IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 17, 2021, accepted November 14, 2021, date of publication November 23, 2021,
date of current version December 8, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3130035

A Generalized Pooling for Brain Tumor Segmentation

KUANKUAN HAO, SHUKUAN LIN, JIANZHONG QIAO, AND YUE TU
School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China

Corresponding author: Shukuan Lin (linshukuan@cse.neu.edu.cn)

This work was supported in part by the Science Research Project of Liaoning Department of Education of China under Grant LJKZ0008, and in part by the Science and Technology Development Project of Liaoning Province of China under Grant 2021JH6/10500127.

This work involved human subjects or animals in its research. The related images are from open datasets.

ABSTRACT As a common malignant disease, brain tumor has high mortality. The automatic segmentation of brain tumor has significance for clinical diagnosis and surgery treatment. With the development of deep learning, CNN (Convolutional Neural Network) achieves remarkable performance in image processing and computer vision. Researchers have proposed a large number of CNN-based segmentation models such as FCN (Fully Convolutional Network) and Unet from the perspective of network architecture, loss function and attention mechanism. However, most of them are based on the traditional pooling operations such as average pooling and maximum pooling, which will lead to the loss of significant features or average features. Especially in brain tumor segmentation, tissues are usually quite small, so feature losing is more serious. More importantly, the fixed pooling patterns such as maximum pooling and average pooling, which cannot accommodate to varying data, may not be able to accurately express their features in down-sampling. In this study, we first unify maximum pooling and average pooling, and then propose a novel generalized pooling (GP) method with adaptive weights. This is the first work to improve models from the perspective of pooling operations for brain tumor segmentation. The experimental results show that our generalized pooling method is effective to segment brain tumors, outperforming the traditional pooling methods.

INDEX TERMS Average pooling, brain tumor segmentation, convolutional neural network, generalized pooling, maximum pooling.

I. INTRODUCTION

Brain tumor is a common disease of nervous system with higher death rate [1]. The segmentation of brain tumor is not only conducive to its early detection, but also important for locating its position and tissues before surgery treatment. So brain tumor segmentation has been a research hotspot in medical image analysis, which utilizes semantic segmentation technologies in computer vision to achieve automatic segmentation for medical images of brains such as MR (Magnetic Resonance) images.

Here brain tumor segmentation is described as the pixel-wise classification for the MR images of a brain, which assigns an identical category label to the pixels belonging to the same tissue of the brain while assigning different category labels for the pixels of various tissues. Formally, given the $H \times W$ MR image of a brain (H and W are respectively the height and width of the image), brain tumor segmentation returns a mapping $X \rightarrow O$, where $X = \{x_1, x_2, \cdots, x_{H\times W}\}$ is the collection of pixels in the image and $O = \{o_1, o_2, o_3, \ldots o_n\}$ is the collection of tissue categories of the brain (n is the number of categories).

With the development of deep learning, CNN is widely used in brain tumor segmentation due to its good capability of image processing [2]–[7], which integrates feature extraction with segmentation not needing hand-crafted features. At present, the typical CNN models, which are successfully applied in brain tumor segmentation, are FCN (Fully Convolutional Network) [8] and Unet [9]. Unlike plain CNN models, FCN without FC (Fully Connected) layers can be fed an image with arbitrary size and up-samples on the feature map from the last convolution layer by

The associate editor coordinating the review of this manuscript and approving it for publication was Vishal Srivastava.

deconvolution operation to recover the original size of the input image and predict the category of each pixel. So FCN can implement pixel-wise classification tasks including brain tumor segmentation. However, FCN executes up-sampling only once, so the obtained feature map is not accurate enough. Unet is a powerful variant of FCN, which is comprised of an encoder and a decoder. The encoder extracts the feature of an input image from lower layers to higher layers by multiple convolution and down-sampling operations, and the decoder is to recover the original resolution of the input image by multiple up-sampling operations during which the features are fused with those of down-sampling with the same resolutions to provide accurate segmentation. The excellent semantic segmentation effect of Unet inspires many researchers to study on Unet-like networks and many variants have been proposed from the perspective of network architecture, loss function and attention mechanism to further enhance segmentation performance. More details will be discussed in Section 2. In 2020, Hou *et al.* [10] proposed a strip pooling method to improve the segmentation effect of strip objects. This is the first time to improve from the perspective of pooling operation. However, strip pooling is not applicable to brain tumor segmentation because of roughly circular brain tumor tissues. In this paper, we propose a novel pooling method to improve segmentation performance according to the characteristics of brain tumor tissues. In CNN models including FCN and Unet, a convolution layer is often followed by a pooling layer, which helps to decrease dimensions to avoid over-fitting and increase receptive fields while strengthening robustness for some disturbance. The commonly used pooling methods are maximum pooling and average pooling. Maximum pooling extracts the most significant feature within a pooling kernel by selecting the maximal value in the kernel while ignoring other features. Average pooling extracts the average feature by calculating the mean value of the pooling kernel while weakening the extremum in the kernel and leading to the loss of significant feature. So, both maximum pooling and average pooling have their respective drawbacks. On the other hand, each of the two pooling methods applies to different situations. However, in practical applications, a pooling layer may be fed various input images or feature maps without fixed distributions. The fixed pooling patterns such as maximum pooling or average pooling, which cannot accommodate to varying data, may not be able to accurately express the features of down-sampling. Specially, compared with general image semantic segmentation tasks, the objects of brain tumor segmentation are usually quite small, so it is necessary to decrease feature losing as far as possible during down-sampling. To this end, we unify the expressions of maximum pooling and average pooling. Moreover, we propose a novel generalized pooling (GP) method, where the weights in a pooling kernel are adaptively calculated conditioned on the input MR images or feature maps to extract more valuable features during down-sampling and improve segmentation performance.

In this study, we first investigate the problems of the traditional pooling methods in brain tumor segmentation by analyzing their forward and back propagations. Then we unify the expressions of maximum pooling and average pooling and propose a novel generalized pooling (GP) method with adaptive pooling weights for brain tumor segmentation. Thus one does not have to choose either maximum pooling or average pooling for down-sampling in CNN models. To the best of our knowledge, this is the first work to unify these two pooling operations and present a generalized pooling for down-sampling, which makes pooling have diversified expressions conditioned on input images or feature maps. Experimental results on BraTS18 and BraTS19 datasets validate the effectiveness of GP for brain tumor segmentation.

The rest of this paper is organized as follows. Section 2 summarizes the related work on brain tumor segmentation. Section 3 analyzes the problems of the traditional pooling methods through their forward and back propagations. Section 4 unifies the traditional pooling expressions and presents our generalized pooling method. Section 5 discusses experimental results. In Section 6, we conclude this paper and introduce the future work.

II. RELATED WORK

With the development of deep learning, CNN has achieved remarkable performance in image processing and computer vision because of its sparse connectivity and weight sharing. It is also widely used in medical image analysis including brain tumor segmentation. In view of the relationship between pooling operations and CNN, next we will discuss the CNN-based related work on brain tumor segmentation.

Reference [2] is an earlier work which directly applied CNN to brain tumor segmentation. Pereira *et al.* [3] employed small convolutional kernels to construct a deep CNN for brain tumor segmentation, which won the second place in BRATS 2015 Challenge. Long *et al.* [8] proposed FCN (Fully Convolutional Networks) making brain tumor segmentation take a leap, which substitutes deconvolution for the fully connected layers of CNN to directly output segmentation mask with the same size as the original input by up-sampling. Because of the good segmentation capability of FCN, some researchers presented its various variants to further improve segmentation performance. Shen *et al.* [11] proposed a tree-structured, multi-task FCN model for brain tumor segmentation. In the same year, Shen *et al.* [12] introduced a boundary-aware FCN to jointly learn boundary and region tasks in brain tumor segmentation. Jesson and Arbel [13] introduced a 3D FCN architecture with a multiscale loss function to combine higher resolution features with the initial segmentation at a lower resolution.

Another breakthrough of brain tumor segmentation is the advent of Unet [9] due to its high-efficiency and good segmentation performance with fewer training images, which is also known as a variant of FCN. It contains a contracting path for extracting features and a symmetric expanding path

for up-sampling to form a U-shaped architecture. Based on Unet, researchers developed a large collection of variants [14]–[31] to further improve segmentation performance. For example, Milletari *et al.* [14] presented V-net, which is the 3D version of Unet, to process 3D data. Chaurasia and Culurciello [16] introduced residual connection into Unet and proposed LinkNet. Zhou *et al.* further proposed Unet++ [21] to dynamically adjust the depth of the network for different segmentation tasks. Cao *et al.* [25] extended DenseNet to Unet to improve segmentation performance. Huang *et al.* [30] proposed a group cross-channel attention (GCA) module on the basis of Unet to focus on the significant feature groups and channels, and introduced a detail recovering path to recover the fine details of brain tumors. Nguyen *et al.* [31] used Attention U-Net architecture to handle the shape variation for brain tumor segmentation and proposed a new loss function based on active contour loss to overcome the limitation of pixel-wise fitting of the segmentation map on the traditional loss functions.

Although researchers have proposed a large number of models for brain tumor segmentation, most of them presented novel ideas from network architecture, loss function and attention mechanism. In 2020, Hou *et al.* [10] proposed a strip pooling method to improve the segmentation effect of strip objects. This is the first work to improve from the perspective of pooling operation. However, strip pooling is not applicable to brain tumor segmentation because of roughly circular brain tumor tissues. In this study, we present a generalized pooling method for brain tumor segmentation. To the best of our knowledge, this is the first work to improve the models of segmenting brain tumors from pooling operation.

III. ANALYSIS OF THE TRADITIONAL POOLING OPERATIONS

Here, we analyze the problems of maximum pooling and average pooling in brain tumor segmentation through their forward and back propagations.

Maximum pooling selects the maximal value within a pooling kernel as the pooling result. Without loss of generality, we assume that pooling stride is 1. Given an $m \times n$ feature map or input image $X = \{x_{11}, x_{12}, \ldots, x_{mn}\}$ and a $p \times q$ pooling kernel $x_{uv} \sim x_{u+p-1,v+q-1}$ ($1 \le u \le m\text{-}p$, $1 \le v \le n\text{-}q$), where x_{ij} ($1 \le i \le m$, $1 \le j \le n$) is the pixel value at Row i and Column j in X, the maximum pooling function over X can be described as.

$$y_{uv} = \max_{\substack{0 \le r \le p-1 \\ 0 \le s \le q-1}} (x_{u+r,v+s}), \tag{1}$$

where y_{uv} ($1 \le u \le m\text{-}p$, $1 \le v \le n\text{-}q$) is the pooling result within the pooling kernel.

By Eq.1, maximum pooling only keeps the maximal value within a pooling kernel while ignoring other information during down-sampling. This will lead to bad up-sampling results in semantic segmentation models such as FCN and Unet.

Next we analyze the back propagation of maximum pooling. If the maximum pooling result over X is denoted by the matrix Y with size $g \times h$ and error calculated through forward propagation is e,

$$\nabla e(Y) = \frac{\mathrm{d}e}{\mathrm{d}Y} = \begin{bmatrix} \frac{\partial e}{\partial y_{11}} & \frac{\partial e}{\partial y_{12}} & \cdots & \frac{\partial e}{\partial y_{1h}} \\ \frac{\partial e}{\partial y_{21}} & \frac{\partial e}{\partial y_{22}} & \cdots & \frac{\partial e}{\partial y_{2h}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial e}{\partial y_{g1}} & \frac{\partial e}{\partial y_{g2}} & \cdots & \frac{\partial e}{\partial y_{gh}} \end{bmatrix}. \tag{2}$$

For any one element $x_{u'v'}$($u \le u' \le u+p-1$, $v \le v' \le v+q-1$) within the pooling kernel $x_{uv} \sim x_{u+p-1,v+q-1}$ ($1 \le u \le m\text{-}p$, $1 \le v \le n\text{-}q$), by the chain rule, we have

$$\frac{\partial e}{\partial x_{u'v'}} = \sum_{i=0}^{g-1} \sum_{j=0}^{h-1} \frac{\partial e}{\partial y_{ij}} \frac{\partial y_{ij}}{\partial x_{u'v'}}, \tag{3}$$

where

$$\frac{\partial y_{ij}}{\partial x_{u'v'}} = \begin{cases} 1, & x_{u'v'} = \max\limits_{\substack{0 \le r \le p-1 \\ 0 \le s \le q-1}} (x_{u+r,v+s}) \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

By Eq. 3 and Eq. 4, we have

$$\frac{\partial e}{\partial x_{u'v'}} = \begin{cases} \frac{\partial e}{\partial y_{ij}}, & x_{u'v'} = \max\limits_{\substack{0 \le r \le p-1 \\ 0 \le s \le q-1}} (x_{u+r,v+s}) \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

From Eq. 5, during back propagation of maximum pooling, the gradient is transmitted to only one pixel in a pooling kernel. If the pixel values in the pooling kernel are close to each other, singularity will occur during back propagation. This will influence the up-sampling results of the pixels around the maximal value, and the feature map derived by up-sampling will contain much noise.

Average pooling calculates the mean value of the $p \times q$ pooling kernel $x_{uv} \sim x_{u+p-1,v+q-1}$ as the pooling result, as shown in Eq. 6.

$$y_{uv} = \frac{1}{p \times q} \sum_{r=0}^{p-1} \sum_{s=0}^{q-1} x_{u+r,v+s}. \tag{6}$$

For any one element $x_{u'v'}$ in the pooling kernel, its gradient is

$$\frac{\partial e}{\partial x_{u'v'}} = \sum_{i=0}^{g-1} \sum_{j=0}^{h-1} \frac{\partial e}{\partial y_{ij}} \frac{\partial y_{ij}}{\partial x_{u'v'}} = \frac{1}{p \times q} \frac{\partial e}{\partial y_{ij}}. \tag{7}$$

From Eq. 7, the gradient of back propagation in average pooling is transmitted to each pixel equally. For a pooling kernel containing one obvious maximal value, average pooling will weaken the maximal value, which may be the key part in segmentation.

In summary, maximum pooling and average pooling have their respective issues. However, they have one common

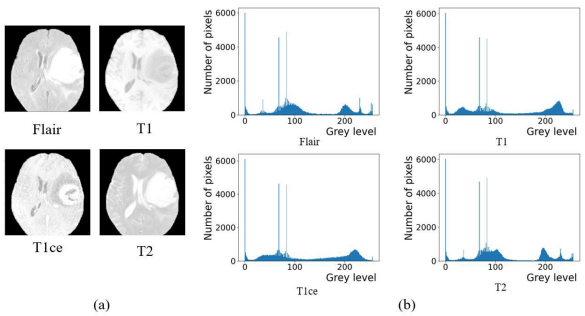

FIGURE 1. The grey level distributions of four modals from one slice. (a) shows the four modals including Flair, T1, T1ce and T2. (b) shows the grey level distribution of each modal.

drawback: their pooling patterns are fixed for various inputs or different pooling kernels over one input although they respectively adapt to different situations, not accommodating to varying data. Once some pooling operation is selected, it will be used over all inputs leading to inaccurate feature extraction. Figure 1 shows the grey level distributions of four modals of one slice (No. BraTS19_CBICA_AOS_1_85) from BraTS19 dataset [32], including Flair, T1, T1ce and T2. We can see that the grey level distribution of each modal is extremely uneven. This means that the pixel value distributions of various pooling kernels even over one input may be different, which needs more flexible pooling pattern to adapt to varying data.

IV. GENERALIZED POOLING OPERATION

In this study, we first unify the expressions of maximum pooling and average pooling, and then we propose a novel generalized pooling method to extract more valuable features during down-sampling and improve segmentation performance.

Given a $p \times q$ pooling kernel, we denote it by a matrix Z for simplicity, as shown in Eq. 8.

$$Z = \begin{bmatrix} z_{11} & z_{12} & \cdots & z_{1q} \\ z_{21} & z_{22} & \cdots & z_{2q} \\ \vdots & \vdots & \vdots & \vdots \\ z_{p1} & z_{p2} & \cdots & z_{pq} \end{bmatrix} \tag{8}$$

For average pooling, the pooling result y is

$$\begin{aligned} y &= \frac{z_{11} + z_{12} + \cdots + z_{pq}}{p \times q} \\ &= \frac{1}{p \times q} z_{11} + \frac{1}{p \times q} z_{12} + \cdots + \frac{1}{p \times q} z_{pq} \end{aligned} \tag{9}$$

The result of maximum pooling y within the pooling kernel Z can be expressed as

$$\begin{aligned} y &= \alpha_{11} \times z_{11} + \alpha_{12} \times z_{12} + \cdots + \alpha_{pq} \times z_{pq} \\ &\begin{cases} \alpha_{rs} = 1, & z_{rs} = max(Z) \\ \alpha_{rs} = 0, & \text{otherwise} \end{cases} \quad 1 \le r \le p, 1 \le s \le q, \end{aligned} \tag{10}$$

where $max(Z)$ expresses the maximal value within the pooling kernel Z.

From Eq. 9 and Eq. 10, we can see that the essence of pooling operations is to find the weighted sum of all elements within a pooling kernel, but the weights are different in different pooling ways. In average pooling, the weights are spread evenly within the pooling kernel. In maximum pooling, the weight 1 is assigned to the maximal value in the pooling kernel and the weights of the remaining elements are all zero. Here, we unify maximum pooling and average pooling and present a generalized pooling way, as shown in Eq. 11.

$$\begin{aligned} y &= \sum_{r=1}^{p} \sum_{s=1}^{q} \alpha_{rs} \cdot z_{rs} \\ 0 \le \alpha_{rs} \le 1 &\text{ and } \sum_{r=1}^{p} \sum_{s=1}^{q} \alpha_{rs} = 1 \end{aligned} \tag{11}$$

Then we can get the gradient of any one element $z_{u'v'} (1 \le u' \le p, 1 \le v' \le q)$ within the pooling kernel Z during back propagation as follows

$$\frac{\partial e}{\partial z_{u'v'}} = \sum_{i=0}^{g-1} \sum_{j=0}^{h-1} \frac{\partial e}{\partial y_{ij}} \frac{\partial y_{ij}}{\partial z_{u'v'}} = \frac{\partial e}{\partial y_{ij}} \alpha_{u'v'} \tag{12}$$

From Eq. 12, we can see that the error is assigned to each element according to its corresponding weight.

In brain tumor segmentation, we utilize this pooling way to avoid compulsorily employing either maximum pooling or average pooling and resulting in the loss of average features or significant features. Moreover, we adaptively generate the weights of the elements within a pooling kernel based on their distribution to decide the most suitable pooling pattern for the current input data. In this way, significant features can be highlighted while other information is not lost during down-sampling. In order to keep average features while not losing significant features, for a $p \times q$ pooling kernel, we first assign an average initial weight α_0 to each element within the pooling kernel, as shown in Eq. 13,

$$\alpha_0 = \frac{1}{p \times q}, \tag{13}$$

and set the mean value $\bar{z}$ of the elements within the pooling kernel as a baseline, as shown in Eq. 14,

$$\bar{z} = \frac{\sum_{r=1}^{p} \sum_{s=1}^{q} z_{rs}}{p \times q}, \tag{14}$$

And then we adjust the weight of each element based on its distance from the baseline to highlight significant features, as shown in Eq. 15.

$$\alpha_{rs} = \alpha_0 + \alpha_0 \cdot \frac{z_{rs} - \bar{z}}{\max\limits_{1 \le r' \le p, 1 \le s' \le q} |z_{r's'} - \bar{z}|}, \tag{15}$$

where α_{rs} is the weight of the element z_{rs} $(1 \le r \le p, 1 \le s \le q)$.

Theorem 1: For $\forall r, s$ such that $1 \le r \le p$ and $1 \le s \le q$, α_{rs} calculated by Eq. 15 meets the requirement of our generalized pooling, i.e., $0 \le \alpha_{rs} \le 1$ and $\sum_{r=1}^{p} \sum_{s=1}^{q} \alpha_{rs} = 1$.

Proof: Let's prove $0 \le \alpha_{rs} \le 1$ first. For $\forall r, s$ such that $1 \le r \le p$ and $1 \le s \le q$, we have

$$z_{rs} - \bar{z} \le \max_{1 \le r' \le p, 1 \le s' \le q} |z_{r's'} - \bar{z}|. \tag{16}$$

So,

$$-1 \leq \frac{z_{rs} - \bar{z}}{\max\limits_{1\leq r'\leq p, 1\leq s'\leq q} |z_{r's'} - \bar{z}|} \leq 1 \tag{17}$$

By Eq. 13, Eq. 15 and Eq. 17,

$$\begin{aligned} 0 \leq \alpha_{rs} &= \alpha_0(1 + \frac{z_{rs} - \bar{z}}{\max\limits_{1\leq r'\leq p, 1\leq s'\leq q} |z_{r's'} - \bar{z}|}) \\ &\leq \alpha_0(1+1) = \frac{2}{p \times q} \end{aligned} \tag{18}$$

Because p and q are the length and width of the pooling kernel, $p \times q \geq 2$. So $0 \leq \alpha_{rs} \leq 1$.

Then we prove $\sum_{r=1}^{p}\sum_{s=1}^{q} \alpha_{rs} = 1$. By Eq. 14,

$$\sum_{r=1}^{p}\sum_{s=1}^{q} z_{rs} - \bar{z}(p \times q) = 0 \tag{19}$$

By Eq. 13,Eq. 15 and Eq. 19,

$$\begin{aligned} &\sum_{r=1}^{p}\sum_{s=1}^{q} \alpha_{rs} \\ &= \sum_{r=1}^{p}\sum_{s=1}^{q} \alpha_0(1 + \frac{z_{rs} - \bar{z}}{\max\limits_{1\leq r'\leq p, 1\leq s'\leq q} |z_{r's'} - \bar{z}|}) \\ &= \alpha_0(p \times q + \frac{\sum_{r=1}^{p}\sum_{s=1}^{q} z_{rs} - \bar{z}(p \times q)}{\max\limits_{1\leq r'\leq p, 1\leq s'\leq q} |z_{r's'} - \bar{z}|}) \\ &= \alpha_0(p \times q) = 1 \end{aligned} \tag{20}$$

□

The pooling algorithm with stride t based on GP is shown in Algorithm 1.

Algorithm 1 Pooling Algorithm Based on GP

Input: an input feature map or image X, pooling kernel size $p \times q$, stride t
Output: pooling result, i.e., a $g \times h$ feature map Y

```
Set the initial weight α0 for each element of X according to Eq. 13
for i = 0 to g-1
    for j = 0 to h-1
        Initialize sum = 0;
        for r = 0 to p-1
            for s = 0 to q-1
                sum = sum + x_{(i×t+r)(j×t+s)}
            end for
        end for
        Find the mean value within the pooling kernel z̄ = sum/(p × q);
        for r = 0 to p-1
            for s = 0 to q-1
                Update the weight α_{r,s} = α0 + α0 · (x_{(i×t+r)(j×t+s)} − z̄) / max_{0≤r'≤p−1, 0≤s'≤q−1} |x_{(i×t+r')(j×t+s')} − z̄|
            end for
        end for
        y_{i,j} = Σ_{r=0}^{p−1} Σ_{s=0}^{q−1} α_{r,s} · x_{(i×t+r)(j×t+s)}
    end for
end for
Return Y
```

From Algorithm 1, we can see that each element in a pooling kernel contributes to the corresponding pooling result and the elements with larger distances from the baseline contribute more. So, in this way, GP can highlight significant features due to their larger weights while not losing other information. The experimental results in Section V validate this point.

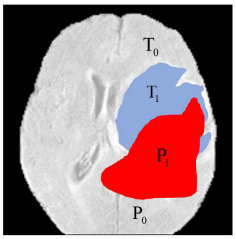

FIGURE 2. True brain tumor region and predicted brain tumor region. The blue part T_1 is true brain tumor region, and the remaining part T_0 except T_1 is the normal region. The red part P_1 is the predicted brain tumor region, and the remaining part P_0 except P_1 is the predicted normal region.

TABLE 1. Comparisons on Dice (Brats18).

	WT-Dice↑	TC-Dice↑	ET-Dice↑
FCN8s	0.8084	0.7584	0.6806
FCN8s_GP	0.8314	0.7649	0.7219
improve	**2.30%↑**	**0.65%↑**	**4.13%↑**
Unet	0.8245	0.7921	0.7209
Unet_GP	0.8347	0.8233	0.7534
improve	**1.02%↑**	**3.12%↑**	**3.25%↑**
Unet++	0.8099	0.5720	0.7069
Unet++_GP	0.8396	0.6594	0.7341
improve	**2.97%↑**	**8.74%↑**	**2.72%↑**

V. EXPERIMENTS

A. EXPERIMENTAL SETUP

The experiments are implemented on a computer with an Intel(R) Core(R) i9-9900k CPU and a NVIDIA GeForce RTX 2080Ti GPU. We adopt Adam optimizer based on Pytorch to train network models, and the initial learning rate and decay rate are set to 0.0001 and 0.9 respectively.

B. DATASETS AND EVALUATION INDICATORS

We conduct the experiments on BraTS18[1] [32]–[34] and BraTS19[2] [32]–[34] datasets, which respectively contain 8952 and 11298 training images, 2888 and 2093 testing images, 2238 and 2825 validating images.

The indicators including Dice, PPV and Sensitivity are used to evaluate our method, which are popular in BraTS Challenge. We illustrate the indicators in Figure 2, where the blue part T_1 indicates the true brain tumor region and the remaining part T_0 except T_1 is the normal region. The red part P_1 indicates the predicted brain tumor region, and the remaining part P_0 except P_1 is the predicted normal region. The samples in $P_1 \cap T_1$ are positive samples and predicted to be positive samples, denoted by TP (True Positive). The samples in $P_0 \cap T_0$ are negative samples and predicted to be negative samples, denoted by TN (True Negative). The samples in $P_1 - P_1 \cap T_1$ are negative samples but predicted to be positive samples, denoted by FP (False Positive). The

[1]https://www.med.upenn.edu/sbia/brats2018/data.html
[2]https://www.med.upenn.edu/cbica/brats2019/data.html

TABLE 2. Comparisons on Dice (Brats19).

	WT-Dice↑	TC-Dice↑	ET-Dice↑
FCN8s	0.8284	0.8528	0.7453
FCN8s_GP	0.8456	0.8634	0.7715
improve	**1.72%↑**	**1.06%↑**	**2.62%**
Unet	0.8192	0.8215	0.7651
Unet_GP	0.8435	0.8736	0.7864
improve	**2.43%↑**	**5.21%↑**	**2.13%↑**
Unet++	0.8424	0.7090	0.7893
Unet++_GP	0.8764	0.7465	0.7926
improve	**3.40%↑**	**3.75%↑**	**0.33%↑**

TABLE 3. Comparisons on PPV (Brats18).

	WT-PPV↑	TC-PPV↑	ET-PPV↑
FCN8s	0.8522	0.8565	0.7435
FCN8s_GP	0.8644	0.8751	0.7645
improve	**0.62%↑**	**1.86%↑**	**2.10%↑**
Unet	0.8644	0.8698	0.7737
Unet_GP	0.8769	0.8733	0.7769
improve	**1.25%↑**	**0.35%↑**	**0.32%↑**
Unet++	0.9074	0.6339	0.7975
Unet++_GP	0.9175	0.6564	0.8175
improve	**1.01%↑**	**2.25%↑**	**2.00%↑**

TABLE 4. Comparisons on PPV (Brats19).

	WT-PPV↑	TC-PPV↑	ET-PPV↑
FCN8s	0.8661	0.8646	0.7708
FCN8s_GP	0.8964	0.8931	0.7724
improve	**3.03%↑**	**2.85%↑**	**0.16%↑**
Unet	0.8561	0.8533	0.7745
Unet_GP	0.8736	0.8861	0.8314
improve	**1.75%↑**	**3.28%↑**	**5.69%↑**
Unet++	0.9239	0.7116	0.8682
Unet++_GP	0.9079	0.7667	0.8801
improve	1.60%↓	**5.51%↑**	**1.19%↑**

samples in $T_1 - P_1 \cap T_1$ are positive samples but predicted to be negative samples, denoted by FN (False Negative).

Dice indicator measures the similarity between the predicted brain tumor region P_1 and the true brain tumor region T_1 by calculating their dice similarly coefficient, as shown in Eq 21. The larger the Dice value, the more similar the two sets $P1$ and $T1$, and the better the segmentation effect.

$$Dice(P_1, T_1) = \frac{|P_1 \cap T_1|}{(|P_1| + |T_1|)/2} = \frac{2TP}{FP + 2TP + FN} \tag{21}$$

PPV indicator measures the precision of prediction by calculating how many samples are correctly predicted (i.e., $P_1 \cap T_1$) in those predicted to be positive ones (i.e., P_1), as shown in Eq 22. The larger the PPV value, the higher the segmentation precision.

$$PPV(T_1, P_1) = \frac{|P_1 \cap T_1|}{P_1} = \frac{TP}{TP + FP} \tag{22}$$

Sensitivity is also called true positive rate. True positive samples contain two parts, the part predicted to be positive samples and the part predicted to be negative samples. Sensitivity expresses the proportion of the correctly predicted samples (i.e., $P_1 \cap T_1$) in true positive ones (i.e., T_1), as shown in Eq 23.

$$Sensitivity(T_1, P_1) = \frac{|P_1 \cap T_1|}{T_1} = \frac{TP}{TP + FN} \tag{23}$$

TABLE 5. Comparisons on Sensitivity (Brats18).

	WT-Se↑	TC-Se↑	ET-Se↑
FCN8s	0.7940	0.8165	0.6886
FCN8s_GP	0.8207	0.8246	0.7514
improve	**2.67%↑**	**0.81%↑**	**6.28%↑**
Unet	0.8087	0.7960	0.7489
Unet_GP	0.8237	0.8264	0.7524
improve	**1.50%↑**	**3.04%↑**	**0.35%↑**
Unet++	0.7867	0.6825	0.7176
Unet++_GP	0.7879	0.7169	0.7367
improve	**0.12%↑**	**3.44%↑**	**1.91%↑**

TABLE 6. Comparisons on Sensitivity (Brats19).

	WT-Se↑	TC-Se↑	ET-Se↑
FCN8s	0.8393	0.8887	0.7857
FCN8s_GP	0.8654	0.8968	0.7964
improve	**2.61%↑**	**0.81%↑**	**1.07%↑**
Unet	0.8150	0.9014	0.7738
Unet_GP	0.8394	0.9167	0.8367
improve	**2.44%↑**	**1.53%↑**	**6.29%↑**
Unet++	0.8381	0.8666	0.7967
Unet++_GP	0.8697	0.8568	0.8169
improve	**3.16%↑**	0.98%↓	**2.02%↑**

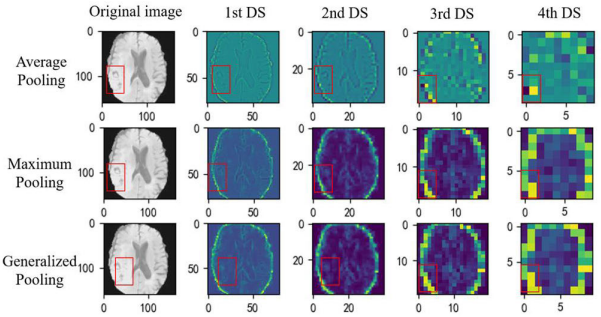

FIGURE 3. Result visualization of different pooling ways on BraTS18, where "DS" is the abbreviation of "down-sampling". The original image is from the slice BraTS18_TCIA04_343_1_41. The parts in the red boxes locate tumor tissues (in the original image) and their corresponding features extracted by 1, 2, 3 and 4 pooling operations.

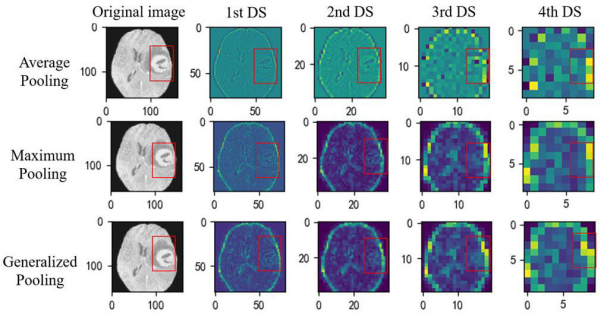

FIGURE 4. Result visualization of different pooling ways on BraTS19. The original image is from the slice BraTS19_CBICA_AOS_1_85.

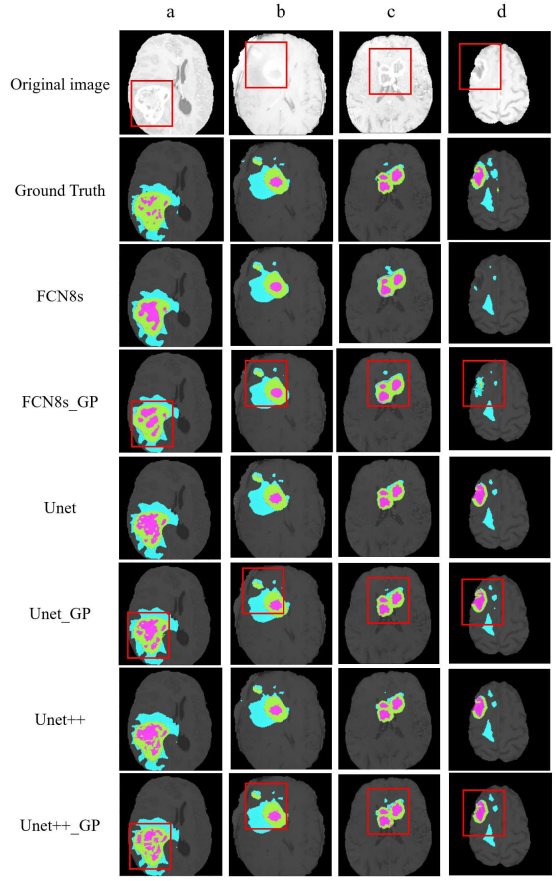

FIGURE 5. The segmentation results of the baseline models and their corresponding models with GP. Columns a and b are from BraTS18, and their slice numbers are BraTS18_TCIA03_474_1_84 and BraTS18_TCIA05_478_1_72, respectively. Columns c and d are from BraTS19, and their slice numbers are BraTS19_CBICA_APK_1_84 and BraTS19_TMC_30014_1_21, respectively.

C. THE EFFECT OF GENERALIZED POOLING

In order to evaluate the effect of our proposed generalized pooling (GP), we substitute GP for the traditional pooling (maximum pooling or average pooling) in some typical brain tumor segmentation models such as FCN8s [8], Unet [9] and Unet++ [21] to form FCN8s_GP (FCN8s with GP), Unet_GP (Unet with GP) and Unet++_GP (Unet++ with GP). Table 1 ∼ Table 6 show their segmentation effect compared with the baseline models FCN8s, Unet and Unet++ for whole tumor (WT), tumor core (TC) and enhance tumor (ET) [31] on BraTS18 and BraTS19 datasets.

From Table 1 ∼ Table 6, we can see that FCN8s_GP, Unet_GP and Unet++_GP outperform FCN8s, Unet and Unet++ respectively on both BraTS18 and BraTS19 datasets. On BraTS18, compared with the baseline models, all the three indicators Dice, PPV and Sensitivity of FCN8s_GP, Unet_GP and Unet++_GP are obviously improved by generalized pooling. Especially, TC-Dice of Unet++_GP increases by 8.74% compared with Unet++, and ET-sensitivity of FCN8s_GP increases by 6.28% compared with FCN8s. From the perspective of segmented tissues, ET has a larger improvement overall because it contains more small objects, whose features are apt to lose during down-sampling, and GP can help to keep more meaningful features. This will be validated in Section 5.4. The experimental results on BraTS19 are similar with BraTS18. These explain that our proposed generalized pooling algorithm is effective for brain tumor segmentation.

D. VISUALIZATION OF GENERALIZED POOLING

Besides the evaluation indicators Dice, PPV and Sensitivity, we also show the hot maps of down-sampling in Figure 3 and Figure 4 to illustrate the effect of GP visually. We can see that compared with the traditional pooling (maximum pooling and average pooling), GP can better keep the features of the tumor regions as the number of pooling operations increases on both BraTS18 and BraTS19 datasets.

Figure 5 shows the segmentation effect of the baseline models and their corresponding models with GP. It can be seen that the segmentation results of the baseline models are obviously improved through integrating with GP. This shows that GP is effective to brain tumor segmentation.

VI. CONCLUSION

The traditional pooling methods such as maximum pooling and average pooling usually result in the loss of average features or significant features. Especially in brain tumor segmentation, object tissues are relatively small, which will lead to more serious feature loss and affect segmentation performance. In this study, we first analyze the problems existing in the traditional pooling operations through their forward and back propagations. And then we unify the expressions of maximum pooling and average pooling, and propose a novel generalized pooling method. Based on generalized pooling, one does not have to use either maximum pooling or average pooling at pooling layers of CNN. Specially, generalized pooling calculates the weights within a pooling kernel conditioned on input images or feature maps to extract pooling features adaptively and improve segmentation performance. The experimental results verify the effectiveness of GP in brain tumor segmentation.

As a general pooling method in CNN, GP can be used in any CNN-based task not limited to brain tumor segmentation. The effect in other applications needs to be verified in our future work.

REFERENCES

[1] A. P. Patel, J. L. Fisher, E. Nichols, and F. Abd-Allah, "Global, regional, and national burden of brain and other CNS cancer, 1990–2016: A systematic analysis for the global Burden of disease study 2016," *Lancet Neurol.*, vol. 18, no. 4, pp. 376–393, 2019.

[2] D. Zikic, Y. Ioannou, and M. Brown, "Segmentation of brain tumor tissues with convolutional neural networks," in *Proc. MICCAI Workshop Multi-modal Brain Tumor Segmentation Challenge (BRATS)*, 2014, pp. 36–39.

[3] 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.

[4] T. Brosch, L. Y. W. Tang, Y. Yoo, D. K. B. Li, A. Traboulsee, and R. Tam, "Deep 3D convolutional encoder networks with shortcuts for multiscale feature integration applied to multiple sclerosis lesion segmentation," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1229–1239, May 2016.

[5] M. Havaei, A. Davy, D. Warde-Farley, 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.

[6] S. Qamar, H. Jin, R. Zheng, and P. Ahmad, "3D hyper-dense connected convolutional neural network for brain tumor segmentation," in *Proc. 14th Int. Conf. Semantics, Knowl. Grids (SKG)*, Sep. 2018, pp. 123–130.

[7] P. Mlynarski, H. Delingette, A. Criminisi, and N. Ayache, "3D convolutional neural networks for tumor segmentation using long-range 2D context," *Computerized Med. Imag. Graph.*, vol. 73, pp. 60–72, Apr. 2019.

[8] J. Long, E. Shellhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[9] 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.*, 2015, pp. 134–137.

[10] Q. Hou, L. Zhang, M.-M. Cheng, and J. Feng, "Strip pooling: Rethinking spatial pooling for scene parsing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1531–1550.

[11] H. Shen, R. Wang, J. Zhang, and S. McKenna, "Multi-task fully convolutional network for brain tumour segmentation," in *Proc. Annu. Conf. Med. Image Understand. Anal.*, 2017, pp. 245–247.

[12] H. Shen, R. Wang, J. Zhang, and S. J. McKenna, "Boundary-aware fully convolutional network for brain tumor segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2017, pp. 433–441.

[13] A. Jesson and T. Arbel, "Brain tumor segmentation using a 3D FCN with multi-scale loss," in *Proc. Int. Brainlesion Workshop (MICCAI)*, 2017, pp. 392–402.

[14] F. Milletari, N. Navab, and S. A. Ahmadi, "V-Net: Fully convolutional neural networks for, vol.umetric medical image segmentation," in *Proc. IEEE Int. Conf. 3D Vis. (3DV)*, 2016, pp. 565–571.

[15] H. T. Le and H. T. T. Pham, "Brain tumor segmentation using U-Net based deep neural networks," in *Proc. Int. Conf. Develop. Biomed. Eng. Vietnam*, 2018, pp. 39–42, doi: 10.1007/978-981-13-5859-3_7.

[16] A. Chaurasia and E. Culurciello, "LinkNet: Exploiting encoder representations for efficient semantic segmentation," 2017, *arXiv:1707.03718*.

[17] J. Zhang, Y. Jin, J. Xu, X. Xu, and Y. Zhang, "MDU-net: Multi-scale densely connected U-net for biomedical image segmentation," 2018, *arXiv:1812.00352*.

[18] Z. Meng, Z. Fan, Z. Zhao, and F. Su, "ENS-unet: End-to-end noise suppression U-Net for brain tumor segmentation," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2018.

[19] J. Dolz, I. Ben Ayed, and C. Desrosiers, "Dense multi-path U-net for ischemic stroke lesion segmentation in multiple image modalities," 2018, *arXiv:1810.07003*.

[20] S. M. Kamrul Hasan and C. A. Linte, "A modified U-net convolutional network featuring a nearest-neighbor Re-sampling-based elastic-transformation for brain tissue characterization and segmentation," in *Proc. IEEE Western New York Image Signal Process. Workshop (WNYISPW)*, Oct. 2018, pp. 1–5.

[21] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A nested U-net architecture for medical image segmentation," in *Proc. Int. Workshop Deep Learn. Med. Image Anal. (DLMIA)*, 2018, pp. 1761–1780.

[22] Y. Chen, Z. Cao, C. Cao, J. Yang, and J. Zhang, "A modified U-net for brain Mr image segmentation," in *Proc. Int. Conf. Cloud Comput. Secur.*, 2018, pp. 233–242.

[23] M. Noori, A. Bahri, and K. Mohammadi, "Attention-guided version of 2D UNet for automatic brain tumor segmentation," in *Proc. 9th Int. Conf. Comput. Knowl. Eng. (ICCKE)*, Oct. 2019, pp. 269–275.

[24] P. K. Gadosey, Y. Li, E. A. Agyekum, T. Zhang, Z. Liu, P. T. Yamak, and F. Essaf, "Sd-unet: Stripping down U-net for segmentation of biomedical images on platforms with low computational budgets," *Diagnostics*, vol. 10, no. 2, p. 110, 2020.

[25] Y. Cao, S. Liu, Y. Peng, and J. Li, "DenseUNet: Densely connected UNet for electron microscopy image segmentation," *IET Image Process.*, vol. 14, no. 12, pp. 2682–2689, Oct. 2020.

[26] P. Ahmad, "Context aware 3D UNet for brain tumor segmentation," 2020, *arXiv:2010.13082v2*.

[27] J. Colman *et al.*, *DR-Unet104 for Multimodal MRI Brain Tumor Segmentation*, 2020, *arXiv:2011.02840*.

[28] H. Wang and J. Yang, "FBUNet: Full convolutional network based on fusion block architecture for biomedical image segmentation," *J. Med. Biol. Eng.*, vol. 41, no. 2, pp. 185–202, Apr. 2021.

[29] J. Yang, J. Zhu, H. Wang, and X. Yang, "Dilated multiresUNet: Dilated multiresidual blocks network based on U-net for biomedical image segmentation," *Biomed. Signal Process. Control*, vol. 68, Jul. 2021, Art. no. 102643.

[30] Z. Huang, Y. Zhao, Y. Liu, and G. Song, "GCAUNet: A group cross-channel attention residual UNet for slice based brain tumor segmentation," *Biomed. Signal Process. Control*, vol. 70, Sep. 2021, Art. no. 102958.

[31] D. T. Nguyen, T. T. Tran, and V. T. Pham, "Attention U-net with active contour based hybrid loss for brain tumor segmentation," in *Soft Computing: Biomedical and Related Applications*, V. Kreinovich and N. H. Phuongr, Eds. Cham, Switzerland: Springer, 2021, pp. 35–45.

[32] B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, and K. Farahani, "The multimodal brain tumor image segmentation benchmark (BRATS)," *IEEE Trans. Med. Imag.*, vol. 34, no. 10, pp. 1993–2024, Oct. 2015.

[33] S. Bakas *et al.*, "Advancing the cancer genome Atlas glioma MRI collections with expert segmentation labels and radiomic features," *Nature Sci. Data*, vol. 4, 2017, Art. no. 170117, doi: 10.1038/sdata.2017.117.

[34] S. Bakas *et al.*, "Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge," 2018, *arXiv:1811.02629*.

KUANKUAN HAO received the B.S. degree in computer science and technology from Hebei University, China, in 2018. He is currently pursuing the M.S. degree in computer science and technology with Northeastern University, China. His current research interest includes medical image analysis.

SHUKUAN LIN was born in Jilin, China, in 1966. She graduated from the Department of Computer Science, Jilin University, in 1988. She is currently a Professor with Northeastern University, China. Her current research interests include medical image analysis and computer vision.

JIANZHONG QIAO was born in Liaoning, China, in 1964. He graduated from the Department of Computer Science, Xi'an Jiaotong University, in 1986. He is currently a Professor with Northeastern University, China. His current research interests include pattern recognition and computer vision.

YUE TU received the B.S. and M.S. degrees in computer science and technology from Northeastern University, China, in 2016 and 2018, respectively, where he is currently pursuing the Ph.D. degree in computer science and technology.