 0.43M, and this value is 2.15M for the original Unet. In other words, compared to the original Unet, the parameters of the proposed basic block are reduced by 4/5 so as to greatly reduce the computational time.

The batch normalization [23] is applied in first layer. The number of stacked levels of the cascade network can be chosen according to the complexity of the task. The proposed network called Stacked Multi-Connection Simple Reducing_Net is shown in Table 1. Except for the last one, the output of all basic blocks contains 32 feature maps, which are connected with the input image by long skip connections (shown light-grey in the figure4. The latter connection has the ability to provide the original information to the basic block so as to refine the previous features by directly accessing information from the input image. The long skip connections will be employed by all basic blocks.

What is more, the first and last blocks (marked as red in the table 1) differ from other basic blocks. The first basic block has one more layer than other basic blocks to reduce

TABLE 1. The hyper-parameters of our model called SMCSRNeT.

BASIC BLOCK ID	LAYER	FILTRRS#	FILTER SIZE	STRIDES	INPUT SIZE	OUTPUT SIZE
1	LAYER 1	32	3*3	2*2	240*240	120*120
1	LAYER 2	32	3*3	1*1	120*120	120*120
1	LAYER 3	64	2*2	2*2	120*120	60*60
1	LAYER 4	64	3*3	1*1	60*60	60*60
1	LAYER 5	128	2*2	2*2	60*60	30*30
1	LAYER 6	128	3*3	1*1	30*30	30*30
1	LAYER 7	256	2*2	2*2	30*30	15*15
1	LAYER 8	128	3*3	2	15*15	30*30
1	LAYER 9	64	3*3	2	30*30	60*60
1	LAYER 10	32	3*3	2	60*60	120*120
			NEXT BLOCKS			
2	LAYER 1	32	3*3	1*1	120*120	120*120
2	LAYER 2	64	2*2	2*2	120*120	60*60
2	LAYER 3	64	3*3	1*1	60*60	60*60
2	LAYER 4	128	2*2	2*2	60*60	30*30
2	LAYER 5	128	3*3	1*1	30*30	30*30
2	LAYER 6	256	2*2	2*2	30*30	15*15
2	LAYER 7	128	3*3	2	15*15	30*30
2	LAYER 8	64	3*3	2	30*30	60*60
2	LAYER 9	32	3*3	2	60*60	120*120
2 < I < N	THE OTHER BLOCKS ARE SAME AS PREVIOUS BLOCKS EXCEPT LAST ONE					
N	LAYER 1	32	3*3	1*1	120*120	120*120
N	LAYER 2	64	2*2	2*2	120*120	60*60
N	LAYER 3	64	3*3	1*1	60*60	60*60
N	LAYER 4	128	2*2	2*2	60*60	30*30
N	LAYER 5	128	3*3	1*1	30*30	30*30
N	LAYER 6	256	2*2	2*2	30*30	15*15
N	LAYER 7	128	3*3	2	15*15	30*30
N	LAYER 8	64	3*3	2	30*30	60*60
N	LAYER 9	32	3*3	2	60*60	120*120
N	LAYER 10	5	3*3	2	120*120	240*240

TABLE 2. The result of baseline without pre-processing.

Network.	Stacking levels	Index					
		Dice			PVP	SCS	Training time
Stacked Unet	one	0.74	0.56	0.53	0.74	0.82	5h_34min
	two	0.76	0.61	0.57	0.74	0.85	10h_02min
Stacked RUnet	one	0.74	0.58	0.53	0.73	0.82	3h_36min
	two	0.76	0.60	0.56	0.74	0.86	4h_29min

the size of the original image into half, then feeding it to the cascade network for training. The last block also has an additional layer compared to other basic blocks, which is used to restore the size of the segmentation result (120*120) to the original image size 240*240. The filters in the last row is set to five because it is a five classification problem in the experiment. And this value can be changed depending on how many categories are needed to be classified for the final result.

Furthermore, the bridge connections have been employed during the stacking process, which are used to improve the cascaded network by comparing with other cascaded networks. The mainly improvement is that during the down-sampling process, the layer not only code the information from the upper level but also utilize the features of the same level from the previous basic block. These bridge connections seem to build a series of bridges within the stacking network. They can provide rich information for the back-end network during forward training, and minimize the problem of vanishing gradient during the back propagation based on the gradient descent direction of search.

For the network training, all networks are trained with the training set which contains 274 patient cases, and are tested with the testing set which contains 110 patient cases. During the network training, some parameters are set as follows: The batch size is set to 10, the learning rate is 4e-5 and the epoch is 12. All models have been implemented by using the TensorFlow framework and were running on one Nvidia GTX 1080 Ti with the INTEL i7-7700k.

D. THE EFFECT OF DIMENSIONALITY REDUCTION

In order to raise the training efficiency of the network, we will discuss the effect of dimensionality reduction in this section. It means that the proposed framework was expected to improve the segmentation efficiency through a dimensionality reduction method and all following experiments plan to reduce the dimension of the input image. Two models have been constructed and total 4 experiments have been evaluated. The first model builds a cascade network with the basic block of original Unet [18] called Stacked Unet, and the second model adopts the reduced-dimensional Unet as the basic block to construct the cascade network called Stacked RUnet. The difference is the step size of the first convolution layer. The step size of the first convolution layer is 1 in the Stacked Unet, while it is set to 2 in the Stacked RUnet. The purpose of this setting is to halve the size of the image before it is entered into the model.

More specifically, the input and output size of each basic block in Stacked Unet is 240-240_240....240_240-240, while the Stacked RUnet reduced the original image size to half as the input of the network, and the input and output size for each block is 240-120_120....120_120-240. The size will be restored to the original size at the last layer in the last basic block. Table 2 shows the experiment results of Staked Unet and Stacked RUnet. It can be found that these two models achieve the almost same performance in terms of disc, pvp and scs. In other words, halving the size of the image does not affect the segmentation accuracy for the lesion area. In opposite, the training time for the Stacked RUnet fell by almost half. It indicates that keeping the original image size as the input will cost a large amount of computing resources with the number of stacked basic blocks increases in the cascade network. Therefore, the forthcoming experiments will follow the idea of dimensionality reduction and reduce the original image size to half as the input of the first basic block. In addition, the Stacked RUet will be adopted as a baseline network in the following experiment.

E. THE EFFECTS OF SRNet

Since the dataset in BRATS 2015 is collected from different machines, and the original distribution of the data is totally different. Therefore, in order to balance the dataset, the z-score normalization operation is adopted as the pre-processing step for optimizing the dataset. The z-score normalization can be defined as follows:

$$y = (x - \text{mean}\,(x))\,/\text{std}(x) \tag{5}$$

All left experiments in the following sections will be discussed both in with pre-processing and without pre-processing.

In this section, the proposed basic block SRNet has been evaluated by comparing with the Stacked RUnet mentioned in the previous section. The main difference is that the basic block in the cascade network. One is the Reducing Unet in baseline and the other is the SRNet in our model. A total of 30 experiments have been evaluated on these two models, which are in terms of disc, pvp and scs.

The Stacked RUnet in Table 3 is only stacked to 7 levels. This is because that the performance of Stacked RUnet

TABLE 3. The results of stacked runet or stacked SRNeT without any pre-processing.

Net.	Stacked RUnet						Stacked RNet					
Stacking levels	Index						Index					
	Dice			PVP	SCS	Paras(M)	Dice			PVP	SCS	Paras(M)
1	0.74	0.58	0.53	0.73	0.82	2.146	0.72	0.58	0.47	0.67	0.85	0.4347
2	0.76	0.60	0.56	0.74	0.86	4.290	0.74	0.61	0.53	0.68	0.88	0.89
3	0.77	0.62	0.56	0.74	0.85	6.435	0.74	0.57	0.51	0.67	0.86	1.3422
4	**0.78**	**0.62**	**0.56**	**0.73**	**0.86**	**8.58**	0.75	0.58	0.54	0.69	0.84	1.7944
5	0.77	0.62	0.56	0.73	0.87	10.725	0.76	0.59	0.53	0.70	0.86	2.2466
6	0.77	0.61	0.56	0.72	0.87	12.869	0.76	0.60	0.54	0.71	0.85	2.6989
7	0.76	0.63	0.56	0.72	0.85	15.014	**0.77**	**0.60**	**0.54**	**0.70**	**0.85**	**3.1511**
8	-	-	-	-	-	-	0.77	0.59	0.53	0.71	0.84	3.6033

TABLE 4. The results of stacked runet or stacked SRNeT with pre-processing.

Net.	Stacked RUnet						Stacked RNet					
Stacking levels	Index						Index					
	Dice			PVP	SCS	Paras(M)	Dice			PVP	SCS	Paras(M)
1	0.812091	0.633909	0.572818	0.777	0.889	2.1461	0.78291	0.61427	0.56445	0.67264	0.87164	0.4377
2	0.818182	0.635818	0.572	0.785091	0.891909	4.2907	0.80736	0.64964	0.58418	0.76782	0.89464	0.89
3	**0.819909**	**0.628**	**0.587818**	**0.791**	**0.891364**	**6.4353**	0.81027	0.65673	0.58909	0.77591	0.89054546	1.3422
4	0.811545	0.654364	0.570727	0.767727	0.898	8.5799	0.814	0.65364	0.58755	0.77627	0.89355	1.7944
5	0.804636	0.651909	0.578818	0.699273	0.908091	10.7245	0.821	0.661	0.58573	0.785	0.8987	2.2466
6	0.797273	0.650818	0.568727	0.757909	0.901364	12.8691	**0.82336**	**0.66127**	**0.59245**	**0.77945**	**0.90718**	**2.6989**
7	0.796745	0.637818	0.569455	0.756545	0.903636	15.0137	0.81818	0.65318	0.58982	0.778	0.90155	3.1511
8	-	-	-	-	-	-	0.81827	0.65764	0.58445	0.77455	0.90591	3.6033

has kept declining from the stacking level 5. Moreover, the amount of parameters in the stacking level 7 has reached to 15 million. There is no need to stack more levels for the Stacked RUnet. The experiment of Stacked RUnet and Stacked SRNet in different stacking level without pre-processing show in table 3. It can be seen that, the Stacked RUnet can achieve a disc up to 0.78 for the complete tumor, 0.62 for the tumor core, and 0.53 for the enhancing tumor, while it is 0.77, 0.60 and 0.54 for Stacked SRNet. The best performance of the Stacked RUnet is also little better than Stacked SRNet in terms of PVP and SCS. In addition, in the same stacking level, the Stacked RUnet achieves a little better performance than Stacked SRNet in terms of all metrics. This is because Stacked SRNet is designed to pursue the efficiency with fewer convolution operations in the basic block so as to result in poor generalization performance. However, it should be noticed that the number of parameters of Stacked SRNet is much less than the Stacked RUnet and it is only up to almost 1/3 of the Stacked RUnet when achieving the best performance. In particularly, when stacking the basic block as the cascade network, the performance of these two models is almost the same at the same stacking level. But the number of parameters of Stacked SRNet is less than 1/5 of the Stacked RUnet. It can be proven that the proposed SRNet can not only maintain the high accuracy, but also greatly improve the computational efficiency.

The Table 4 shows the segmentation result of Stacked RUnet and Stacked SRNet with pre-processing. As shown in Table 4, the best disc of Stacked RUnet were 0.819, 0.654, and 0.587, respectively, while the proposed Stacked SRNet were 0.823, 0.661, and 0.592. It can be found that the best performance of Stacked SRNet exceeded the Stacked RUnet after adopting the pre-processing. It means that it has the ability to capture more useful information, but its generalization performance is not very well. Specifically, the Stacked SRNet achieves the best performance when stacking at the sixth level, and the Stacked RUnet achieves the best performance at the third level. But the performance of Stacked SRNet at the fifth level has been exceeded the best performance of the Stacked RUnet. What's more important, the parameters of Stacked SRNet are about 1/3 of the Stacked RUnet.

Overall, according to these experiments, four findings can be summarized as follows:

Firstly, the most important point is that the proposed SRNet reduces by 4/5 parameters compared to the original Unet when being adopted as the basic block to construct the

TABLE 5. The results of stacked runet or stacked SRNeT with pre-processing.

Net.	Stacked RUnet						SMCSRNet					
Stacking levels	Index						Index					
	Dice			PVP	SCS	Paras(M)	Dice			PVP	SCS	Paras(M)
1	0.74	0.58	0.53	0.73	0.82	2.146	0.72	0.58	0.48	0.67	0.85	0.4347
2	0.77	0.61	0.56	0.74	0.86	4.312	0.75	0.6	0.54	0.70	0.86	1.1296
3	**0.79**	**0.63**	**0.57**	**0.73**	**0.87**	**6.477**	0.77	0.62	0.55	0.74	0.85	1.7017
4	0.77	0.62	0.57	0.72	0.87	8.643	0.78	0.61	0.57	0.75	0.85	2.4857
5	0.78	0.61	0.55	0.75	0.86	10.808	0.79	0.62	0.57	0.76	0.86	3.2051
6	0.77	0.63	0.56	0.73	0.87	12.973	**0.80**	**0.63**	**0.58**	**0.74**	**0.86**	**3.8969**
7	0.77	0.63	0.57	0.73	0.85	15.139	0.78	0.63	0.57	0.74	0.86	4.5888
8	-	-	-	-	-	-	0.79	0.63	0.58	0.75	0.85	5.2806

cascade network. Moreover, it also improves the segmentation performance after adopting the pre-processing step. Both the effectiveness and efficiency of the proposed basic block SRNet have been demonstrated through these experiments.

Secondly, we find that as the stacking level of network increases, the accuracy becomes more and more better. But after the best performance is achieved, the network becomes stable and then the performance gradually decreases. It is easy to understand that after the network reaches a certain scale, the overfitting problem can't be avoided and the generalization ability of the model will be decreased.

Thirdly, even with the pre-processing, the performance of Stacked SRNet is still not as good as the Stacked RUnet in the first three layers. The reason is that the basic block SRNet only performs one convolution operation per layer, but the Unet has three convolution operations on each layer. Fewer convolution operations represent the model itself is restricted in the ability of capturing information.

Fourthly, compared to the Stacked Unet, although the efficiency has been greatly improved, the effectiveness of Stacked SRNet seems to be more dependent on the preprocessing. In other words, the proposed Stacked SRNet is with poor generalization ability in these experiment.

Therefore, in order to further improve the effectiveness of proposed model in brain tumor segmentation tasks, we have investigated and developed the bridge connections to enhance the generalization ability for the proposed model. More experiments have been done to evaluate the proposed bridge connection in the following sections.

F. THE EFFECTS OF BRIDGE CONNECTIONS

In this section, we mainly discuss the effects of the proposed bridge connections (B_Conns). These bridged connections seem to build a series of bridges inside the stacking cascade network to improve the information propagation, so as to enhance the generalization performance of the model. These bridged connections can provide more abundant information for the subsequent basic blocks during the forward training and also try to avoid vanishing gradient problem during the back propagation. In addition, it has the ability to remain some information to enhance the relationship between the previous layer and the current layer in different stacking blocks.

Table 5 presents the experiment results of Stacked RUnet and SMCSRNet by adopting the bridged connection and be without the pre-processing. Because the proposed basic block SRNet is very simple, the bridge connections are useful for capturing the information. Compared to the result in Table 3, where the Stacked RUnet and the Stacked SRNet are without bridged connections, it can be found that when stacking the basic block at the same level, the performance of Stacked RUnet has been raised by about 1% to 2% after adopting the bridged connections, while the performance has been raised by about 1% to 3% for the SMCSRNet. In detailed, the best performance of Stacked RUnet is 0.79, 0.63, and 0.57, respectively. And the proposed SMCSRNet can achieve the best result of 0.80, 0.63, and 0.58. In addition, the parameters of the proposed model are 1/3 of the Stacked RUnet when both achieving the best performance. According to this experiment, it can be clearly seen that the performance of SMCSRNet is superior to the Stacked RUnet in the most stacking level after adopting the bridged connection method. In other words, it can be indicated that the bridge connection has the ability to enhance the generalization performance both for the Stacked RUnet and the proposed model.

However, the SMCSRNet still perform worse than the Stacked RUnet in the first three layers. For the number of parameters, even if stacking to three layers, parameters of SMCSRNet are still less than the original Unet without any stacking. Overall, the parameter of SMCSRNet is only 1/4 or 1/3 of the Stacked RUnet. On the other hand, it is well known that too fewer convolution operations result in a limited ability for shallow networks to capture the useful information. After adopting the bridge connection method, the proposed model has been arisen by 1% to 3%. It can be proven that the proposed bridge connection has the ability of capturing the useful information, enhancing the relationships

TABLE 6. The comparison results of stacked runet or SMCSRNeT with pre-processing.

Net.	Stacked RUnet						SMCSRNet					
Stacking levels	Index						Index					
	Dice			PVP	SCS	Paras(M)	Dice			PVP	SCS	Paras(M)
1	0.812091	0.633909	0.572818	0.777	0.889	2.1461	0.78291	0.61427	0.56445	0.67264	0.87163636	0.4377
2	0.821273	0.641182	0.582818	0.721	0.891	4.312	0.81818	0.63582	0.572	0.78509	0.89190909	1.1296
3	**0.823364**	**0.661273**	**0.592455**	**0.779455**	**0.907182**	**6.477**	0.819	0.65682	0.583	0.78173	0.89755	1.7017
4	0.81	0.641273	0.573636	0.768091	0.897273	8.643	0.82355	0.65982	0.588	0.77545	0.89845455	2.4857
5	0.808455	0.653091	0.567636	0.757818	0.904818	10.808	0.82345	0.66127	0.59382	0.78955	0.89527273	3.2051
6	0.805545	0.644364	0.564455	0.748455	0.910364	12.973	0.824	0.65709	0.58155	0.78864	0.90045455	3.8969
7	0.794455	0.648273	0.569182	0.768273	0.902	15.139	0.82336	0.66991	0.587	0.788	0.90209091	4.5888
8	-	-	-	-	-	-	**0.82491**	**0.67209**	**0.58264**	**0.78336**	**0.89963636**	**5.2806**

between different blocks, and making full use of the information obtained by each block in a network structure.

The experiments in Table 6 is similar to Table 5 respect to the bridge connection but these new experiments adopt the pre-processing step. The best performance of SMCSRNet was improved to the 0.82, 0.67 and 0.58 respectively. Interestingly, the proposed model only needs to stack to 4 levels (marked with purple color in the table) to achieve a better performance compared to the best results of the Stacked RUnet. And when continuous increasing the stacking level, the performance of the proposed model keeps to become better and better. Furthermore, although there are 4 levels stacked in our mode1, the amount of parameters at this time is about 1/3 of the Stacked RUnet, which obtains the best results at the stacking level 3.

Through these comparative experiments, it can be found that the bridge connection can help the cascaded network to better learn the features whether they are with or without pre-processing. When adopting the bridged connections for both the Stacked RUnet and the proposed model, the segmentation accuracy of the proposed model is more than about 1% to 2% above the Stacked RUnet in the same stacking level. Moreover, this additional calculation produced by the bridge connection is almost negligible. In summary, there are four key points in these experiments:

First, after adopting the proposed bridge connection method, both the Stacked RUnet and the proposed model achieve a significant improvement, especially being with the pre-processing step. It can be proven that this bridge connection method has the ability to enhance the robustness of the model. In other words, the effects of bridge connection can be proved.

Second, the bridge connection method has a bigger impact on the proposed model whether it is with or without pre-processing. And the best performance under without-processing has exceeded the Stacked RUnet, which also adopts the bridge connection method. It means that the aforementioned generalization problem of the stacked SRNet has been improved by the bridge connection method.

Third, after adopting the bridge connection method, the number of parameters of the proposed model is still much less than the Stacked RUnet. It still has huge advantages on the computational efficiency.

At last but not the least, when stacking the SRNet at the 8 level, the performance of the proposed model still keeps an increasing trend. Therefore, it is necessary to further stack the SRNet to a deeper level so as to explore its optimal stacking level or its best performance. On the other hand, in this experiment, the bridge connection is just used to build connections from the previous block to the current block.

However, if more bridge connections are built to connect the current block from all previous blocks, the performance will become better or not? And How about the computation efficiency for different number of bridge connections? These questions will be further discussed in the next section.

G. DISCUSSION ON THE STACKING LEVEL AND THE NUMBER OF BRIDGE CONNECTIONS

In this section, in order to obtain the best performance of the proposed model, some experiments have been done to discuss how many levels should be stacked and how many previous basic blocks should be connected for the bridge connections.

Table 7 shows the experiment result of stacked SRNet in different stacking levels, which is without the bridge connection. It can be seen that the disc reaches the highest value of 0.823, 0.661 and 0.592 when stacking the basic SRNet blocks with level 6 in the cascade network. In addition, the amount of parameters at this time is only equivalent to an original Unet. But when stacking more basic blocks (from level 7 to 12) in the cascade network, the performance gradually decreases and the number of parameters continuous increases.

Overall, the performance of stacked SRNet will become better with the stacking levels increasing, and the best

TABLE 7. The result of stacked SRNeT in different levels.

Stacking	Index					
levels	Dice			PPV	SCS	Paras(M)
1	0.78291	0.61427	0.56445	0.67264	0.87164	0.4377
2	0.80736	0.64964	0.58418	0.76782	0.89464	0.89
3	0.81027	0.65673	0.58909	0.77627	0.89355	1.3422
4	0.814	0.65364	0.58755	0.78173	0.89755	1.7944
5	0.821	0.661	0.58573	0.785	0.8987	2.2466
6	**0.82336**	**0.66127**	**0.59245**	**0.77945**	**0.90718**	**2.6989**
7	0.81818	0.65318	0.58982	0.778	0.90155	3.1511
8	0.81827	0.65764	0.58445	0.77455	0.90591	3.6033
9	0.819	0.65891	0.59227	0.786	0.89673	4.0555
10	0.81655	0.65482	0.58464	0.77755	0.89955	4.5077
11	0.8187	0.662	0.57645	0.784	0.895	4.96
12	0.81227	0.65173	0.588	0.77127	0.89764	5.4122

performance can be achieved at the stacking level 6 with high efficiency. Over all, the performance of stacked SRNet will become better with the stacking levels increasing, and the best performance can be achieved at the stacking level 6 with high efficiency. However, because of the overfitting problem, the performance of stacked SRNet becomes a litter lower than the best performance and keeps stable from the stacking level 7 to 12, where the number of parameters keeps growing. Except for the staking levels, we also verify how many previous blocks should be connected when adopting the bridge connection method so as to build a cascade network that is more suitable for segmenting brain tumors.

TABLE 8. The result of smcsrnet in different levels.

Levels of	Index					
stacked	Dice			PPV	SCS	Paras(M)
1	0.78291	0.61427	0.56445	0.67264	0.87163636	0.4377
2	0.81818	0.63582	0.572	0.78509	0.89190909	1.1296
3	0.819	0.65682	0.583	0.77591	0.89054546	1.7017
4	0.82355	0.65982	0.588	0.77545	0.89845455	2.4857
5	0.82345	0.66127	0.59382	0.78955	0.89527273	3.2051
6	0.824	0.65709	0.58155	0.78864	0.90045455	3.8969
7	0.82336	0.66991	0.587	0.788	0.90209091	4.5888
8	0.82491	0.67209	0.58264	0.78336	0.89963636	5.2806
9	0.82645	0.66091	0.57709	0.783	0.90227273	5.9725
10	**0.83122**	**0.67118**	**0.59191**	**0.78964**	**0.90245455**	**6.6643**
11	0.82164	0.66573	0.58236	0.78482	0.90081818	7.3561
12	0.81545	0.66327	0.58709	0.77991	0.89490909	8.048

As shown in Table 8, the performance of the SMCSRNet on different stacking levels has been presented, where the bridge connection is used to construct connections from each layer in previous block to the same layer in current block between two adjacent blocks. The best performance for the SMCSRNet is 0.831, 0.663, 0.593 at the stacking level 10. What's more important, the number of parameters for the best performance is equivalent to the parameters of Stacked RUnet stacked to three levels. This also benefits from the fact that the basic blocks SRNet is very efficiency. Therefore we can stack more basic blocks to further increase the depth of the network so as to obtain a better performance. In addition, the bridge connections are used to explore the relationship between block-to-block and to enhance the effectiveness and robustness of the SMCSRNet.

When adopting the bridge connection method, except for constructing connections just from the previous blocks, another way is to connect from all previous blocks, just like the connections in Dense Network. The aim for these two connection ways is both to make full use of hierarchical information. Compared to the experiments in Table 8, the Table 9 shows the experiment result of the SMCSRNet, where the bridge connection is used to build the connections for each layer in current block from the same layer in all previous blocks. These bridge connections are used to concatenate features at the same level in different blocks.

TABLE 9. The result of SMCSRNeT_dense_bridge in different levels.

Levels of	Index					
stacked	Dice			PPV	SCS	Paras(M)
1	0.78291	0.61427	0.56445	0.67264	0.87163636	0.4377
2	0.80855	0.64145	0.587	0.77045	0.89154546	1.131
3	0.81673	0.66036	0.581	0.77818	0.89527273	1.8214
4	0.81491	0.65927	0.58191	0.77818	0.895	2.5133
5	0.81564	0.66227	0.58109	0.77964	0.89527273	3.5845
6	**0.82305**	**0.66319**	**0.58573**	**0.785**	**0.9087**	**5.0243**
7	0.82217	0.66243	0.59173	0.7801	0.8987	6.8052
8	0.82013	0.6629	0.5827	0.7998	0.9013	8.9271
9	0.81918	0.663	0.58536	0.78	0.899	11.3901
10	0.819	0.66282	0.58545	0.78218	0.89927273	14.1942

As shown in Table 9, it can be clearly seen that the segmentation performance is not improved by building more bridge connections for the each layer in current block instead of only from the previous blocks. Moreover, the number of parameters increased rapidly. In detailed, the highest score was 0.823, 0.663 and 0.591 in this experiment. With the increment of the stacking level, the parameters are growing faster, and the parameters reach 14 million when stacked on the 10th level. In summary, constructing more bridge connections is not benefit for improving the performance, but lead to the explosive growth of parameters.

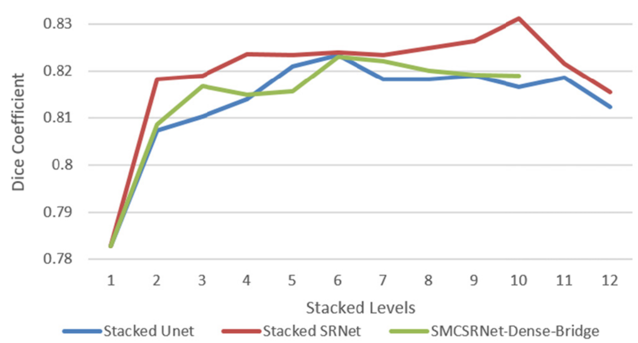


FIGURE 5. The performance comparison on different stacking levels and different bridge connections.

The Figure 5 and 6 represent the performance comparison and efficiency comparison on different stacking levels and on different bridge connection ways respectively.

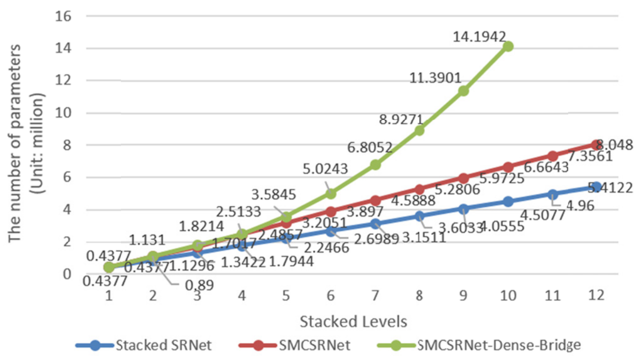


FIGURE 6. **The efficiency comparison on different stacking levels and different bridge connections.**

The "Stacked SRNet" presents the stacked SRNet model without any bridge connection among different blocks. The "SMCSRNet" indicates the stacked SRNet model with the bridge connection, but the bridge connection in this model is just used to build the connection between two adjacent blocks. It means that each layer in one block has a connection from the same layer in the previous block. The "SMCSRNet-Dense-Bridge" means the bridge connections of each layer in one block are concatenated from the same layer in all previous blocks. As shown in Figure 5, both the "SMCSRNet" and "SMCSRNet-Dense-Bridge" obtain the better performance than the "Stacked RNet" after the 6th stacking level, especially the "SMCSRNet" is always the best from the beginning. That's meant that the proposed bridge connection is useful for cascade network to improve the segmentation performance. There are two reasons for this: on the one hand, the proposed bridge connection can increase the effective use of information, and on the other hand, it can alleviate the problem of gradient disappearance due to the increase of stacked blocks in the cascade network.

The number of parameters for these three model in different stacking levels can be seen in Figure 6. It can be found that too many bridge connections in "SMCSRNet-Dense-Bridge" were prone to information redundancy for the model and greatly decrease the computational efficiency. Overall, considering both the effectiveness and efficiency for the model, the SMCSRNet is the best choice for segmenting the brain tumor.

H. THE PERFORMANCE OF SMCSRNET

The comparison chart for the disc of the different models mentioned in this paper is shown in Figure 7. Stacked Unet is the original model, and the Stacked RUnet represents the stacked Unet with dimensionality reduction, which is adopted as the Stacked RUnet in this paper. The "Bridge" following the Stacked Unet and Stacked RUnet indicate that the model adopts the bridge connection method. It can be found the proposed SMCSRNet model achieves the best performance at the stacking level 10 when segmenting the brain tumor. In addition, the SMCSRNet and stacked SRNet can be stacked to

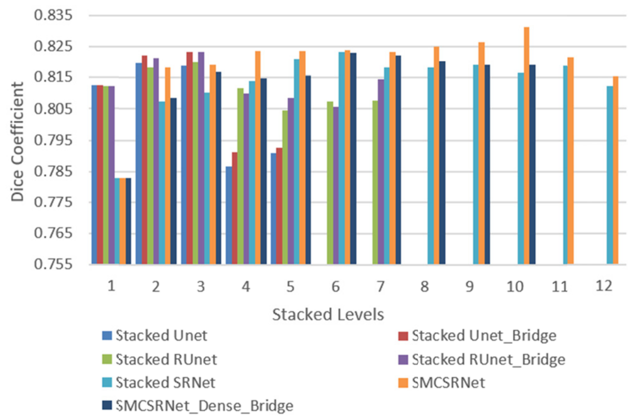


FIGURE 7. **The comparison of different models in terms of dsc.**

a deeper level to achieve a better performance by comparing with other models, especially for the original stacked Unet and the stacked RUnet (w/o the bridge connection), which are only stacked at the stacking level 7.

The reason behind this is that with the increases of stacking levels, these four models lead to an overfitting problem, the performance was coming to a head and gradually decrease. Therefore, it is not necessary to stack more levels for these models. Another reason is that there are too many parameters for these models and it consumes a lot of computational resources and time when stacking too many blocks.

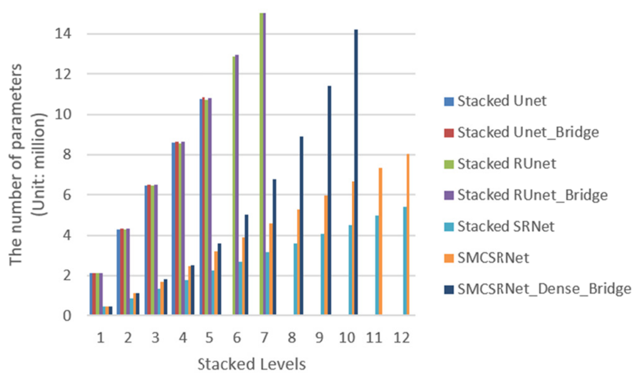


FIGURE 8. **The comparison of different models in terms of parameters.**

Figure 8 shows the efficiency comparison among different models based on the parameters. It can be clearly seen that the stacked SRNet and SMCSRNet always has much less parameters than the stacked Unet at the same stacking level. It can be proven that the efficiency of stacked Unet has been improved by the proposed method. But the parameters of SMCSRNet-Dense raised substantially after the 5th stacking level but are still less than the stacked Unet at the same stacking level. For example, parameters of the SMCSRNet-Dense at the stacking level 10 is less than the original stacked Unet at the stacking level 7.

TABLE 10. Compared with other state-of-the-art models.

Method	Complete tumor	Tumor core	Enhancing tumor	Segmentation time
Pereira[7]	0. 78	0. 65	0. 75	2min
Havaei[8]	0. 79	0. 58	0. 69	25s~3min
Kamnitsas[25]	0. 847	0. 667	0. 634	2min~3min
Our model	0. 831	0. 671	0. 592	Less than 10. 0s

It means that there are too many bridge connections in the SMCSRNet-Dense model because of ''connections from all previous blocks''. Moreover, through adopting the bridge connection in the original stacked Unet and the stacked Unet with dimensionality reduction, it can be found the disc of these two models has been improved but the number of parameters added by the bridge connection is almost negligible. From this point, the effectiveness of the bridge connection can be further proven.

I. COMPARISONS WITH OTHERS

Except comparing with the original stacked Unet, other state-of-the-art brain tumor segmentation methods should be employed as the comparison method to evaluate the effectiveness and efficiency of the proposed framework. These methods are listed as follows: a patch-based segmentation net with a small filter kernel presented by Pereira *et al.* [7]; a multi-scale CNN for patch-based segmentation proposed by Havaei *et al.* [8]. And a 3D multi-scale CNN for patch-based segmentation, in this method, Kamnitsas *et al.* [25] proposed a 3D CRF to optimize the softmax probability maps.

The comparison between the proposed method and counterpart methods is shown in Table 10 in terms of segmentation disc and time. Our model called SMCSRNet outperforms the Pereira method and the Havaei method in terms of complete tumor and tumor core segmentation. But the segmentation performance of our model is slightly worse than the Kamnitsas [25].

In addition, the proposed method performs not well on the enhancing tumor compared to other three methods and is deserved to further reflection. There are two main reasons for this. The first is because the model we proposed is end-to-end, which is used to predict the entire image, and other counterpart methods are mainly based on the patch. The second reason is due to the small number of ''enhanced tumors'' samples. It becomes hard for the network to obtain better performance for the ''enhanced tumors''. However, in term of the computational time, the proposed model costs about 40min for each epoch when training. For the testing time, our network only needs 9.6s to segment one patient brain with 155 slices. As a contrast, Pereira' method takes about 2min [7]. Havaei's network costs 25s∼3min to process a patient brain [8], and Kamnitsas's framework costs 2min∼3min [25] to process a patient brain. In other words, our network outperforms the counterpart methods in terms of efficiency. To sum up, by both considering the effectiveness and efficiency, it can be proven that the proposed model is a state-of-the-art method and also a good choice for segmenting the brain tumor.

V. CONCLUSION

In this paper, in order to solve the problem of large amount of parameters in the stacked Unet, a basic block called SRNet has been proposed, which is improved based on the Unet. Compared to the original Unet, the parameters of basic block SRNet are reduced by 4/5 while the performance keeps the same. Therefore, it is more suitable for building the cascade network. And also, in order to further make full use of hierarchical information among stacked network and improve the performance, the bridge connection method has been investigated and developed to further explore the relationship between two adjacent blocks in the cascade network. Moreover, based on the SRNet and bridge connection, we proposed a novel framework called Stack Multi-Connection Simple Reducing_Net (SMCSRNet). By evaluating the proposed framework on the BRATS2015 datasets, it can be found that the SMCSRNet achieves a better performance with much less parameters by comparing with the original stacked Unet and other counterpart method. In the future, how to introduce the relevant content of migration learning into the bridge connection and further improve the transmission capacity of effective information, so as to achieve better SMCSRNet performance? In addition, it is also a challenging task to apply this network to different types of medical image segmentation tasks and improve the generalization performance of the model by utilizing the correlation between them.

REFERENCES

[1] Y. Ding, C. Zhang, T. Lan, Z. Qin, X. Zhang, and W. Wang, ''Classification of alzheimer's disease based on the combination of morphometric feature and texture feature,'' in *Proc. IEEE Int. Conf. Bioinf. Biomed. (BIBM)*, Nov. 2015, pp. 409–412.

[2] Y. Ding, R. Dong, T. Lan, X. Li, G. Shen, H. Chen, and Z. Qin, ''Multi-modal brain tumor image segmentation based on SDAE,'' *Int. J. Imag. Syst. Technol.*, vol. 28, no. 1, pp. 38–47, Mar. 2018.

[3] I. E. Kaya, A. A. Pehlivanl, E. G. Sekizkardeş, and T. Ibrikci. ''PCA based clustering for brain tumor segmentation of T1w MRI images,'' *Comput. Methods Programs Biomed.*, vol. 140, pp. 19–28, Mar. 2017.

[4] H. Hooda, O. P. Verma, and T. Singhal, ''Brain tumor segmentation: A performance analysis using K-Means, Fuzzy C-means and region growing algorithm,'' in *Proc. IEEE Int. Conf. Adv. Commun., Control Comput. Technol. (ICACCCT)*, May 2014, pp. 1621–1626.

[5] L. Szilgyi, L. Lefkovits, and B. Benyó, ''Automatic Brain Tumor Segmentation in multispectral MRI volumes using a fuzzy c-means cascade algorithm,'' in *Proc. 12th Int. Conf. Fuzzy Syst. Knowl. Discovery (FSKD)*, 2016, pp. 174–181.

[6] G. Mirajkar and B. Barbadekar, ''Automatic segmentation of brain tumors from MR images using undecimated wavelet transform and gabor wavelets,'' in *Proc. 17th IEEE Int. Conf. Electron., Circuits Syst.*, Dec. 2011, pp. 702–705.

[7] S. Pereira, A. Pinto, V. Alves, and C. A. Silva, ''Brain tumor segmentation using convolutional neural networks in MRI images,'' *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1240–1251, May 2016.

[8] M. Havaei, A. Davy, D. Wardefarley, A. Biard, A. Courville, Y. Bengio, C. Pal, P. M. Jodoin, and H. Larochelle, ''Brain tumor segmentation with deep neural networks,'' *Med. Image Anal.*, vol. 35, pp. 18–31, Jan. 2017.

[9] K. Kamnitsas, C. Ledig, V. F. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, and B. Glocker, ''Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation,'' *Med. Image Anal.*, vol. 36, pp. 61–78, Feb. 2016.

[10] J. Long, E. Shelhamer, and T. Darrell, ''Fully convolutional networks for semantic segmentation,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[11] J. Dai, K. He, and J. Sun, ''Instance-aware semantic segmentation via multi-task network cascades,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 3150–3158.

[12] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, ''Large kernel matters—Improve semantic segmentation by global convolutional network,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4353–4361.

[13] T. M. Hsieh, Y.-M. Liu, C.-C. Liao, F. Xiao, I.-J. Chiang, and J.-M. Wong, ''Automatic segmentation of meningioma from non-contrasted brain mri integrating fuzzy clustering and region growing,'' *BMC Med. Inform. Decis. Making*, vol. 11, no. 1, p. 54, Aug. 2011.

[14] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, ''Rethinking atrous convolution for semantic image segmentation,'' 2017, *arXiv:1706.05587*. [Online]. Available: https://arxiv.org/abs/1706.05587

[15] P. F. Christ, M. E. A. Elshaer, F. Ettlinger, S. Tatavarty, M. Bickel, P. Bilic, M. Rempfler, M. Armbruster, F. Hofmann, M. D'Anastasi, W. H. Sommer, S.-A. Ahmadi, and B. H. Menze, ''Automatic liver and lesion segmentation in CT using cascaded fully convolutional neural networks and 3D conditional random fields,'' in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2016, pp. 415–423.

[16] K. Ma, Z. Shu, X. Bai, J. Wang, and D. Samaras, ''DocUNet: Document image unwarping via a stacked U-Net,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 4700–4709.

[17] X. Pan, P. Luo, J. Shi, and X. Tang, ''Two at once: Enhancing learning and generalization capacities via IBN-net,'' in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 484–500. doi: 10.1007/978-3-030-01225-0_29.

[18] O. Ronneberger, P. Fischer, and T. Brox, ''U-net: Convolutional networks for biomedical image segmentation,'' in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer 2015, pp. 234–241.

[19] K. Simonyan and A. Zisserman, ''Very deep convolutional networks for large-scale image recognition,'' 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556

[20] V. Badrinarayanan, A. Kendall, and R. Cipolla, ''SegNet: A deep convolutional encoder-decoder architecture for image segmentation,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2015. doi: 10.1109/TPAMI.2016.2644615.

[21] G. E. Hinton and R. R. Salakhutdinov, ''Reducing the dimensionality of data with neural networks,'' *Science*, vol. 28, no. 5786, pp. 504–507, Jul. 2006.

[22] X. Yuan, B. Huang, Y. Wang, C. Yang, and W. Gui, ''Deep learning-based feature representation and its application for soft sensor modeling with variable-wise weighted SAE,'' *IEEE Trans. Ind. Inform.*, vol. 14, no. 7, pp. 3235–3243, 2018.

[23] S. Ioffe and C. Szegedy, ''Batch normalization: Accelerating deep network training by reducing internal covariate shift,'' 2015, *arXiv:1502.03167*. [Online]. Available: https://arxiv.org/abs/1502.03167

[24] D. P. Kingma, J. Ba, ''Adam: A method for stochastic optimization,'' 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980

[25] K. Kamnitsas, C. Ledig, V. F. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, and B. Glocker, ''Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation, *Med. Image Anal.*, vol. 36, pp. 61–78, Feb. 2017.

[26] L. Ale, N. Zhang, H. Wu, D. Chen, and T. Han, ''Online proactive caching in mobile edge computing using bidirectional deep recurrent neural network,'' *IEEE Internet Things J.*, vol. 6, no. 3, pp. 5520–5530, Jun. 2019. doi: 10.1109/JIOT.2019.2903245.

[27] Z. Qin, L. Hu, N. Zhang, D. Chen, K. Zhang, Z. Qin, and K.-W. R. Choo, ''Learning aided user identification using smartphone sensors for smart homes,'' *IEEE Internet Things J.*, to be published. doi: 10.1109/JIOT.2019.2900862.

[28] H. Xiong, H. Zhang, and J. Sun, ''Attribute-based privacy-preserving data sharing for dynamic groups in cloud computing,'' *IEEE Syst. J.*, to be published. doi: 10.1109/JSYST.2018.2865221.

[29] D. Chen, N. Zhang, R. Lu, N. Cheng, K. Zhang, and Z. Qin, ''Channel precoding based message authentication in wireless networks: Challenges and solutions,'' *IEEE Network*, vol. 33, no. 1, pp. 99–105, Jan./Feb. 2019.

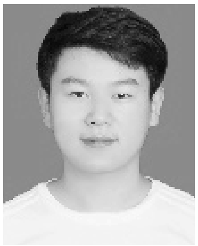
YI DING received the B.S. degree in software engineering from the University of Electronic Science and Technology of China (UESTC), and the Ph.D. degree in computer science from the Dublin Institute of Technology, in 2012. He is currently an Associate Professor with the School of Information and Software Engineering, UESTC, and a Researcher of the Network and Data Security Key Laboratory of Sichuan Province, China. His research interests include machine learning, deep learning, medical image processing, and computer-aided diagnosis.

FUJUAN CHEN is currently pursuing the master's degree with the School of Information and Software Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu, Sichuan, China. Her main research interests include machine learning and deep learning.

YANG ZHAO received the Ph.D. degree from the School of Information and Software Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu, Sichuan, China, where he is currently an Associate Professor. He has long been engaged in research work on network and information security, mobile Internet applications, and smart city top-level design. As a Principal Researcher, he has participated in more than 20 major projects, such as the National Major Science and Technology Special Project, the National Natural Science Foundation, the National Natural Science Foundation of China, the Guangdong Province Industry, Education and Research Key Project, and the Sichuan Science and Technology Plan Project. He also presided over the Basic Research Business Fee Project of Central University, the Sichuan Production and Research Project, the Chongzhou Smart City Top Design Project, the Emeishan Smart City Top Design Project, and the Wenchuan County Smart City Top Design Project. He has made certain contributions in research and development in related fields and has achieved various technological achievements. At present, 13 patents have been applied, of which six have been authorized, more than 20 academic papers have been published, and three textbooks have been published.

ZHIXING WU is currently pursuing the master's degree with the School of Information and Software Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu, Sichuan, China. His main research interests include machine learning and deep learning.

CHAO ZHANG is currently pursuing the master's degree with School of Information and Software Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu, Sichuan, China. His main research interests include machine learning and deep learning.



DONGYUAN WU is currently pursuing the master's degree with the School of Information and Software Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu, Sichuan, China. His main research interests include machine learning and deep learning.